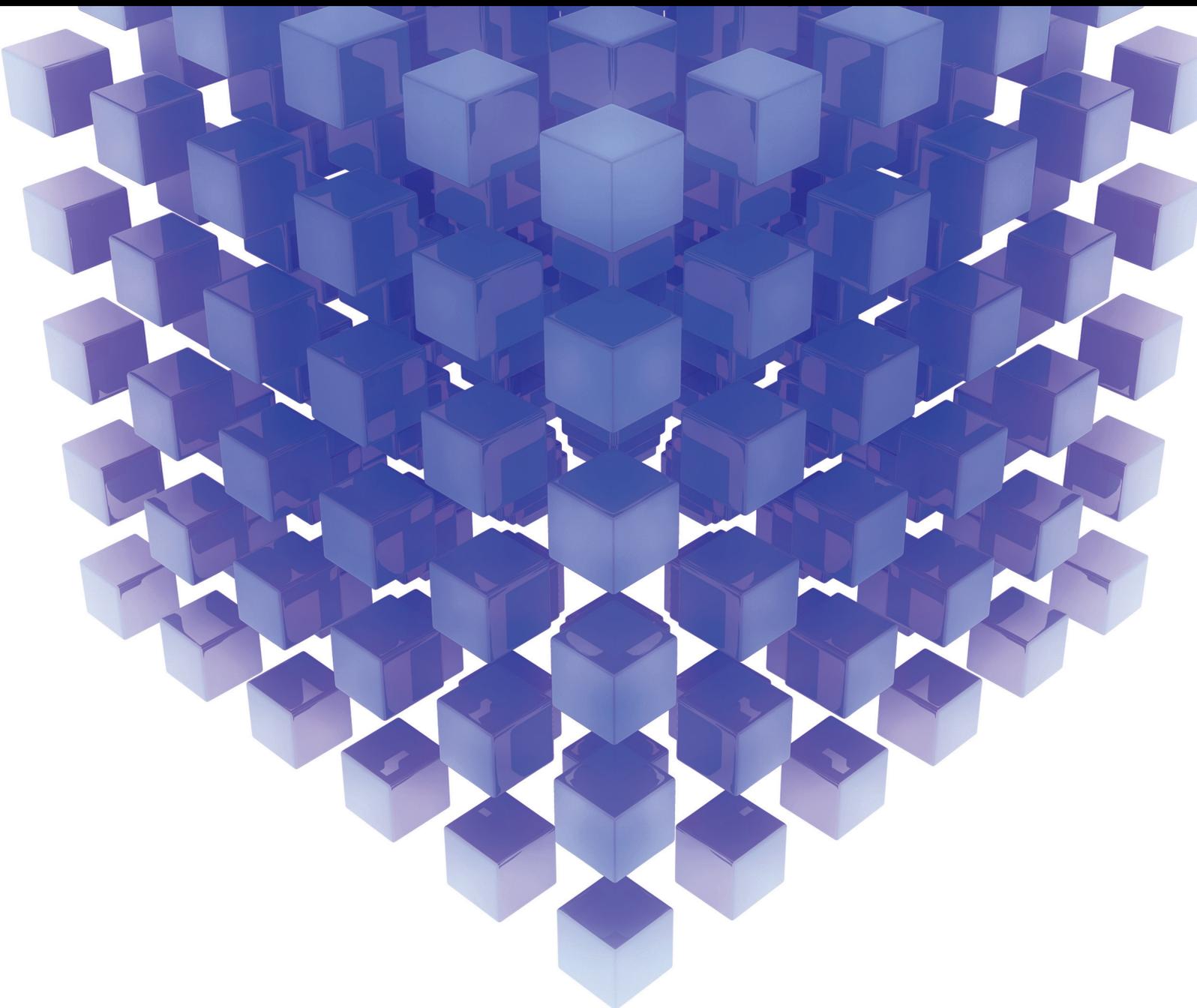


Mathematical Problems in Engineering

# Computational Intelligence in Image Processing 2014

Guest Editors: Erik Cuevas, Daniel Zaldívar, Gonzalo Pajares,  
Marco Perez-Cisneros, and Raúl Rojas





---

# **Computational Intelligence in Image Processing 2014**

Mathematical Problems in Engineering

---

## **Computational Intelligence in Image Processing 2014**

Guest Editors: Erik Cuevas, Daniel Zaldívar, Gonzalo Pajares,  
Marco Perez-Cisneros, and Raúl Rojas



---

Copyright © 2015 Hindawi Publishing Corporation. All rights reserved.

This is a special issue published in “Mathematical Problems in Engineering.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

## Editorial Board

- Mohamed Abd El Aziz, Egypt  
Farid Abed-Meraim, France  
Silvia Abrahão, Spain  
Paolo Addresso, Italy  
Claudia Adduce, Italy  
Ramesh Agarwal, USA  
Juan C. Agüero, Australia  
Ricardo Aguilar-López, Mexico  
Tarek Ahmed-Ali, France  
Hamid Akbarzadeh, Canada  
Muhammad N. Akram, Norway  
Mohammad-Reza Alam, USA  
Salvatore Alfonzetti, Italy  
Francisco Alhama, Spain  
Juan A. Almendral, Spain  
Lionel Amodeo, France  
Igor Andrianov, Germany  
Sebastian Anita, Romania  
Renata Archetti, Italy  
Felice Arena, Italy  
Sabri Arik, Turkey  
Fumihiko Ashida, Japan  
Hassan Askari, Canada  
Mohsen Asle Zaeem, USA  
Francesco Aymerich, Italy  
Seungik Baek, USA  
Khaled Bahlali, France  
Laurent Bako, France  
Stefan Balint, Romania  
Alfonso Banos, Spain  
Roberto Baratti, Italy  
Martino Bardi, Italy  
Azeddine Beghdadi, France  
Abdel-Hakim Bendada, Canada  
Ivano Benedetti, Italy  
Elena Benvenuti, Italy  
Jamal Berakdar, Germany  
Enrique Berjano, Spain  
Jean-Charles Beugnot, France  
Simone Bianco, Italy  
David Bigaud, France  
Jonathan N. Blakely, USA  
Paul Bogdan, USA  
Daniela Boso, Italy  
A.-O. Boudraa, France
- Francesco Braghin, Italy  
Michael J. Brennan, UK  
Maurizio Brocchini, Italy  
Julien Bruchon, France  
Javier Buldu, Spain  
Tito Busani, USA  
P. Cacciola, UK  
Salvatore Caddemi, Italy  
Jose E. Capilla, Spain  
Ana Carpio, Spain  
Miguel E. Cerrolaza, Spain  
Mohammed Chadli, France  
Gregory Chagnon, France  
Ching-Ter Chang, Taiwan  
Michael J. Chappell, UK  
Kacem Chehdi, France  
Chunlin Chen, China  
Xinkai Chen, Japan  
Francisco Chicano, Spain  
Hung-Yuan Chung, Taiwan  
Joaquim Ciurana, Spain  
John D. Clayton, USA  
Carlo Cosentino, Italy  
Paolo Crippa, Italy  
Erik Cuevas, Mexico  
Peter Dabnichki, Australia  
Luca D'Acerno, Italy  
Weizhong Dai, USA  
P. Damodaran, USA  
Farhang Daneshmand, Canada  
Fabio De Angelis, Italy  
Stefano de Miranda, Italy  
Filippo de Monte, Italy  
Xavier Delorme, France  
Luca Deseri, USA  
Yannis Dimakopoulos, Greece  
Zhengtao Ding, UK  
Ralph B. Dinwiddie, USA  
Mohamed Djemai, France  
Alexandre B. Dolgui, France  
George S. Dulikravich, USA  
Bogdan Dumitrescu, Finland  
Horst Ecker, Austria  
Ahmed El Hajjaji, France  
Fouad Erchiqui, Canada
- Anders Eriksson, Sweden  
Giovanni Falsone, Italy  
Hua Fan, China  
Yann Favennec, France  
Giuseppe Fedele, Italy  
Roberto Fedele, Italy  
Jacques Ferland, Canada  
Jose R. Fernandez, Spain  
Simme D. Flapper, Netherlands  
Thierry Floquet, France  
Eric Florentin, France  
Francesco Franco, Italy  
Tomonari Furukawa, USA  
Mohamed Gadala, Canada  
Matteo Gaeta, Italy  
Zoran Gajic, USA  
Ciprian G. Gal, USA  
Ugo Galvanetto, Italy  
Akemi Gálvez, Spain  
Rita Gamberini, Italy  
Maria Gandarias, Spain  
Arman Ganji, Canada  
Xin-Lin Gao, USA  
Zhong-Ke Gao, China  
Giovanni Garcea, Italy  
Fernando Garcia, Spain  
Laura Gardini, Italy  
A. Gasparetto, Italy  
V. Gattulli, Italy  
Oleg V. Gendelman, Israel  
M. H. Ghayesh, Australia  
A. M. Gil-Lafuente, Spain  
Hector Gómez, Spain  
Rama S. R. Gorla, USA  
Oded Gottlieb, Israel  
Antoine Grall, France  
Jason Gu, Canada  
Quang Phuc Ha, Australia  
Ofer Hadar, Israel  
Masoud Hajarian, Iran  
Frédéric Hamelin, France  
Zhen-Lai Han, China  
Thomas Hanne, Switzerland  
Takashi Hasuike, Japan  
Xiao-Qiao He, China

M.I. Herreros, Spain  
Vincent Hilaire, France  
Eckhard Hitzer, Japan  
Jaromir Horacek, Czech Republic  
Muneo Hori, Japan  
András Horváth, Italy  
Gordon Huang, Canada  
Sajid Hussain, Canada  
Asier Ibeas, Spain  
Giacomo Innocenti, Italy  
Emilio Insfran, Spain  
Nazrul Islam, USA  
Payman Jalali, Finland  
Reza Jazar, Australia  
Khalide Jbilou, France  
Linni Jian, China  
Bin Jiang, China  
Zhongping Jiang, USA  
Ningde Jin, China  
Grand R. Joldes, Australia  
Joaquim Joao Judice, Portugal  
Tadeusz Kaczorek, Poland  
Tamas Kalmar-Nagy, Hungary  
Tomasz Kapitaniak, Poland  
Haranath Kar, India  
K. Karamanos, Belgium  
C. M. Khalique, South Africa  
Do Wan Kim, Korea  
Nam-Il Kim, Korea  
Oleg Kirillov, Germany  
Manfred Krafczyk, Germany  
Frederic Kratz, France  
Jurgen Kurths, Germany  
K. Kyamakya, Austria  
Davide La Torre, Italy  
Risto Lahdelma, Finland  
Hak-Keung Lam, UK  
Antonino Laudani, Italy  
Aime' Lay-Ekuakille, Italy  
Marek Lefik, Poland  
Yaguo Lei, China  
Thibault Lemaire, France  
Stefano Lenci, Italy  
Roman Lewandowski, Poland  
Qing Q. Liang, Australia  
Panos Liatsis, UK  
Peide Liu, China  
Peter Liu, Taiwan

Wanquan Liu, Australia  
Yan-Jun Liu, China  
Jean J. Loiseau, France  
Paolo Lonetti, Italy  
Luis M. López-Ochoa, Spain  
V. C. Loukopoulos, Greece  
Valentin Lychagin, Norway  
Fazal M. Mahomed, South Africa  
Yassir T. Makkawi, UK  
Noureddine Manamanni, France  
Didier Maquin, France  
Paolo Maria Mariano, Italy  
Benoit Marx, France  
Gépard A. Maugin, France  
Driss Mehdi, France  
Roderick Melnik, Canada  
Pasquale Memmolo, Italy  
Xiangyu Meng, Canada  
Jose Merodio, Spain  
Luciano Mescia, Italy  
Laurent Mevel, France  
Yuri V. Mikhlin, Ukraine  
Aki Mikkola, Finland  
Hiroyuki Mino, Japan  
Pablo Mira, Spain  
Vito Mocella, Italy  
Roberto Montanini, Italy  
Gisele Mophou, France  
Rafael Morales, Spain  
Aziz Moukrim, France  
Emiliano Mucchi, Italy  
Domenico Mundo, Italy  
Jose J. Muñoz, Spain  
Giuseppe Muscolino, Italy  
Marco Mussetta, Italy  
Hakim Naceur, France  
Hassane Naji, France  
Dong Ngoduy, UK  
Tatsushi Nishi, Japan  
Ben T. Nohara, Japan  
Mohammed Nouari, France  
Mustapha Nourelfath, Canada  
Sotiris K. Ntouyas, Greece  
Roger Ohayon, France  
Mitsuhiro Okayasu, Japan  
Eva Onaindia, Spain  
Javier Ortega-Garcia, Spain  
A. Ortega-Moñux, Spain

Naohisa Otsuka, Japan  
Erika Ottaviano, Italy  
Alkiviadis Paipetis, Greece  
Alessandro Palmeri, UK  
Anna Pandolfi, Italy  
Elena Panteley, France  
Manuel Pastor, Spain  
Pubudu N. Pathirana, Australia  
Francesco Pellicano, Italy  
Haipeng Peng, China  
Mingshu Peng, China  
Zhike Peng, China  
Marzio Pennisi, Italy  
Matjaz Perc, Slovenia  
Francesco Pesavento, Italy  
Maria do Rosário Pinho, Portugal  
Antonina Pirrotta, Italy  
Vicent Pla, Spain  
Javier Plaza, Spain  
Jean-Christophe Ponsart, France  
Mauro Pontani, Italy  
Stanislav Potapenko, Canada  
Sergio Preidikman, USA  
Christopher Pretty, New Zealand  
Carsten Proppe, Germany  
Luca Pugi, Italy  
Yuming Qin, China  
Dane Quinn, USA  
Jose Ragot, France  
K. R. Rajagopal, USA  
Gianluca Ranzi, Australia  
Sivaguru Ravindran, USA  
Alessandro Reali, Italy  
Oscar Reinoso, Spain  
Nidhal Rezg, France  
Ricardo Riaza, Spain  
Gerasimos Rigatos, Greece  
José Rodellar, Spain  
Rosana Rodriguez-Lopez, Spain  
Ignacio Rojas, Spain  
Carla Roque, Portugal  
Aline Roumy, France  
Debasish Roy, India  
Rubén Ruiz García, Spain  
Antonio Ruiz-Cortes, Spain  
Ivan D. Rukhlenko, Australia  
Mazen Saad, France  
Kishin Sadarangani, Spain

Mehrdad Saif, Canada  
Miguel A. Salido, Spain  
Roque J. Saltarén, Spain  
Francisco J. Salvador, Spain  
Alessandro Salvini, Italy  
Maura Sandri, Italy  
Miguel A. F. Sanjuan, Spain  
Juan F. San-Juan, Spain  
Roberta Santoro, Italy  
Ilmar Ferreira Santos, Denmark  
José A. Sanz-Herrera, Spain  
Nickolas S. Sapidis, Greece  
E. J. Sapountzakis, Greece  
Andrey V. Savkin, Australia  
Valery Sbitnev, Russia  
Thomas Schuster, Germany  
Mohammed Seaid, UK  
Lotfi Senhadji, France  
Joan Serra-Sagrasta, Spain  
Leonid Shaikhet, Ukraine  
Hassan M. Shanechi, USA  
Sanjay K. Sharma, India  
Bo Shen, Germany  
Babak Shotorban, USA  
Zhan Shu, UK  
Dan Simon, USA  
Luciano Simoni, Italy  
Christos H. Skiadas, Greece  
Michael Small, Australia  
F. Soldovieri, Italy  
Raffaele Solimene, Italy

Ruben Specogna, Italy  
Sri Sridharan, USA  
Ivanka Stamova, USA  
Yakov Strelniker, Israel  
Sergey A. Suslov, Australia  
Thomas Svensson, Sweden  
Andrzej Swierniak, Poland  
Yang Tang, Germany  
Sergio Teggi, Italy  
Alexander Timokha, Norway  
Rafael Toledo, Spain  
Gisella Tomasini, Italy  
E. Tornabene, Italy  
Antonio Tornambe, Italy  
Fernando Torres, Spain  
Fabio Tramontana, Italy  
Sébastien Tremblay, Canada  
Irina N. Trendafilova, UK  
George Tsiatas, Greece  
Antonios Tsourdos, UK  
Vladimir Turetsky, Israel  
Mustafa Tutar, Spain  
E. Tzirtzilakis, Greece  
Filippo Ubertini, Italy  
Francesco Ubertini, Italy  
Hassan Ugail, UK  
Giuseppe Vairo, Italy  
K. Vajravelu, USA  
Robertt A. Valente, Portugal  
Pandian Vasant, Malaysia  
Miguel E. Vázquez-Méndez, Spain

Josep Vehi, Spain  
Kalyana C. Veluvolu, Korea  
Fons J. Verbeek, Netherlands  
Franck J. Vernerey, USA  
Georgios Veronis, USA  
Anna Vila, Spain  
Rafael J. Villanueva, Spain  
Uchechukwu E. Vincent, UK  
Mirko Viroli, Italy  
Michael Vynnycky, Sweden  
Junwu Wang, China  
Shuming Wang, Singapore  
Yan-Wu Wang, China  
Yongqi Wang, Germany  
Desheng D. Wu, Canada  
Yuqiang Wu, China  
Guangming Xie, China  
Xuejun Xie, China  
Gen Qi Xu, China  
Hang Xu, China  
Xinggang Yan, UK  
Luis J. Yebra, Spain  
Peng-Yeng Yin, Taiwan  
Ibrahim Zeid, USA  
Huaguang Zhang, China  
Qingling Zhang, China  
Jian Guo Zhou, UK  
Quanxin Zhu, China  
Mustapha Zidi, France  
Alessandro Zona, Italy

# Contents

**Computational Intelligence in Image Processing 2014**, Erik Cuevas, Daniel Zaldívar, Gonzalo Pajares, Marco Perez-Cisneros, and Raúl Rojas  
Volume 2015, Article ID 782826, 3 pages

**Sealing Clay Text Segmentation Based on Radon-Like Features and Adaptive Enhancement Filters**, Xia Zheng, Wei Wei, Houbing Song, and Wei Li  
Volume 2015, Article ID 983601, 8 pages

**An Image Watermarking Scheme Using Arnold Transform and Fuzzy Smooth Support Vector Machine**, Lin Sun, Jiucheng Xu, Xingxing Zhang, and Yun Tian  
Volume 2015, Article ID 931672, 14 pages

**An Image Filter Based on Shearlet Transformation and Particle Swarm Optimization Algorithm**, Kai Hu, Aiguo Song, Min Xia, Zhiyong Fan, Xiaoying Chen, Ruidong Zhang, and Zhichen Zheng  
Volume 2015, Article ID 414561, 9 pages

**Stamps Detection and Classification Using Simple Features Ensemble**, Paweł Forczmański and Andrzej Markiewicz  
Volume 2015, Article ID 367879, 15 pages

**Multiscale Hybrid Nonlocal Means Filtering Using Modified Similarity Measure**, Zahid Hussain Shamsi and Dai-Gyoung Kim  
Volume 2015, Article ID 318341, 17 pages

**Applying BAT Evolutionary Optimization to Image-Based Visual Servoing**, Marco Perez-Cisneros, Gerardo Garcia-Gil, Sabrina Vega-Maldonado, J. Arámburo-Lizárraga, Erik Cuevas, and Daniel Zaldívar  
Volume 2015, Article ID 590138, 11 pages

**Multithreshold Segmentation by Using an Algorithm Based on the Behavior of Locust Swarms**, Erik Cuevas, Adrián González, Fernando Fausto, Daniel Zaldívar, and Marco Pérez-Cisneros  
Volume 2015, Article ID 805357, 25 pages

**Face Recognition Using MLP and RBF Neural Network with Gabor and Discrete Wavelet Transform Characterization: A Comparative Study**, Fatma Zohra Chelali and Amar Djeradi  
Volume 2015, Article ID 523603, 16 pages

**Inferring Visual Perceptual Object by Adaptive Fusion of Image Salient Features**, Xin Xu, Nan Mu, and Hong Zhang  
Volume 2015, Article ID 973241, 9 pages

**Multiframe Superresolution Reconstruction Based on Self-Learning Method**, Shao-Shuo Mu, Ye Zhang, Ping Jia, Xun Yang, and Xiao-Feng Qiu  
Volume 2015, Article ID 181864, 12 pages

**Automatic Segmentation of Nature Object Using Salient Edge Points Based Active Contour**, Shangbing Gao, Yunyang Yan, Youdong Zhang, Jingbo Zhou, Suqun Cao, and Jianxun Xue  
Volume 2015, Article ID 174709, 12 pages

## Editorial

# Computational Intelligence in Image Processing 2014

**Erik Cuevas,<sup>1</sup> Daniel Zaldívar,<sup>1</sup> Gonzalo Pajares,<sup>2</sup> Marco Perez-Cisneros,<sup>3</sup> and Raúl Rojas<sup>4</sup>**

<sup>1</sup>*Departamento de Electrónica, Universidad de Guadalajara, CUCEI, Avenida Revolución 1500, 44430 Guadalajara, JAL, Mexico*

<sup>2</sup>*Departamento de Ingeniería de Software e Inteligencia Artificial, Facultad Informática, Universidad Complutense, 28040 Madrid, Spain*

<sup>3</sup>*Centro Universitario de Tonalá, Universidad de Guadalajara, Avenida Nuevo Periférico No. 555, Ejido San José Tatepozco, 48525 Tonalá, JAL, Mexico*

<sup>4</sup>*Institut für Informatik, Freie Universität Berlin, Arnimallee 7, 14195 Berlin, Germany*

Correspondence should be addressed to Erik Cuevas; [erik.cuevas@cucei.udg.mx](mailto:erik.cuevas@cucei.udg.mx)

Received 2 July 2015; Accepted 8 July 2015

Copyright © 2015 Erik Cuevas et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Image processing is a dynamic and fast moving field of research. Recent advances in image processing have led to an explosion in the use of images in a variety of scientific and engineering applications. Therefore, each new approach that is developed by engineers, mathematicians, and computer scientists is quickly identified, understood, and assimilated in order to be applied to image processing problems.

Such growing interest has forged the need for including computational intelligence as a potential tool to provide novel solutions to challenging image processing problems. One of the most demanding issues is the handling of image uncertainties that cannot be otherwise eliminated, including various sorts of information that is incomplete, noisy, imprecise, fragmentary, not fully reliable, vague, contradictory, deficient, and overloading. They result in a lack of the full and precise knowledge of the system including the determining and selection of evaluation criteria, alternatives, weights, assignment scores, and the final integrated decision result. Computational intelligent techniques including fuzzy logic, neural networks, and evolutionary methods have shown great potential to solve such image processing problems as alternative to the existing classical techniques.

The importance of computational intelligence in image processing has increased in all engineering areas. Such a fact is evident from a quick look at special issues, congresses, and specialized journals that focus on such a topic. The main objective of this special issue is to bridge the gap between computational intelligence techniques and challenging image

processing applications. Since this idea was first conceived, the goal has aimed at exposing the readers to the cutting-edge research and applications that are going on across the domain of image processing, particularly those whose contemporary computational intelligence techniques can be or have been successfully employed.

The special issue received several high-quality submissions from different countries all over the world. All submitted papers have followed the same standard of peer-reviewing by at least three independent reviewers, just as it is applied to regular submissions to this journal. Due to the limited space, a very short number of papers have been finally included. The primary guideline has been to demonstrate the wide scope of computational intelligence algorithms and their applications to image processing problems.

The paper authored by P. Forczmański and A. Markiewicz presents a method for the detection and classification of rubber stamp instances in scanned documents. The approach works on typical stamps of different colors and shapes. For color images, color space transformation is applied in order to find potential color stamps. Monochrome stamps are detected through shape specific algorithms. Following the feature extraction stage, identified candidates are subjected to classification task using a set of shape descriptors. The authors perform two-tier classification in order to discriminate between stamps and no-stamps and then classify stamps in terms of their shape. The experiments carried out on a considerable set of real documents gathered from the Internet showed high potential of the proposed method.

K. Hu et al. propose an image filter method which combines the merits of Shearlet transformation and particle swarm optimization (PSO) algorithm. Firstly, the authors use the classical Shearlet transform to decompose noised image into many subwavelets under multiscale and multiorientation. Secondly, they gave weighted factor to those subwavelets obtained. Then, using classical Shearlet inverse transform, authors obtained a composite image which gathers those weighted subwavelets. After that, they have designed a fast and rough evaluation method to evaluate noise level of the new image; by using this method as fitness, authors adopted PSO to find the optimal weighted factor; after lots of iterations through the optimal factors and Shearlet inverse transform, the best denoised images are defined. Experimental results demonstrate that proposed algorithm eliminates noise effectively and yields good peak signal noise ratio (PSNR).

S.-S. Mu et al. present a multiframe superresolution reconstruction method based on self-learning methods. In the approach, first, multiple images from the same scene are selected to be both input and training images, and larger-scale images, which are also involved in the training set, are constructed from the learning dictionary. Then, different larger-scale images are constructed via repetition of the first step and the initial high-resolution (HR) sets whose scale closely approximates that of the target HR image are finally obtained. Lastly, initial HR images are fused into one target HR image. The simulation results demonstrate that the proposed algorithm produces more accurate reconstructions than those produced by other general superresolution algorithms, while, in real scene experiments, the proposed algorithm can run well and create clearer HR images from input images captured by cameras.

The paper by Z. H. Shamsi and D.-G. Kim proposed a new algorithm for image denoising. The approach involves two steps: the first step is a multiscale implementation of an accelerated nonlocal means filtering in the discrete stationary wavelet domain to obtain a refined version of the noisy patches for later comparison. The next step is to apply the proposed modification of standard nonlocal means filtering to the noisy image using the reference patches obtained in the first step. These refined patches contain less noise, and consequently the computation of normal vectors and partial derivatives is more precise. Experimental results show equivalent or better performance of the proposed algorithm compared to various state-of-the-art algorithms.

S. Gao et al. present a novel model for unsupervised segmentation of viewer's attention object from natural images based on localizing region-based active contour (LRAC) model. Considering a Harris detector and the core saliency map, authors get the salient object edge points. Then, these points are employed as the seeds of initial convex hull. Finally, this convex hull is improved by the edge-preserving filter to generate the initial contour for our automatic object segmentation system. Extensive experiments on a large variety of natural images demonstrate that their algorithm consistently outperforms the popular existing salient object segmentation methods, yielding higher precision and better recall rates.

The paper by F. Z. Chelali and A. Djeradi proposed a face recognition system using multilayer perceptron (MLP) and radial basis functions (RBF). In the approach, Gabor and discrete wavelet are considered for the extraction of features from facial images. The experiments over two standard facial databases demonstrate that the proposed method outperforms standard methods in terms of robustness and accuracy.

X. Xu et al. present a data driven approach to adaptively select proper features for different kinds of images. This method exploits low-level features containing the most distinguishable salient information per image. Then the image saliency can be calculated based on the adaptive weight selection scheme. A large number of experiments are conducted on a standard database to compare the performance of the proposed method against state-of-the-art saliency computational models.

The paper by E. Cuevas et al. presents an algorithm for the automatic selection of pixel classes for segmentation proposes. The approach combines a novel evolutionary method with the definition of a new objective function that appropriately evaluates the segmentation quality with respect to the number of classes. The new evolutionary algorithm, called Locust Search (LS), is based on the behavior of swarms of locusts. Different to the most of existent evolutionary algorithms, it explicitly avoids the concentration of individuals in the best positions, avoiding critical flaws such as the premature convergence to suboptimal solutions and the limited exploration-exploitation balance. Experimental tests over several benchmark functions and images validate the efficiency of the proposed technique with regard to accuracy and robustness.

X. Zheng et al. introduce a new method for text segmentation based on Radon-like features and adaptive enhancement filters. In the approach, first, an adaptive enhancement LM filter bank is used to get the maximum energy image; second, the edge image of the maximum energy image is calculated; finally, a set of Radon-like feature images is generated by combining maximum energy image and its edge image. The average image of Radon-like feature images is segmented by a classical image thresholding method. Compared with 2D Otsu, GA, and Fast FCM, the experiment results show that this method can perform better in terms of accuracy and completeness of the text.

The paper by L. Sun et al. presents a new image watermarking scheme based on Arnold Transform (AT) and Fuzzy Smooth Support Vector Machine (FSSVM). Compared with other watermarking techniques, the approach can promote the security by adding more secret keys, and the imperceptibility of watermark is improved by introducing fuzzy rules. Experimental results show that the proposed method outperforms many existing methods against various types of attacks.

M. Perez-Cisneros et al. introduce a predictive control strategy for an image-based visual servoing scheme that employs evolutionary optimization. The visual control task is approached as a nonlinear optimization problem that naturally handles relevant visual servoing constraints such as workspace limitations and visibility restrictions. As the predictive scheme requires a reliable model, this paper uses

a local model that is based on the visual interaction matrix and a global model that employs 3D trajectory data extracted from a quaternion-based interpolator.

### **Acknowledgments**

Finally, we would like to express our gratitude to all of the authors for their contributions and the reviewers for their efforts to provide valuable comments and feedback. We hope that this special issue offers a comprehensive and timely view of the area of applications of computational intelligence in image processing and that it will grant stimulation for further research.

*Erik Cuevas  
Daniel Zaldívar  
Gonzalo Pajares  
Marco Perez-Cisneros  
Raúl Rojas*

## Research Article

# Sealing Clay Text Segmentation Based on Radon-Like Features and Adaptive Enhancement Filters

Xia Zheng,<sup>1</sup> Wei Wei,<sup>2</sup> Houbing Song,<sup>3</sup> and Wei Li<sup>4</sup>

<sup>1</sup>Department of Culture Heritage & Museology, Zhejiang University, Hangzhou, Zhejiang 310028, China

<sup>2</sup>School of Computer Science and Engineering, Xi'an University of Technology, Xi'an, Shanxi 710048, China

<sup>3</sup>Department of Electrical and Computer Engineering, West Virginia University Institute of Technology, Montgomery, WV 25136, USA

<sup>4</sup>School of Information Technologies, The University of Sydney, Sydney, NSW 2006, Australia

Correspondence should be addressed to Xia Zheng; zhengxia@zju.edu.cn

Received 9 October 2014; Revised 9 January 2015; Accepted 12 January 2015

Academic Editor: Daniel Zaldivar

Copyright © 2015 Xia Zheng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Text extraction is a key issue in sealing clay research. The traditional method based on rubbings increases the risk of sealing clay damage and is unfavorable to sealing clay protection. Therefore, using digital image of sealing clay, a new method for text segmentation based on Radon-like features and adaptive enhancement filters is proposed in this paper. First, adaptive enhancement LM filter bank is used to get the maximum energy image; second, the edge image of the maximum energy image is calculated; finally, Radon-like feature images are generated by combining maximum energy image and its edge image. The average image of Radon-like feature images is segmented by the image thresholding method. Compared with 2D Otsu, GA, and FastFCM, the experiment result shows that this method can perform better in terms of accuracy and completeness of the text.

## 1. Introduction

In recent years, with the development of computer, network, and multimedia technology, digital preservation of culture relics has become richer in its connotations: from the initial real-time recording and permanent preservation to multiple levels such as preservation studies, dissemination, and utilization. Digital preservation of culture relics aims at effectively protecting the body of culture relics in the process of exploring and taking their historical, artistic, and scientific value. Among culture relics, sealing clay plays an important role in revising and supplementing for ancient documents that record official systems and geographic data. As a result, text extraction and identification have become key issues in sealing clay research. With the promoting of digital preservation of culture relics in China, more and more sealing clays are presented and archived by images. Sealing clay research based on images can be conducted without direct contact with the body of culture relics, which makes a balance between preservation and utilization of culture relics. Depending on image processing and analyzing technology, many researches

are possible. In these researches, text extraction is the most significant one, which will affect following steps, such as feature quantification and text recognition.

As for the sealing clay text extraction, traditional method is identifying the text by ink rubbing. This method needs to make rubbings on a piece of paper from the sealing clay, in which the sealing clay is directly used and contacted, increasing the risks of damaging. For the seriously incomplete or damaged sealing clay whose preservation condition is not so good, especially, ink rubbings are not advisable from the perspective of culture relic preservation. However, taking sealing clay image as the research objective can directly extract text from the image, without the process of making ink rubbings. Since the sealing clay surface is simple, mainly including the background and the text, thresholding segmentation method is available to the research. Currently, thresholding segmentation method includes histogram-shape-based method, such as peak-valley thresholding [1] and shape modeling thresholding [2]; cluster-based method, such as minimum error thresholding [3] and fuzzy clustering thresholding [4]; entropy-based method, such as cross entropy

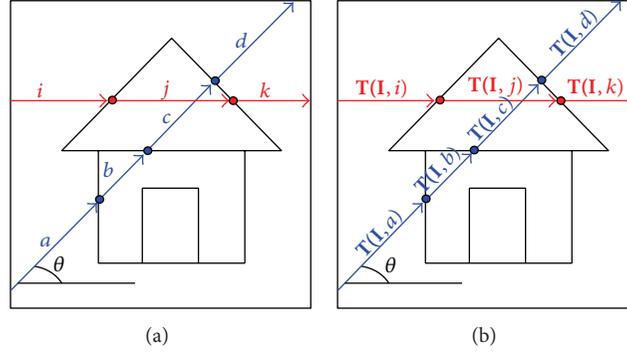


FIGURE 1: The diagram of Radon-like features.

thresholding [5] and fuzzy entropy thresholding [6]; and similar characteristic-based method, such as fuzzy similar thresholding [7] and semantic thresholding [8]. All of the thresholding segmentation methods above can achieve better effects with their own appropriate images. Nevertheless, there are rare systemic studies on the segmentation of sealing clay images. Thus, on the basis of previous studies on image segmentation and insistent demands of sealing clay research and preservation, the present study puts forward a new method for text segmentation based on radon-like features and adaptive enhancement filters compared with 2D Otsu, GA, and FastFCM. Experimental results show that the method proposed by this research cannot only extract text completely and accurately, but also reduce the interference of incomplete or damaged surface and uneven inward of sealing clay, which provides an appropriate solution to the problem of sealing clay text segmentation.

## 2. Radon-Like Features and Adaptive Enhancement Filters

**2.1. Radon-Like Features.** Radon-like features were originated from Radon transform [9], which is a method of integral transform. The Radon transform formula of 2D continuous function  $f(x, y)$  is as follows:

$$\mathbf{R}(m, \tau) = \int_{-\infty}^{\infty} f(x, y = \tau + mx) dx. \quad (1)$$

$m$  and  $\tau$  are, respectively, the slope and intercept of the straight line and Radon transform is a method of evaluating integrals along a straight line which is determined by  $m$  and  $\tau$ . Radon transform is widely used in tomography, such as MRI (magnetic resonance imaging), and its inverse transformation is often used to reconstruct the original images. The discrete performance of (1) is

$$\mathbf{R}(m, \tau) = \sum_x f(x, y = \tau + mx). \quad (2)$$

The Radon-like features just borrowed from the main ideas of Radon transform [9]. Given an image of  $\mathbf{I}(x, y)$ , the parameterized expression of line  $\mathbf{L}$  is  $\mathbf{L}(t) = (x(t), y(t))$ , which is not to make the information about image  $\mathbf{I}$  completely map to line  $\mathbf{L}$  to get an accumulated value, but to make

the information distribute to the different line segments of line  $\mathbf{L}$ . The line segments are determined by a collection of knots. Supposing that the collection of crunodes is  $(t_1, \dots, t_n)$ , so point  $P$  on line  $\mathbf{L}$  which located between  $(x(t_i), y(t_i))$  and  $(x(t_{i+1}), y(t_{i+1}))$  is the class Radon-like feature. And its value is determined by

$$\Psi(p, \mathbf{L}, t_i, t_{i+1}) [\mathbf{I}(x, y)] = \mathbf{T}(\mathbf{I}, \mathbf{L}(t)), \quad t \in [t_i, t_{i+1}]. \quad (3)$$

$\mathbf{T}$  is an extraction function which determines how to calculate the values of Radon-like features. When fixing the rotation angle  $\theta$  of line  $\mathbf{L}$  and changing its intercept, according to the (3), we can get the Radon-like features image of  $\mathbf{I}'(x, y)$  which contains the values of Radon-like features and is the same size as  $\mathbf{I}(x, y)$ . If rotation angle  $\theta$  is also varied, every pixel  $(x, y)$  of image  $\mathbf{I}$  will produce a vector of Radon-like features. As shown in Figure 1(a), two lines from different angles will intersect with the picture to form nodes and line segments. According to the extract function,  $\mathbf{T}$ , the value of Radon-like features image is highlighted in Figure 1(b).

In Figure 1(a), the line segments are represented by  $a, b, c,$  and  $d$  which are formed by the blue line intersecting with the picture; the line segments are represented by  $i, j,$  and  $k$  which are formed by the red line intersecting with the picture.

In Figure 1(b), the value of Radon-like features will be defined by the extraction function  $\mathbf{T}$ .

The usefulness of Radon-like feature depends on knots and the extraction function. For the application at hand, image analysis, knots derived from the edges in the image, provides a useful guide to the structures of the image. When a line scans through the input image, its intersections with the image's edge map define the knots and the line segments. The choice of the extraction function provides enough flexibility to differentiate among various structures of image.

**2.2. The Adaptive Enhancement Filters.** The linear combinations of the Gaussian model and its derivatives are suitable for the human visual system, which has been well applied in the areas of image denoising, edge detection, and so on. The first and second derivative of Gaussian function filters are mainly used for strengthening the edges. Based on above theories, this paper adopted Leung-Malik filter bank to strengthen the structural information of images. Leung-Malik filter is the composition of polygon filter, bar filter, and punctate filter

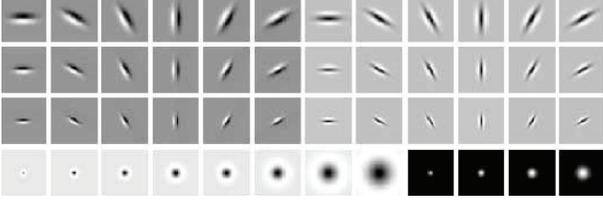


FIGURE 2: Leung-Malik filter bank.



FIGURE 3: The five basic strokes and their direction in Chinese characters.

which are multiscale and multidirection filters, and it can be divided into 36 first and second derivatives of Gaussian function filters (6 directions and 3 scales) shown in first three lines of Figure 2, 8 Gaussian Laplacian filters, and 4 Gaussian filters, shown in the last line of Figure 2 [10].

As the strokes of Chinese character generally include five primitive strokes, point, horizontal stroke, vertical stroke, left slant, and right slant (as shown in Figure 3), we use the 36 second Gaussian derivative filters and 8 Gaussian Laplacian filters of Leung-Malik filter bank to strengthen the spatial direction information of image.

The parameters of second derivative of Gaussian filters are redefined as 12 directions ( $0^\circ\text{--}165^\circ$ , an interval of  $15^\circ$ ) and 3 dimensions ( $(\sigma_x, \sigma_y) = \{(\sqrt{2}, 3\sqrt{2}), (2, 6), (2\sqrt{2}, 6\sqrt{2})\}$ ). The scales of 8 Gaussian Laplacian filters are  $\sqrt{2}, 2, 2\sqrt{2}, 4, 3\sqrt{2}, 6, 6\sqrt{2}$ , and 12. After strengthening of image  $\mathbf{I}$  by these filters from Leung-Malik filter bank, it will achieve 44 energy images  $\mathbf{E}_{\sigma,\theta}$ , as shown in

$$\mathbf{E}_{\sigma,\theta} = \mathbf{I} * \mathbf{F}_{\sigma,\theta}, \quad (4)$$

where  $*$  is a convolution symbol and  $\mathbf{F}_{\sigma,\theta}$  is the filter from 36 second derivatives of Gaussian filters and 8 Gaussian Laplacian filters.

The construction of the actual LM filter bank needs to determine the maximum window's size of the filter bank:  $K \times K$ . The value of  $K$  decides the fine level of enhancement. Therefore, this paper adopts an adaptive enhancement mechanism to decide the window size by estimating noise of the original image. The detailed information of a sealing clay image is not complicated, which is only interfered by damage, stains, and so on. Thus, in order to boost the system speed,

this paper adopts the method of fast noise variance estimation [11]. For an image measuring  $W \times H$  as seen in  $\mathbf{I}$ , the formula of variance for noise is presented as

$$\sigma_n = \sqrt{\frac{\pi}{2}} \frac{1}{6(W-2)(H-2)} \sum_{x=1}^W \sum_{y=1}^H |\mathbf{I}(x, y) * \mathbf{N}|, \quad (5)$$

where  $\sigma_n$  is the noise variance;  $*$  is the convolution symbol;  $\mathbf{N}$  is a  $3 \times 3$  template. Consider

$$\mathbf{N} = \begin{array}{c|c|c} 1 & -2 & 1 \\ \hline -2 & 4 & -2 \\ \hline 1 & -2 & 1. \end{array} \quad (6)$$

On the basis of distribution of noise variance, we determine threshold parameters  $\mathbf{T}_1$  and  $\mathbf{T}_2$ . According to (7), we get the value of  $K$ :

$$K = \begin{cases} 5, & \sigma_n > \mathbf{T}_1 \\ 9, & \mathbf{T}_2 < \sigma_n < \mathbf{T}_1 \\ 13, & \sigma_n < \mathbf{T}_2. \end{cases} \quad (7)$$

### 3. The Process and Description of Algorithm

When calculating the Radon-like feature values of an image, the function is as follows:

$$\mathbf{T}(\mathbf{I}, \mathbf{L}(t)) = \frac{\int_{t_i}^{t_{i+1}} \mathbf{E}(\mathbf{L}(t)) \partial t}{\|\mathbf{L}(t_{i+1}) - \mathbf{L}(t_i)\|_2}, \quad t \in [t_i, t_{i+1}]. \quad (8)$$

This function calculates the mean values of image  $\mathbf{E}$ 's pixel values locating along line  $\mathbf{L}$  and situating between knots  $t_i$  and  $t_{i+1}$ .  $\mathbf{E}$  is the maximum energy image:

$$\mathbf{E}(x, y) = \max_{\sigma,\theta} \mathbf{E}_{\sigma,\theta}(x, y). \quad (9)$$

Besides, this paper extracts the edge of  $\mathbf{E}$  by using Canny operator to generate knots. 12 lines with different direction ( $0^\circ\text{--}360^\circ$ , with an interval of  $30^\circ$ ) are used to scan images to locate knots and determine Radon-like features. The basic process of algorithm is as follows.

*Step 1.* For an input image  $\mathbf{I}$ , the method of fast noise variance estimation is used to calculate  $\sigma_n$ . Based on the values of  $\sigma_n$  and  $\mathbf{T}_1$  and  $\mathbf{T}_2$ , windows parameter  $K$  is determined.

*Step 2.* Enhancement is achieved by using the 36 second Gaussian derivative filters and 8 Gaussian Laplacian filters in the LM filter bank. 44 energy images  $\mathbf{E}_{\sigma,\theta}$  are obtained and the maximum energy image  $\mathbf{E}$  is got.

*Step 3.* Canny operator is used to extract the edge of  $\mathbf{E}$ , and an edge graph is got.

*Step 4.* Based on the edge graph, 12 lines with different direction are used to scan image  $\mathbf{E}$ . The knots are determined by the intersections of the scan line and the edge graph.

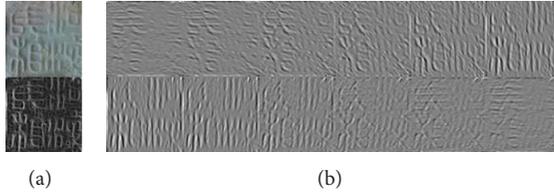


FIGURE 4: Some results in Step 2. (a) Top: sealing clay image; (a) bottom: the maximum energy image  $E$ ; (b) the energy images from 12 directions at scale  $(\sqrt{2}, 3\sqrt{2})$ .

The pixels between two knots in  $E$  are the found Radon-like features. Their values are calculated according to the extraction function which is defined in formula (8). 12 scan lines will make 12 different Radon-like feature pictures.

*Step 5.* Thresholding is conducted on the mean value of the 12 Radon-like feature pictures, and the segmentation result is got.

In Step 2, the adaptive LM filter bank is used to enhance the texture features and it also means to characterize a texture by its responses to a set of filters. In order to reduce the dimensionality of the filter response space, the maximum energy image  $E$  over both scales and orientations is selected from  $E_{\sigma,\theta}$ . Figure 4 shows some results in Step 2. In Figure 4, the top of (a) is a sealing image and its some energy images from 12 directions at scale  $(\sqrt{2}, 3\sqrt{2})$  are listed in (b). The maximum energy image is shown in lower (a). We can see that maximum energy image contains enhanced main parts and local details.

In Step 3, canny operator is used to get edge of  $E$ . Though the operator can be directly applied to input image  $I$ , its result is less complete. It can be seen in Figure 5 that the result of  $E$  has more complete edges.

In Step 4, based on the edge graph and  $E$ , Radon-like features are found. After that, these features' values are calculated according to the extraction function and make 12 different pictures. Figure 6 shows the 12 Radon-like feature pictures of sealing clay image shown in Figure 4. It can be seen that the features of text in 12 directions are selected and enhanced.

#### 4. Experimental Results and Analysis

Based on the above algorithm, this paper selects 30 sealing clay images to conduct segmentation experiment. These images are obtained from three types of sealing clay with different materials and color. Because the algorithm proposed by this paper is to divide the text of sealing clay image, before the experiment, preprocessing of the original sealing clay image is conducted. The central text area is extracted from the sealing clay images and different areas are unified into the same size measuring 128 in length and 128 in width. To demonstrate the advantages of the proposed method we implemented it with Matlab R2010b and compared it with 2D Otsu [12], Genetic Algorithm [13], and FastFCM

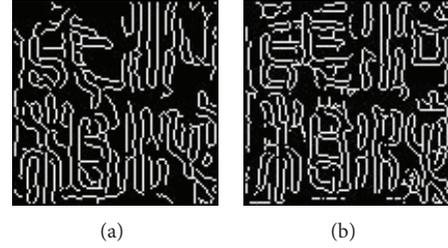


FIGURE 5: Edge extraction results using canny operator. (a) The edge information of input image  $I$ ; (b) the edge information of the maximum energy image  $E$ .



FIGURE 6: 12 Radon-like feature pictures of sealing clay image in Figure 4.

[14]. Figure 7 lists five sealing clay images using different segmentation methods, among which the last line is the result yielded by manual division under the guidance of experts [15].

We can see that 2D Otsu and GA yield similar results which do not completely separate the text information from the noise, with many areas integrating with each other especially in Figures 7(a) and 7(b). In comparison, FastFCM not only extracts the major form of characters but also suppresses parts of the noise, yielding more detailed information comparing with the former two ways, which is shown in Figures 7(a), 7(b), 7(c), and 7(d). However, looking at every single character, the specific strokes and the intactness of structures are not satisfactory and Figures 7(a), 7(b), 7(c), and 7(d) can illustrate this problem. It should be noticed that, comparing the methods of 2D Otsu, GA, and FastFCM, only GA can obtain the approximate form of the Chinese character “司” in Figure 7(b). Therefore, GA has a better recognizability. For Figure 7(e), GA and FastFCM yield similar results which contain much noise. Though 2D Otsu can obtain better result, the form of characters is incomplete. The method proposed by this paper can solve the problems mentioned above, which not only clearly separates the major body of a character but also has a better performance in terms of richness and intactness. What is more, this method can greatly filter the noise so as to yield clearer and more definite results [16–20].

As is shown in Table 1, by comparing the operation time, we found that 2D Otsu method requires the minimum time. The time required by FastFCM method is longer than that of 2D Otsu method, but is shorter than that of the method used in this paper. GA method required the longest operation time. The method we used in this paper was not advantageous in operating the time, which needed twice the time of FastFCM method and three times the time of 2D Otsu method. Time



FIGURE 7: The segmentation results using different methods including 2D Otsu, GA, FastFCM, and the proposed method.

TABLE 1: Operation time of different method.

Method	$a$ (ms)	$b$ (ms)	$c$ (ms)	$d$ (ms)	$e$ (ms)
2D Otsu	86	93	95	86	89
GA	541	527	520	513	518
FastFCM	248	249	243	232	241
The method in this paper (with 12 scanning directions)	476	481	472	468	473

TABLE 2: Average values of ME, TPR, FPR, and EP of 4 methods.

	2D Otsu	GA	FastFCM	The proposed method
Average value of ME	0.2163	0.2545	0.1536	0.0982
Average value of TPR	87.3%	84.7%	91.2%	96.5%
Average value of FPR	10.1%	12.7%	7.3%	4.7%
Average value of EP	12.3%	15.2%	8.6%	4.2%

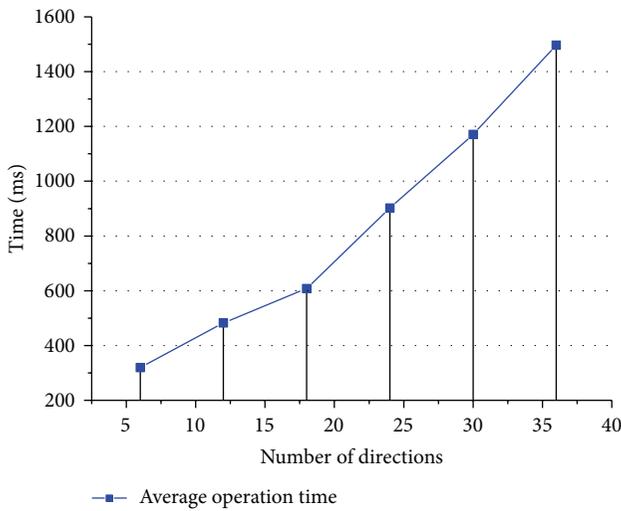


FIGURE 8: The relationship between average operation time and the number of directions in the proposed method.

was mainly spent in scanning and searching the Radon-like features in different directions. The more directions we have scanned, the more detailed Radon-like features we would find and the longer time it would take. Figure 8 shows the relationship between average operation time and the number of directions we scanned. According to several experiments in this paper and with the principle to keep the operation time as short as possible, we found that it could achieve preferable result when the number of directions was 12.

To measure the result of segmentation and further validate effectiveness of the method, the ground-truth image, which was manually segmented under the guidance of experts, was used as a standard reference diagram to compare with the results made by other 4 methods. Misclassification error (ME) was adopted in the measurement of specific differences, which was defined as follows [21]:

$$ME = 1 - \frac{|\mathbf{B}_O \cap \mathbf{B}_T| + |\mathbf{F}_O \cap \mathbf{F}_T|}{|\mathbf{B}_O| + |\mathbf{F}_O|}, \quad (10)$$

where  $\mathbf{F}_O, \mathbf{F}_T$  refer to pixels of the text area in standard reference figure and test pattern, respectively, and  $\mathbf{B}_O, \mathbf{B}_T$  stand for pixels of the background region in standard reference image and test pattern, respectively.  $|\mathbf{F}_O \cap \mathbf{F}_T|$  represents the pixel of statistical texts which are segmented correctly, while  $|\mathbf{B}_O \cap \mathbf{B}_T|$  means the pixel of statistical backgrounds which are segmented correctly. The value of ME ranges from 0 to 1, with 0 standing for completely correct segmentation while 1 means completely incorrect segmentation. Therefore, the smaller the ME value is, the more precise the segmentation is. Moreover, the average values of true positive rate (TPR), false positive rate (FPR), and error probability (EP) are also used to measure the performance, which are calculated the following, respectively:

$$TPR = \left( \frac{TP}{TP + FN} \right) \times 100\%, \quad (11)$$

$$FPR = \left( \frac{TN}{FP + TN} \right) \times 100\%, \quad (12)$$

$$EP = \left( \frac{FP + FN}{TP + FN + FP + TN} \right) \times 100\%, \quad (13)$$

where TP, TN, FP, and FN are true positives, true negatives, false positive, and false negatives, respectively. In this paper, 30 images are tested, and the average values of ME, TPR, FPR, and EP obtained from 4 methods are shown in Table 2.

It is apparent from Table 2 that, in all experiments, the proposed method is superior compared to other methods, with the least ME (0.0982), FPR (4.7%), and EP (4.2%) and the maximum value of TPR (96.5%), while FastFCM method has a slightly lower accuracy on segmentation, with its values of ME (0.1536), TPR (91.2%), FPR (7.3%), and EP (8.6%). By comparison, 2D Otsu method and GA method have the lowest accuracy on segmentation with higher values of ME (0.2163, 0.2545), FPR (10.1%, 12.7%), and EP (12.3%, 15.2%). All the results indicate that the proposed method is the most efficient to segment the text of sealing clay in terms of accuracy and stability.

## 5. Conclusion

Traditional text extraction method of sealing clay needs to contact the sealing clay object directly, which increases the risk of damage. Due to the gradually mature digital technology of cultural relics, this paper proposes a text extraction method based on digital images to facilitate sealing clay research. This method combines Radon-like features and adaptive enhancement LM filter bank and is implemented by two major steps of enhancing the filtering and calculating Radon-like feature images. In experimental testing phase, the result obtained from the method used in this paper is compared with those of 2D Otsu method, GA method, and FastFCM method. According to the result, the method proposed in this paper could not only effectively reduce noise like breakage and muddiness, but also extract text with higher definition, completeness, and accuracy. The disadvantage of this method is its long execution time. The major reason is that the directional scanning process which identifies Radon-like features is relatively time consuming. The more directions it scans, the longer execution time it takes. Therefore, we should promote the scanning strategy to improve the execution speed. Besides, to identify the Radon-like feature, more abundant partial statistical information can be further utilized to extract function so as to acquire more detailed information to enhance the representativeness of Radon-like features. As for applying the method, we can extend the method to solve the problem of segmentation in other types of images, such as blood vessels image segmentation, cells image segmentation, and road image segmentation.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (Grant no. 61202198) and Social Science Foundation of Zhejiang Province (Grant no. I1JCWH13YB). This program is supported by Scientific Research Program Funded by Shaanxi Provincial Education Department (Program no. 2013JK1139) and supported by China Postdoctoral Science Foundation (no. 2013M542370) and supported by the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant no. 20136118120010).

## References

- [1] F. Kurugollu, B. Sankur, and A. E. Harmanci, "Color image segmentation using histogram multithresholding and fusion," *Image and Vision Computing*, vol. 19, no. 13, pp. 915–928, 2001.
- [2] A. El Zaart, D. Ziou, S. Wang, and Q. Jiang, "Segmentation of SAR images," *Pattern Recognition*, vol. 35, no. 3, pp. 713–724, 2002.
- [3] J. W. Long, X. J. Shen, and H. P. Chen, "Adaptive minimum error thresholding algorithm," *Acta Automatica Sinica*, vol. 38, no. 7, pp. 1134–1144, 2012.
- [4] L. Ma, S. Zhao, L. Zhu et al., "A modified fuzzy C-Means segmentation algorithm for illumination pattern-based images under bias field," *Application Research of Computers*, vol. 23, no. 4, pp. 135–136, 2006.
- [5] M.-H. Horng and R.-J. Liou, "Multilevel minimum cross entropy threshold selection based on the firefly algorithm," *Expert Systems with Applications*, vol. 38, no. 12, pp. 14805–14811, 2011.
- [6] L. Bo, L. Rong, F. Jiulun et al., "Fast algorithm for maximum fuzzy entropy thresholding method," *Pattern Recognition and Artificial Intelligence*, vol. 23, no. 6, pp. 867–873, 2010.
- [7] J. Li, B. Dai, K. Xiao, and A. E. Hassanien, "Density based fuzzy thresholding for image segmentation," in *Advanced Machine Learning Technologies and Applications*, vol. 322 of *Communications in Computer and Information Science*, pp. 118–127, Springer, Berlin, Germany, 2012.
- [8] J. Lázaro, J. L. Martín, J. Arias, A. Astarloa, and C. Cuadrado, "Neuro semantic thresholding using OCR software for high precision OCR applications," *Image and Vision Computing*, vol. 28, no. 4, pp. 571–578, 2010.
- [9] R. Kumar, A. Vázquez-Reina, and H. Pfister, "Radon-like features and their application to connectomics," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '10)*, pp. 186–193, 2010.
- [10] T. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," *International Journal of Computer Vision*, vol. 43, no. 1, pp. 29–44, 2001.
- [11] J. Immerkær, "Fast noise variance estimation," *Computer Vision and Image Understanding*, vol. 64, no. 2, pp. 300–302, 1996.
- [12] J. Liu and W. Li, "Automatic thresholding of gray-level pictures via two-dimensional OTSU method," *Acta Automatica Sinica*, vol. 19, no. 1, pp. 101–105, 1993.
- [13] P.-Y. Yin, "A fast scheme for optimal thresholding using genetic algorithms," *Signal Processing*, vol. 72, no. 2, pp. 85–95, 1999.
- [14] Y. X. Qing, H. Z. Hua, and X. Qiang, "Histogram based fuzzy C-mean algorithm for image segmentation," in *Proceedings of the 11th IAPR International Conference on Pattern Recognition. Vol.III. Conference C: Image, Speech and Signal Analysis, Proceedings*, pp. 704–707, The Hague, The Netherlands, August–September 1992.
- [15] W. Wei and Y. Qi, "Information potential fields navigation in wireless Ad-Hoc sensor networks," *Sensors*, vol. 11, no. 5, pp. 4794–4807, 2011.
- [16] W. Wei, X.-L. Yang, P.-Y. Shen, and B. Zhou, "Holes detection in anisotropic sensor networks: topological methods," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 135054, 9 pages, 2012.
- [17] W. Wei, X.-L. Yang, B. Zhou, J. Feng, and P.-Y. Shen, "Combined energy minimization for image reconstruction from few views," *Mathematical Problems in Engineering*, vol. 2012, Article ID 154630, 15 pages, 2012.
- [18] W. Wei, Q. Xu, L. Wang et al., "GI/Geom/1 queue based on communication model for mesh networks," *International Journal of Communication Systems*, vol. 27, no. 11, pp. 3013–3029, 2014.
- [19] W. Wei, H. M. Srivastava, Y. Zhang, L. Wang, P. Shen, and J. Zhang, "A local fractional integral inequality on fractal

space analogous to Anderson's inequality," *Abstract and Applied Analysis*, vol. 2014, Article ID 797561, 7 pages, 2014.

- [20] W. Wei, Y. Qiang, and J. Zhang, "A bijection between lattice-valued filters and lattice-valued congruences in residuated lattices," *Mathematical Problems in Engineering*, vol. 2013, Article ID 908623, 6 pages, 2013.
- [21] W. A. Yasnoff, J. K. Mui, and J. W. Bacus, "Error measures for scene segmentation," *Pattern Recognition*, vol. 9, no. 4, pp. 217–231, 1977.

## Research Article

# An Image Watermarking Scheme Using Arnold Transform and Fuzzy Smooth Support Vector Machine

Lin Sun,<sup>1,2,3</sup> Jiucheng Xu,<sup>1,2,3</sup> Xingxing Zhang,<sup>1</sup> and Yun Tian<sup>4</sup>

<sup>1</sup>College of Computer and Information Engineering, Henan Normal University, Henan 453007, China

<sup>2</sup>Engineering Technology Research Center for Computing Intelligence and Data Mining, Henan 453007, China

<sup>3</sup>Engineering Laboratory of Intellectual Business and Internet of Things Technologies, Henan 453007, China

<sup>4</sup>College of Information Science and Technology, Beijing Normal University, Beijing 100875, China

Correspondence should be addressed to Lin Sun; [linsunok@gmail.com](mailto:linsunok@gmail.com)

Received 10 October 2014; Revised 2 January 2015; Accepted 5 January 2015

Academic Editor: Daniel Zaldivar

Copyright © 2015 Lin Sun et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the development of information security, the traditional encryption algorithm for image has been far from ensuring the security of image in the transmission. This paper presents a new image watermarking scheme based on Arnold Transform (AT) and Fuzzy Smooth Support Vector Machine (FSSVM). First of all, improved AT (IAT) is obtained by adding variables and expanding transformation space, and FSSVM is proposed by introducing fuzzy membership degree. The embedding positions of watermark are obtained from IAT, and the pixel values are embedded in carrier image by quantization embedding rules. Then, the watermark can be embedded in carrier image. In order to realize blind extraction of watermark, FSSVM model is used to find the embedding positions of watermark, and the pixel values are extracted by using quantization extraction rules. Through using improved Arnold inverse transformation for embedding positions, the watermark coordinates can be calculated, and the extraction of watermark is carried out. Compared with other watermarking techniques, the presented scheme can promote the security by adding more secret keys, and the imperceptibility of watermark is improved by introducing quantization rules. The experimental results show that the proposed method outperforms many existing methods against various types of attacks.

## 1. Introduction

Application and popularization of multimedia technologies and computer networks have made duplication and distribution much easier for multimedia contents [1, 2]. The digital media (such as video, image, audio, and text) can be modified easily by attackers who can claim their ownership [3]. Then copyright protection of intellectual properties has become an important and challenging task. One way for copyright protection is digital watermarking, which means embedding certain specific information about the copyright holder into the protected media [4]. In the last decade, digital watermarking methods for images from different purposes are usually categorized into two types: robust watermarking and fragile watermarking. And the watermarking techniques can also be classified into two groups based on the domain in which the watermark is inserted: spatial domain techniques

and frequency domain techniques [2, 5–7]. The methods of spatial domain have advantages of easy implementation and low cost of operation, but they are generally not robust to geometrical attacks and image processing. Nowadays, there are many transform domain watermarking techniques [8, 9], such as discrete cosine transform, singular value decomposition, discrete Fourier transform, and discrete wavelet transform. In practice, the performances of watermarking methods are further improved by combining two or more transforms [5]. The image encryption using gyrator transform based on two-step phase-shifting interferometry and AT has been proposed in [10]. Liu et al. [11] proposed an image encryption algorithm by using AT and discrete cosine transform. Abuturab [12] proposed the color information encryption in the gyrator transform domain not only based on discrete cosine transform and radial Hilbert phase mask but also based on AT. Sui and Gao [10] presented a color image

encryption scheme by using gyrator transform and AT. Chen et al. [13] designed a new image encryption algorithm based on singular value decomposition and AT. Chen et al. [14] offered a watermarking scheme based on Arnold cat map. AT has periodicity and the transform is simple, but the periodicity depends on image size. Moreover, it is the main concern of the watermarking schemes that the embedded watermark should not degrade the quality of carrier image and the inserted watermark must be as much invisible as possible. The distortion of the watermarked image should be negligible without degrading its robustness under attacking conditions. In order to generate the watermarked images of high quality and extract high fidelity watermarks in attacking environments, the overall watermarking procedure depends on a set of configuration parameters which need to be optimized. Most existing image watermarking algorithms take scaling factor as a single value which needs the proper fine tune. However, when each singular value has a different tolerance limit, a single scaling factor is not suitable in case of singular value decomposition. It is known that imperceptibility and robustness are two important issues of image watermarking schemes. Then the tradeoff between imperceptibility and robustness in watermarking problems is viewed as an optimization problem.

Recently, attacks against image watermarking systems have become more sophisticated [15, 16]. A simple solution consists in embedding the watermark in a geometrical invariant subspace, and another strategy for coping with geometric distortions is to insert an additional watermark or template into the carrier image [17, 18]. However, the implementation difficulties hinder the research of image watermarking schemes based on this principle above. Nasir et al. [19] proposed feature-based image watermarking schemes. Wang et al. [20] proposed a feature-based digital watermarking method for halftone images. However, some drawbacks indwelled in current feature-based schemes restrict the performance of watermarking systems. To address the issues, support vector machine (SVM) is introduced to the image watermarking domain. Fu et al. [21] embedded template and watermark into original images; then the output of SVM models was constructed and the watermark was extracted. Tsai and Sun [22] employed the classification technique based on SVMs to extract a watermark in spatial domain. Li et al. [23] used support vector regression for watermark embedding and extracting in spatial domain. Peng et al. [24] proposed an image watermarking method in multiwavelet domain based on SVMs. Tsai et al. [25] presented a robust lossless watermarking technique based on  $\alpha$ -trimmed mean algorithm and SVM. Li et al. [26] introduced a semifragile watermarking scheme based on SVM. Yang et al. [17] proposed a new geometrically invariant image watermarking algorithm based on fuzzy SVM correction. However, by virtue of the good learning ability of SVM and AT, most of the existing watermarking schemes mentioned above still have some shortcomings as follows. (1) The standard SVM is always used as a learning scheme, while the speed and precision of training samples are not very ideal. Then, the distortion of final extracted watermark is more serious. (2) As many AT schemes only have single secret key, this

characteristic has the limitations of being susceptible to one or several joint attacks and breaks, and then the difficulty of watermark extraction will be increased. (3) Because the extracted image feature is not very stable, hyperplane and generalization capability in SVM of the learned models are affected. Moreover, all samples in training data set are treated uniformly in the same class during the learning process of SVM, but this is not always true. (4) Due to the poor feature vectors, these methods are not very robust against some attacks. In watermark detection procedure, the original carrier image is usually needed, so it is unfavorable to practical application. Furthermore, some of them lack blind detection features; then they cannot balance imperceptibility and robustness effectively.

In this paper, embedding and extraction models of watermark based on AT and FSSVM are proposed with corresponding algorithms. The watermark image is first processed by IAT, and the pixel values are embedded by using quantization embedding rules. The watermarked carrier image can be obtained. Then the embedding method is presented by increasing secret keys to enhance the security degree. The quantization embedding rules can improve imperceptibility of watermark well. FSSVM model is constructed by training some embedding positions to find out positions embedded with watermark in the extraction process. The pixel values of watermark are extracted by introducing quantization extraction rules. FSSVM model not only enhances the training speed and precision of image characteristic values, but also realizes the blind extraction of watermark with the help of quantization extraction rules. The watermark coordinates are calculated by improving Arnold inverse transformation for embedding positions. The original watermark image is extracted, and the processing does not need original carrier image. Moreover, the combination of spatial domain and SVM can balance imperceptibility and robustness of watermark well.

The rest of this paper is organized as follows. In Section 2, AT and SVM techniques are briefly described. In Section 3, the proposed techniques are illustrated. In Section 4, the experimental results are presented and the comparative analysis of our scheme with other methods is given. Finally, the conclusion is drawn in Section 5.

## 2. Preliminaries

**2.1. Arnold Transform (AT).** In practical applications, AT not only scrambles the pixel position by encoding the iterative number of the process, but also reduces the key spaces of storage and transmission. Although there are many ways for scrambling, here we will discuss only the AT in [10, 14], which is an iterative process of moving the pixel position. Suppose that the original image is a  $N \times N$  array and the coordinate of the pixel is  $F = \{(x, y) \mid x, y = 0, 1, 2, \dots, N - 1\}$ . The generalized two-dimension AT is denoted by

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab + 1 \end{bmatrix} \begin{bmatrix} x_{n-1} \\ y_{n-1} \end{bmatrix} \bmod N, \quad (1)$$

where  $x_n$  and  $y_n$  are the transformed coordinates corresponding to  $x_{n-1}$  and  $y_{n-1}$  after  $n$  iterations, respectively,  $a$  and  $b$  are

positive integers, and  $N$  is the height or width of the square image processed.

Since the transformation is an iterative process, if the location  $(x, y)$  is transformed several times, it returns to its original position after  $T$  iterations.  $T$  is called the period of the transformation and depends on parameters  $a, b$ , and  $N$ . These parameters ( $a, b$ , and  $N$ ) can be used as secret keys in this paper. The pixels will continue to move until they return back to their original positions. Here, the moving time is  $T$ , and the size of pixel space is  $n = 0, 1, 2, \dots, N - 1$ . Pixels move with periodicity.  $T, a, b$ , and  $N$  (the size of original watermark) are correlated. Whenever the values change, it generates a completely different Arnold cat map. After being multiplied a few times, the correlation among the pixels will be completely chaotic. To get back to the original image, periodicity is required. Suppose that the scrambling has performed  $n$  iterations; then one can get back to the original image by performing  $(T - n)$  iterations.

**2.2. Support Vector Machine (SVM).** SVM has been successfully applied to classification and function estimation problems introduced by Vapnik within the context of statistical learning theory and structural risk minimization [21, 22], and it can be used as regression prediction. Suppose that  $(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)$  are training samples, and samples have nonlinear relationship in many cases. The regression function can be denoted by  $f(x) = w\phi(x) + b$ , where  $x \in R^n$  is an input vector,  $w \in R^n$  is a weight vector,  $\phi$  is a nonlinearity mapping function, and  $b \in R$  is an offset.

In order to obtain the last two parameters  $w$  and  $b$ , the structural risk minimization rule is used and the original problem is transformed as follows:

$$\begin{aligned} \min \quad & \left( \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*) \right), \\ \text{s.t.} \quad & y_i - w_i - b \leq \varepsilon + \xi_i, \\ & w_i + b - y_i \leq \varepsilon + \xi_i^*, \\ & \xi_i, \xi_i^* \geq 0, \end{aligned} \quad (2)$$

where  $\|w\|^2$  describes the complexity of the function  $f$ ,  $C > 0$  is a constant to determine the complex rate of model and moderation of experimental risk,  $\xi_i$  and  $\xi_i^*$  are slack variables,  $y_i$  is a sample,  $w_i$  is a weight vector,  $b \in R$  is an offset, and  $\varepsilon$  is an insensitive loss function.

To solve the above convex optimization problem, the core idea is to transform the optimization question into dual form by using Lagrange multiplier method as follows:

$$\begin{aligned} \min \quad & \left( -\frac{1}{2} \sum_{i,j=1}^l (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) (x_i \bullet x_j) \right. \\ & \left. - \varepsilon \sum_{i=1}^l (\alpha_i - \alpha_i^*) + \sum_{i=1}^l y_i (\alpha_i - \alpha_i^*) \right), \\ \text{s.t.} \quad & \sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0, \quad \alpha_i, \alpha_i^* \in [0, C], \end{aligned} \quad (3)$$

where  $\alpha_i, \alpha_i^*, \alpha_j$ , and  $\alpha_j^*$  are the weight coefficients and  $\bullet$  denotes the inner product operation. In the above formulas, kernel function is introduced to complete the inner product computation for inputting data of high-dimensional feature space while the  $\phi$  is not known. The kernel function must satisfy Mercer theorem.

### 3. Proposed Techniques

**3.1. Fuzzy Smooth Support Vector Machine (FSSVM).** In order to improve the efficiency and precision of prediction, SVM is introduced, and through combining with fuzzy mathematics and transforming the problem into unconstrained optimization problem, one can optimize the object function and transform the risk function into fuzzy dual extreme problems. Then, it can effectively reduce the errors between the predicted pixel values and the actual pixel values in the carrier image. The FSSVM model is used to train the specific pixels in some positions and find the embedding positions of watermark in carrier image, which is constructed in detail as follows.

*Step 1.* Select  $k$  coordinates from  $M \times K$  embedding position coordinates  $(x_i, y_i)$  at random, where  $i = 1, 2, \dots, k$ . The pixel values corresponding to the original carrier image  $I$  are denoted by  $I(x_i, y_i)$ , where  $I = \{I(x_i, y_i) \mid i = 1, 2, \dots, k\}$ .

*Step 2.* Take each position coordinate  $(x_i, y_i)$  as the center of the original carrier image  $I$  for each selected watermark position  $(x_i, y_i)$ , select an image block with the size  $3 \times 3$ , and then receive  $k$  image blocks.

*Step 3.* Calculate the corresponding eigenvalues of the image block in the carrier image for each position coordinate  $(x_i, y_i)$ . Namely, the mean value of pixels except the central point  $(x_i, y_i)$  is calculated as follows:

$$X(x_i, y_i) = \frac{1}{8} \left( \sum_{m=-1}^1 \sum_{l=-1}^1 I(x_i + l, y_i + m) - I(x_i, y_i) \right). \quad (4)$$

And the pixel variance except the central point  $(x_i, y_i)$  is calculated as follows:

$$\begin{aligned} D(x_i, y_i) = \frac{1}{8} \left( \sum_{m=-1}^1 \sum_{l=-1}^1 (I(x_i + l, y_i + m) - X(x_i, y_i))^2 \right. \\ \left. - (I(x_i, y_i) - X(x_i, y_i))^2 \right). \end{aligned} \quad (5)$$

Thus, one can totally get  $k$  groups of eigenvalues  $\{X(x_i, y_i), D(x_i, y_i)\}_{i=1,2,\dots,k}$ .

*Step 4.* Let each feature vector  $\{X(x_i, y_i), D(x_i, y_i)\}$  be a training data set and the corresponding pixel value  $I(x_i, y_i)$  of the original carrier image the target value of the training. It constitutes  $k$  couples of training sample sets, shown as

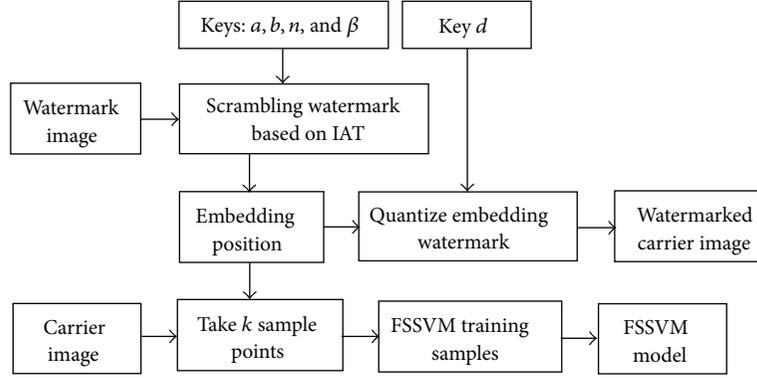


FIGURE 1: Process of watermark embedding.

$\{\{X(x_i, y_i), D(x_i, y_i)\} \rightarrow I(x_i, y_i)\}_{i=1,2,\dots,k}$ , to present a FSSVM training process for the above sample sets as follows.

*Step 4.1.* Introduce fuzzy membership degree  $\mu_i$  to each training sample, and blur the input sample set  $\{\{X(x_i, y_i), D(x_i, y_i)\} \rightarrow I(x_i, y_i)\}$ . Fuzzy membership degree  $\mu_i$  is determined by the relationship between sample input sets  $\{X(x_i, y_i), D(x_i, y_i)\}$  and optimization values  $I(x_i, y_i)$ .

*Step 4.2.* Divide the sets into two categories through fuzzy C-means clustering to all eigenvalue sets  $\{X(x_i, y_i), D(x_i, y_i)\}$ , find out the center of the two classes, and then calculate the distance  $d_i$  from the feature vector  $\{X(x_i, y_i), D(x_i, y_i)\}$  to the center of the corresponding class. Then the membership degree can be expressed as

$$\mu_i = \frac{d_i}{d_{\max}}, \quad (6)$$

where  $d_{\max}$  represents the maximum distance from feature point to class center for the class of feature vector  $\{X(x_i, y_i), D(x_i, y_i)\}$ .

*Step 4.3.* Introduce the kernel function  $K(x, y)$  to map the sample points into high-dimensional feature space.

*Step 4.4.* Carry out linear regression in high-dimensional feature space, and obtain the nonlinear regression effect in the original space. Then its regressive function  $f$  can be expressed as

$$f(x, x_i) = \sum_{i=1}^k (\alpha_i - \widehat{\alpha}_i) K(x, x_i) + \beta. \quad (7)$$

Here, the kernel function mainly uses Gaussian radial basis function as follows:

$$K(x, x_i) = \exp\left(-\frac{\|x - x_i\|^2}{\sigma^2}\right), \quad (8)$$

$$K(x, x_i) = \Phi(x) \bullet \Phi(x_i),$$

where  $x, x_i \in R_n$  represent the input vector  $\{X(x_i, y_i), D(x_i, y_i)\}$ ,  $\alpha_i$  and  $\widehat{\alpha}_i$  are the weight coefficients after training,

$\beta \in R$  is the deviation,  $\Phi$  is a nonlinearity function, and  $\bullet$  denotes the inner product operation.

*Step 5.* Use the principle of structural risk minimization to determine the parameters  $(\alpha_i, \widehat{\alpha}_i, \text{ and } \beta)$ . Namely, transform the original regression equation into solving unconstrained programming problem. Then, the objective function is denoted by

$$\begin{aligned} \min \frac{1}{2} \left( \sum_{i=1}^k \alpha_i^2 + \beta^2 \right) \\ + \frac{C\mu_i}{2} \sum_{i=1}^k P \left( \left( 1 - I_i \left( \sum_{i=1}^k I_i \alpha_i \Phi + \beta \right) \right), \lambda \right), \end{aligned} \quad (9)$$

where  $P(\Delta, \lambda) = \Delta + (1/\lambda) \ln(1 + e^{-\lambda\Delta})$ ,  $I_i = I(x_i, y_i)$  simply,  $\lambda$  is an adjustable parameter, and  $C > 0$  is a constant which is used to determine the complexity of the model and the folded moderation of empirical risk.

*3.2. Watermark Embedding Algorithm.* The watermark embedding procedure, participating in the optimization flow chart of Figure 1, consists of the processing modules depicted as follows.

*Step 1.* Input a carrier image  $I$  with size  $N \times N$  and a binary symbol image  $W$  with size  $M \times K$ . As the original carrier image and the watermark image, they are, respectively, described as  $I = \{I(i, j) \mid 1 \leq i \leq N, 1 \leq j \leq N\}$ ,  $W = \{w(i, j) \mid 1 \leq i \leq M, 1 \leq j \leq K\}$ , where  $M$  and  $K$  are the height and width of binary symbol image, respectively,  $I(i, j)$  is the pixel value of the position  $(i, j)$  in carrier image, and  $w(i, j)$  is the pixel value of the position  $(i, j)$  in watermark image.

*Step 2.* Expand the pixel matrix of binary image into square matrix for making  $M = K$ , and satisfy the conditions of AT. That is, if  $M < K$ , then  $w(i, j) = 1$ , where  $M < i \leq K$ , and if  $M > K$ , then  $w(i, j) = 1$ , where  $K < j \leq M$ . Through adding 1 to pixel matrix of binary symbol image  $W$ , it is expanded into square matrix; that is,  $M = K$ . AT algorithm can be improved by adding the number of variables and expanding transformation space. Then, the coordinate

$(x_0, y_0)$  of the watermark is taken as initial value to do  $n$  times Arnold iteration transformation in binary symbol image. On the basis of such an above consideration, IAT can be denoted by

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \left( \left( \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \right) \bmod M \right) \beta, \quad (10)$$

and it follows that

$$\begin{aligned} x_n &= ((x_0 + ay_0) \bmod M) \beta, \\ y_n &= ((bx_0 + (ab+1)y_0) \bmod M) \beta, \end{aligned} \quad (11)$$

where  $1 \leq x_0 \leq M$ ,  $1 \leq y_0 \leq K$ ,  $1 \leq x_n \leq N$ ,  $1 \leq y_n \leq N$ ,  $\beta = \text{floor}(N/M)$ , floor is down to an integral function,  $M$  is the current order of watermark image matrix,  $N$  is the order of original carrier image,  $a$ ,  $b$ , and  $n$  are all positive integers, and  $0 < n < M^2/2$ . Suppose that (11) are taken  $n$  times iteration; then the corresponding watermark embedding position  $(x_n, y_n)$  is got. So  $M \times K$  position coordinates  $(x_i, y_i)$  can be obtained easily, where  $i = 1, 2, \dots, M \times K$ .

In the above conversion formulas, independent parameters  $a$ ,  $b$ , and  $n$  are defined by the copyright owners themselves, and the three secret key parameters can be used to recover the watermark.  $n$  is the times of iteration, which generally ranges from 1 to 20. Here, if  $n$  is too large, it will affect the running speed of the program and increase the time-space complexity of IAT.  $\beta$  is the enlargement factor to the value of  $(x_0, y_0)$  after  $n$  times iteration transformation. When the watermark is embedded, one needs to make  $(x_n, y_n)$  adapt to the size of the whole carrier image instead of focusing on a certain square. Namely, all of the embedding positions are dispersed into the whole original carrier image.

*Step 3.* Let  $M \times K$  position coordinates be the embedding positions of watermark in the carrier image. That is, embed the pixel of the watermark position  $(x_0, y_0)$  into the position  $(x_n, y_n)$  of carrier image after getting  $M \times K$  embedding positions  $(x_i, y_i)$ , where  $i = 1, 2, \dots, M \times K$ . Then select  $k$  position coordinates  $(x_i, y_i)$  at random from embedding positions to do FSSVM training, where  $i = 1, 2, \dots, k$ .

*Step 4.* Determine the pixel value  $I'(x_n, y_n)$  of the embedded position in the original carrier image by using quantitative embedding rules.  $d = 10$  is used as the quantifying step size of the pixel value  $I(x_n, y_n)$ , and  $k = \text{round}(I(x_n, y_n)/d)$ , where round is rounded to the nearest integral function. Then, to determine the pixel value  $I'(x_n, y_n)$ , the several cases are given as follows.

- (1) If the watermark pixel value  $w(x_0, y_0) = 1$  and  $k = 2m + 1$  hold, where  $m \in N$ ,  $k$  is an odd number, then one has that

$$I'(x_n, y_n) = \left( \text{round} \frac{I(x_n, y_n)}{d} \right) d + 5. \quad (12)$$

- (2) If the watermark pixel value  $w(x_0, y_0) = 0$  and  $k = 2m + 1$  hold, where  $m \in N$ ,  $k$  is an odd number, then one has that

$$I'(x_n, y_n) = \left( \text{round} \frac{I(x_n, y_n)}{d} \right) d + 5. \quad (13)$$

- (3) If the watermark pixel value  $w(x_0, y_0) = 1$  and  $k = 2m$  hold, where  $m \in N$ ,  $k$  is an even number, then one has that

$$I'(x_n, y_n) = \left| \left( \text{round} \frac{I(x_n, y_n)}{d} \right) d - 5 \right|. \quad (14)$$

- (4) If the watermark pixel value  $w(x_0, y_0) = 0$  and  $k = 2m$  hold, where  $m \in N$ ,  $k$  is an even number, then one has that

$$I'(x_n, y_n) = \left| \left( \text{round} \frac{I(x_n, y_n)}{d} \right) d - 5 \right|. \quad (15)$$

Here, compared with the traditional embedding rules, the magnitude of pixel value in the original carrier image, which is changed by quantitative rule, is relatively small, and the biggest changed value is 10. Thus it can realize the imperceptibility after embedding the watermark. According to the above four rules, the pixel values of  $M \times K$  embedding coordinate positions  $(x_n, y_n)$  of watermark can be calculated in the watermarked carrier image easily.

*Step 5.* Replace the original pixel values in the carrier image with those of the calculated  $M \times K$  embedding coordinate positions  $(x_n, y_n)$ ; namely,  $I(x_n, y_n) \rightarrow I'(x_n, y_n)|_{i=1,2,\dots,M \times K}$ . Then, the watermarked carrier image can be obtained, whose size is still  $N \times N$ , and only the pixel values of a minority of points change slightly.

**3.3. Watermark Extraction Algorithm.** The watermark extraction procedures are illustrated with the help of block diagrams in Figure 2. The watermark extraction process can be summarized as follows.

*Step 1.* Denote the watermarked carrier image by  $I'$ , whose size is  $N \times N$ .  $I'(i, j)$  represents the pixel value of the position  $(i, j)$  in watermarked carrier image, where  $I' = \{I'(i, j) \mid 1 \leq i \leq N, 1 \leq j \leq N\}$ .

*Step 2.* Test  $I'$  using FSSVM model obtained from the process of watermark embedding as follows. Firstly, calculate the eigenvalues of each pixel in  $(x_i, y_i)$  of  $I'$ , where the mean pixel value of eight points except the central point  $(x_i, y_i)$  is calculated by

$$X(x_i, y_i) = \frac{1}{8} \left( \sum_{m=-1}^1 \sum_{l=-1}^1 I'(x_i + l, y_i + m) - I'(x_i, y_i) \right). \quad (16)$$

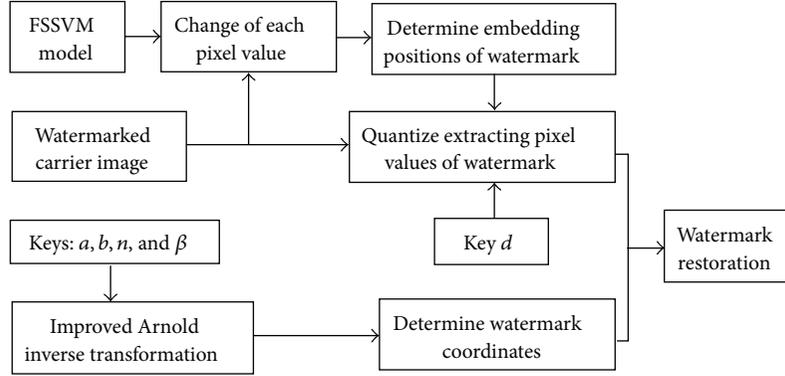


FIGURE 2: Process of watermark extraction.

And the pixel variance of eight points except the central point  $(x_i, y_i)$  is calculated by

$$D(x_i, y_i) = \frac{1}{8} \left( \sum_{m=-1}^1 \sum_{l=-1}^1 (I'(x_i + l, y_i + m) - X(x_i, y_i))^2 - (I'(x_i, y_i) - X(x_i, y_i))^2 \right). \quad (17)$$

Then,  $N \times N$  groups of eigenvalues  $\{X(x_i, y_i), D(x_i, y_i)\}_{i=1,2,\dots,N \times N}$  can be obtained totally. Let the  $N \times N$  groups of eigenvalues  $\{X(x_i, y_i), D(x_i, y_i)\}_{i=1,2,\dots,N \times N}$  be the input data sets  $x$  and  $y$  of FSSVM model test. Namely,  $X(x_i, y_i)$  and  $D(x_i, y_i)$  substitute  $x$  and  $y$  into the model, respectively, and the result is calculated by using the following regressive function:

$$f(x, y) = \sum_{i=1}^k (\alpha_i - \hat{\alpha}_i) K(x, y) + \beta. \quad (18)$$

In (18),  $\alpha_i$ ,  $\hat{\alpha}_i$ , and  $\beta$  have been derived from the training in the process of watermark embedding. So the predicted pixel values  $I'_0(i, j)$  of  $N \times N$  positions in the watermarked carrier image can be got easily.

The eigenvalue selection refers to one-order moment and second-order moment of the probability statistics. In fact, one can also increase the type of eigenvalue, such as the third-order moment, which is beneficial for improving the prediction accuracy. Here, the  $n$ -order moment  $A$  can be briefly expressed as

$$A = E \left( \sum_{i,j} (I'(i, j) - \bar{X})^n \right), \quad (19)$$

where  $\bar{X}$  represents the average values of pixel points except the central point in selected image blocks and  $E$  is to solve the average.

*Step 3.* Compare the pixel values  $I'(i, j)$  of  $I'$  with the predictive pixel values  $I'_0(i, j)$  of the  $N \times N$  positions, and calculate their difference matrix  $D$ , which is denoted by

$$D = |I'(i, j) - I'_0(i, j)|, \quad (20)$$

where  $1 \leq i \leq N$ ,  $1 \leq j \leq N$ . Then sort each element and take out the top  $M \times K$  pixel values corresponding to  $D$ , which are progressively becoming smaller. Their corresponding position coordinates  $(i, j)$  in carrier image, which are embedded with watermark, are obtained. Process the  $M \times K$  pixel coordinates  $(i, j)$  by improving Arnold inverse transformation. Namely, take the pixel coordinate  $(i, j)$  as initial value to do  $n$  times iteration. Thus, the corresponding position coordinate  $(x_0, y_0)$  for  $(i, j)$  in the watermark image can be got from the following formula:

$$\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \left( \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix}^{-1} \begin{bmatrix} i \\ j \\ \beta \end{bmatrix} \right) \bmod M, \quad (21)$$

and it follows that

$$\begin{aligned} x_0 &= \left( (ab+1) \frac{i}{\beta} - b \frac{j}{\beta} \right) \bmod M, \\ y_0 &= \left( -a \frac{i}{\beta} + \frac{j}{\beta} \right) \bmod M, \end{aligned} \quad (22)$$

where  $1 \leq x_0 \leq M$ ,  $1 \leq y_0 \leq K$ ,  $M = K$ ,  $\beta = \text{floor}(N/M)$ ,  $-1$  describes the inverse matrix,  $M$  is the order of watermark matrix,  $N$  is the order of carrier image,  $a$ ,  $b$ , and  $n$  are all positive integers, and  $0 < n < M^2/2$ . Suppose that, in every iteration,  $i/\beta$  is used by the assignment of  $x_0$ ,  $j/\beta$  is used by the assignment of  $y_0$ , and (22) are taken  $n$  times iteration, and then the position coordinates  $(x_0, y_0)$  of watermark corresponding to  $(i, j)$  can be obtained. Finally,  $M \times K$  position coordinates  $(x_i, y_i)$  can be obtained easily, where  $i = 1, 2, \dots, M \times K$ , which are the coordinate values of the original watermark.

*Step 4.* Compute the corresponding pixel values in the watermark, when the position coordinates  $(i, j)$  of the watermarked



FIGURE 3: (a) Original carrier image; (b) original watermark image; (c) the watermarked image; (d) the extracted watermark without attack.

carrier image have been determined and their corresponding position coordinates  $(x_0, y_0)$  in the watermark have also been calculated. Namely, according to the change rules of the pixel values in the watermarked carrier image, the pixel recovery rules of the current watermark are deduced in the process of watermark extraction.

Suppose that  $\lambda = \text{floor}(I'(i, j)/d)$ ; then  $d = 10$  is still the size of quantifying step of pixel value, and  $I'(i, j)$  is the pixel value at the position  $(i, j)$  of the watermarked carrier image. If  $\lambda = 2m + 1$ , where  $m \in N$ , then the  $\lambda$  is an odd number, and  $w(i, j) = 1$ . It follows that the pixel value of watermark is 1. If  $\lambda = 2m$ , where  $m \in N$ , then the  $\lambda$  is an even number, and  $w(i, j) = 0$ . It follows that the pixel value of watermark is 0. However, the current position coordinate  $(i, j)$  is not the position coordinate corresponding to the watermark. The recovery rules of the pixel values just embody the advantages of binary watermark image, because the pixel value is either 0 or 1. So it does not depend on the pixel characteristics, and the pixel values of watermark can be judged only by the quantitative extraction rules of the pixel value in the watermarked carrier image.

*Step 5.* Restore the pixel values of original watermark image according to the values of  $w(i, j)$ , which are calculated at the position of  $M \times K$  pixel coordinates and selected from  $I'$ . Then, each position coordinate  $(i, j)$  is replaced by  $(x_0, y_0)$  through improved Arnold inverse transformation; that is,  $w(i, j) \rightarrow w(x_0, y_0)$ . The coordinate values of watermark and the corresponding pixel values have been determined. According to the above two results, the original watermark image  $W$  can be recombined.

#### 4. Experimental Works

For the experimental purposes, specific software has been developed to implement the proposed methods using MATLAB. Figure 3(a) shows an original carrier image with size  $300 \times 300$ , which is Lena gray image of unit 8. Figure 3(b) depicts an original watermark image with size  $150 \times 150$ , which is the school badge image of Henan Normal University with binary sequence. When the watermark is embedded, the positions embedded into the carrier image are determined by IAT, and the secret key parameters are set as  $a = 1, b = 2,$

$n = 1$ , and  $\beta = 2$ . Then, the pixel values of the embedding positions are determined. When the watermark is extracted, the embedded positions should be found out by using FSSVM model, and the pixel values are restored. Thus, the embedding position is transformed inversely back to the original watermark by using improved Arnold inverse transformation. The above two processes can complement each other. Figure 3(c) is the Lena digital image embedded with actual watermark image by using the watermark embedding algorithm in this paper.

The quality and perceptibility of digital image embedded with actual watermark are judged by using the value of peak signal-to-noise ratio (PSNR), which presents the damaged degree of embedded watermark information to carrier quality. The bigger the PSNR is, the smaller the damaged degree is. Then, PSNR is denoted by

$$\text{PSNR} = 10 \log_{10} \frac{mn \max_{m,n} \left( (I_{i,j})^2 \right)}{\sum_{i=1}^m \sum_{j=1}^n (I_{i,j} - I'_{i,j})^2}, \quad (23)$$

where  $m = 300$ ,  $n = 300$ , and  $I_{i,j}$  and  $I'_{i,j}$  describe the pixel values of the original carrier image and the watermarked carrier image, respectively.

The bit error rate (BER) between extracted watermark and original watermark is employed to evaluate the extracted watermark image objectively. The closer the distance between BER and 0 is, the higher the robustness of the watermark system is, and the stronger the antiattack capability is. Then, BER is denoted by

$$\text{BER} = \frac{\sum_{i=1}^M \sum_{j=1}^K w(i,j) \oplus w'(i,j)}{MK}, \quad (24)$$

where  $M = 150$ ,  $K = 150$ ,  $w(i,j)$  and  $w'(i,j)$  describe the pixel values of the original watermark and the extracted watermark in corresponding positions, respectively, and  $\oplus$  represents XOR operation by bit.

The objective evaluation of the test result of watermark can also use normalized correlation coefficient (NCC) to evaluate the similarity degree, which describes the change before and after embedding watermark in the carrier image. The greater the similarity degree is, the higher the robustness of watermark is. Then, NCC is denoted by

$$\text{NCC} = \frac{\sum_{m,n} I_{m,n} I'_{m,n}}{\sum_{m,n} (I_{m,n})^2}. \quad (25)$$

It follows from Figure 3(c) that PSNR is very high and gets 36.3697 dB, and the quality of the watermarked image has no change, which is consistent with Figure 3(a). Hence, it fully satisfies the requirement of imperceptibility of watermark. In what follows, the watermark shown in Figure 3(d) is extracted from Figure 3(c) by using the proposed extraction algorithm. When the watermarked image has not got any attack, the actual watermark can almost be extracted losslessly, BER = 0.0059 is approximately equal to 0, and NCC = 0.9785 is very close to 1. Therefore, the extracted watermark is almost the same as the original watermark.

TABLE 1: Experimental results of watermark imperceptibility and robustness.

Attacks	PSNR	BER
Image brightening (+75)	21.6031	0.0124
Image darkening (-50)	12.5791	0.0074
Image histogram equalization	15.9505	0.0088
Gaussian noise ( $\mu = 0$ and $\sigma = 0.02$ )	23.9825	0.0730
Median filter ( $9 \times 9$ )	31.5116	0.0270
JPEG compression (10%)	22.7442	0.0737
Geometric cutting (left $100 \times 300$ )	12.5184	0.0126
Geometric rotation $10^\circ$	22.0357	0.0142

In order to further investigate the robustness of our proposed scheme, the watermarked image shown in Figure 3(c) is distorted by taking various attacks, respectively, like (1) image brightening, (2) image darkening, (3) histogram equalization, (4) superposition of Gaussian noise, (5) median filter, (6) JPEG compression, (7) geometric cutting, and (8) geometric rotation. Then, Table 1 shows the experimental data under the above eight types of attacks, and Figures 4(a)–4(h) display the watermarks extracted from the watermarked images which are manipulated by the above attack operations, respectively. From Table 1, the results show that the actual embedded watermarks still have ideal capability of antiattacks, and the BER values are equal to 0 approximately. It follows from Figure 4 that these embedded watermarks can be extracted well and are close to the result of no attacks. Thus, the proposed method in this paper has better robustness against various attacks.

The following part of our experiments continues testing the proposed algorithms in a new watermark image with size  $35 \times 35$ , shown in Figure 5(a), while the carrier image is still Figure 3(a). By using the embedding algorithm in this paper, the quality of the watermarked image has not changed too much, and PSNR is very high, up to 50.0612 dB, which is consistent with Figure 3(a). Then, it fully satisfies the requirements of the watermark invisibility. Figure 5(b) is the watermark extracted by using the proposed extraction algorithm. It shows that when the watermarked image has not suffered any attack processing, the embedded watermark can be extracted losslessly, BER = 0.0049 is approximate to 0, and NCC = 0.9956 is very close to 1. So the extracted watermark is almost the same as Figure 5(a). What is more, the watermarked image further suffers the above eight types of attacks, and then the corresponding extracted watermark images are shown in Figures 5(c)–5(j). With respect to PSNR and BER, our proposed scheme (IAT + FSSVM) is compared with the other two methods, including the fact that IAT is replaced by AT to combine AT and FSSVM (AT + FSSVM) method and FSSVM is replaced by Least Squares Support Vector Machine (LSSVM) to combine IAT and LSSVM (IAT + LSSVM) method. The experimental results are illustrated in Table 2.

The third part of our experiments is to test the proposed algorithms in a camera man image with size  $300 \times 300$ , shown in Figure 6, while the watermark is still Figure 3(b).

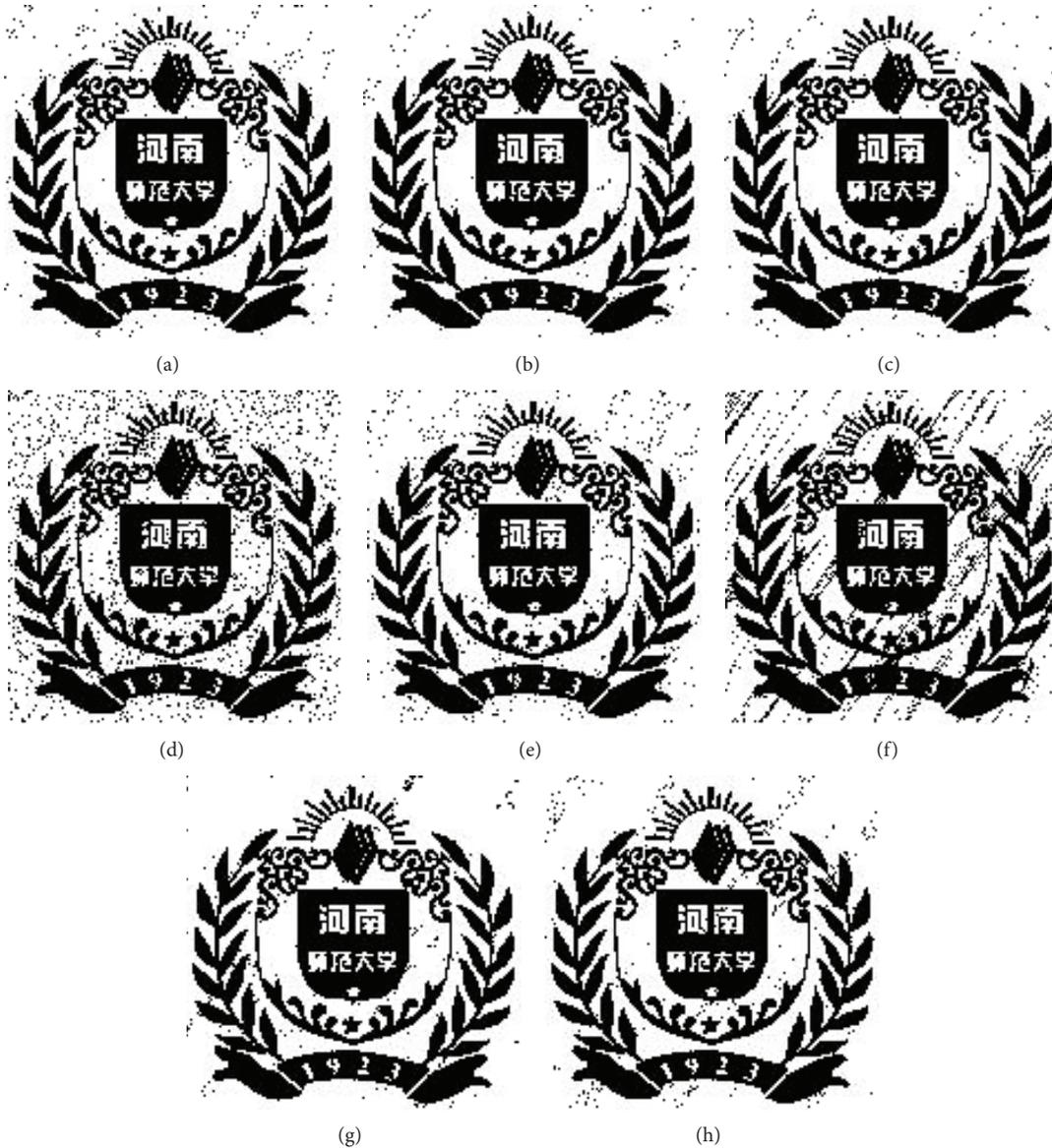


FIGURE 4: (a) The extracted watermark after brightening (+75); (b) the extracted watermark after darkening (-50); (c) the extracted watermark after equalization; (d) the extracted watermark after adding Gaussian noise; (e) the extracted watermark after median filter; (f) the extracted watermark after compression; (g) the extracted watermark after cutting; (h) the extracted watermark after rotation.

The camera man image is embedded with Figure 3(b) by using the proposed embedding algorithm, and then PSNR is very high, up to 73.9578 dB. Figure 7(a) is the watermark extracted from the watermarked camera man image by using the proposed extraction algorithm. When the watermarked camera man image has not suffered any attack processing, the embedded actual watermark can be almost extracted losslessly, BER = 0.0035 is approximate to 0, and NCC = 1. Hence, Figure 7(a) is almost the same as Figure 3(b). In what follows, the watermarked camera man image further suffers the above eight attacks, and then the corresponding extracted watermark images are shown in Figures 7(b)–7(i). Similarly, with respect to PSNR and BER, the experimental results of

three methods (AT + FSSVM, IAT + LSSVM, and IAT + FSSVM) are shown in Table 3.

From Tables 2 and 3, it can be concluded that the combination of IAT and FSSVM proposed in this paper makes the BER and PSNR very ideal. On the one hand, though the PSNR values between AT + FSSVM and IAT + FSSVM are approximate, the BER values in our scheme are obviously smaller; then it is obvious that IAT has more advantages in the watermark extraction. On the other hand, though the PSNR values between IAT + LSSVM and IAT + FSSVM are identical, half of the BER values in our scheme are slightly smaller and the others are approximate, so it can be concluded that the FSSVM model is more powerful in the watermark

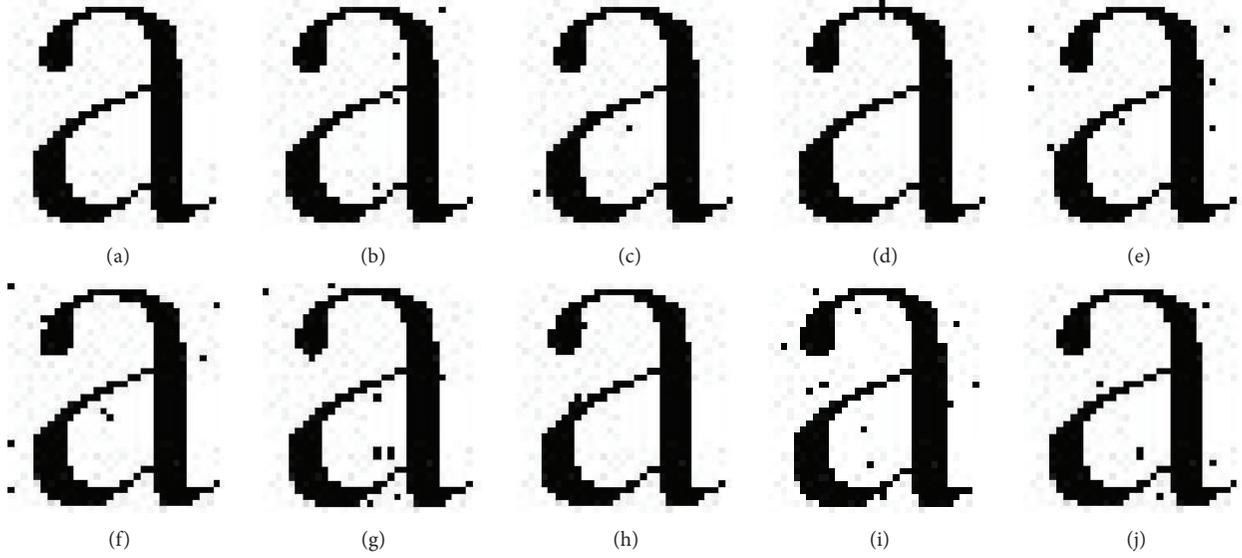


FIGURE 5: (a) The original changed watermark; (b) the extracted watermark without attack; (c) the extracted watermark after brightening (+75); (d) the extracted watermark after darkening (-50); (e) the extracted watermark after equalization; (f) the extracted watermark after adding Gaussian noise; (g) the extracted watermark after median filter; (h) the extracted watermark after compression; (i) the extracted watermark after cutting; (j) the extracted watermark after rotation.

TABLE 2: Experimental results of watermark imperceptibility and robustness after changing watermark.

Attacks	AT + FSSVM		IAT + LSSVM		IAT + FSSVM	
	PSNR	BER	PSNR	BER	PSNR	BER
Image brightening (+75)	15.6836	0.1706	21.5951	0.0024	21.5951	0.0041
Image darkening (-50)	13.8697	0.1739	12.9168	0.0016	12.9168	0.0033
Image histogram equalization	15.9308	0.1249	15.9258	0.0016	15.9258	0.0065
Gaussian noise ( $\mu = 0$ and $\sigma = 0.02$ )	24.0350	0.4710	23.9928	0.0155	23.9928	0.0122
Median filter ( $[9 \times 9]$ )	31.4799	0.4531	31.4850	0.0155	31.4850	0.0123
JPEG compression (10%)	22.7381	0.4702	22.7412	0.0073	22.7412	0.0059
Geometric cutting (left $100 \times 300$ )	31.5740	0.3045	11.6074	0.0049	11.6074	0.0080
Geometric rotation $10^\circ$	22.4917	0.3086	18.8479	0.0106	18.8479	0.0110



FIGURE 6: The original changed carrier image.

extraction. Therefore, the performances of IAT and FSSVM are better than those of AT and LSSVM.

The fourth part of our experiments is to give more details and justifications of our scheme by taking six carrier images against thirteen kinds of attacks. The watermark is still Figure 3(b), and the carrier images include Clock, Aerial, Girl, Truck, Airport, and Man standard images, respectively, which are downloaded from USC-SIPI image library. The thirteen kinds of attack indexes present free, JPEG, low-pass filtering, median filtering, average filtering, salt peppers noise, Gaussian noise, cropping 25%, rotation  $15^\circ$ , sharpening, brightening, darkening, and histogram equalization, respectively. The BER values between original and extracted watermark for the six carrier images against thirteen attacks are shown in Figure 8, which shows that the Airport and Man with size  $1024 \times 1024$  have the least BER, the Clock and Aerial with size  $256 \times 256$  have the most BER, and the sizes of Girl and Truck are

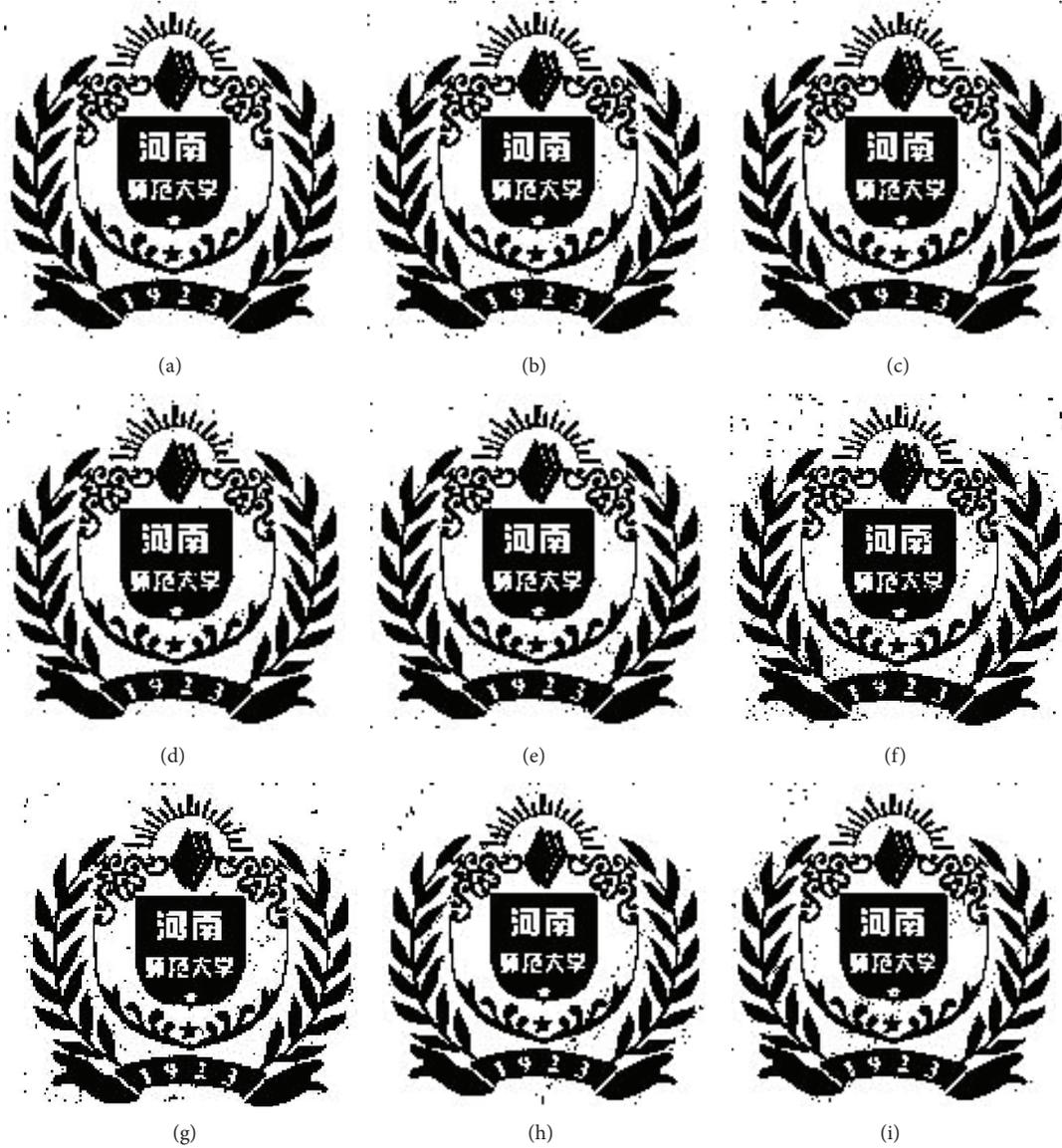


FIGURE 7: (a) The extracted watermark without attack; (b) the extracted watermark after brightening (+75); (c) the extracted watermark after darkening (-50); (d) the extracted watermark after equalization; (e) the extracted watermark after adding Gaussian noise; (f) the extracted watermark after median filter; (g) the extracted watermark after compression; (h) the extracted watermark after cutting; (i) the extracted watermark after rotation.

TABLE 3: Experimental results of watermark imperceptibility and robustness after changing carrier image.

Attacks	AT + FSSVM		IAT + LSSVM		IAT + FSSVM	
	PSNR	BER	PSNR	BER	PSNR	BER
Image brightening (+75)	14.8985	0.0347	15.0029	0.0095	15.0029	0.0091
Image darkening (-50)	16.5895	0.0158	16.4844	0.0039	16.4844	0.0084
Image histogram equalization	21.9937	0.0439	21.9832	0.0101	21.9832	0.0067
Gaussian noise ( $\mu = 0$ and $\sigma = 0.02$ )	38.8298	0.0431	37.8283	0.0054	37.8283	0.0076
Median filter ( $9 \times 9$ )	33.0709	0.1410	34.5537	0.0381	34.5537	0.0277
JPEG compression (10%)	36.1963	0.0379	34.7436	0.0131	34.7436	0.0135
Geometric cutting (left $100 \times 300$ )	23.6165	0.0193	12.8017	0.0049	12.8017	0.0106
Geometric rotation $10^\circ$	18.8558	0.0530	22.5508	0.0135	22.5508	0.0073

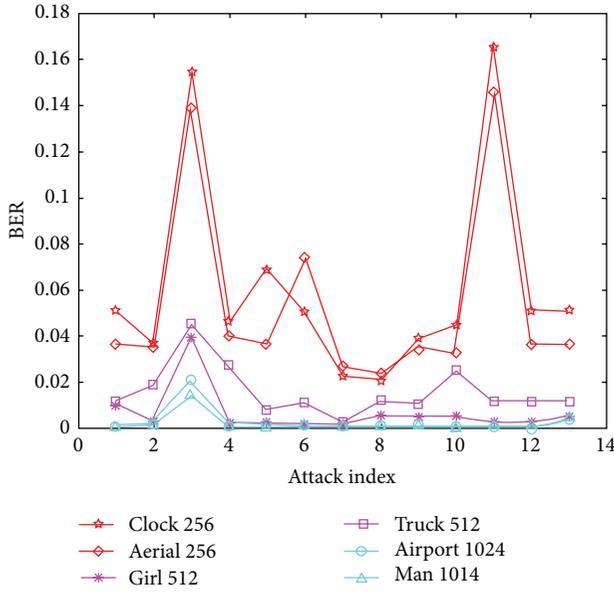


FIGURE 8: The BER for six carrier images against thirteen attacks.

$512 \times 512$ . That is, the larger the carrier is, the lesser the BER is. The reason is the parameter  $\beta$  of IAT in our scheme, which makes the watermark pixels evenly dispersed in the whole carrier and minimizes the damage of being attacked, while the larger the carrier is, the larger the  $\beta$  is, the more the pixels disperse, the less the damage is, and as a result, the lesser the BER is. So the proposed embedding and extraction methods can be considered reasonable and feasible.

The last part of our experiments is to compare our proposed scheme with the other state-of-the-art watermarking methods based on SVM, which are Peng's method [24], Tsai's method [22], and Li's method [23], respectively. The watermark is still Figure 3(b), and the carrier image is Boat with size  $512 \times 512$ , downloaded from USC-SIPI image library. The top-ten kinds of the above thirteen kinds of attacks are given to the watermarked Boat image. Then the BER values of our scheme compared with the other three methods against ten attacks are shown in Figure 9. It can be concluded from Figure 9 that our scheme outperforms the other three methods as a whole, though the BER values of our scheme are a little bigger than those of others against the low-pass filtering and Gaussian noise.

From the above experimental results, the comparative analysis of our scheme with other schemes and the advantages of our algorithms are further summarized as follows.

(1) The embedding method presented in this paper is realized by using IAT and quantization rules, and more parameters ( $a$ ,  $b$ ,  $n$ ,  $\beta$ , and  $d$ ) are used, where  $a$ ,  $b$ ,  $n$ , and  $\beta$  are introduced in IAT and  $d$  appears in quantization rules. These parameters are used as secret keys to extract watermark in the process of watermark extraction, which can be obtained from the embedding process. Because the number of secret keys increases to five and every key has many possible values, our extraction scheme is very difficult to break; then the security of the proposed embedding and extraction algorithms is

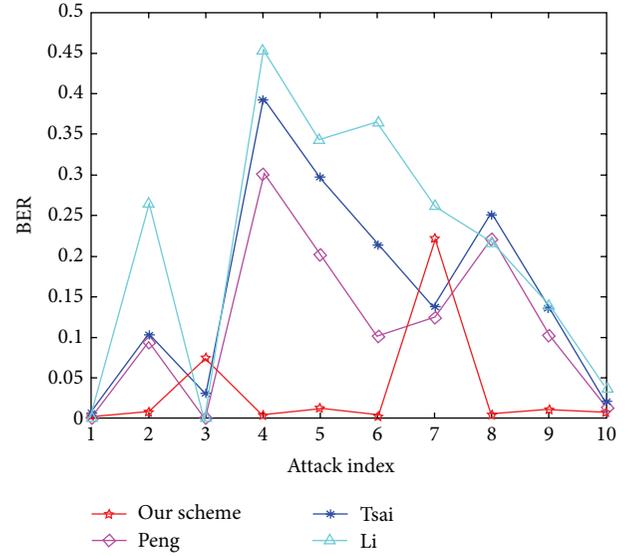


FIGURE 9: The BER of four watermarking schemes for Boat against ten attacks.

improved greatly. Besides, the watermark can be extracted without the original carrier by using our scheme; then the blind extraction is realized, which expands the watermarking applications and improves the security degree. In addition, the change range of pixel values in the carrier image with the quantitative embedding rules is very small. Therefore, compared with the traditional embedding techniques of watermark, the proposed scheme not only realizes the complete imperceptibility of the watermark, but also causes minor damage to the original carrier image. Thus, the problem of image degradation does not exist, so it maintains the integrity of original information of the carrier and the imperceptibility of the carrier image with watermark.

(2) Compared with the existing SVM-based watermarking techniques, the FSSVM model proposed in this paper introduces the concept of fuzzy membership degree and combines the fuzzy mathematics with smooth SVM to simulate the visual features of the human eyes for the watermarked carrier image instead of the standard SVM. The application method not only provides a new solution for digital image watermark technique, but also quickens the training speed and improves the efficiency of image characteristic value. The predicted pixel results of test sample are closer to the actual values than the standard SVM. Thus, the extracted watermark is very close to the original watermark.

(3) Based on IAT, the proposed watermarking scheme has implemented mutual conversion of coordinates from watermarked carrier and original watermark, and then it makes full use of the chaotic scrambling feature of AT to distribute the watermark into the carrier image. In the combination of the new space domain transformation and FSSVM, the watermarked carrier image still can remember the relationship among the local pixels after experiencing various attacks and realize the correct extraction of watermark. Therefore, the embedding and extraction methods have strong robustness to

all kinds of conventional image attacks, and they also balance imperceptibility and robustness of watermark well.

## 5. Conclusion

In this paper, a detailed investigation of image watermarking process by handling it as an optimization procedure based on IAT and FSSVM is presented. The watermarking technique of spatial domain by using IAT is introduced to determine the positions embedded with watermark. Then, one can make full use of the scrambling feature of IAT to evenly distribute the watermark into the whole space of carrier image. Through increasing the secret key parameters, the security degree is improved efficiently. By using quantization and round methods in mathematics to change the pixels of embedding positions, it not only realizes the imperceptibility of watermark, but also deduces the quantitative extraction rules reversely. So the process realizes the blind extraction of watermark without depending on the original carrier image. Furthermore, FSSVM model is constructed to predict the original pixel values of watermarked carrier when the watermark is extracted. Through comparing the predictable values with the actual pixel values of the watermarked carrier image, the positions embedded with watermark can be found out easily. It makes full use of spatial domain features of the image and improves the accuracy and efficiency in predicting outcomes. What is more, the combination of spatial domain and SVM not only efficiently improves the robustness of watermark obviously, but also realizes the blind extraction of watermark. Thus, it achieves the efficacy of advantageous complementarities. Hence, the proposed scheme in this paper is different from the traditional watermark embedding and extraction methods. Theoretical analysis and computer simulations indicate the feasibility of our proposed algorithms. Therefore, our proposed scheme has satisfied the blind extraction, robustness, imperceptibility, and security requirements. In addition, to extend the proposed idea to color video watermarking is another future work.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61370169, 61402153, and 61472042), the Key Project of Science and Technology Department of Henan Province (no. 142102210056), the Science and Technology Research Key Project of Educational Department of Henan Province (nos. 13A520529, 12A520027), and the Education Fund for Youth Key Teachers of Henan Normal University.

## References

- [1] G. A. Papakostas, E. D. Tsougenis, and D. E. Koulouriotis, "Moment-based local image watermarking via genetic optimization," *Applied Mathematics and Computation*, vol. 227, pp. 222–236, 2014.
- [2] B.-Z. Li and Y.-P. Shi, "Image watermarking in the linear canonical transform domain," *Mathematical Problems in Engineering*, vol. 2014, Article ID 645059, 9 pages, 2014.
- [3] N. M. Makbol and B. E. Khoo, "Robust blind image watermarking scheme based on redundant discrete wavelet transform and singular value decomposition," *AEU—International Journal of Electronics and Communications*, vol. 67, no. 2, pp. 102–112, 2013.
- [4] Y. Naderahmadian and S. Hosseini-Khayat, "Fast and robust watermarking in still images based on QR decomposition," *Multimedia Tools and Applications*, vol. 72, no. 3, pp. 2597–2618, 2014.
- [5] M. Ali, C. W. Ahn, and M. Pant, "A robust image watermarking technique using SVD and differential evolution in DCT domain," *Optik—International Journal for Light and Electron Optics*, vol. 125, no. 1, pp. 428–434, 2014.
- [6] A. Mishra, C. Agarwal, A. Sharma, and P. Bedi, "Optimized gray-scale image watermarking using DWT-SVD and firefly algorithm," *Expert Systems with Applications*, vol. 41, no. 17, pp. 7858–7867, 2014.
- [7] E. Nezhadarya and R. K. Ward, "Multiscale derivative transform and its application to image watermarking," *Digital Signal Processing*, vol. 33, pp. 148–155, 2014.
- [8] M. Mundher, D. Muhamad, A. Rehman, T. Saba, and F. Kausar, "Digital watermarking for images security using discrete slantlet transform," *Applied Mathematics & Information Sciences*, vol. 8, no. 6, pp. 2823–2830, 2014.
- [9] A. K. Singh, M. Dave, and A. Mohan, "Hybrid technique for robust and imperceptible image watermarking in DWT-DCT-SVD domain," *National Academy Science Letters*, vol. 37, no. 4, pp. 351–358, 2014.
- [10] L. Sui and B. Gao, "Color image encryption based on gyration transform and Arnold transform," *Optics and Laser Technology*, vol. 48, pp. 530–538, 2013.
- [11] Z. Liu, L. Xu, T. Liu et al., "Color image encryption by using Arnold transform and color-blend operation in discrete cosine transform domains," *Optics Communications*, vol. 284, no. 1, pp. 123–128, 2011.
- [12] M. R. Abaturab, "Securing color information using Arnold transform in gyration transform domain," *Optics and Lasers in Engineering*, vol. 50, no. 5, pp. 772–779, 2012.
- [13] L. F. Chen, D. M. Zhao, and F. Ge, "Image encryption based on singular value decomposition and Arnold transform in fractional domain," *Optics Communications*, vol. 291, pp. 98–103, 2013.
- [14] Y.-L. Chen, H.-T. Yau, and G.-J. Yang, "A maximum entropy-based chaotic time-variant fragile watermarking scheme for image tampering detection," *Entropy*, vol. 15, no. 8, pp. 3170–3185, 2013.
- [15] S. Sadasivam, P. Moulin, and T. P. Coleman, "A message-passing approach to combating desynchronization attacks," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 894–905, 2011.
- [16] X.-Y. Wang, E.-N. Miao, and H.-Y. Yang, "A new SVM-based image watermarking using Gaussian-Hermite moments," *Applied Soft Computing*, vol. 12, no. 2, pp. 887–903, 2012.

- [17] H.-Y. Yang, X.-Y. Wang, and C.-P. Wang, "A robust digital watermarking algorithm in undecimated discrete wavelet transform domain," *Computers and Electrical Engineering*, vol. 39, no. 3, pp. 893–906, 2013.
- [18] L. D. Li, S. S. Li, A. Abraham, and J. S. Pan, "Geometrically invariant image watermarking using Polar Harmonic Transforms," *Information Sciences*, vol. 199, pp. 1–19, 2012.
- [19] I. Nasir, F. Khelifi, J. Jiang, and S. Ipson, "Robust image watermarking via geometrically invariant feature points and image normalisation," *IET Image Processing*, vol. 6, no. 4, pp. 354–363, 2012.
- [20] X. Wang, Z. Xu, and P. Niu, "A feature-based digital watermarking scheme for halftone image," *AEU: International Journal of Electronics and Communications*, vol. 64, no. 10, pp. 924–933, 2010.
- [21] Y. Fu, R. Shen, and H. Lu, "Watermarking scheme based on support vector machine for colour images," *Electronics Letters*, vol. 40, no. 16, pp. 986–987, 2004.
- [22] H.-H. Tsai and D.-W. Sun, "Color image watermark extraction based on support vector machines," *Information Sciences*, vol. 177, no. 2, pp. 550–569, 2007.
- [23] C. H. Li, Z. D. Lu, and K. Zhou, "An image watermarking technique based on support vector regression," in *Proceedings of the International Symposium on Communications and Information Technologies (ISCIT '05)*, pp. 177–180, October 2005.
- [24] H. Peng, J. Wang, and W. X. Wang, "Image watermarking method in multiwavelet domain based on support vector machines," *Journal of Systems and Software*, vol. 83, no. 8, pp. 1470–1477, 2010.
- [25] H.-H. Tsai, H.-C. Tseng, and Y.-S. Lai, "Robust lossless image watermarking based on  $\alpha$ -trimmed mean algorithm and support vector machine," *Journal of Systems and Software*, vol. 83, no. 6, pp. 1015–1028, 2010.
- [26] C.-H. Li, H.-F. Ling, and Z.-D. Lu, "Semi-fragile watermarking based on SVM for image authentication," in *IEEE International Conference on Multimedia and Expo (ICME '07)*, pp. 1255–1258, July 2007.

## Research Article

# An Image Filter Based on Shearlet Transformation and Particle Swarm Optimization Algorithm

**Kai Hu,<sup>1,2</sup> Aiguo Song,<sup>2</sup> Min Xia,<sup>1</sup> Zhiyong Fan,<sup>1</sup> Xiaoying Chen,<sup>3</sup>  
Ruidong Zhang,<sup>1</sup> and Zhichen Zheng<sup>1</sup>**

<sup>1</sup>Nanjing University of Information Science and Technology, Nanjing 210044, China

<sup>2</sup>Southeast University, Nanjing 210000, China

<sup>3</sup>PLA University of Science and Technology, Nanjing 210000, China

Correspondence should be addressed to Aiguo Song; [a.g.song@seu.edu.cn](mailto:a.g.song@seu.edu.cn)

Received 26 August 2014; Revised 18 October 2014; Accepted 19 October 2014

Academic Editor: Erik Cuevas

Copyright © 2015 Kai Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Digital image is always polluted by noise and made data postprocessing difficult. To remove noise and preserve detail of image as much as possible, this paper proposed image filter algorithm which combined the merits of Shearlet transformation and particle swarm optimization (PSO) algorithm. Firstly, we use classical Shearlet transform to decompose noised image into many subwavelets under multiscale and multiorientation. Secondly, we gave weighted factor to those subwavelets obtained. Then, using classical Shearlet inverse transform, we obtained a composite image which is composed of those weighted subwavelets. After that, we designed fast and rough evaluation method to evaluate noise level of the new image; by using this method as fitness, we adopted PSO to find the optimal weighted factor we added; after lots of iterations, by the optimal factors and Shearlet inverse transform, we got the best denoised image. Experimental results have shown that proposed algorithm eliminates noise effectively and yields good peak signal noise ratio (PSNR).

## 1. Introduction

Images are frequently contaminated by noise on the processes of formation, transmission, and reception and make following processes such as segmentation, recognition difficult. This phenomenon makes noise reduction one of the most important problems in image processing.

Basically, there are three common methods to solve this problem, such as transform domain method [1, 2], spatial domain method [3], and partial differential equations (PDE) [4] method; here, our work is one branch of transform domain method. In transform domain method, because of its good performance in both time domain and frequency domain, wavelet transform has become one of the most active research fields in image enhancing. In this field, multiscale geometric analysis theory is the hottest and the most advanced research area. Under this theory, many famous achievements are proposed, for example, ridgelet [5], curvelet [6], contourlet [7], and bandlet [8]. However, with these achievements, there are also many problems to overcome.

Recently, Labate et al. proposed a novel class of multidimensional representation systems, which is called Shearlet. One advantage of this approach is that these systems can be constructed using generalized multiresolution analysis and implemented efficiently using a classical cascade algorithm [1, 9–15]. Due to the fact that Shearlet transforms have multiscale and multidirection feature, simple hard threshold denoising method could not yield bad performance in its practical application of image denoising [16]. Fan and Zhao [17–19] researched the problem of how to calculate the best threshold in this denoising method; they proposed a good idea that used optimal algorithm to get best threshold, but, in their papers, they did not research the fitness function. We hold this opinion that this last step is worth studying and the finding has theoretical value and practical directive meaning.

Among many optimal algorithms [20–33], PSO algorithm has been the most famous global optimization algorithm which is based on swarm intelligence. And it is easy to be realized and already applied to many fields. In this paper,

we adopted PSO algorithm to optimize the problem we mentioned above. In this process, as an important part of PSO algorithm, an objective fitness function should be built to evaluate quality of reconstructed image, so we adopted a fast and rough method to evaluate noise level of image, and this method is just asked to be able to reflect the change trend of noise level in image. Many experiments data have shown that proposed algorithm can achieve better performance than classical Shearlet transform, but like all optimization algorithms, it also has drawback of slow computation speeds.

The remaining paper is organized as follows. Section 2 introduces related theories. Section 3 explains our algorithm, including workflow, Section 4 presents the experiment results of proposed algorithm, and Section 5 concludes this paper.

## 2. Related Theories

*2.1. Shearlet Transform.* Labate et al. [1, 2] proposed Shearlet transform based on wavelet. In dimension  $n = 2$ , affine system

$$\begin{aligned} \Psi_{AB}(\psi) \\ = \{ \psi_{j,l,k}(x) = |\det A|^{j/2} \psi(B^l A^j x - k) : j, l \in \mathbb{Z}, k \in \mathbb{Z}^2 \}; \end{aligned} \quad (1)$$

here,  $\psi \in L^2(\mathbb{R}^2)$  and  $A, B$  are  $2 * 2$  invertible matrices with  $|\det B| = 1$ .

If  $\Psi_{AB}(\psi)$  satisfied Parseval  $L^2(\mathbb{R}^2)$ , then those elements of  $\Psi_{AB}(\psi)$  are called composite wavelets.

Shearlet is a special example of  $L^2(\mathbb{R}^2)$ , for only when

$$A = A_0 = \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix}, \quad B = B_0 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}, \quad (2)$$

here  $A = A_0$  is the anisotropic dilation matrix and  $B = B_0$  is the shear matrix.

For  $\xi = (\xi_1, \xi_2) \in \widehat{\mathbb{R}}^2$ ,  $\xi_1 \neq 0$ , when  $\psi^{(0)}$ ,  $\widehat{\psi}_1$ ,  $\widehat{\psi}_2$  satisfy

$$\widehat{\psi}^{(0)}(\xi) = \widehat{\psi}^{(0)}(\xi_1, \xi_2) = \widehat{\psi}_1(\xi_1) \widehat{\psi}_2\left(\frac{\xi_2}{\xi_1}\right)$$

$$\text{supp } \widehat{\psi}_1 \subset \left[-\frac{1}{2}, -\frac{1}{16}\right] \cup \left[\frac{1}{16}, \frac{1}{2}\right], \quad \text{supp } \widehat{\psi}_2 \subset [-1, 1]$$

$$\widehat{\psi}_1, \widehat{\psi}_2 \in C^\infty(\widehat{\mathbb{R}}),$$

$$\sum_{j \geq 0} |\widehat{\psi}_1(2^{-2j}\omega)|^2 = 1 \quad \text{for } |\omega| \geq \frac{1}{8}, j \geq 0$$

$$\sum_{l=-2^j}^{2^j-1} |\widehat{\psi}_2(2^j\omega - l)|^2 = 1 \quad \text{for } |\omega| \leq 1. \quad (3)$$

Then, we get

$$\begin{aligned} \sum_{j \geq 0} \sum_{l=-2^j}^{2^j-1} |\widehat{\psi}^{(0)}(\xi A_0^{-j} B_0^{-l})|^2 \\ = \sum_{j \geq 0} \sum_{l=-2^j}^{2^j-1} |\widehat{\psi}_1(2^{-2j}\xi_1)|^2 \left| \widehat{\psi}_2\left(\frac{2^j\xi_2}{\xi_1} - l\right) \right|^2 = 1. \end{aligned} \quad (4)$$

Then,  $\{\widehat{\psi}^{(0)}(\xi A_0^{-j} B_0^{-l})\}$  form a tiling of the set

$$D_0 = \left\{ (\xi_1, \xi_2) \in \widehat{\mathbb{R}}^2 : |\xi_1| \geq \frac{1}{8}, \left| \frac{\xi_2}{\xi_1} \right| \leq 1 \right\}. \quad (5)$$

From the condition on the support of  $\widehat{\psi}_1$ ,  $\widehat{\psi}_2$ , it is easily deduced that  $\widehat{\psi}_{j,l,k}$  have frequency support contained in the set

$$\begin{aligned} \text{supp } \widehat{\psi}_{j,l,k}^{(0)} \subset \left\{ (\xi_1, \xi_2) : \xi_1 \in [-2^{2j-1}, -2^{2j-4}] \right. \\ \left. \cup [2^{2j-4}, 2^{2j-1}], \left| \frac{\xi_2}{\xi_1} + l2^{-2j} \right| \leq 2^{-j} \right\}. \end{aligned} \quad (6)$$

Thus, every element in  $\psi_{j,l,k}$  is supported on a pair of trapezoids of approximate size  $2^{2j} \times 2^j$ , oriented along lines of slope  $l2^{-j}$ .

For  $L^2(D_1)^\vee$ , here  $D_1$  is the vertical cone, when formula (7) was satisfied:

$$D_1 = \left\{ (\xi_1, \xi_2) \in \widehat{\mathbb{R}}^2 : |\xi_2| \geq \frac{1}{8}, \left| \frac{\xi_1}{\xi_2} \right| \leq 1 \right\}$$

$$A_1 = \begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

$$\widehat{\psi}^{(1)}(\xi) = \widehat{\psi}^{(1)}(\xi_1, \xi_2) = \widehat{\psi}_1(\xi_2) \widehat{\psi}_2\left(\frac{\xi_1}{\xi_2}\right). \quad (7)$$

Then, collection  $\{\psi_{j,l,k}^{(1)}(x) = 2^{3j/2} \psi^{(1)}(B_1^l A_1^j x - k) : j \geq 0, -2^j \leq l \leq 2^j - 1, k \in \mathbb{Z}^2\}$  is a Parseval frame for  $L^2(D_1)^\vee$ .

*2.2. Particle Swarm Optimization Algorithm.* The classical PSO algorithm is described as follows [34].

*Step 1.* Initialize a population of particles with random positions and velocities in a  $n$ -dimension problem space.

*Step 2.* For each particle, evaluate its fitness value.

*Step 3.* Compare each particle's fitness evaluation with the current particle's *pbest*. If current value is better than *pbest*,

set its  $pbest$  value to the current value and the  $pbest$  location to the current location in  $n$ -dimensional space.

*Step 4.* Compare fitness evaluation with the population's overall previous best. If current value is better than  $gbest$ , then reset  $gbest$  to the current particle's array index and value.

*Step 5.* Change the velocity and position of particle according to Step 1 and Step 2, respectively:

$$\begin{aligned} v_i &= \omega v_i + c_1 \text{rand} (p_i - x_i) + c_2 \text{rand} (p_g - x_i) \\ x_{i+1} &= (x_i + v_i). \end{aligned} \quad (8)$$

*Step 6.* Loop to Step 2 until a stopping criterion is met.

*Step 7.* Over.

The vector  $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T$  is the position of  $i$ th particles,  $v_i = [v_{i1}, v_{i2}, \dots, v_{in}]^T$  is the velocity of the  $i$ th particles, and  $p_i = [p_{i1}, p_{i2}, \dots, p_{in}]^T$  is the best previous position (the position giving the best fitness value) of the  $i$ th particles. The index  $g$  is the index of the best particle among all the particles in the swarm. Variable  $\omega$  is the inertia weight,  $c_1$  and  $c_2$  are positive constants, and  $\text{rand}$  is random number in range  $[0, 1]$  generated according to a uniform probability distribution. Particles' velocities along each dimension are clamped to a maximum velocity  $V_{\max}$ . If the sum of acceleration causes the velocity on that dimension to exceed  $V_{\max}$ , which is a parameter specified by the user, then the velocity on that dimension is limited to  $V_{\max}$ .

The inertia weight  $\omega$  represents the degree of the momentum of the particles. The second part is the "cognition" part, which represents the independent behavior of the particle itself. The third part is the "social" part, which represents the collaboration among the particles. The constants  $c_1$  and  $c_2$  represent the weighting of "cognition" and "social" parts that pull each particle towards  $pbest$  and  $gbest$  positions.

### 3. Proposed Algorithm

This chapter is divided into 3 parts. Section 3.1 introduced the new threshold rule we designed which is based on classical Shearlet theory. And the fitness function we quoted is shown in Section 3.2. Then, our all algorithm workflow (Figure 1) is detailed and introduced in Section 3.3.

*3.1. Threshold Rule.* Threshold rule is the most important problem in image denoising of transform domain, and the hard threshold and the soft-threshold approach are two options. Fan et al. [18] proposed a threshold rule:

$$\delta = \sigma \sqrt{2 \ln(N)}. \quad (9)$$

Here,  $N$  is the pixels number of image and  $\sigma$  is noise level.

Research shows that Fan et al. threshold is not the optimal threshold. Considering this, Fan et al. [18] proposed improved threshold rules:

$$\delta_k = \sigma \sqrt{2 \ln(N)} * 2^{(k-K)/2}, \quad k = 0, 1, \dots, K. \quad (10)$$

All formula (9) and formula (10) are intended to get better denoising performance by setting good threshold to each subwavelet, but they can not find the best threshold. Based on this idea, here, we proposed a new idea that adopts optimization algorithm to find the best performance; that is, give weighted factor to each subwavelet; then, use PSO algorithm to find the best factors.

However, in optimization algorithm, the key problem is how to design an objective and good fitness.

*3.2. Fitness Function.* Fan and Zhao [17–19] proposed the way to get best threshold by optimal algorithm, but in their papers, they did not introduce where to get fitness function.

Wang [35] proposed a fast and rough evaluated method to noise level of noisy image, and we used as our fitness parameter of PSO algorithm. We defined our fitness as follows:

$$\text{SNR}' = \frac{I_s}{\hat{\delta}_w}, \quad (11)$$

where  $I_s$  is estimate of signal intensity in image and  $\hat{\delta}_w$  is estimate of noise intensity in image.

Define  $x(m, n)$  as an unpolluted gray image with size of  $M \times N$ , and define an area  $D_o = \{(m, n) \mid m \in [1, M] \wedge n \in [1, N]\}$ ; with the same form, we can define a noisy image  $y(m, n)$  as follows:

$$y(m, n) = x(m, n) + w(m, n), \quad \forall (m, n) \in D_o, \quad (12)$$

where  $w(m, n)$  is noise model; in this paper, it is Gaussian noise model. From formula (12), we designed and evaluated methods of  $I_s$  and  $\hat{\delta}_w$ :

$$\begin{aligned} I_s &= \frac{\max(y * G_\rho) - \min(y * G_\rho)}{2} \\ G_\rho &= \frac{1}{2\pi\rho^2} \exp\left(-\frac{x_1^2 + x_2^2}{2\rho^2}\right). \end{aligned} \quad (13)$$

Here, let us suppose the classical Gaussian filter is perfect; its output is noiseless, so we can suppose the median data of signal to be the estimate of signal intensity in image.

Then, we suppose signal in image is at low frequencies, and  $H_{\text{grad}}$  is perfect; its output is signal-free totally, so we can suppose the standard variance  $\hat{\delta}_w$  in image to be the estimate of noise intensity in image:

$$H_{\text{grad}} = \frac{1}{2} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \quad (14)$$

$$\hat{\delta}_w = \text{std}(H_{\text{grad}} * y).$$

Here,  $\text{std}(\cdot)$  is standard variance.

Based on this definition of fitness, we proposed our algorithm.

*3.3. Proposed Algorithm.* The most critical problem is how to find the best weighted factor  $\delta_{k,j}$ . Here, we proposed our

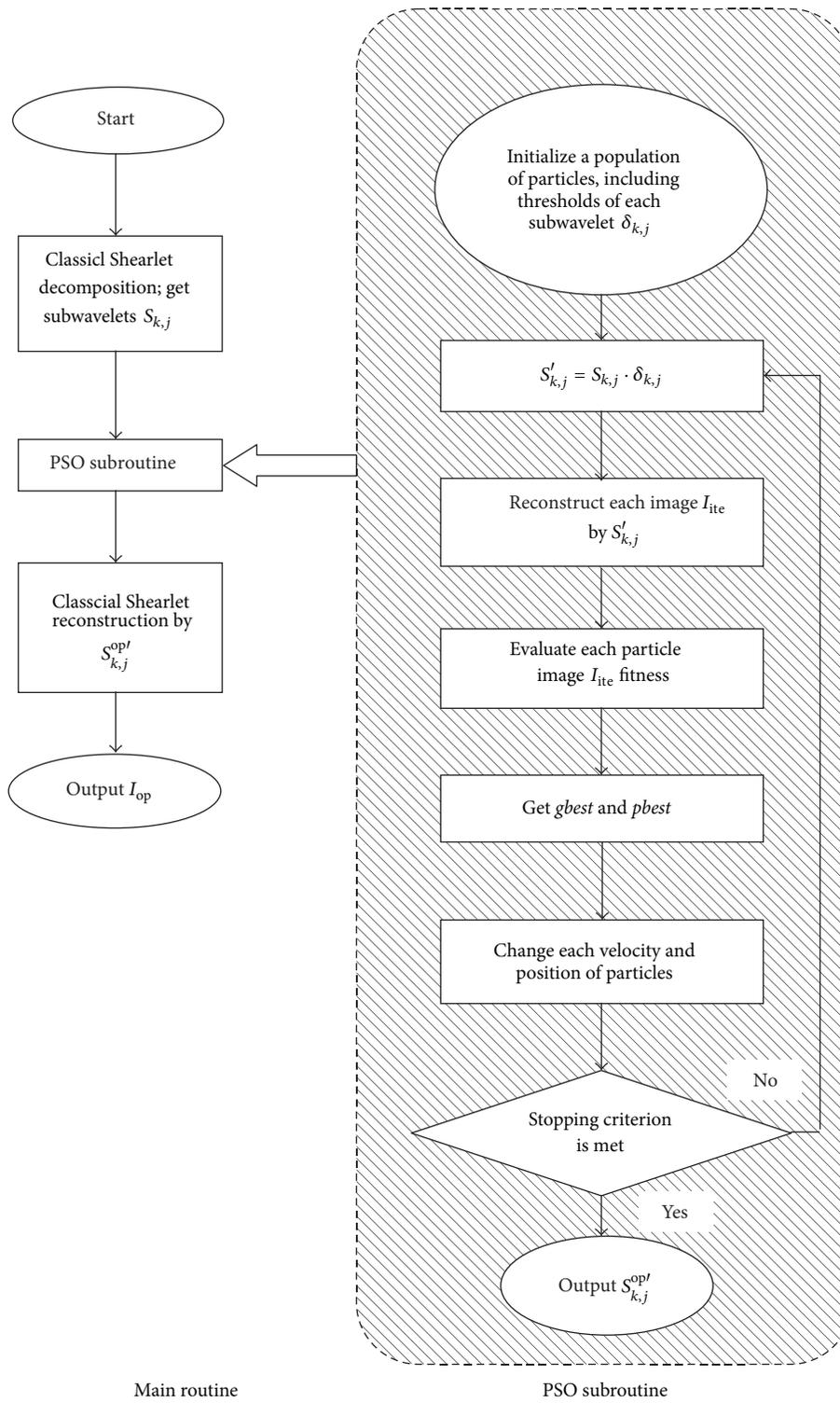


FIGURE 1: Workflow of proposed algorithm.

TABLE 1: Improved PSNR of Figure 3.

Average of Figure 3(f)	Average of Figure 3(h)	Average of Figure 3(j)	Average of Figure 3(l)
0.057	0.072	0.127	0.119
Best of Figure 3(f)	Best of Figure 3(h)	Best of Figure 3(j)	Best of Figure 3(l)
0.132	0.243	0.281	0.542
Worst of Figure 3(f)	Worst of Figure 3(h)	Worst of Figure 3(j)	Worst of Figure 3(l)
-0.043	0.005	-0.076	-0.081

algorithm which adopts PSO algorithm to decide coefficients  $\delta_{k,j}$  of each subwavelet  $S_{k,j}$  in different scale and direction of Shearlet transform, intended to get optimal denoising performance.

Our algorithm main routine is as follows.

*Step 1.* Use classical Shearlet transform to decompose noised image into many pieces under multiscale and multiorientation, get subwavelet  $S_{k,j}$ ,  $k$  is scale, and  $j$  is direction.

*Step 2.* Give weighted factor  $\delta_{k,j}$  to those subwavelets  $S_{k,j}$  obtained, reconstruct image, design a fitness function to evaluate noise level of new image, and use PSO algorithm to find the best  $\delta_{k,j}^{\text{op}}$  and output  $S_{k,j}^{\text{op}} = S_{k,j} \cdot \delta_{k,j}^{\text{op}}$ .

*Step 3.* Output the best image  $I_{\text{op}}$  which is reconstructed by  $S_{k,j}^{\text{op}}$ .

Our PSO subroutine is shown as follows.

*Step 1.* Initialize a population of particles with random positions and velocities.

*Step 2.* For each particle,  $S'_{k,j} = S_{k,j} \cdot \delta_{k,j}$ .

*Step 3.* For each particle, reconstruct image  $I_{\text{ite}}$  by  $S'_{k,j}$ .

*Step 4.* For each  $I_{\text{ite}}$ , evaluate its fitness by formula (11).

*Step 5.* Compare each particle's fitness evaluation with the current particle's  $pbest$ . If current value is better than  $pbest$ , set its  $pbest$  value to the current value and the  $pbest$  location to the current location in 2-dimensional space.

*Step 6.* Compare fitness evaluation with the population's overall previous best. If current value is better than  $gbest$ , then reset  $gbest$  to the current particle's array index and value.

*Step 7.* Change the velocity and position of particle, respectively:

$$\begin{aligned} v_i &= wv_i + c_1 \text{rand}(p_i - x_i) + c_2 \text{rand}(p_g - x_i) \\ x_{i+1} &= (x_i + v_i). \end{aligned} \quad (15)$$

*Step 8.* Loop to Step 2 until a stopping criterion is met. If one stopping condition in the following is satisfied, go to Step 9; if not, return to Step 2.

(i) Maximum iterations are exceeded.

(ii) The optimal target value is achieved.

*Step 9.* Exit optimal algorithm, output  $S_{k,j}^{\text{op}}$ .

## 4. Experimental Results and Analysis

*4.1. Fitness Function Performance Test.* For the sake of evaluating the performance of fitness function in proposed algorithm, we added Gaussian noise in several classical images with different variance and then calculated their fitness function data and plotted in Figure 2.

In Figure 2, as we can see from this output, despite the existence of some small fluctuation, those lines can basically reflect the trend that image's quality is moving in the worse direction, and that is exactly what we need in our optimal algorithm.

These two images are 512 \* 512 pixels, and the test platform is window 7, Matlab 2012a; CPU is i7-3770 3.4 GHz, 16 G RAM. By using "tic" and "toc" function, its average elapsed time is 0.001612 seconds, so this fitness function can be calculated very fast.

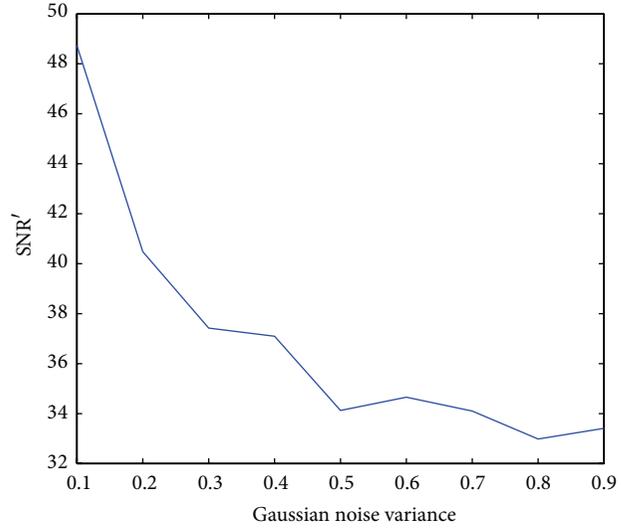
Based on the above, we can draw a conclusion that this fitness function (formula (11)) is able to be applied in our algorithm basically.

*4.2. Total Algorithm Performance Test.* In order to validate the performance of this algorithm, 4 images were filtered by proposed algorithm and classical Shearlet algorithm that we downloaded code from website [36]. These 4 test images are head MRI (Figure 3(a)), baboon (Figure 3(b)), Barbara (Figure 3(c)), and brain MRI (Figure 3(d)) image; we added different noise level into them, filtered these noisy images by two algorithms, and listed their filtered results PSNR (peak signal noise ratio) in Figure 2. Here, PSNR has the same classical definition in [3]; it has no relationship with SNR' which we defined by ourselves.

Figures 3(e), 3(g), 3(i), and 3(k) are comparison charts of two algorithms PSNR output and we can see that all red line is higher than all blue line totally, though all two lines are similar. In order to get a better view, we listed these differences of two-group data by bar charts of Figures 3(f), 3(h), 3(j), and 3(l). From these bar charts, we can directly find that our proposed algorithm is better than classical Shearlet algorithm, and this better performance will decrease when noise level is higher.



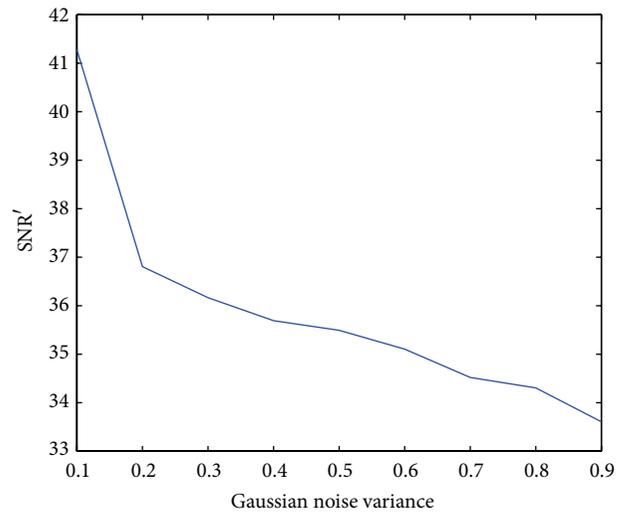
(a) Boat



(b) Our fitness function of image (a) in different noise level



(c) Barbara



(d) Our fitness function of image (c) in different noise level

FIGURE 2: Performance of our fitness function.

TABLE 2: Some PSO parameters.

Population size	24
Local best influence	2
Initial inertia weight	0.9
Minimum global error gradient	$1e-10$
VarRange	[0~1]
Maximum number of iterations	100
Global best influence	2
Final inertia weight	0.4
Max particle velocity	0.1

So the result of these experiments data had shown that our proposed algorithm has better denoising performance than classical Shearlet algorithm, and this advantage will decrease when noise level increased.

To show the “improved PSNR” data in Figure 3 better, Table 1 is show; in this table, from those average data, we can see that our algorithm has good performance, but those worst data reflect our algorithm that has drawback; we guess that is caused by the small fluctuation of our fitness function.

In all above experiments, 9 main PSO algorithm’s parameters adopted are shown in Table 2.

## 5. Conclusions

In order to eliminate Gaussian noise and preserve image details as much as possible, this paper proposed a novel image filter algorithm which combined merit of PSO algorithm and classical Shearlet transform. It decomposes noised image into many subwavelets under multiscale and multiorientation by classical Shearlet transform; design a fitness functions in PSO and give weighted factors to each subwavelet; use PSO to

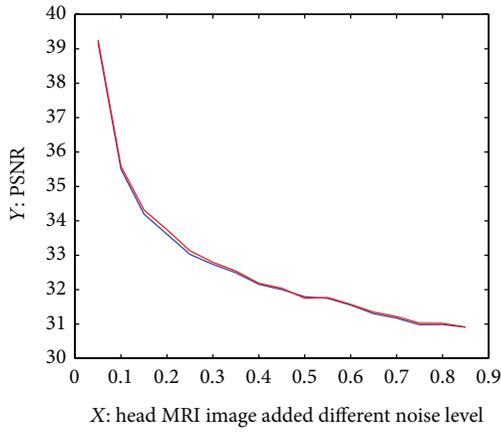


(a) Head MRI

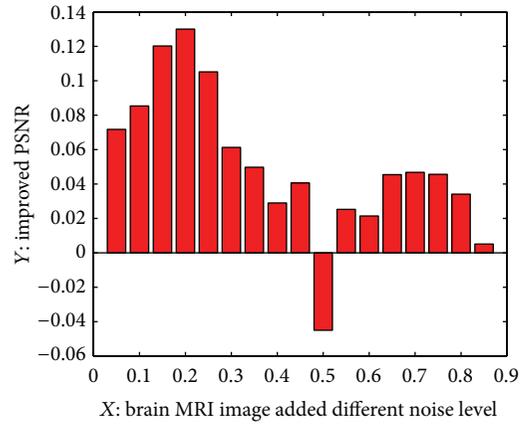
(b) Baboon

(c) Barbara

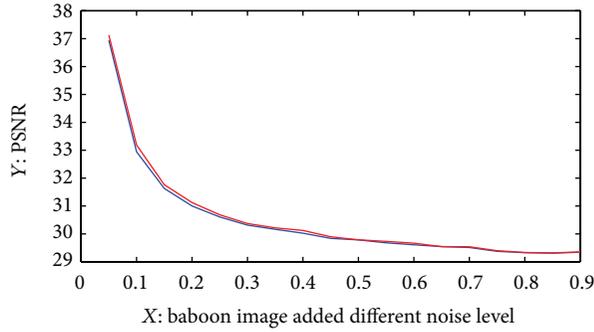
(d) Brain MRI



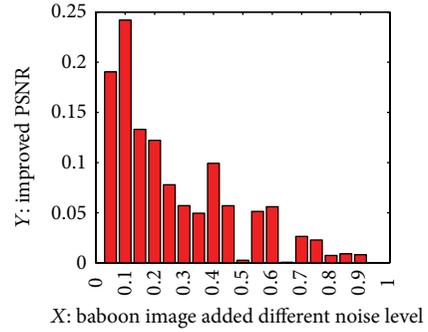
(e) Comparison chart of two algorithms on (a)



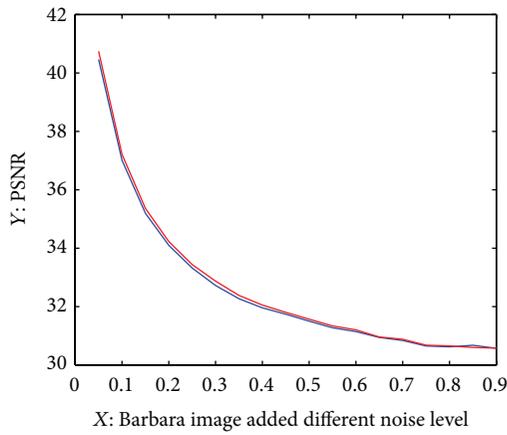
(f) Improved PSNR of (e)



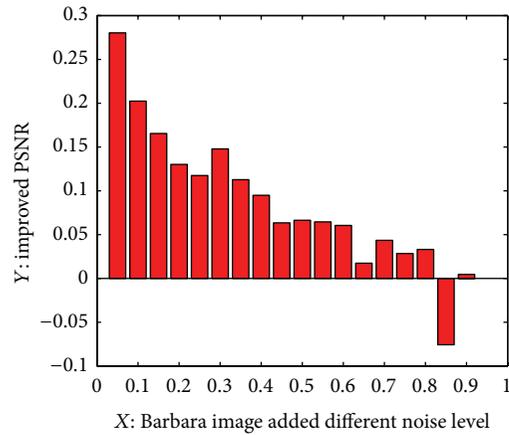
(g) Comparison chart of two algorithms on (b)



(h) Improved PSNR of (g)



(i) Comparison chart of two algorithms on (c)



(j) Improved PSNR of (i)

— Classical Shearlet  
— Proposed

FIGURE 3: Continued.

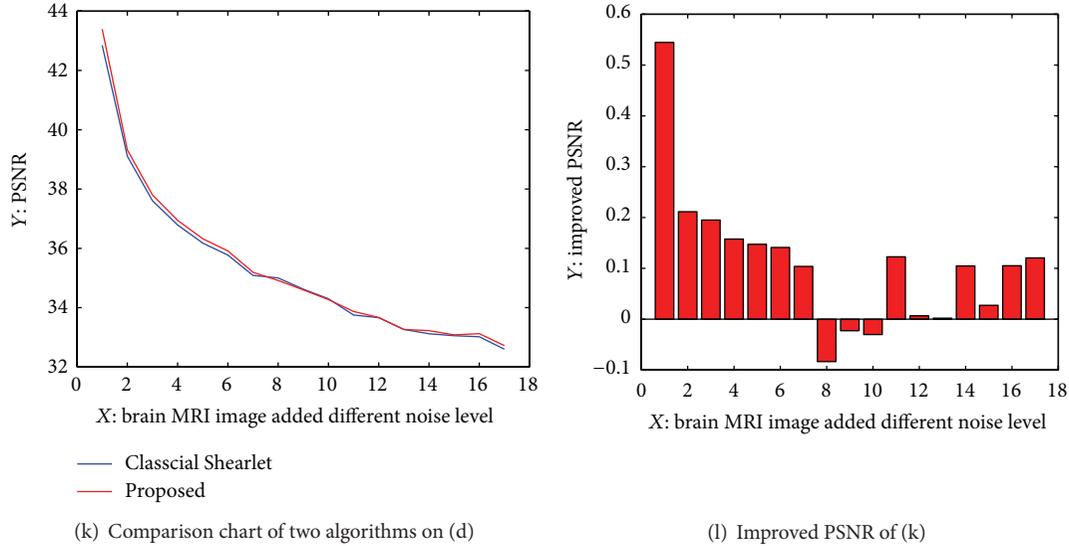


FIGURE 3: Comparison chart of two algorithms on some images.

find optimal weighted factors; after that, reconstruct new and denoised image. It has two contributions: one is proposing PSO to optimal Shearlet transform and another is quoting a rough, objective, fast fitness function to measure images quality. Computer simulations results are given to verify the effectiveness of this algorithm.

### Conflict of Interests

The authors declare that they have no conflict of interests.

### Authors' Contribution

All authors drafted the paper and read and approved the final paper.

### Acknowledgments

Whatever implementation the authors come up with needs to be based on classical Shearlet algorithm, and they cannot execute their work without Professor Labate's earlier work and his open-source code online. Thanks go to Professor Labate. The authors would like to thank the anonymous reviewers whose comments greatly improved the paper. This paper was funded under a Grant from the National Natural Science Foundation of China (no. 61105115), "Six Talent Peaks Program" of Jiangsu Province of China (2014-XXRJ-007), a Grant from the Jiangsu Province Natural Science Foundation (nos. BK20131000, S7013028001), National Training Program for Undergraduate on Innovation and Entrepreneurship (no. 201310300039Z), Industry Academia Research Cooperative Innovation Fund of Jiangsu Province (BY2013007-01), and Open Program based on Laboratory of Nanjing University of Information Science and Technology (nos. 2014KF001, 2014KF006, and 2014KF0019).

### References

- [1] D. Labate, W. Lim, G. Kutyniok, and G. Weiss, "Sparse multidimensional representation using shearlets," in *11th Wavelets*, vol. 5914 of *Proceedings of SPIE*, pp. 254–262, San Diego, Calif, USA, September 2005.
- [2] D. Labate and G. Weiss, "Wavelets associated with composite dilations," in *Matemáticas: Investigación y Educación. Un homenaje a Miguel de Guzmán*, Universidad Complutense de Madrid, Madrid, Spain, 2005.
- [3] K. Hu, A. Song, M. Xia, X. Ye, and Y. Dou, "An adaptive filtering algorithm based on genetic algorithm-backpropagation network," *Mathematical Problems in Engineering*, vol. 2013, Article ID 573941, 8 pages, 2013.
- [4] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, pp. 259–268, 1992.
- [5] M. N. Do and M. Vetterli, "The finite ridgelet transform for image representation," *IEEE Transactions on Image Processing*, vol. 12, no. 1, pp. 16–28, 2003.
- [6] E. J. Candes and D. L. Donoho, "Curvelets: a surprisingly effective nonadaptive representation of objects with edges," in *Curves and Surfaces Fitting*, pp. 105–120, Vanderbilt University Press, Nashville, Tenn, USA, 2000.
- [7] M. N. Do and M. Vetterli, *Contourlets Beyond Wavelets*, edited by G. V. Welland, Academic Press, 2003.
- [8] S. Mallat and G. Peyré, "A review of bandlet methods for geometrical image representation," *Numerical Algorithms*, vol. 44, no. 3, pp. 205–234, 2007.
- [9] D. Labate, W. Q. Lim, G. Kutyniok, and G. Weiss, "Sparse multidimensional representation using shearlets," in *Wavelets XI*, vol. 5914 of *Proceedings of SPIE*, pp. 254–262, San Diego, Calif, USA, 2005.
- [10] S. Yi, D. Labate, G. R. Easley, and H. Krim, "A shearlet approach to edge analysis and detection," *IEEE Transactions on Image Processing*, vol. 18, no. 5, pp. 929–941, 2009.
- [11] S. Yi, D. Labate, G. R. Easley, and H. Krim, "Edge detection and processing using shearlets," in *Proceedings of the IEEE*

- International Conference on Image Processing (ICIP '08)*, pp. 1148–1151, San Diego, Calif, USA, October 2008.
- [12] G. R. Easley, D. Labate, and F. Colonna, “Shearlet-based total variation diffusion for denoising,” *IEEE Transactions on Image Processing*, vol. 18, no. 2, pp. 260–268, 2009.
- [13] S. Yi, D. Labate, G. R. Easley, and H. Krim, “A shearlet approach to edge analysis and detection,” *IEEE Transactions on Image Processing*, vol. 18, no. 5, pp. 929–941, 2009.
- [14] G. R. Easley, D. Labate, and W.-Q. Lim, “Optimally sparse image representations using shearlets,” in *Proceedings of the 40th Asilomar Conference on Signals, Systems, and Computers (ACSSC '06)*, pp. 974–978, IEEE, November 2006.
- [15] M. A. Borgi, D. Labate, M. ElArbi, and C. Ben Amar, “Sparse multi-regularized shearlet-network using convex relaxation for face recognition,” in *Proceedings of the International Conference on Pattern Recognition (ICPR '14)*, 2014.
- [16] G. R. Easley, D. Labate, and V. M. Patel, “Directional multiscale processing of images using wavelets with composite dilations,” *Journal of Mathematical Imaging and Vision*, vol. 48, no. 1, pp. 13–34, 2014.
- [17] Z.-Y. Fan, Q.-S. Sun, Z.-X. Ji, and K. Hu, “An image filter based on multiobjective genetic algorithm and shearlet transformation,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 463760, 7 pages, 2013.
- [18] Z. Fan, Q. Sun, and F. Ruan, “An improved image denoising algorithm based on Shearlet,” *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 6, no. 4, 2013.
- [19] J. Zhao, H. Sun, C.-Z. Deng, and X. Chen, “Particle swarm optimization based adaptive image denoising in shearlet domain,” *Journal of Chinese Computer Systems*, vol. 32, no. 6, pp. 1147–1150, 2011 (Chinese).
- [20] X. Chen, C. Deng, and S. Wang, “Shearlet-based adaptive shrinkage threshold for image denoising,” in *Proceedings of the International Conference on E-Business and E-Government (ICEE '10)*, pp. 1616–1619, IEEE, May 2010.
- [21] M. Xia and W. K. Wong, “A seasonal discrete grey forecasting model for fashion retailing,” *Knowledge-Based Systems*, vol. 57, pp. 119–126, 2014.
- [22] S. Aiguo and L. Jiren, “Evolving Gaussian RBF network for nonlinear time series modelling and prediction,” *Electronics Letters*, vol. 34, no. 12, pp. 1241–1243, 1998.
- [23] K. Hu, A. G. Song, Y. C. Zhang, and W. L. Wang, “Delay-range-dependent stability criteria of neural networks with time-varying discrete and distributed delays,” *International Journal of Advanced Robotic Systems*, vol. 11, no. 1, article 23, 2014.
- [24] K. Hu, A. Song, W. Wang, Y. Zhang, and Z. Fan, “Fault detection and estimation for non-Gaussian stochastic systems with time varying delay,” *Advances in Difference Equations*, vol. 2013, article 22, 2013.
- [25] X. Wen and A. Song, “An immune evolutionary algorithm for sphericity error evaluation,” *International Journal of Machine Tools and Manufacture*, vol. 44, no. 10, pp. 1077–1084, 2004.
- [26] A. Song, Y. Han, H. Hu, L. Tian, and J. Wu, “Active perception-based haptic texture sensor,” *Sensors and Materials*, vol. 25, no. 1, pp. 1–15, 2013.
- [27] M. Xia, Y. Zhang, L. Weng, and X. Ye, “Fashion retailing forecasting based on extreme learning machine with adaptive metrics of inputs,” *Knowledge-Based Systems*, vol. 36, pp. 253–259, 2012.
- [28] M. Xia, Z. Wang, and J. Fang, “Temporal association based on dynamic depression synapses and chaotic neurons,” *Neurocomputing*, vol. 74, no. 17, pp. 3242–3247, 2011.
- [29] M. Xia, L. Weng, and X. Ye, “Sequence memory based on ordered pattern interrelation,” *Advanced Science Letters*, vol. 5, no. 2, pp. 547–551, 2012.
- [30] Q. Liu, H. Lu, and S. Ma, “Improving Kernel Fisher discriminant analysis for face recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 42–49, 2004.
- [31] Q. Liu, R. Huang, H. Lu, and S. Ma, “Kernel-based nonlinear discriminant analysis for face recognition,” *Journal of Computer Science and Technology*, vol. 18, no. 6, pp. 788–795, 2003.
- [32] Q. Liu, S. Ma, and H. Lu, “Head tracking using shapes and adaptive color histograms,” *Journal of Computer Science and Technology*, vol. 17, no. 6, pp. 859–864, 2002.
- [33] M. A. Duarte-Mermoud, N. H. Beltrán, and S. A. Salah, “Probabilistic adaptive crossover applied to chilean wine classification,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 734151, 10 pages, 2013.
- [34] R. A. Krohling, “Gaussian swarm: a novel particle swarm optimization algorithm,” in *Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems*, vol. 1, pp. 372–376, December 2004.
- [35] W.-Y. Wang, “Selecting the optimal Gaussian filtering scale via the SNR of image,” *Journal of Electronics and Information Technology*, vol. 31, no. 10, pp. 2483–2487, 2009 (Chinese).
- [36] <http://www.math.uh.edu/~dlabate/software.html>.

## Research Article

# Stamps Detection and Classification Using Simple Features Ensemble

**Paweł Forczmański and Andrzej Markiewicz**

*Faculty of Computer Science and Information Technology, West Pomeranian University of Technology,  
Szczecin, Żołnierska Street 52, 71-210 Szczecin, Poland*

Correspondence should be addressed to Paweł Forczmański; [pforczmanski@wi.zut.edu.pl](mailto:pforczmanski@wi.zut.edu.pl)

Received 1 August 2014; Accepted 23 September 2014

Academic Editor: Erik Cuevas

Copyright © 2015 P. Forczmański and A. Markiewicz. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The paper addresses a problem of detection and classification of rubber stamp instances in scanned documents. A variety of methods from the field of image processing, pattern recognition, and some heuristic are utilized. Presented method works on typical stamps of different colors and shapes. For color images, color space transformation is applied in order to find potential color stamps. Monochrome stamps are detected through shape specific algorithms. Following feature extraction stage, identified candidates are subjected to classification task using a set of shape descriptors. Selected elementary properties form an ensemble of features which is rotation, scale, and translation invariant; hence this approach is document size and orientation independent. We perform two-tier classification in order to discriminate between stamps and no-stamps and then classify stamps in terms of their shape. The experiments carried out on a considerable set of real documents gathered from the Internet showed high potential of the proposed method.

## 1. Introduction

Nowadays, computer analysis of digitized documents is one of key areas of digital image processing and pattern recognition. Computer-based processing of documents joins many fields of research, especially computer vision, image analysis, pattern recognition, and artificial intelligence. The detection and recognition of rubber stamps (seals) on digital images are still very important problems. As we face a significant change in the technology, namely, a conversion from paper documents into digital ones, the need for solutions for automatic segmentation and extraction of important document's elements is very high. The problem dates back to the '80s last century, when an issue of stamp authenticity has been addressed [1]. However, despite considerable progress in this field, this problem still remains open [2–4].

The literature survey shows that algorithms of stamp detection and recognition employ different features of digital images. Most of the methods can be classified as ones using shape information [4, 5] or ones based on color features [3, 6]. There are only a few algorithms that join features from those domains (e.g., [7]).

While stamp detection can be realized by means of both shape and color features, the problem of stamp classification is purely shape-based. It has been proved many times that shape (silhouette, contour) is one of the most important low level image features since it is a very important attribute to human perception [8]. Humans tend to perceive complex scenes as being composed of elementary objects, which can be identified by their shapes. Shape is a particularly useful, especially when we want to identify specific objects in image, for example, in medicine or biology (e.g., cell shapes in microscopic images) [9, 10] or surveillance (e.g., optical character recognition of license plates [11]) and many other systems, where objects easy to segment may be determined. While general shape recognition and classification problems have a significant place in the scientific literature, often with reported accuracies of over 90% (e.g., [12, 13]), the specific topic of stamps retrieval by means of low-dimensional feature sets is much less represented (with significantly lower accuracies).

A recent study shows that few different methods oriented at automatic stamp detection and classification have been proposed. They work either on color stamps [3] or detect

objects of particular shape [4, 14]. Unfortunately, there is no general approach aimed at detection and classification of all the diversity of stamps. There is also a large group of algorithms oriented at logo retrieval, which is a similar problem [15, 16]. The newest ones include application of modern shape descriptors, for example, SIFT/SURF/FAST [17–19] or ART [20]. It should be, however, noted that such approaches aimed at logo detection can not be directly employed to stamp detection because of different properties of both objects' classes. Such geometrical features that are common for both object classes [15, 21, 22] are heavily influenced by the process of imprinting (noise, inconsistency, gaps, stains, etc.) in the stamp representation. Because of this fact and the diversity of stamps, even within a single class (in terms of shape), a straightforward application of such descriptors is not possible.

In this paper we present a new solution for detecting stamps of different shapes and colors. The experiments showed that they can be extracted properly even if they are overlapped with signature or text. Because of the high variance in the shape domain we focus on rather standardized stamps (with well-defined shape) of any particular color. In opposition to [3], we do not rely on color properties only; hence the detection of black stamps is also possible. Moreover, shapes being detected are not limited to ovals and squares, like in [4]. The experiments showed also some good results of detecting stamps in documents containing other similar objects such as logos and texts.

*1.1. Existing Approaches.* As it has been noted, the algorithms of stamp detection and recognition can employ different features of digital images. These features are based, in most cases, on color or edge information. The algorithms proposed in [3, 6] use  $YC_bC_r$  color representation. Candidate areas are extracted using the projection of  $C_b$  and  $C_r$  matrices onto both axes. As an effect, a binary mask of bounding boxes (instead of the original areas) is generated. Unfortunately, it negatively influences further processing aimed at determining their shapes. Although, the reported detection accuracy of color stamps is rather high (over 83%), such a solution can not be employed in case of grayscale images or black-colored stamps.

The method proposed in [23] allows detecting stamps even in highly degraded documents. It does not take into consideration the color information, since the main principle of the proposed method is so called connected edges that creates contour of the stamp. In the first step, an input image is subjected to scaling and edge detection. Then, candidate regions are evaluated using a set of rules that are characteristic to pixels within stamps. Selected elements are finally verified using Hough Transform. Unfortunately, the detection algorithm can deal with circular- and oval-shaped stamps only. The reported accuracy is higher than 35% and it is not sufficient for practical implementations.

Another method that employs edge information is presented in [5]. It uses a two-stage approach. The first step is based on letters detection. The letters being often a part of a stamp have specific spatial properties, which are employed

in presented method. The second stage involves the usage of a Generalized Hough Transform in order to verify the result obtained in the first step. GHT has a clear advantage over the basic Hough Transform version since it allows finding any particular shapes. The experiment performed by the authors showed that the accuracy is close to 70%.

The solution proposed in [15] addresses the problem of logo detection, which can be considered similar to the tasks of stamp detection and recognition. Proposed approach is divided into three stages: segmentation, in which a document is decomposed into smaller parts (text regions, images, logos, etc.), detection of logotypes based on the features extracted from output segments, and matching logotypes by means of template matching. The main principles of the proposed method are projection of binarized image, analysis of connected components, and a classification stage. Unfortunately, this solution can not be directly used for the detection of stamps. Features used by the authors of [15] do not allow getting such good accuracy of classification in case of stamps, which differ significantly from logotypes.

The whole group of methods is based on recently proposed features, like Scale-Invariant Feature Transform (SIFT) or Speeded Up Robust Features (SURF). In the paper [17] a fast and efficient logo detector is proposed. The solution involves the use of feature vectors obtained by SURF. It is designed to detect a set of local features for each key point in the image. The final features are reduced using Principal Component Analysis. The reference database contains a large collection of samples. Despite a quite high complexity, the accuracy of the proposed algorithm is equal to 67%.

The algorithm proposed in [20] is also a multitier approach. Firstly, it performs a noise reduction (median filtering) on the input image. Then the image is binarized and subjected to horizontal dilation. Extracted candidate regions are described using a set of geometrical features. The final verification is done using a tree. The next step is verification. It is done using a set of rules. After successful detection, logos are classified using  $k$ -Nearest Neighbor classifier. Reported accuracy is equal to 92%; however the method works only on logotypes that are not overlapped by text or any other objects.

It is worth noticing that in the method presented in this paper we joined several elementary approaches in order to create more flexible and more accurate algorithm.

*1.2. Data Characteristics.* The problem with stamp detection and classification comes mainly from the lack of templates or general models. The in-depth analysis shows that this is due to the absence of standard and commonly used stamp's representation. Stamps are complex objects, containing graphical and textual elements that can be located anywhere within documents. Their diversity comes from varying orientation, color, fonts, ornaments, and quality of imprinting. Even two imprints of the same physical stamp can look very different. The same applies to other objects that were described in [21]. Taking into consideration above limitations, object's shape is considered one of the most stable features in terms of classification. Exemplary stamps extracted from real documents divided into shape-specific classes are provided



FIGURE 1: Sample official stamps, often regular and without decorations (first four rows), and unofficial stamps, more complex, with many decorative motives (the last row).

in Figure 1. The last row of Figure 1 shows stamps that can not be categorized in terms of shape. Such stamps are mostly unofficial and less common.

The color information is very important when it comes to document analysis since original documents in most cases include color stamps. The black stamps are less frequently used for official stamping, and the lack of color in many cases suggests a copy (not original document). It is also worth noticing that the stamps on official documents do not cover large area (often not more than 3%–5% of the total image area for A4 page) [24]. Above observation can ease the stage of stamp detection.

## 2. Algorithm Overview

The input for the proposed approach is given as a scanned or photographed (paper) document. In the scanning process the imaging plane is parallel to the document plane; hence we preserve geometrical features of all elements. In order to reduce the artifacts it is also recommended that the input image has quite high spatial resolution and is stored in a file format that provides minimal loss of quality after decompression. The algorithm presented further is developed to work on both color and monochrome images. It is worth noticing also that the input images are not subjected to any brightness alternation (neither histogram stretching nor equalization). Moreover, we do not assume the number of stamps as well as their location and orientation. The output provides information about the number of stamps detected, their shapes, colors, and coordinates.

The algorithm (see Figure 2) consists of the following steps.

- (1) Load an image.
- (2) Detect candidates (in monochrome images or in each color channel in case of color image):

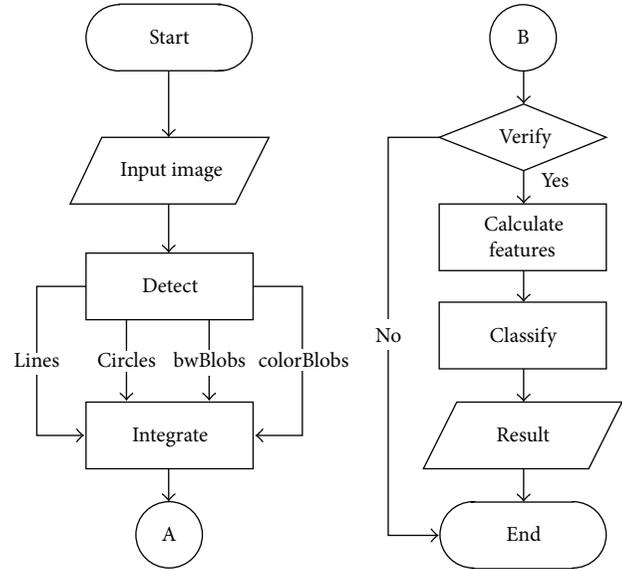


FIGURE 2: Stages of processing.

- (i) detect lines;
- (ii) detect circles;
- (iii) detect other shapes (*bwBlobs* or *colorBlobs*).

- (3) Integrate locations and dimensions.
- (4) Verify stamps/no-stamps.
- (5) Classify and report the results.

Firstly, in case of color images, we perform a color conversion  $RGB \rightarrow YC_bC_r$  and work on  $C_b$  and  $C_r$  planes in an independent manner [6]. Then, in order to detect areas of potential stamps, we look for elementary image structures, like lines, circles, and any other consistent areas. Next, we obtain candidate areas which are later verified using a number of features that characterize specific shapes and classify them into stamps and no-stamps (objects with similar shape, yet with different raster features). The further recognition (in terms of shape) is performed using shape descriptors and dedicated classification methods.

Several experiments on color-based detection have also been described in our previous works [6, 25].

**2.1. Image Preprocessing.** It is known that  $RGB$  color space is not optimal in terms of color image segmentation due to the correlation between channels [6]. Hence we convert an input image into  $YC_bC_r$  representation. Since the color of stamps are mainly in the blue or red range (according to our observations, less than 4% of the stamps are represented by other colors), it is especially usable because of the red/blue separation in that color space. What is more, it is a native color format for JPEG/JFIF files. However, in general case, we assume that an input image of a document is stored in a file with possibly lossless compression, high spatial resolution, and full color range (24-bit  $RGB$ ). Hence, the input image is



FIGURE 3: Sample document containing a stamp (a),  $C_b$  (b) and  $C_r$  (c) components of its  $YCbCr$  representation together with the binarized output image (d).

converted into  $YCbCr$  color space (ITU-R BT.709 standard) in order to expose the above shown color properties:

$$\begin{aligned} Y &= 16 + (65.481 \cdot R + 128.553 \cdot G + 24.966 \cdot B), \\ C_b &= 128 + (-37.797 \cdot R - 74.203 \cdot G + 112 \cdot B), \\ C_r &= 128 + (112 \cdot R - 93.786 \cdot G - 18.214 \cdot B), \end{aligned} \quad (1)$$

where  $R$ ,  $G$ ,  $B$  and  $Y$ ,  $C_b$ ,  $C_r$  are appropriate color components.

According to the above observations, we examine each channel  $Y$ ,  $C_b$ , and  $C_r$  in an independent manner. (see Figure 3).

As a next preprocessing stage, above matrices are filtered using simple averaging mask in order to reduce noise. The filter uses a mask whose size is equal to 3% of input image's shorter edge. This ensures favorable results in the aspect of computation cost versus quality.

In the next step, each image is binarized and consistent areas are determined. Additionally, before filling holes, each area is subjected to morphological operation, region growing. Binary image is then labeled and each candidate region is passed to the module responsible for stamp detection (verification). Our previous experiments [6] showed also that stamps are objects that are not smaller than 5% of shorter edge of input image and not larger than 40% of its longer edge. Hence, we look for such areas only.

In the case presented below (Figure 3), several potential areas are detected and passed to the verification/classification

stage. Presented binarized image is obtained as a superposition of thresholded  $C_b$  and  $C_r$  channels; hence it contains areas of high blue and red channels intensity.

**2.2. Stamp Detection.** Different stamp shapes are detected by independent modules (see Figure 2). Firstly, the input image with dimensions of  $H_{\text{doc}} \times W_{\text{doc}}$  pixels is binarized using adaptive thresholding [26], so it contains a number of closed areas.

**2.2.1. Circular Shapes.** Detection of circles is based on Circular Hough Transform (CHT) performed on intensity image. At first, an image is binarized and then edge detection using Canny filtering is performed; then morphological opening in order to eliminate noise is employed. On such image, classical Hough Transform is used. Using a voting strategy, it finds circles of different radii. The details of the algorithm can be found, for example, in [27]. In our experiments we used radii from interval [15, 120]. Exemplary results of circular stamps detection are presented in Figure 4.

**2.2.2. Shapes Consisting of Straight Lines.** For the extraction of shapes containing straight lines (e.g., squares, rectangles, and triangles), we employ a line detector based on Hough Transform. It works on image containing pixel gradients. Firstly, the input image is averaged in order to remove noise. Then, we select pixels of high intensity (above a certain threshold) and for each one we perform the following sequence of operations. Each pixel is represented in the

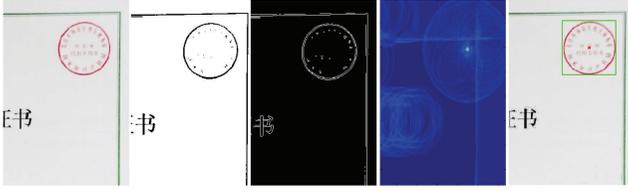


FIGURE 4: Sample document containing circular stamp (left), binarized, after edge detection, accumulator matrix and detection results, respectively.

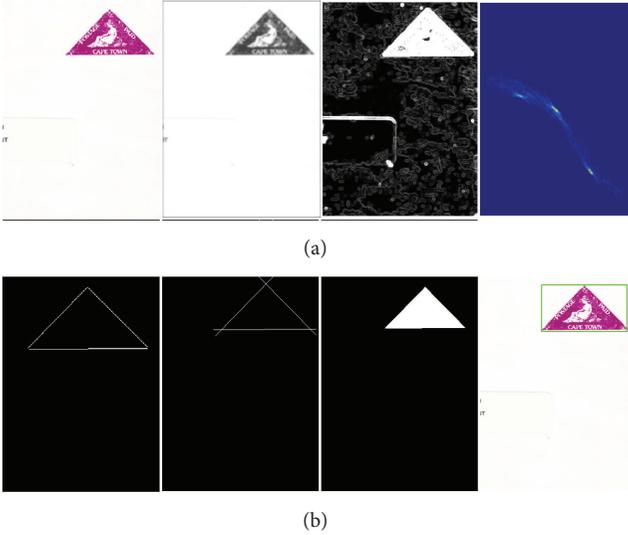


FIGURE 5: Sample document containing triangular stamp, blurred and grayscaled, after gradient detection and accumulator matrix in (a), respectively. Detected straight lines, lines extended to create closed area, filled area and the detection result in (b), respectively.

parametric Hough Space as a sinusoid. After checking all selected pixels we get a bunch of intersecting curves. The values at the intersections give an information about the possible lines in the original image. Since values of both slope and intercept in original Hough approach are unbounded and slope value for vertical lines is huge, application of this original technique is complicated. Hence we use an approach, presented in [28], which is an alternative to the original Hough approach.

Then we extend each detected line using 20 pixels at both of its ends in order to create closed area, which is later filled. Further we eliminate introduced extensions. Exemplary results of stamps containing straight lines in their contours are presented in Figure 5.

**2.2.3. Nonregular Shapes.** In order to extract any other shapes we introduce a heuristic approach presented below. It uses the results of our observations (based on the analysis of our documents database) and very basic algorithms from the image processing area. For each consistent area (blob) found in the image its bounding box is determined. Its area is given as  $A_{\text{box}} = H_{\text{box}} * W_{\text{box}}$ , where  $W_{\text{box}}$  and  $H_{\text{box}}$  are its width

and height, respectively. We assume that the background pixels are zeros; hence, additionally, we calculate the number of pixels (other than zero) in each area  $A_p$ . It gives us an information about complexity of each blob. Candidates that satisfy the following conditions are passed to the further processing:

$$400 < A_p < 5000,$$

$$\min \{H_{\text{box}}, W_{\text{box}}\} > 0.05 * \min \{H_{\text{doc}}, W_{\text{doc}}\}, \quad (2)$$

$$\max \{H_{\text{box}}, W_{\text{box}}\} < 0.40 * \max \{H_{\text{doc}}, W_{\text{doc}}\}.$$

As it can be seen, each object's area must fall into range (400, 5000) pixels. This constraint is not denominated by any relative value and is used just as a filter since other two conditions could easily select objects smaller or larger than expected.

The alternative way of detecting stamps, not taking into consideration their geometrical properties, was presented in [24]. However, that approach rejects longitudinal blobs and promotes objects of high contrast only.

**2.3. Stamp/No-Stamp Verification.** Extracted objects are verified using a set of 11 object's characteristics captured in the spatial domain. We compute seven direct features: average pixel intensity  $f_{\text{avg}}$ , intensity standard deviation  $f_{\text{std}}$ , median value of intensity, pixel contrast, brightness to contrast ratio, intensity of edges, brightness to edges intensity ratio, and four features calculated from Gray-Level Cooccurrence Matrix (GLCM) [29]: contrast, correlation, energy, and homogeneity.

Firstly, image intensity matrix  $I$  of size  $H_{\text{box}} \times W_{\text{box}}$  containing a single blob is converted into a vector  $f$  of length  $l = H_{\text{box}} W_{\text{box}}$ . Aforementioned features (aside from those which are self-explanatory) are calculated according to following formulas:

$$f_{\text{avg}} = \frac{1}{l} \sum_{i=1}^l f(i), \quad (3)$$

$$f_{\text{std}} = \sqrt{\frac{1}{l} \sum_{i=1}^l (f(i) - f_{\text{avg}})^2}. \quad (4)$$

Median value is calculated over sorted vector  $f$ , so that  $f(i) \leq f(i+1)$ ,  $i = 1 \dots l-1$ . The result is equal either to  $f((l+1)/2)$  or  $(1/2)(f(l/2) + f(l/2+1))$  depending on  $l$  parity. Calculation of remaining features depends on edge detection operator. In our case Sobel algorithm was applied [30]. Intensity of edges is then computed in accordance with (3).

GLCM matrix of size  $P \times Q$  is created by calculating how often a pixel of intensity  $p$  has a pixel of intensity  $q$  in its closest horizontal neighbourhood. However, this could be changed based on  $\Delta x$ ,  $\Delta y$  offset parameter. For a discrete image,  $p = 1, \dots, P$  and  $q = 1, \dots, Q$ . Its size is determined by number of gray-levels; in this case both  $P$  and  $Q$  are equal to 8. It also means that all values in image matrix  $I$  are scaled

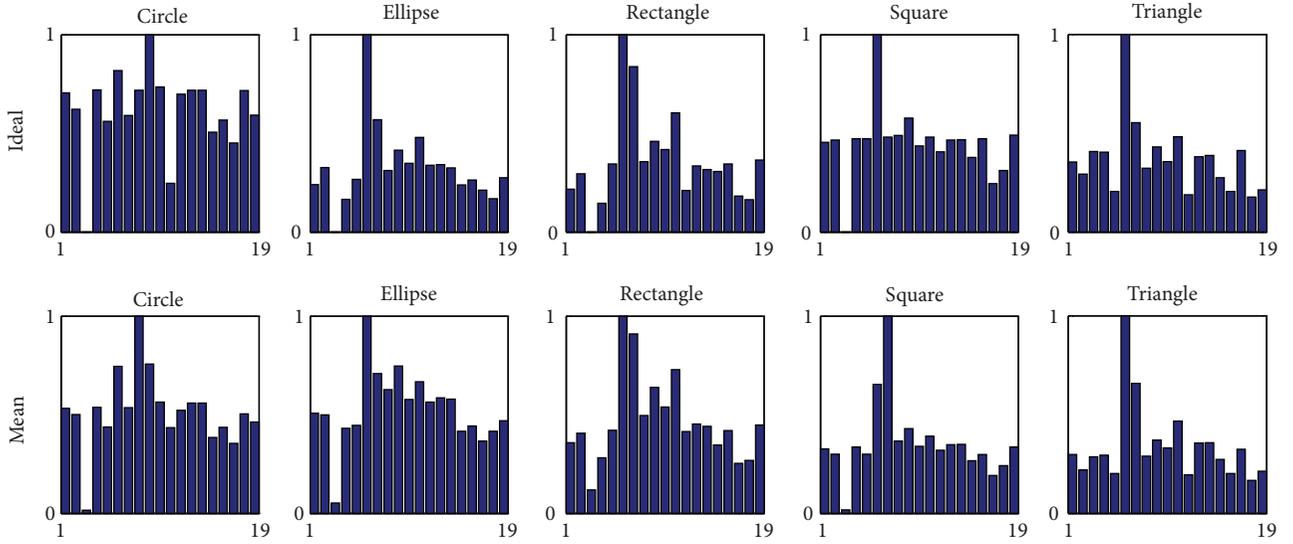


FIGURE 6: Features for each of five distinctive classes.

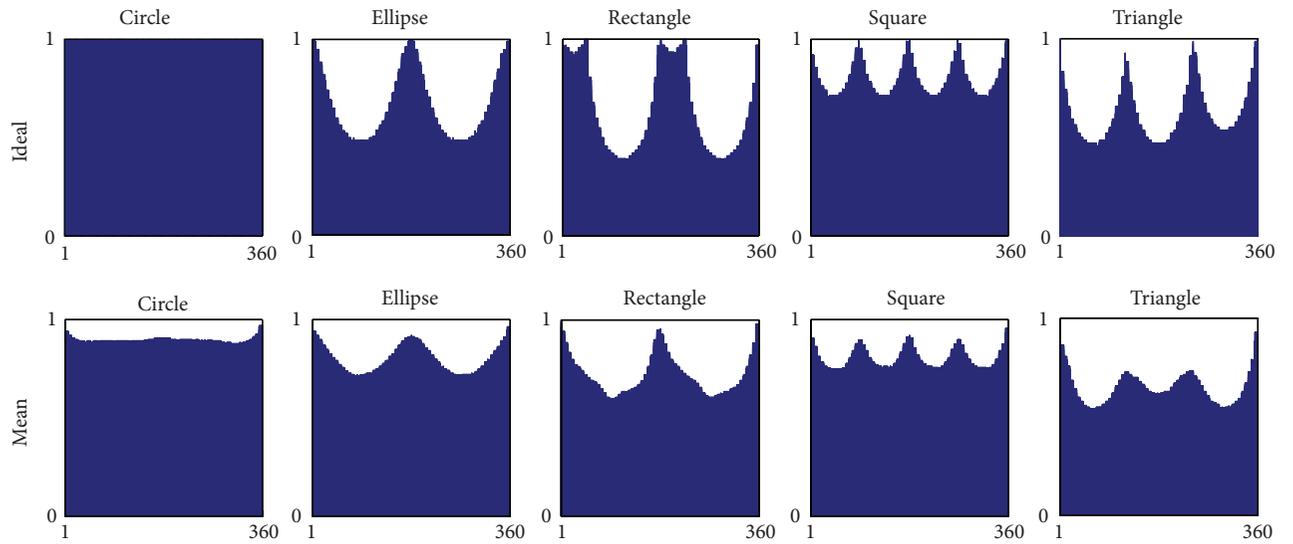


FIGURE 7: Shape signatures of five distinctive classes.

to [1, 8] interval. The GLCM is calculated according to the following formula [31]:

$$\begin{aligned} & \text{GLCM}_{\Delta x, \Delta y}(p, q) \\ &= \sum_{x=1}^M \sum_{y=1}^N \begin{cases} 1, & \text{if } I(x, y) = p, I(x + \Delta x, y + \Delta y) = q \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (5)$$

The contrast of GLCM is given as [31]

$$\text{GLCM}_C = \sum_p \sum_q (p - 1)^2 \text{GLCM}(p, q). \quad (6)$$

The energy is calculated as [31]

$$\text{GLCM}_E = \sum_p \sum_q \text{GLCM}(p, q)^2. \quad (7)$$

The homogeneity is given as [31]

$$\text{GLCM}_H = \sum_p \sum_q \frac{\text{GLCM}(p, q)}{1 + |p - q|}. \quad (8)$$

The correlation of GLCM is calculated as [31]

$$\text{GLCM}_H = \sum_p \sum_q \frac{(p - \mu_p)(q - \mu_q) \text{GLCM}(p, q)}{\sigma_p \sigma_q}, \quad (9)$$

where  $\mu_p$ ,  $\sigma_p$  and  $\mu_q$ ,  $\sigma_q$  are mean and standard deviation calculated in row and column directions, respectively.

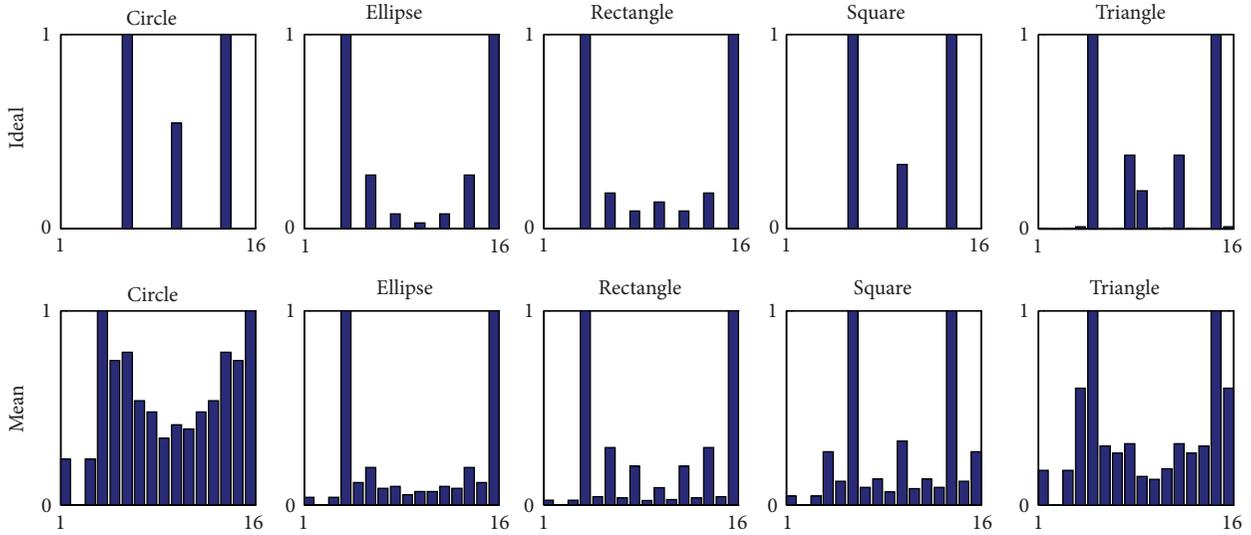


FIGURE 8: Fourier descriptors of five distinctive classes.

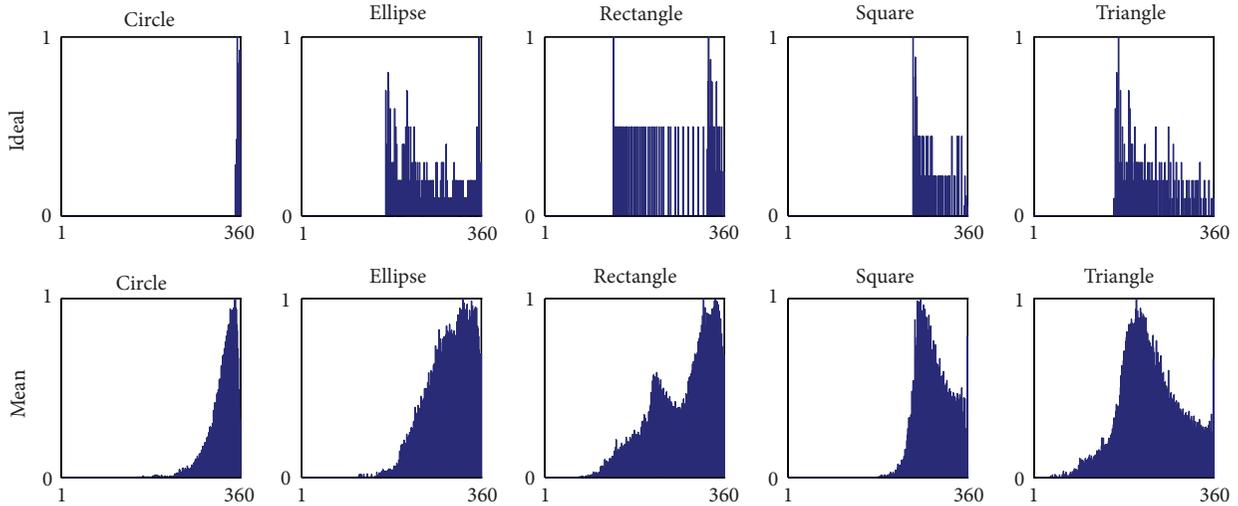


FIGURE 9: Point distance histograms of five distinctive classes.

### 3. Stamps Shape Features

In this paper we focus on the low-level image features employed in classification as it seems to be, according to our opinion, the most challenging problem. Such low-level properties can be easily computed and stored in limited memory. This two criteria are crucial when it comes, for example, for hardware implementation. Moreover, features presented below were selected, since they showed their abilities in other computer vision tasks.

In case of stamps, shape is one of the most valuable properties. Shape of stamp used for recognition can be considered as a binary object stored in a matrix, which can be represented by a specific number of points, including its interior or as a boundary (outer contour). Compact representation of shape is often known as shape descriptor. It is crucial for the recognition or classification to uniquely characterize shape

and stay invariant to as many transformations as possible (i.e., translation, scaling, rotation, presence of noise, and occlusion). These distortions are considered as differences between object under recognition and the reference object belonging to the same class, stored in a database. In practical recognition of stamps one has to take into consideration the following distortions divided into three main categories. The first one includes spatial transformations of an object, mainly translation, rotation in the image plane, and change of scale. The second category includes distortions introduced by imaging system, for example, variable number of captured points, presence of noise, discontinuity of contour, and occlusion. The third category of problems comes from contour representation and contour evaluation. The elements of the second group are the most challenging and difficult to solve.

Shape descriptors can be classified in various ways. The first taxonomy is based on mentioned earlier difference



FIGURE 10: Sample scanned documents used for the performance evaluation.

between object boundary and the whole shape. The second very popular classification (as described in [32]) divides descriptors into global approaches (shape represented as a whole) or structural methods (set of primitives). The third one discriminates spatial and transform domains [13]. Since shape is one of the most important features in problems related to content-based image retrieval, there are many known shape representations and retrieval methods; however, most of those methods neither represent shape in a sufficiently precise manner nor are relatively easy in matching. Among them, methods based on moments, polar coordinate system representation, and histogram of distances from the centroid achieve both good representation and easy normalization.

As it was mentioned earlier, there are two main classes of shape descriptors which capture different features, namely, region-based shape representation and contour-based ones. In region based techniques, all pixels within a shape are taken into consideration to obtain final shape representation. Most of region-based methods employ different variants of moments calculation. Contour-based shape representation

exploits shape boundary information. Such methods can be classified into global shape descriptors, shape signatures, and spectral descriptors. Although global descriptors such as area, circularity, eccentricity, and axis orientation are simple to compute and also robust in representation, they can only discriminate shapes with large dissimilarities and therefore are usually suitable for filtering purpose. Most shape signatures such as complex coordinates, curvature, and angular representations are essentially local representations of shape features; they are sensitive to noise and not robust. In addition, shape representation using shape signatures requires intensive computation during similarity calculation, due to the hard normalization of rotation invariance. As a result, these representations need often further processing. On the other hand, there are plenty of methods (e.g., MPEG-7 descriptors [33]) that are reported to be very efficient in shape retrieval. Unfortunately, their computational complexity is not justified in the task addressed in this paper.

**3.1. Simple Scalar Features.** In the shape analysis module we calculate several elementary measures that are later evaluated

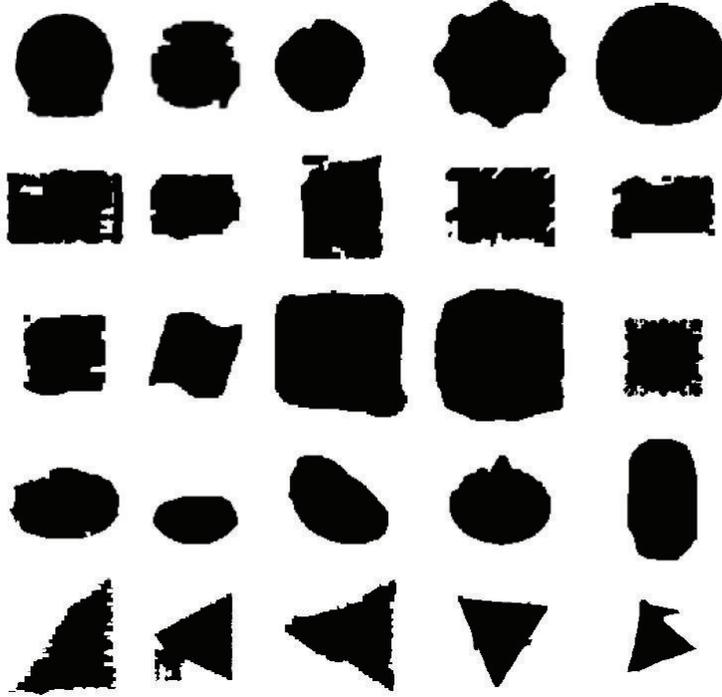


FIGURE 11: Sample shapes used for the performance evaluation: circle, rectangle, square, ellipse, and triangle (in rows, resp.).

at the decision stage. These simple properties are later concatenated in a single vector having 19 elements [34]. All of them use the following measures as common values ( $M_1$ : minor axis length,  $M_2$ : major axis length,  $A_p$ : object area defined by the number of pixels, and  $P$ : object perimeter), which are later employed to build more complex characteristics. Above characteristics were chosen intentionally, since they give a maximal discriminative power in terms of shape classification. All of them are normalized to the interval  $[0, 1]$ . It should be also noted that in case of a bounding box (associated with an analysed blob), that is not rotated,  $M_1$  is equal to  $\min(W_{\text{box}}, H_{\text{box}})$  and  $M_2$  is equal to  $\max(W_{\text{box}}, H_{\text{box}})$ .

**3.1.1. Roundness.** It is computed as an average value of the three following measures  $R_1$ ,  $R_2$ , and  $R_3$ :

$$\begin{aligned} R_1 &= \left| 1 - \frac{|M_2 - M_1|}{\max(M_1, M_2)} \right|, \\ R_2 &= \left| 1 - \frac{|\pi((M_1 + M_2)/2)^2/2 - A_p|}{A_p} \right|, \\ R_3 &= \left| 1 - \frac{|\pi \max(M_1, M_2) - P|}{P} \right|, \end{aligned} \quad (10)$$

where  $R_1$  expresses normalized difference between longest and shortest diameter;  $R_2$  incorporates information about object's area measured in two different ways, as a number of pixels belonging to the area and calculated according to geometrical formula;  $R_3$  is analogous to the  $R_2$  measure but perimeter is used instead of area.

**3.1.2. Squareness.** Squareness, as a measure of similarity between analysed object and a perfect square, is calculated according to the following formula:

$$S = \left( \frac{\min(\sqrt{A_p}, P/4)}{\max(\sqrt{A_p}, P/4)} \right)^2. \quad (11)$$

**3.1.3. Number of Vertices.** It is calculated as a function of object's extreme points (in terms of geometry); that is, if an object contains three extremes, then it is considered to be a triangle. The observations show, however, that binary objects extracted from real documents are often noised or distorted; hence finding extremes is very difficult or the number of detected extremes can differ from the actual one. In order to count the extremes, an input object is binarized, and then we select all potential extreme points (their coordinates) and create a new binary matrix containing zeros. Next, we put ones in the cells that match found coordinates. Further we cluster the points to their groups based on their spatial location. At this stage we perform a dilation procedure in order to fill gaps and create groups of neighboring pixels. Then, based on those groups, the number of extremes is computed.

**3.1.4. Aspect Ratio.** It is a very elementary property that is calculated as the proportion of object's width and height (in the similar manner to the one presented in Section 2.2):

$$R = \frac{M_1}{M_2}. \quad (12)$$



FIGURE 12: Sample stamps correctly recognized at the shape analysis stage.



FIGURE 13: Sample stamps misclassified at the shape analysis stage.

**3.1.5. Extent.** This property represents a ratio of the number of pixels inside a rectangle that is circumscribed around an object to the total number of pixels belonging to the object itself:

$$E_X = \frac{H_{\text{box}} W_{\text{box}}}{A_p}. \quad (13)$$

**3.1.6. Moment-Based Properties.** Further measures that are later employed include ellipticity, elliptic variance, circular variance, triangularity coefficient (based on central moments), and minimal bounding figure, a coefficient defining a ratio between the area of a smallest shape that is wrapped around the object and this object's area. They involve the use of moment invariants  $\mu_{pq}$  (of second order), where each contour point  $i \in 1, \dots, N$  is described by its coordinates  $p_i = (x, y)$ . The centroid is calculated as a mean of all points coordinates:  $\mu = (1/N) \sum_i p_i$ , and the mean radius is equal to  $\mu_r = (1/N) \sum_i \|p_i - \mu\|$ .

*Ellipticity* proposed in [35] is calculated using the following formula:

$$I_1 = \frac{\mu_{20}\mu_{02} - \mu_{11}^2}{\mu_{00}^2}. \quad (14)$$

The moment for a unit radius circle allows computing  $I_1 = 1/16\pi^2$  for a perfect figure. Ellipticity value ranges over  $[0, 1]$  and is computed as follows [35]:

$$E_I = \begin{cases} 16\pi^2 I_1 & \text{if } I_1 \leq \frac{1}{16\pi^2} \\ 1 & \text{otherwise.} \end{cases} \quad (15)$$

*Circular variance*, as a similarity between an analysed object and a circle, was described in [36]. It is calculated using the following formula:

$$C_{\text{var}} = \frac{1}{N\mu_r^2} \sum_i (\|p_i - \mu\| - \mu_r)^2. \quad (16)$$

*Elliptic variance* responsible for describing ellipses employs the following formula [36]:

$$E_{\text{var}} = \frac{1}{N\mu_{rC}} \sum_i \left( \sqrt{(p_i - \mu)^T C^{-1} (p_i - \mu)} - \mu_{rC} \right)^2, \quad (17)$$

where  $C = (1/N) \sum_i (p_i - \mu)(p_i - \mu)^T$  is a covariance matrix, while  $\mu_{rC} = (1/N) \sum_i \sqrt{(p_i - \mu)^T C^{-1} (p_i - \mu)}$ .

*Triangularity* is based on moment invariants [35] and for an ideal right-angled triangle is equal to  $I_1 = 1/108$ ; thus

the triangularity of an analysed object is computed as follows (again, giving values from the interval  $[0, 1]$ ):

$$T_I = \begin{cases} 108I_1 & \text{if } I_1 \leq \frac{1}{16\pi^2} \\ \frac{1}{108I_1} & \text{otherwise.} \end{cases} \quad (18)$$

*Minimal bounding figure coefficient* is calculated as a ratio of object's actual area to the area of an ideal figure that is circumscribed on it. It is called minimal bounding figure and the method of calculation is similar to [35]. The algorithm checks the similarity between each perfect figure (one of the predefined five classes) and an analysed object.

**3.1.7. Other Properties.** In the shape analysis we employ also a coefficient  $A_p$  related to the object's area understood as the number of pixels belonging to it. The following group of geometrical features is calculated on the basis of object's area, its shortest radius (span)  $R_s$  and its longest radius (span)  $R_l$ . If the difference  $S_{\text{cir}} = R_s - R_l$  is small enough (in our case, lower than 10 pixels), we assume that the object has circular shape. On the other hand, an object is considered to be a square if a value of  $S_{\text{sqr}}$  from the following equation is close to one:

$$S_{\text{sqr}} = \frac{A_p}{4R_s^2}. \quad (19)$$

An ellipse is described as a coefficient:

$$S_{\text{eli}} = \frac{A_p}{\pi R_s R_l}. \quad (20)$$

If it is close to one, then the object is recognized as an ellipse. In the same manner, the coefficients related to other shapes, like rectangle, triangle, and rhombus (which, for classification purposes, is treated as a square) can be calculated, respectively, as

$$\begin{aligned} S_{\text{rct}} &= \frac{A_p}{4R_s \sqrt{R_l^2 - R_s^2}}, \\ S_{\text{tri}} &= \frac{A_p \sqrt{3}}{(R_l + R_s)^2}, \\ S_{\text{rho}} &= \frac{A_p \sqrt{R_l^2 - R_s^2}}{2R_l^2 R_s}. \end{aligned} \quad (21)$$

All of above scalar features are joined into single vector and further used at the classification stage. The feature values for ideal shapes (circle, ellipse, rectangle, square, and triangle) and mean values for shapes from our experimental database are presented in Figure 6. In each plot, the horizontal axis is related to the feature number, while the vertical axis is related to the feature value. The ideal shapes were extracted from images created in painting software, while the mean shapes are averaged objects from our database.

**3.2. Vector Features.** As for the comparison we selected three popular vector shape descriptors: shape signature, Fourier descriptor, and point distance histogram. All of them have been successfully applied in many shape recognition tasks; hence the comparison is justified.

**3.2.1. Shape Signature.** Shape signature (SSig) is one of the most popular representations that belong to the contour-based class of descriptors. There are several variants of SSig which employ different features [32]. Here, we use the so-called Centroid Distance Function (CDF). It is easy to calculate and after some normalization can be invariant to scaling and rotation. However it should be noted that, in addition to the high matching cost, shape signatures are sensitive to noise, and slight changes in the boundary can cause large errors in matching. Hence, shape signatures should be stored in a reduced form. In this paper we calculate SSig according to the following algorithm.

- (1) Calculate the centroid of an object.
- (2) Detect outer contour of object and store its coordinates in a polar system of coordinates  $(r, t)$ .
- (3) Find the maximal distance  $r$  and perform circular shift so the  $t$  related to this maximum occupies the first position in the vector.
- (4) Discard the information about  $t$ ; hence remember only distances  $r$ .
- (5) Normalize the vector to the maximal value of  $r$ .
- (6) Interpolate the vector containing  $r$  to the final length of  $b$  elements.

The shape signature feature vectors with 360 bins for ideal shapes (circle, ellipse, rectangle, square, and triangle) and mean vectors for shapes from our experimental database are presented in (Figure 7). In each plot, the horizontal axis is related to the angle between radius vector and the horizontal direction, while the vertical axis is related to the distance from the origin. The descriptors for ideal shapes were extracted from images created in painting software, while the mean shapes are averaged objects from our database.

**3.2.2. Fourier Descriptor.** There is a whole family of descriptors called Fourier descriptors (FDs). Different shape signatures have been exploited to derive such descriptors. It should be noticed that FDs derived from different signatures can have significantly different effect on the result of retrieval [37]. In this paper we calculate FD according to the following algorithm.

- (1) Calculate the centroid of an object.
- (2) Detect outer contour of object and store its coordinates  $(x, y)$  as complex numbers, where  $x$  coordinate is a real part and  $y$  coordinate is an imaginary part.
- (3) Perform fast Fourier transform (FFT) on these values.
- (4) Normalize FFT spectrum to its maximal magnitude.
- (5) Remember first  $b$  elements (related to low frequency components).

TABLE 1: Classifiers comparison in terms of stamp/no-stamp verification.

Classifier	TP rate	FP rate	Classifier	TP rate	FP rate
NB	91.8%	49.6%	TR	79.0%	21.9%
LLR	84.1%	37.7%	CART	82.9%	23.8%
MLP	86.5%	26.9%	$k$ NN, $k = 1$	83.9%	16.0%
SVM	45.6%	54.6%	$k$ NN, $k = 3$	84.9%	21.3%

The Fourier descriptor feature vectors with 16 bins for ideal shapes and mean vectors for our experimental database are presented in (Figure 8). In each plot, the horizontal axis is related to the spectral component, while the vertical axis is related to its amplitude. The descriptors for ideal shapes were extracted from images created in painting software, while the mean objects were averaged over the whole database.

**3.2.3. Point Distance Histogram.** The point distance histogram (PDH) is a histogram of contour point distances represented in the polar system of coordinates. The algorithm of calculating PDH is as follows [21].

- (1) Calculate the centroid of an object.
- (2) Detect outer contour of object and store its coordinates in a polar system of coordinates  $(r, t)$ .
- (3) Discard the information about  $t$ ; hence remember only distances  $r$ .
- (4) Calculate the histogram  $H$  of distances  $r$  with  $b$  bins.
- (5) Normalize  $H$  to the maximal value.

The PDH feature vectors with 360 bins for ideal shapes and mean vectors for our experimental database are presented in (Figure 9). In each plot, the horizontal axis is related to the distance (or distance interval), while the vertical axis is related to the number of points within such distance. As above, the descriptors for ideal shapes were calculated for images created in painting software, while the mean objects are averaged over the whole our database.

## 4. Experiments

**4.1. Experimental Setup.** The experiments were aimed at the evaluation of the performance of both stages of processing, namely, the stamp detector and stamp's shape classifier. Since there are no benchmark databases oriented at this specific problem, the experiments were performed on our own benchmark database consisting of a number of images collected from the Internet. It contains scanned or photographed documents of different origin and variable quality. The details of this database are as follows: documents with no stamps: 294 (41%), with single stamp: 309 (43%), with multiple stamps: 116 (16%), and with logotypes: 367 (51%). Exemplary documents gathered in the database are presented in Figure 10.

**4.2. Stamp/No-Stamp Verification.** The experiments on stamp verification were performed on 2925 graphical objects extracted from the above presented documents. There are 1589 stamps and 1336 no-stamps (logos, ornaments, and

TABLE 2: Confusion matrix with True Positive and False Positive indicators for INN classifier.

Analysed	Predicted		False Positives	True Positives
	Stamps	No-stamps		
Stamps	1326	263	15.4%	83.4%
No-stamps	206	1130	16.6%	84.6%

large letters) in the set. They were divided into learning and testing subsets according to the 10-fold cross-validation.

In order to select a most suitable classifier we performed several experiments involving several state-of-the-art approaches, namely, Naive Bayes Classifier (NBC), Linear Logistics Regression (LLR), Multilayer Perceptron (MLP), Support Vector Machines (SVM), Classification and Regression Tree (CART), Random Tree (RT), and  $k$ -Nearest Neighbors ( $k$ NN) with  $k = 1$  and  $k = 3$ . The rates of True Positive and False Positive for *Stamp* class are given in Table 1. The analysis of the results showed that the accuracy of SVM is significantly lower than any other classifiers (probably due to high variance). On the other hand, the high True Positive rates for NB and MLP are accompanied by also high False Positive rates. Based on these results at the stage of stamp/no-stamp verification, we selected  $k$ NN with  $k = 1$  as a method with the lowest FP rate. During experiments employing INN, the algorithm was able to recognize 2456 objects, while the rest 469 were misclassified. For such experimental setup, the detailed results are presented in Table 2.

**4.3. Shape Classification.** We performed a shape classification on a database created from stamps extracted from the aforementioned documents. The set consists of 2101 binary objects. Exemplary objects are presented in Figure 11. For each object we calculated a scalar-based feature vector of 19 properties and three vector descriptors shown above. In order to select the most discriminative method we tested several modern classifiers and, as for comparison, simple voting classifier. The later uses simple scalar weights equal to one for all simple features, except the following, which have weights equal to 3:  $R$ ,  $S_{\text{sq}}$ ,  $S_{\text{rec}}$ ,  $S_{\text{eli}}$ ,  $T_I$ , minimal bounding triangle, minimal bounding circle, and minimal bounding square (from the group of minimal bounding figure). In Table 3 one can observe a comparison of different classifiers versus selected features. The column *Simple* contains the results of simple shape features described in Section 3.1.

As it can be observed, the performance of all classifiers in case of simple scalar features is superior to selected vector features. It is also visible that the classifiers ensembles based

TABLE 3: Classification performance [%].

Classifier	Simple	SSig	FD	PDH
Bayes network (K2 learning rule)	<b>91.9</b>	83.3	87.4	81.7
MLP (single hidden layer)	<b>95.7</b>	94.7	94.1	88.9
SVM (Sequential Minimal Optimization, polynomial kernel)	<b>95.0</b>	84.6	91.7	82.1
INN (1-Nearest Neighbor, Euclidean Distance)	<b>96.6</b>	95.5	95.2	89.9
KStar	<b>97.0</b>	95.1	94.0	90.3
Bagging (Fast Decision Tree Learner)	<b>95.4</b>	91.6	93.5	89.0
RandomCommittee (RandomTree Classifiers)	<b>97.7</b>	95.9	95.4	92.1
RotationForest (C4.5 Decision Tree, Principal Components Analysis)	<b>97.3</b>	96.1	95.6	90.4
Nearest Neighbor with Generalization	<b>96.4</b>	91.8	93.8	89.5
RandomForest (10 trees)	<b>97.4</b>	95.0	95.1	90.4

TABLE 4: Confusion matrix for shape analysis module based on a Committee of RandomTree Classifiers.

Analysed shape	Predicted					Class accuracy
	Circle	Ellipse	Square	Triangle	Rectangle	
Circle	434	9	3	0	0	97%
Ellipse	14	400	0	0	3	96%
Square	2	0	393	0	7	98%
Triangle	0	0	0	321	0	100%
Rectangle	0	4	6	1	504	98%

TABLE 5: The performance of the whole algorithm.

	Detected	Missed	Accepted (TP)	Rejected (FN)
Stamp	417/70.44%	175/29.56%	313/75.06%	104/24.94%
	Detected	Missed	Accepted (FP)	Rejected (TN)
No-stamp	377	—	129/34.22%	248/65.78%

on nearest neighbours can give significantly higher accuracy than any other approach.

Table 4 shows the confusion matrix for shape classification in terms of particular types of stamps. The detailed results show that the highest confusion is observable for shapes pairs, for example, circle versus ellipse and square versus rectangle. It comes from the fact that the boundaries of those pairs of classes may be fuzzy.

**4.4. Overall Performance.** In the final experiment we wanted to evaluate the overall performance of the proposed approach. It included both stages: stamp detection and shape classification. It was performed on a set of 719 scanned or photographed documents. The results are presented in Table 5.

Tables 4 and 5 prove that the proposed set of algorithms is able to detect and classify most of stamps. A considerable number of missed stamps comes from the fact that the database consists of many problematic cases, that is, stamps which are hardly visible, only partially imprinted, or overlap objects such as pictures, post stamps, or printed text (see examples in Figures 12 and 13).

A special case is stamps containing small tables or just texts. It is almost impossible to distinguish them from actual

tables or parts of text. Hence, it leads to the further misclassification. The decrease in classification performance is caused also by the fact that the database contains documents completely without stamps, documents with objects similar to stamps, or actual stamps that do not meet conditions specified in Section 2.2. As it can be seen from the observations, a large part of the original documents has a low contrast. Moreover, many wrongly detected areas (later rejected by the classifier) are simply the artifacts coming from strong compression of the input image.

**4.5. Comparison with State-of-the-Art.** We compared our approach with some of the works presented in the literature. Since it was impossible to repeat all the experiments presented in the papers, we took the results provided by the authors. They are gathered in Table 6. Unfortunately, not all details of compared algorithms could be obtained. Such cases are marked with a “—” sign. The comparison is divided into three areas: stamp detection, stamp verification, and stamp shape classification. In our comparison we included the numbers of input documents, verified objects (stamps/no-stamps), and extracted stamps. Actually, not all elements can be properly compared, since as for now a comprehensive method has not been presented and the methods given in the literature focus on individual actions only.

A full comparison of stamp detection is hard to perform, since several other methods allow detecting only color stamps. The reported accuracy of detection of our method is lower; however our database is the largest and we detect stamps not taking into account their color or overlapping. On the other hand, the actual accuracy of method presented in [3] is in fact equal to 69%.

TABLE 6: Comparison with other approaches, detect refers to detection stage, verify refers to verification stage, and shape refers to shape analysis stage.

	Indicator	Algorithm					
		GSA [6]	PDH & DCT [25]	[3]	GHT & CPG [4]	[5]	Our
Detect	Number of doc.	170	170	400	—	127	719
	Accur. [%]	82	82	83.4 (69)	—	—	70.44
Verify	Number of obj.	—	—	—	—	—	1326
	Accur. [%]	—	—	83.8	—	—	83.4
Shape	Number of stamps	—	140	—	220	—	2101
	Accuar. [%]	93	77	—	98.58	92.03	97.7

As it can be noticed, no method found in the literature performs an appropriate verification step (discrimination between stamps and no-stamps). Our method employs a set of image features to distinguish stamps from objects like logotypes, small tables, and other graphical ornaments.

Finally, our shape recognition algorithm was tested on the largest database among all methods and gave a very competitive results.

## 5. Summary

In the paper we propose a novel approach for detecting and classifying stamp instances in scanned documents. To our best knowledge, this is the first comprehensive algorithm that deals with color and monochrome stamps of any shape. It incorporates several methods of image processing, pattern recognition, and some heuristic. The algorithm is multistage, consisting of object detection, stamp verification, and shape classification. During experiments we selected a set of low-complexity features that are invariant to stamp's color, orientation, scale, and position in the document; hence this approach is independent of the document size and orientation. Such basic low-level features containing elementary shape properties occurred to be superior to several, more complex descriptors. The results of experiments showed also that at the stage of stamp/no-stamp verification the best results gave simple INN classifier and at the stage of shape classification, classifiers ensembles utilizing Random Trees. The experiments performed on a large number of real scanned and photographed documents show that the performance is sufficient for practical implementation. Future works may include the investigations on the additional features of stamps, which can give an increase in the overall robustness, especially at the stage of stamp/no-stamp verification. Above approach could also be employed for recognizing different parts of documents (e.g., texts, tables, signatures, logotypes, etc.). The potential area of application of this algorithm includes office software for processing paper documents, content-based document retrieval systems, and various postal services. It could be also employed in cases, when stamps should be automatically covered in order to preserve privacy.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] K. Ueda and Y. Nakamura, "Automatic verification of seal impression patterns," in *Proceedings of the 7th International Conference on Pattern Recognition.*, pp. 1019–1021.
- [2] S. Dey, J. Mukherjee, S. Sural, and P. Bhowmick, "Colored rubber stamp removal from document images," in *Pattern Recognition and Machine Intelligence*, vol. 8251 of *Lecture Notes in Computer Science*, pp. 545–550, Springer, Berlin, Germany, 2013.
- [3] B. Mícenková and J. van Beusekom, "Stamp detection in color document images," in *Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR '11)*, pp. 1125–1129, Beijing, China, September 2011.
- [4] P. P. Roy, U. Pal, and J. Lladós, "Document seal detection using GHT and character proximity graphs," *Pattern Recognition*, vol. 44, no. 6, pp. 1282–1295, 2011.
- [5] P. P. Roy, U. Pal, and J. Lladós, "Seal detection and recognition: an approach for document indexing," in *Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR '09)*, pp. 101–105, July 2009.
- [6] P. Forczmański and D. Frejlichowski, "Robust stamps detection and classification by means of general shape analysis," in *Computer Vision and Graphics*, vol. 6374 of *Lecture Notes in Computer Science*, pp. 360–367, Springer, Berlin, Germany, 2010.
- [7] L. Cai and L. Mei, "A robust registration and detection method for color seal verification," in *Proceedings of the International Conference on Intelligent Computing (ICIC '05)*, pp. 97–106, August 2005.
- [8] S. Dickinson and Z. Pizlo, *Shape Perception in Human and Computer Vision, An Interdisciplinary Perspective*, Advances in Computer Vision and Pattern Recognition, Springer, London, UK, 2013.
- [9] M. del Pozo-Baños, J. Ticay-Rivas, J. Cabrera-Falcon et al., "Image processing for pollen classification," in *Biodiversity Enrichment in a Diverse World*, G. A. Lameed, Ed., InTech, Rijeka, Croatia, 2012.
- [10] M. Rodríguez-Damián, E. Cernadas, A. Formella, and P. Sá-Otero, "Pollen classification using brightness-based and shape-based descriptors," in *Proceedings of the 17th International*

- Conference on Pattern Recognition (ICPR '04)*, pp. 212–215, August 2004.
- [11] D. Frejlichowski, “An algorithm for the automatic analysis of characters located on car license plates,” in *Image Analysis and Recognition*, M. Kamel and A. Campilho, Eds., vol. 7950 of *Lecture Notes in Computer Science*, pp. 774–781, Springer, Berlin, Germany, 2013.
- [12] G. McNeill and S. Vijayakumar, “2D shape classification and retrieval,” in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI '05)*, pp. 1483–1488, August 2005.
- [13] B. M. Mehre, M. S. Kankanhalli, and W. F. Lee, “Shape measures for content based image retrieval: a comparison,” *Information Processing and Management*, vol. 33, no. 3, pp. 319–337, 1997.
- [14] J. He and A. C. Downton, “Configurable text stamp identification tool with application of fuzzy logic,” in *Document Analysis Systems VI*, vol. 3163 of *Lecture Notes in Computer Science*, pp. 134–151, Springer, Berlin, Germany, 2004.
- [15] S. Seiden, M. Dillencourt, S. Irani, R. Borrey, and T. Murphy, “Logo detection in document images,” in *Proceedings of the International Conference on Imaging Science, Systems, and Technology*, pp. 446–449, 1997.
- [16] G. Zhu and D. Doermann, “Automatic document logo detection,” in *Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR '07)*, pp. 864–868, September 2007.
- [17] R. Jain and D. Doermann, “Logo retrieval in document images,” in *Proceedings of the 10th IAPR International Workshop on Document Analysis Systems (DAS '12)*, pp. 135–139, March 2012.
- [18] K. Jeong and H. Moon, “Object detection using FAST corner detector based on smartphone platforms,” in *Proceedings of the 1st ACIS/JNU International Conference on Computers, Networks, Systems, and Industrial Engineering (CNSI '11)*, pp. 111–115, May 2011.
- [19] J. Kleban, X. Xie, and W.-Y. Ma, “Spatial pyramid mining for logo detection in natural scenes,” in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '08)*, pp. 1077–1080, Hannover, Germany, June 2008.
- [20] S. Hassanzadeh and H. Pourghassem, “A novel logo detection and recognition framework for separated part logos in document images,” *Australian Journal of Basic and Applied Sciences*, vol. 5, no. 9, pp. 936–946, 2011.
- [21] D. Frejlichowski, “An experimental comparison of seven shape descriptors in the general shape analysis problem,” in *Image Analysis and Recognition—ICIAR 2010, Part I*, A. Campilho and M. Kamel, Eds., vol. 6111 of *Lecture Notes in Computer Science*, pp. 294–305, Springer, Berlin, Germany, 2010.
- [22] T. D. Pham, “Unconstrained logo detection in document images,” *Pattern Recognition*, vol. 36, no. 12, pp. 3023–3025, 2003.
- [23] G. Zhu, S. Jaeger, and D. Doermann, “A robust stamp detection framework on degraded documents,” in *Proceedings of the International Conference on Document Recognition and Retrieval XIII*, pp. 1–9, 2006.
- [24] D. Frejlichowski and P. Forczmański, “General shape analysis applied to stamps retrieval from scanned documents,” in *Artificial Intelligence: Methodology, Systems, and Applications: 14th International Conference, AIMSA 2010, Varna, Bulgaria, September 8–10*, vol. 6304 of *Lecture Notes in Computer Science*, pp. 251–260, Springer, Berlin, Germany, 2010.
- [25] P. Forczmański and D. Frejlichowski, “Efficient stamps classification by means of point distance histogram and discrete cosine transform,” in *Computer Recognition Systems 4*, vol. 95 of *Advances in Intelligent and Soft Computing*, pp. 327–336, Springer, Berlin, Germany, 2011.
- [26] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [27] J. Illingworth and J. Kittler, “A survey of the hough transform,” *Computer Vision, Graphics, and Image Processing*, vol. 44, no. 1, pp. 87–116, 1988.
- [28] R. O. Duda and P. E. Hart, “Use of the Hough transformation to detect lines and curves in pictures,” *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.
- [29] R. M. Haralick, K. Shanmugam, and I. Dinstein, “Textural features for image classification,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 3, no. 6, pp. 610–621, 1973.
- [30] I. Sobel and G. Feldman, “A  $3 \times 3$  isotropic gradient operator for image processing,” Presentation to Stanford Artificial Intelligence Project 01/1968, 1968.
- [31] F. Albrechtsen, “Statistical texture measures computed from gray level cooccurrence matrices,” Image Processing Laboratory, Department of Informatics, University of Oslo, pp. 1–14, 1995.
- [32] D. Zhang and G. Lu, “Review of shape representation and description techniques,” *Pattern Recognition*, vol. 37, no. 1, pp. 1–19, 2004.
- [33] M. Bober, “MPEG-7 visual shape descriptors,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 716–719, 2001.
- [34] P. Forczmański and A. Markiewicz, “Low-Level image features for stamps detection and classification,” *Advances in Intelligent Systems and Computing*, vol. 226, pp. 383–392, 2013.
- [35] P. L. Rosin, “Measuring shape: ellipticity, rectangularity, and triangularity,” *Machine Vision and Applications*, vol. 14, no. 3, pp. 172–184, 2003.
- [36] M. Peura and J. Iivarinen, “Efficiency of simple shape descriptors,” in *Aspects of Visual Form*, pp. 443–451, World Scientific, 1997.
- [37] D. Zhang and G. Lu, “A Comparative Study on Shape Retrieval using fourier descriptors with different shape signatures,” in *Proceedings of the 5th Asian Conference on Computer Vision (ACCV '02)*, pp. 646–651, 2002.

## Research Article

# Multiscale Hybrid Nonlocal Means Filtering Using Modified Similarity Measure

Zahid Hussain Shamsi<sup>1,2</sup> and Dai-Gyoung Kim<sup>1</sup>

<sup>1</sup> Department of Applied Mathematics, Hanyang University (ERICA), Ansan 426-791, Republic of Korea

<sup>2</sup> Department of Mathematics, University of the Punjab, Lahore 54590, Pakistan

Correspondence should be addressed to Dai-Gyoung Kim; dgkim@hanyang.ac.kr

Received 3 September 2014; Accepted 7 November 2014

Academic Editor: Daniel Zaldivar

Copyright © 2015 Z. H. Shamsi and D.-G. Kim. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A new multiscale implementation of nonlocal means filtering (MHNLM) for image denoising is proposed. The proposed algorithm also introduces a modification of the similarity measure for patch comparison. Assuming the patch as an oriented surface, the notion of a normal vectors patch is introduced. The inner product of these normal vectors patches is defined and then used in the weighted Euclidean distance of intensity patches as the weight factor. The algorithm involves two steps: the first step is a multiscale implementation of an accelerated nonlocal means filtering in the discrete stationary wavelet domain to obtain a refined version of the noisy patches for later comparison. The next step is to apply the proposed modification of standard nonlocal means filtering to the noisy image using the reference patches obtained in the first step. These refined patches contain less noise, and consequently the computation of normal vectors and partial derivatives is more precise. Experimental results show equivalent or better performance of the proposed algorithm compared to various state-of-the-art algorithms.

## 1. Introduction

The phenomenon of image degradation is quite natural due to the digitization and quantization processes in signal acquisition devices. Although various image denoising techniques have been extensively studied and effectively employed in the last two decades, the preservation of texture, edges, and fine details during image denoising is an open problem and needs rigorous treatment.

In advance of the nonlocal methodology, a variety of variational [1–6] PDE [7–12] and wavelet based [13–16] approaches were proposed for image denoising that rely on the local features of image data. A major shift towards nonlocal filtering was initiated by the usage of bilateral filtering [17], which exploits spatial and intensity domains for image denoising. In this approach, spatially proximate pixels are given more weights in the similarity measure. Later on, the wavelet-based BLS-GSM [18] method provided the best results in terms of PSNR measure; however, these denoised images contain ringing artifacts and have low visual quality. More recently, Buades et al. [19] introduced nonlocal

means filtering for image denoising. Although the PSNR of nonlocal means filtering was found to be less than the wavelet-based BLS-GSM [18], the notion of the patch-based approach combined with the idea of nonlocality has led to an entirely new way of attacking the problem. Earlier, approaches similar to nonlocal means were used for image inpainting [20] and texture synthesis [21]. Ever since the nonlocal means were proposed, more rigorous research for better estimation of parameters or finding suitable similarity measures has improved the performance of nonlocal means filtering for a variety of noise models in various image processing applications.

Kervrann et al. [22] provided a theoretical foundation for an intuitive nonlocal means approach using Bayesian statistics. In this approach, refined adaptive dictionaries of similar patches are obtained using Bayesian estimation, while the irrelevant patches are rejected. Afterwards, these learned dictionaries are used for patch-based comparison using a modified similarity measure. Elad and Aharon [23] proposed learned dictionaries of patches using K-means SVD algorithm and then employed sparse representation using  $L_0$

pseudonorm over the learned dictionaries for construction of denoised image. Tasdizen [24] proposed using the similarity of patches in the principal component analysis domain. Extensive research in patch-based denoising has resulted in state-of-the-art algorithms [25, 26]. Dabov et al. [25] achieved enhanced sparsity for similar patches in the predefined 3D dictionaries like discrete cosine transform (DCT) or discrete Haar wavelets transform. These dictionaries are separable and can be extended from two-dimensional to three-dimensional form using tensor product. Furthermore, Mairal et al. [26] extended the dictionary learning approach [23] using  $L_1$  norm for learning phase and  $L_0$  pseudonorm for the final construction phase. The major distinction between this approach and [25] is the use of learning phase to construct a dictionary rather than to use a predefined dictionary. The learned dictionary is developed either from the given noisy image or from a database of patches extracted from large collection of natural images.

Several attempts have been made to relate nonlocal means filtering with the variational and PDE based image denoising techniques. Gilboa and Osher [27] provided an elegant interpretation of the nonlocal means approach as a generalization of variational and PDE-based formulations. Brox et al. [28] proposed a computationally efficient algorithm for nonlocal filtering that arranged the data in a cluster tree. This special arrangement further helps in preselection of similar patches. Also, an iterative version of nonlocal filtering based on the variational framework was suggested in [28]. Pizarro et al. [29] introduced a discrete version of variational formulation that exploits the nonlocal data and smoothness (NDS) constraints for a variety of generalized dissimilarity measures defined on the space of image patches. This formulation results in a new similarity measure that considers not only the patch similarity of the two selected pixels but also the similarity of the respective neighbors. Moreover, the formulation emphasized the connection between diffusion based approaches and NDS formulation. Yang and Jacob [30] proposed a unified variational framework for nonlocal regularization by introducing robust distance norm to determine interpatch distances. Furthermore, by using this formulation for inverse problems, theoretical justification is provided for heuristic iterative nonlocal means approaches. Tschumperlé and Brun [31] defined a patch space for implementing PDE-based diffusion or smoothing process.

In addition to the photometric similarity, which is used for patch-based comparison, the patches contain much more information that requires the attention of researchers. With this motivation, we propose the notion of a normal vectors patch corresponding to each intensity patch. By employing this new notion in the second step of our algorithm, we achieve remarkably better results than most of the state-of-the-art algorithms in the presence of moderate or severe noise. Inspired by the special treatment of the central patch in [32], we also employ a slightly higher weight than in the standard nonlocal means approach. The weight value in our approach is associated with the central patch empirically through experiments. This modification has further improved our results.

Earlier, Mahmoudi and Sapiro [33] have proposed the average gradient orientation difference in combination with nonlocal means filtering which seems to be similar to the second step of our proposed method. However, the approach of [33] significantly differs from the proposed approach due to two main reasons. Firstly, for [33], the average gradient information is employed only for preclassification of the intensity patches to increase the computational efficiency of standard nonlocal filtering without the modification of the classical similarity measure. Whereas, the proposed approach exploits the interactions of normal vectors patches for the modification of the similarity weights without any preclassification step. Secondly, the average gradient orientations of patches are used in [33] on the basis of the assumption that the average gradient direction is expected to be similar in the presence of additive white Gaussian noise. This assumption is somewhat arguable in the presence of severe noise level.

The rest of the paper is organized as follows. A brief review of methodologies used in this paper is provided in Section 2. The new multiscale algorithm and its implementation are explained in Section 3. Experimental results are described and discussed in Section 4, and conclusions are drawn in Section 5.

## 2. Preliminaries

*2.1. Nonlocal Means Filtering.* Consider a noisy gray-scale intensity image

$$u = u_0 + v(0, \sigma^2 \mathbf{I}), \quad (1)$$

where  $u_0$  is the true noise free image to be recovered,  $v$  represents the zero mean additive white Gaussian noise of known variance  $\sigma^2$  and  $\mathbf{I}$  is the Identity matrix with the same dimensions as those of the given image. Instead of the nonlocal filtering, it is preferable to use the notion of semilocal filtering. It was mentioned in [32] that considering the whole image to search for similar patches has no major benefit, with the exception of periodic or quasiperiodic images. Also, searching the whole image for each pixel is computationally too expensive. Therefore, in the rest of this paper, the term nonlocal means refers specifically to semilocal filtering. The nonlocal means filtering is defined [19] as follows:

$$u_{\text{nlm}}(i) = \frac{1}{K(i)} \sum_{j \in \Delta_i} w(i, j) u(j), \quad (2)$$

where  $u_{\text{nlm}}(i)$  denotes the denoised value at pixel location  $i$  and  $w(i, j)$  is the weight obtained by determining similarity of noisy intensity patches  $\mathbf{P}(i)$  and  $\mathbf{P}(j)$  around the central pixel  $i$  and its neighboring pixel  $j$ , respectively, within the search window  $\Delta_i$ . The normalization factor is given by  $K(i) = \sum_{j \in \Delta_i} w(i, j)$ .

The weights,  $w(i, j)$ , are obtained using the Gaussian weighted  $L_2$  norm [19]:

$$w(i, j)_{i \neq j} = \exp\left(-\frac{\|\mathbf{P}(i) - \mathbf{P}(j)\|_{2,\alpha}^2}{h^2}\right), \quad (3)$$

where  $\alpha$  is the standard deviation of Gaussian function to consider the spatial proximity of the central patch and its neighboring patches in the search window. The patches, in the search window, spatially closer to the central patch are assigned higher weights than those for the distant patches. The nonlocal means approach involves three parameters: the size of the patch  $\mathbf{P}(i)$ , the size of the search window  $\Delta_i$ , and the filtering or smoothing parameter  $h$ . Detailed discussions on the suitable choices of these parameters can be found in [32, 34].

Note that the similarity measure, given in (3), trivially assigns the largest value of 1 to the patch when compared to itself, so that the central patch has the weight 1 as  $\|\mathbf{P}(i) - \mathbf{P}(j)\|_{2,\alpha}^2 = 0$  for  $i = j$ . However, the weight value assigned to the central patch may not be reliable for the noisy image data. To avoid this unreliable self-similarity weight, Buades et al. [19] offered the maximum value of all the weights computed for  $j \neq i$  around the central patch for the center self-similarity weight. The standard similarity measure can then be given by

$$w(i, j) = \begin{cases} \exp\left(-\frac{\|\mathbf{P}(i) - \mathbf{P}(j)\|_{2,\alpha}^2}{h^2}\right) & j \neq i \\ \max_{k \in \Delta_i, k \neq i} \{w(i, k)\} & j = i. \end{cases} \quad (4)$$

**2.2. Accelerated Nonlocal Means Filtering.** To increase the computational efficiency of nonlocal means filtering, several approaches [28, 33, 34, 36] have been introduced. Buades et al. [34] suggested an accelerated version (block-based) of nonlocal means algorithm that, instead of using the central pixel, replaces the whole patch around the central pixel with a weighted average of patches around the neighboring pixels in the search window. Mathematically this filtering process is defined as [34]

$$\mathbf{P}_{\text{nlm}}(i) = \frac{1}{K(i)} \sum_{j \in \Delta_i} w(i, j) \mathbf{P}(j), \quad (5)$$

where the similarity measure,  $w(i, j)$ , is the same as defined in (3). Finally, the pixel value at central location  $i$  is recovered by averaging all the resulting estimators (patches) containing that pixel location. However, the cost for the computational improvement of the accelerated nonlocal means is a slight degradation in the visual quality of the denoised image.

Mahmoudi and Sapiro [33] proposed a fast implementation of nonlocal means filtering through preclassification step to remove unrelated patches prior to the computation of similarity weights. This preclassification step is implemented on the basis of either the comparison of the local average gray values computed from intensity patches or the comparison of average gradients computed from the gradient patches. In either case, due to the presence of noise, this preclassification step is followed by thresholding process. Finally, the weights are computed only with the preclassified patches. We employ the block-based nonlocal means filtering [34] in the first step of our algorithm exploiting its relative simplicity as compared to other fast implementations.

**2.3. Adaptive Similarity Measure Approaches.** In this section, we briefly discuss the recent approaches proposing effective modifications to the similarity measure used in the classical nonlocal means filtering. For further details of these modifications, we refer to [37, 38]. The weighted patch-wise photometric distance between patches  $\mathbf{P}(i)$  and  $\mathbf{P}(j)$  can be expressed in vectorial notation as

$$\|\mathbf{P}(i) - \mathbf{P}(j)\|^2 = (\mathbf{P}(i) - \mathbf{P}(j))^t \mathbf{G} (\mathbf{P}(i) - \mathbf{P}(j)), \quad (6)$$

where  $\mathbf{G}$  denotes a fixed diagonal matrix which contains Gaussian weights. Despite the fact that the weighted  $L_2$  norm represented by weight matrix  $\mathbf{G}$  performs quite well, nonlocal means filtering results in oversmoothing in a certain area of a given image due to fixed weight matrix. One of the most effective approaches to incorporate the locally adaptive similarity weights is based on finding the local covariance matrix. The locally adaptive photometric distance is therefore defined as

$$\|\mathbf{P}(i) - \mathbf{P}(j)\|^2 = (\mathbf{P}(i) - \mathbf{P}(j))^t \mathbf{C}_{\mathbf{P}_j}^{-1} (\mathbf{P}(i) - \mathbf{P}(j)), \quad (7)$$

where the  $\mathbf{C}_{\mathbf{P}_j}$  is the local covariance matrix for the patch centered at pixel  $j$ . However, it is not possible to obtain reliable covariance matrix from a single observation that contains noise as well. In order to overcome this problem, a certain redundancy in the given image is used as an implicit prior which is generally true in case of natural images. Exploiting this assumption, the noisy patches in the observed image similar to the reference patch are sampled and grouped to obtain an estimate of covariance matrix. Several methods [35, 39, 40] have been proposed to obtain robust and reliable estimates of locally adaptive covariance matrix which are briefly described below.

Dabov et al. [39] proposed further enhancement for the original BM3D method [25] by introducing shape adaptive transform basis (BM3D-SAPCA) instead of fixed DCT or wavelet transform basis. This shape adaptivity is achieved by incorporating the following modifications. Firstly, the shape adaptive 3D groups of patches similar to the reference patch are constructed as described in [41]. Secondly, the local covariance matrix is estimated from this group provided that the number of similar patches is large enough to ensure the reliable estimation of the local covariance matrix. Thirdly, PCA transform is obtained by the eigen-decomposition of the estimated covariance matrix and only those eigenvectors are retained as principal components whose eigenvalues are greater than a fixed threshold value. Next, the truncated PCA transform is applied on the 3D group of the shape adaptive patches with 1D orthogonal wavelet transform along the third direction of the group. All the remaining process is same as that of the original BM3D [25].

Zhang et al. [40] proposed LGP-PCA approach where PCA is performed on the local covariance matrix estimated by local pixel grouping (LPG) of the patches. Afterwards, the linear minimum mean square-error estimation (LMMSE) technique is used for coefficient shrinkage in the PCA transform domain obtained from diagonalization of the covariance matrix. This approach is similar to BM3D-SAPCA [39]

in the sense that it involves two steps denoising process. However, in contrast to the BM3D, it does not depend on basic estimate (oracle) obtained in first iteration. Recently, Lebrun et al. [35] has proposed NL-Bayes algorithm to merge the Fourier transform domain like methods with Bayesian framework. In this case, a reliable estimate of the covariance matrix is obtained by applying Bayes' rule for 3D groups of noisy patches followed by MAP (maximum *a posteriori*) estimation. The LPG-PCA method [40] can be realized as a special case of this more general approach.

As will be described in the next section, the proposed modification to the classical similarity measure significantly differs from the above mentioned approaches. It assigns the adaptive similarity weights based on the correlation of normal vectors patches in addition to the similarity of intensity patches. Also, unlike the above techniques, the proposed approach does not apply 2D or 3D grouping of the similar patches.

**2.4. Discrete Stationary Wavelet Transform.** Wavelets have numerous applications, in particular, in signal and image processing, thanks to the efficient time-frequency localization and multiresolution properties. The wavelet transform is generated by the convolution integral with a wavelet  $\psi$  which defines a Riesz basis for  $L_2(\mathbb{R})$  with a two parameter family  $\{\psi_{j,l}\}_{j,l \in \mathbb{Z}}$ , where

$$\{\psi_{j,l}(x) = 2^{j/2} \psi(2^j x - l)\}_{j,l \in \mathbb{Z}}. \quad (8)$$

The wavelet decomposition of  $f \in L_2(\mathbb{R})$  is given by

$$f(x) = \sum_{j,l \in \mathbb{Z}} d_{j,l} \psi_{j,l}(x) \quad \text{with } d_{j,l} = \langle f, \tilde{\psi}_{j,l} \rangle, \quad (9)$$

where  $\tilde{\psi}_{j,l}$  is the dual wavelet to  $\psi_{j,l}$ . For the orthogonal wavelet case,  $\tilde{\psi}_{j,l} = \psi_{j,l}$ . The wavelet decomposition provides the perfect reconstruction and multiresolution analysis of  $f$ , as well. In the practical situation,  $f$  is a signal with finite resolution scale. In this case, the wavelet decomposition has the following representation:

$$f(x) = \sum_l c_{J_0,l} \phi_{J_0,l} + \sum_{j=J_0}^J \sum_l d_{j,l} \psi_{j,l}(x) \quad (10)$$

with  $c_{J_0,l} = \langle f, \tilde{\phi}_{J_0,l} \rangle$ ,

where  $J$  is the finest resolution scale determined by the size of the signal  $f$  and  $J_0 < J$  is a chosen coarsest resolution scale. Also,  $\phi$  is the scaling function associated with the wavelet  $\psi$ . The standard discrete wavelet transform (DWT) decomposes a signal  $f$  into low and high pass subbands. At a given resolution scale  $j$ ,  $(c_{j,l})$  is the approximate (low-passed) data and  $(d_{j,l})$  is the detail (high-passed) data of  $f$ . For the properties of the wavelet transform, we refer to [42].

The one-dimensional wavelet can be applied through separable extension to obtain a two-dimensional wavelet. One easy way to construct two-dimensional wavelet is to use the tensor product of one-dimensional wavelet. In this

case, there are one scaling function  $\phi$  and three kinds of wavelet,  $\psi^k$  ( $k = 1, 2, 3$ ):  $\psi^1$ ,  $\psi^2$ , and  $\psi^3$  are horizontally, vertically, and diagonally oriented wavelets, respectively. For a given image data  $u$ , the wavelet transform provides the subband data through the wavelets and scaling function at the resolution scale  $j$  by

$$W_j^k(u) = (\langle u, \tilde{\psi}_{j,l}^k \rangle), \quad k = 1, 2, 3, \quad (11)$$

$$S_j(u) = (\langle u, \tilde{\phi}_{j,l} \rangle). \quad (12)$$

So,  $S_j(u)$  represents the low-passed data and  $W_j^k(u)$  are high-passed detail data:  $W_j^1(u)$ ,  $W_j^2(u)$ , and  $W_j^3(u)$  are horizontal detail, vertical detail, and diagonal detail subband data, respectively. Note that the standard DWT generates the downsampled data due to low and high pass filtering followed by the decimation in the nature of DWT. Also, even if DWT has the perfect reconstruction property, it is not shift invariant as the wavelet coefficients of the given data and its shifted version may not be same [43]. So, the transformed data loses some detail information.

To overcome this kind of disadvantage, the discrete stationary wavelet transform (DSWT) was proposed [42, 43]. One of the main ideas of DSWT is to perform low and high pass filtering without decimation. In this approach, multiresolution analysis is achieved by simply inserting zeroes between every adjacent pair of the coefficients in low and high pass filters associated with the wavelet. The insertion of zeroes makes the dimensions of filters at the resolution scale  $j$  same as those at the resolution scale  $j+1$ . Subsequently, the DSWT provides a redundant representation of a data so that the resulting wavelet decomposition is overcomplete. We notice the following advantages of the DSWT in the viewpoint of image denoising.

- (i) The redundant representation of data contains more information than the case of nonredundant one.
- (ii) The shift invariance representation has smooth regularity and also does not contain the error due to downsampling.
- (iii) The size of the subband data is same as that of the input image data, so that the spatial information of those data can be directly compared.

In the proposed method, we implicitly exploit these advantages to obtain the similarity measure of normal vectors patches. For our notational convenience, we will use the same notation for DSWT as the case of DWT. That is,  $S_j(u)$ ,  $W_j^k(u)$ , and  $k = 1, 2, 3$  are the subband data at the resolution scale  $j$  generated by the DSWT.

### 3. The Proposed Algorithm

In this section, we adopt a hybrid approach to nonlocal means filtering. Our scheme is composed of two steps, as shown in Figure 1. In the first step, we apply the accelerated (block-based) version in the wavelet transform domain to obtain a predenoised image as shown in Figure 2. In the second

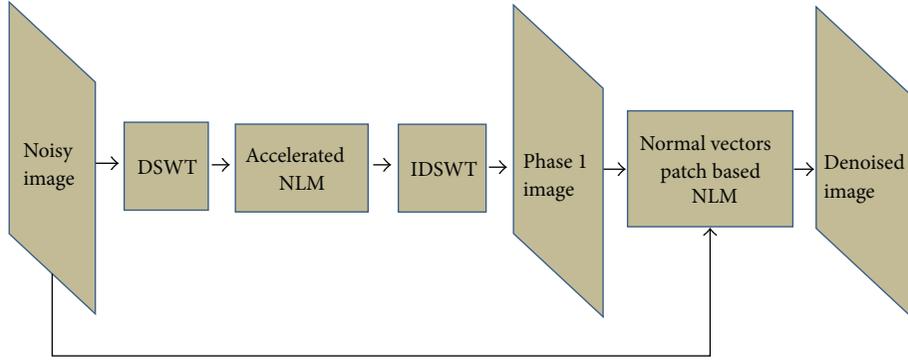


FIGURE 1: Schematic depiction of the proposed nonlocal means filtering algorithm. Phase 1 image is obtained in the first step. Afterwards, in the second step, phase 1 image is used in combination with the given noisy image to obtain a final denoised image.

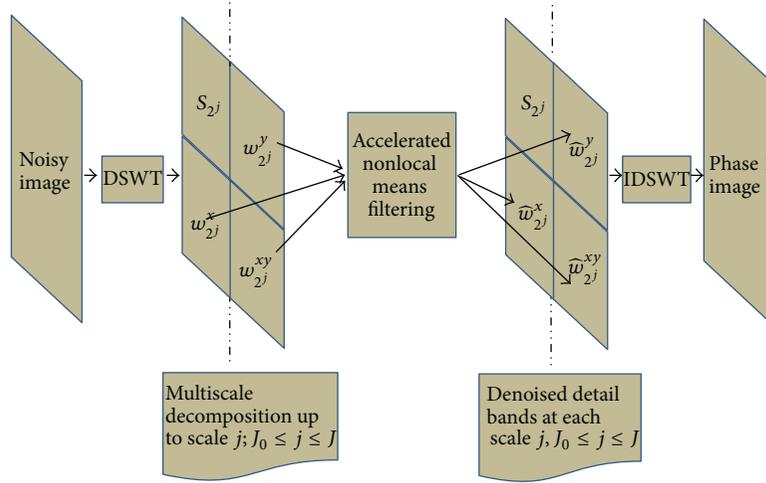


FIGURE 2: Schematic depiction of the first step of the proposed algorithm at the scale  $j$  with  $J_0 \leq j \leq J$ .

step, we employ a modified version of conventional nonlocal means filtering on the given noisy image. In this step, the weights are computed using the predenoised image obtained in the first step.

**3.1. Multiscale Accelerated Nonlocal Means Filtering.** Inspired by various approaches [22, 23, 25], based on using predenoised images to obtain refined patches, we seek to obtain a predenoised image in our approach. However, in contrast to those techniques, we adopt a very simple approach as shown in Figure 2. We first decompose the image  $u$  using the two-dimensional discrete stationary wavelet transform [43] with a chosen coarsest level  $J_0$ ,

$$\text{DSWT}(u) = \left( (W_j^1), (W_j^2), (W_j^3) \right)_{J_0 \leq j \leq J}, (S_{J_0}) u, \quad (13)$$

where  $W_j^1(u)$ ,  $W_j^2(u)$ , and  $W_j^3(u)$  represent the horizontal, vertical, and diagonal detailed data, respectively, obtained with (11) at the scale  $j$  with  $j = J_0, \dots, J$ . Also,  $S_{J_0}(u)$  is the approximate data given by (12) at the coarsest resolution scale  $J_0$ .

The accelerated (block-based) version of nonlocal means filtering [34] is performed on each of the detail data  $W_j^1(u)$ ,  $W_j^2(u)$ , and  $W_j^3(u)$  at each scale  $j$ . To be precise, the estimated wavelet coefficients patch (or block)  $\hat{w}_j^k u(t)$  at location  $t$  from the detailed data  $W_j^k(u)$  is obtained by applying (5) as

$$\hat{w}_j^k u(t) = \frac{1}{K(r)} \sum_{s \in \Delta_t} w(t, s) w_j^k u(s), \quad (14)$$

where the weights  $w(t, s)$  are computed using wavelet coefficients patches  $w_j^k u(s)$  instead of using intensity patches  $\mathbf{P}(s)$  in (4). Subsequently, the denoised wavelet coefficient  $\hat{w}_j^k u(t)$  at central location  $t$  is recovered by averaging all the resulting estimated blocks containing that location. This process is implemented on each detailed subband data independently and finally the predenoised image is obtained using the inverse discrete stationary wavelet transform. Thanks to overcomplete, shift invariant, and sparse representation of an image in the DSWT domain, the image can be efficiently denoised using accelerated nonlocal means filtering without

severe degradation of texture, edges, and fine details. This predenoised image will further be used as a reference image along with the given noisy image for computation and comparison of normal vectors patches in the subsequent section. It is worth noticing that several recent methods [44, 45] have implemented nonlocal means filtering in wavelet domain for super-resolution of images and video sequences.

**3.2. Normal Vectors Patches and Weight Factors.** Assuming the patches as surfaces, to each intensity patch of size  $m \times m$  at pixel location  $i$ , we associate a normal vectors patch  $\tilde{\mathbf{N}}_i$  of size  $2m \times 2m$ . In order to construct this normal vectors patch, we first compute the gradient of the image at each pixel location and then form the patches of horizontal and vertical components of the gradient vectors around each pixel  $i$ . These partial derivative patches are denoted by  $\mathbf{N}_{i,x}$  and  $\mathbf{N}_{i,y}$ , where the subscripts  $x$  and  $y$  represent partial derivatives in the horizontal and vertical directions, respectively. The size of  $\mathbf{N}_{i,x}$  and  $\mathbf{N}_{i,y}$  patches is the same as that of the intensity patch.

*Definition 1.* The normal vectors patch  $\tilde{\mathbf{N}}$  is defined as

$$\tilde{\mathbf{N}} \equiv (\mathbf{N}_x)_{m \times m} \oplus (\mathbf{N}_y)_{m \times m} = \begin{pmatrix} \mathbf{N}_x & \mathbf{O} \\ \mathbf{O} & \mathbf{N}_y \end{pmatrix}_{2m \times 2m}, \quad (15)$$

where  $\oplus$  denotes the direct sum of matrices and  $\mathbf{O}$  is the zero matrix of size  $m \times m$ .

For a similarity measure between two normal vectors patches  $\tilde{\mathbf{N}}_i$  and  $\tilde{\mathbf{N}}_j$ , we employ the inner product for square real matrices induced by the trace as follows:

$$\langle \tilde{\mathbf{N}}_i, \tilde{\mathbf{N}}_j \rangle = \text{tr}(\Gamma_{ij}), \quad (16)$$

where we set the matrix  $\Gamma_{ij}$  as

$$\begin{aligned} \Gamma_{ij} &= \tilde{\mathbf{N}}_i^t \tilde{\mathbf{N}}_j = (\mathbf{N}_{i,x} \oplus \mathbf{N}_{i,y})^t (\mathbf{N}_{j,x} \oplus \mathbf{N}_{j,y}) \\ &= (\mathbf{N}_{i,x}^t \mathbf{N}_{j,x} \oplus \mathbf{N}_{i,y}^t \mathbf{N}_{j,y}). \end{aligned} \quad (17)$$

Recall that for square real matrices  $A$  and  $B$  with same size,  $\langle A, B \rangle = \text{tr}(A^t B)$  is a well-defined inner product. The inner product defined above signifies the geometric correlation between the central normal vectors patch  $\tilde{\mathbf{N}}_i$  at central location  $i$  and the neighboring patch  $\tilde{\mathbf{N}}_j$  at the position  $j$  in the search window. The higher the absolute value of inner product the more the similarity between the normal vector patches and vice versa. In order to obtain the individual normal vector in the neighboring patch that may have the highest correlation with that of the central patch, we define the diagonal matrix as follows:

$$\Lambda_{ij} = \text{diag}(\mathbf{N}_{i,x}^t \mathbf{N}_{j,x} + \mathbf{N}_{i,y}^t \mathbf{N}_{j,y}). \quad (18)$$

The maximum individual correlation, denoted by  $\tau_{ij}$ , is defined as

$$\tau_{ij} = \|\Lambda_{ij}\|_{\infty} = \max_{1 \leq p \leq m} \sum_{q=1}^n |\lambda_{pq}|, \quad (19)$$

where  $\|A\|_{\infty}$  norm denotes the maximum of absolute row sum of a matrix  $A$ . Since the matrix  $\Lambda_{ij}$  is the diagonal matrix, this norm yields the maximum absolute value of all diagonal entries of  $\Lambda_{ij}$ . In order to obtain the modified similarity measure between normal vectors patches in the next section, we define a weight factor as follows.

*Definition 2.* A weight factor  $\eta(i, j)$  based on two normal vectors patches  $\tilde{\mathbf{N}}_i$  and  $\tilde{\mathbf{N}}_j$  is defined as

$$\begin{aligned} \eta(i, j)_{i \neq j} &= \exp\left(-\frac{\langle \tilde{\mathbf{N}}_i, \tilde{\mathbf{N}}_j \rangle}{\tau_{ij}(m^2 - 1)}\right) \\ &= \exp\left(-\frac{\text{tr}(\Gamma_{ij})}{\tau_{ij}(m^2 - 1)}\right). \end{aligned} \quad (20)$$

*Remark 3.* It can be easily verified that the weight factor  $\eta$  satisfies the following properties to measure the similarity between normal vectors patches.

- (i)  $\eta(i, j) > 0 \quad \forall j \in \Delta_i$  (Positivity).
- (ii)  $\eta(i, j) = \eta(j, i) \quad \forall j \in \Delta_i$  (Symmetry).
- (iii)  $\eta(i, j) \leq M < \infty \quad \forall j \in \Delta_i$  (Bounded).

It can be noticed that in case of more similar normal vectors patches,  $\text{tr}(\Gamma_{ij})/\tau_{ij}(m^2 - 1)$  is a large value resulting in the small value of  $\eta(i, j)$  and vice versa. The intuitive motivation for introducing this factor is to acquire the degree of similarity of the original intensity patches, based on the similarity of the corresponding normal vectors patches, as shown in Figure 3. Note that the weight factor involves the computation of derivatives which is sensitive to the noise level in the image. Since in the predenoised image obtained in the first step, the noise level is much lower than in the given noisy image, the computation of partial derivatives is more reliable. Therefore, we use the weight factor in the second step of our algorithm.

**3.3. Modified Nonlocal Means Filtering.** After obtaining predenoised image as defined in Section 3.1, we perform the modified version of standard nonlocal means filtering on a given noisy image, based on the similarity of reference patches from the predenoised image.

*Definition 4.* The modified similarity measure between distinct intensity patches  $\mathbf{P}(i)$  and  $\mathbf{P}(j)$  is defined as

$$w(i, j)_{i \neq j} = \exp\left(-\frac{\|\mathbf{P}(i) - \mathbf{P}(j)\|_2^2 \eta(i, j)}{h^2}\right), \quad (21)$$

where the weight factor  $\eta(i, j)$  is given in Definition 2.

It can be noticed that the Gaussian weight  $G_{\alpha}$  in (3) is replaced by the weight factor  $\eta(i, j)$  obtained with (20). The Gaussian weights were used to take into account the spatial

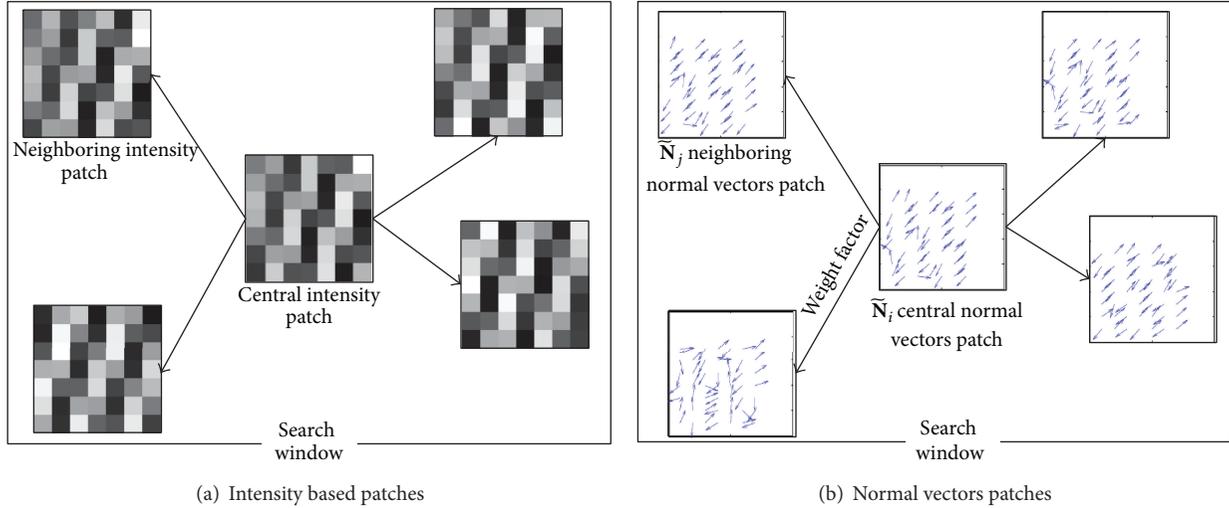


FIGURE 3: Schematic depiction for normal vectors patches, corresponding to the respective intensity patches, extracted from Barbara image.

proximity of the patches around the central patch. However, the proposed weight factor is based upon the similarity or correlation of the mean normal orientation of the central patch with other patches in the search window. The smaller the value of the weight factor  $\eta(i, j)$ , the higher the similarity weight  $w(i, j)$  assigned to the corresponding intensity patch and vice versa. The proposed metric performs better than the standard  $L_2$  norm. This is due to the fact that the standard metric, using spatial proximity weights,  $G_\alpha$ , may assign a small weight value to a distant but similar intensity patch in the search window. On the other hand, the new metric assigns more weight to the distant but similar intensity patch based on the similarity of normal vectors patches.

Our second modification to standard nonlocal means filtering is made in the assignment of self-similarity weight. The standard nonlocal means method assigns the maximum of all the computed weights as obtained in (4) for  $j = i$ . This choice of weight for self-comparison is intuitive and arbitrary. Salmon [32] discussed various choices for assigning self-similarity weight. One of the choices assumed was based on Stein unbiased risk estimation (SURE) for the central patch weight that does not affect the remaining similarity weights. It has been shown in [32] that SURE choice of self-similarity weight is at par with the arbitrary choice [19] for higher levels of noise. However, we follow the approach adopted in [19] with a little modification based on experimental results for various arbitrary values. For low or medium noise levels in the image, we assign a slightly higher self-similarity weight than that in (4), that is, the proposed self-similarity weight is defined by

$$w(i, i) = \frac{4}{3} \max_{j \in \Delta_i; j \neq i} \{w(i, j)\}. \quad (22)$$

The intuitive justification for assigning higher self-similarity weight is based on the assumption that the low or medium levels of noise do not produce a significance effect on self-similarity. Therefore this choice yields slightly better results

for low levels of noise. However, for severe noise ( $\sigma > 20$ ), the performance of the self-similarity weights defined in (4) and (22) is found to be the same. We use (22) in our simulations for both the low and high levels of noise. The second modification of self-similarity measure is then optional. The distinction in visual appearance with or without this modification is imperceptible.

**3.4. Summary of the Proposed Algorithm.** The flow chart of the proposed denoising algorithm is shown in Figure 4 and it can be summarized as follows.

- (i) Construct multiscale data in the wavelet domain by decomposing the noisy image using discrete stationary wavelet transform up to the coarsest scale  $J_0$ .
- (ii) Perform the accelerated nonlocal means filtering on the multiscale data, as described in Section 3.1, to obtain the denoised wavelet transform components.
- (iii) Reconstruct the predenoised image using inverse of the DSWT. This image will be used as a reference in the next step for patch-based comparison.
- (iv) Denoise the given noisy image using the modified nonlocal means filtering as proposed in Section 3.3. The patch comparison is performed on the refined patches obtained in the previous step.

**3.5. Extension to Color Image Denoising.** Although the proposed algorithm is primarily designed for gray-scale images, a possible extension to color images is also concerned here. For this purpose, we consider a natural color image in  $RGB$  color domain with additive i.i.d. zero mean Gaussian noise and variances  $\sigma_R^2 = \sigma_G^2 = \sigma_B^2 = \sigma^2$  in each of  $RGB$  channels. Nonlocal means filtering can be extended in a straightforward fashion to joint or simultaneous filtering of all color channels without disturbing the inherent correlations of color channels. However, due to 2D DSWT transform in the first step of

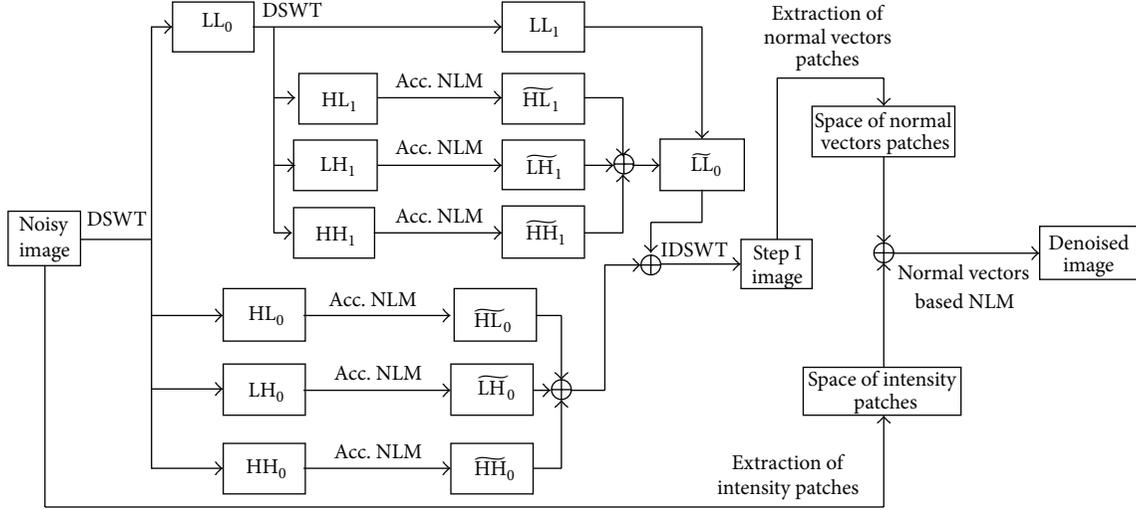


FIGURE 4: The flow chart of the proposed MHNLM algorithm. The given image is decomposed up to two scale levels using DSWT and accelerated nonlocal means [34] is applied on the detail data (HL, LH, and HH) at each scale. Afterwards, the predenoised image is obtained using IDSWT. Finally, the modified similarity measure based nonlocal means filtering is employed on the given noisy image using reference normal vector patches extracted from the predenoised image.

the proposed algorithm, we follow the approach as described in [25, 35, 41]. For the first step only, The given noisy image is transformed to  $YUV$  color domain using the transformation defined by

$$A_{YUV} = \begin{pmatrix} 0.30 & 0.59 & 0.11 \\ -0.15 & -0.29 & 0.44 \\ 0.61 & -0.51 & -0.10 \end{pmatrix}, \quad (23)$$

where  $Y$  denotes the luminance channel and  $U$  and  $V$  are two chrominance channels. The resulting channels are sufficiently decorrelated. Further, due to this transformation, the luminance channel has higher SNR than the other two chrominance channels. The noise variances  $\sigma_Y^2$ ,  $\sigma_U^2$ , and  $\sigma_V^2$  can be determined using the relation [41]:

$$[\sigma_Y^2 \ \sigma_U^2 \ \sigma_V^2] = [\sigma_R^2 \ \sigma_G^2 \ \sigma_B^2] T, \quad (24)$$

where  $T$  denotes the transpose of  $A_{YUV}$  with element wise square of its entries.

The first step of the algorithm, described in Section 3.1, is applied to each channel independently. Consequently, a predenoised image in  $YUV$  color space is transformed back to original  $RGB$  color space. For the second step of the proposed algorithm, let the normal vectors patch,  $\tilde{\mathbf{N}}_i$ , centered at pixel  $i$  in  $RGB$  color space be denoted by

$$\tilde{\mathbf{N}}_i := \left\{ \tilde{\mathbf{N}}_i^{(c)} \right\}_{c=1}^3, \quad (25)$$

where the superscript  $c = 1, 2, 3$  denotes the individual color channel  $R, G, B$ , respectively, and  $\tilde{\mathbf{N}}_i^{(c)}$  represents the normal vectors patch in that color channel defined by (15).

The similarity of normal vectors patches, denoted by  $d_{ij}^{(N)}$ , is performed jointly on all the color channels as

$$\begin{aligned} d_{ij}^{(N)} &= \frac{1}{3(m^2 - 1)} \sum_{c=1}^3 \frac{\langle \tilde{\mathbf{N}}_i^{(c)}, \tilde{\mathbf{N}}_j^{(c)} \rangle}{\tau_{ij}^{(c)}} \\ &= \frac{1}{3(m^2 - 1)} \sum_{c=1}^3 \frac{\text{tr}(\Gamma_{ij}^{(c)})}{\tau_{ij}^{(c)}}, \end{aligned} \quad (26)$$

where  $\Gamma_{ij}^{(c)}$  and  $\tau_{ij}^{(c)}$  are defined for each channel using the relations (17) and (19), respectively. The weight factor,  $\eta_{ij}$ , used in the proposed similarity measure can then be obtained with

$$\eta_{ij} = \exp(-d_{ij}^{(N)}). \quad (27)$$

Finally, the modified photometric similarity measure can be defined as

$$w(i, j) = \exp(-d_{ij}^2 \eta_{ij}), \quad (28)$$

where  $d_{ij}$  denotes the classical nonlocal means distance for  $RGB$  color patches  $\mathbf{P}(i)$  and  $\mathbf{P}(j)$  given by

$$d_{ij}^2 = \frac{1}{3} \sum_{c=1}^3 \frac{\|\mathbf{P}^{(c)}(i) - \mathbf{P}^{(c)}(j)\|^2}{(h^{(c)})^2}. \quad (29)$$

## 4. Results and Discussion

**4.1. Gray-Scale Image Denoising.** To evaluate the results of the proposed method, the benchmark gray-scale intensity images of Lena, Barbara, Boats, Peppers, and House are

considered. We use  $9 \times 9$  patches,  $15 \times 15$  search windows, and  $h = 6\sigma$  as the filtering parameter in the first step. In the second step, we select  $7 \times 7$  patches with filtering parameter  $h = 4\sigma$ . The filtering parameter is empirically reduced to  $4\sigma$  from  $6\sigma$  to avoid the oversmoothing phenomenon. The finest resolution scale for  $512 \times 512$  gray-scale image is  $J = 9$ . The proposed algorithm, denoted by MHNLM, is compared with various state-of-the-art algorithms: those by Dabov et al. [25] (BM3D), Lebrun et al. [35] (NL-Bayes), Portilla et al. [18] (BLS-GSM), Buades et al. [19] (NLM), Elad and Aharon [23] (K-SVD), and Kervrann et al. [22] (BNLM) (Table 3). The denoising results for NL-Bayes and the standard NLM methods are obtained using source codes [46, 47], respectively.

**4.2. Computational Complexity Comparison.** In what follows, we present the comparison of computational complexity in terms of the number of operations required. In our proposed algorithm, the computational time is proportional to  $N^2 n^2 m^2 (1 + 3(J - J_0)/m^2)$ , where  $N^2$ ,  $n^2$ , and  $m^2$  are the sizes of image, search window, and patch, respectively. The first term in this expression refers to the number of operations required in classical nonlocal means filtering. The second term refers to complexity of the accelerated nonlocal mean filtering on three detail subbands at each scale. The coarsest level of multiscale decomposition in DSWT domain is  $J_0 = 7$ . In our simulations, we employ biorthogonal spline wavelet transform for DSWT decomposition. The time complexity of the nonlocal means filtering is  $N^4 n^2$  where the search window is set as the whole image. However, for semilocal version of the standard NLM [19], compared here, the time complexity is given by  $N^2 n^2 m^2$ . In case of BNLM [22], the computational time depends upon the sub-sampling size  $L$  of the image and is given by  $N^2 n^2 m^2 / L^2$ . In the absence of sub-sampling scheme ( $L = 1$ ), the computational complexity of BNLM is the same as that of NLM. For BM3D [25], the time complexity, excluding the 3D transformation overhead, is given by  $2(n_1^2 + n_2)m^2 N^2 / L^2$ , where the sum,  $n_1^2 + n_2$  represents the number of operations required for grouping the patches in 3D transform domain. The multiplication factor, 2, is used to indicate the two step image denoising nature of BM3D algorithm. The sub-sampling parameter,  $L$  is the sliding window size used for block matching and is set to 1 when the grouped patches do not overlap. The computational complexity of NL-Bayes [35] is similar to that of BM3D. This is due the fact that NL-Bayes algorithm employs Bayes' rule to the the 3D groups obtained with BM3D approach. Finally, the computational cost of K-SVD [23] is given by  $(4Km^2L + 2K(L + m^2) + K^3)N^2$  [48], where  $K$  denotes the desired sparsity level and  $L$  represents the number of atoms (columns) in the dictionary. The remaining terms,  $m^2$  and  $N^2$  have the usual meanings of patch and image size, respectively. Among all the methods compared here, K-SVD algorithm is the most expensive due to the sequential update of each of the  $L$  atoms (columns) in the dictionary. From the above analysis, it can be noticed that the computational complexity of the proposed algorithm,

somehow, lies between those of the classical nonlocal means filtering and BM3D.

**4.3. Visual Quality Comparison.** Here, we discuss and compare the denoising capability of the proposed algorithm with that of the above mentioned algorithms. The peak signal-to-noise ratio (PSNR) is used for comparing the denoising capability of each scheme. For an original gray-scale image  $u_o$  of size  $M \times N$ , let  $\hat{u}$  be the denoised image obtained as its estimate using certain denoising algorithm. The PSNR is then defined as

$$\text{PSNR}(\hat{u}) = 10 \log \left( \frac{(255)^2}{(MN)^{-1} \sum_{i=1}^M \sum_{j=1}^N [u_o(i, j) - \hat{u}(i, j)]^2} \right). \quad (30)$$

Table 1 provides the PSNR comparison values for the above cited approaches; the bold-faced values represent the best performance among the last five columns. The results of BM3D and NL-Bayes are quite competitive regardless of the noise level and are better than the rest of the compared algorithms. The reason for these outstanding results is the enhanced sparse representation obtained with 3D transform domain and collaborative filtering as discussed earlier. NL-Bayes is similar to BM3D in its structure. However, it exploits Bayesian framework rather than hard thresholding or Wiener filtering used in BM3D. For the rest of the algorithms, KSVD performance is better when the noise level is low. On the other hand, the proposed method yields better results in the presence of moderate and severe noise ( $\sigma \geq 20$ ). It can be noticed that the results of the proposed algorithm are always better than the classical nonlocal means filtering. Our results can also be confirmed according to the average of the performances at various noise levels. Table 2 indicates that, on average, our algorithm outperforms KSVD, BLS-GSM, and BNLM in the presence of moderate or severe noise. However, the performance of the proposed method is slightly lower than those of the KSVD, BLS-GSM, and BNLM for low level of noise ( $\sigma = 10$ ) as depicted in Figure 8. Therefore, the proposed algorithm is more suitable in the scenarios where the noise levels are high.

Apart from the quantitative analysis with the PSNR values, the quality of visual appearance for the test images can also be assessed in terms of fine details and textures. For instance, in case of Lena's face shown in Figure 5(d) weak fine marks on the chin are preserved at the moderate noise level of  $\sigma = 20$ . Barbara image is a well-known example that contains a lot of texture. As shown at the magnified scale in Figures 6(d) and 7(d), our algorithm efficiently retains the texture on the table cover, scarf, and chair. Similarly, the noise free image of house, shown in Figure 7(e) contains very fine regular pattern of bricks that is quite intermingled with the noise. The proposed algorithm has managed to recover the brick structure while removing the noise effectively. Lastly, the boats image is the representative of the benchmark images with very fine details, sharp edges, and discontinuities. Although, our algorithm effectively removes

TABLE 1: PSNR comparison of the proposed and various state-of-the-art algorithms for gray-scale images.

Image	$\sigma$	BM3D [25]	NL-Bayes [35]	GSM [18]	NLM [19]	KSVD [23]	BNLM [22]	MHNLM
Lena	10	35.93	35.88	<b>35.61</b>	34.35	35.47	35.25	35.40
	15	34.27	34.02	33.90	32.10	33.70	33.68	<b>33.93</b>
	20	33.05	32.88	32.66	31.56	32.38	32.63	<b>32.75</b>
	25	32.05	31.89	31.69	30.44	31.32	31.55	<b>31.76</b>
	50	28.86	28.83	<b>28.61</b>	27.35	27.79	27.51	28.47
Barbara	10	34.98	34.95	34.03	33.18	<b>34.42</b>	33.83	33.80
	15	33.11	32.70	31.86	30.77	<b>32.37</b>	32.21	32.35
	20	31.78	31.55	30.32	30.27	30.83	30.88	<b>31.13</b>
	25	30.72	30.39	29.13	29.01	29.60	29.77	<b>30.07</b>
	50	27.17	26.84	25.48	25.62	25.47	24.91	<b>26.41</b>
Boats	10	33.92	33.93	33.58	32.93	<b>33.64</b>	33.18	32.94
	15	32.14	31.93	31.70	30.70	<b>31.73</b>	31.45	31.52
	20	30.88	30.71	30.38	29.73	30.36	30.16	<b>30.42</b>
	25	29.91	29.68	29.37	28.56	29.28	29.11	<b>29.43</b>
	50	26.64	26.40	<b>26.38</b>	25.25	25.95	25.13	26.22
Peppers	10	34.68	34.77	33.77	33.51	<b>34.28</b>	33.87	33.53
	15	32.70	32.53	31.74	31.07	<b>32.22</b>	32.06	32.03
	20	31.29	31.31	30.31	30.04	<b>30.82</b>	30.75	<b>30.82</b>
	25	30.16	30.16	29.21	28.68	29.73	29.77	<b>29.78</b>
	50	26.41	26.49	25.90	24.17	26.13	23.84	<b>26.28</b>
House	10	36.71	36.18	35.35	34.91	<b>35.98</b>	35.67	35.78
	15	34.94	34.33	33.64	32.70	<b>34.32</b>	34.23	34.23
	20	33.77	33.33	32.39	32.43	33.20	<b>33.24</b>	33.11
	25	32.86	32.48	31.40	31.18	32.15	<b>32.30</b>	32.15
	50	29.37	29.04	28.26	27.62	27.95	27.64	<b>28.38</b>

TABLE 2: Average PSNR comparison for various noise levels.

$\sigma$	BM3D [25]	NL-Bayes [35]	GSM [18]	NLM [19]	KSVD [23]	BNLM [22]	MHNLM
10	35.24	35.14	34.47	33.78	<b>34.76</b>	34.36	34.29
15	33.43	33.10	32.57	31.47	<b>32.87</b>	32.73	32.81
20	32.15	31.96	31.21	30.81	31.52	31.53	<b>31.65</b>
25	31.14	30.92	30.16	29.58	30.42	30.50	<b>30.63</b>
50	27.69	27.52	26.93	26.00	26.66	25.81	<b>27.15</b>

the noise, very thin ropes and strings have not been fully recovered. Moreover, at the magnified scale, there are very weak oscillations across the sharp discontinuities such as edges of ropes and rods. This oscillatory phenomenon may be inherent to wavelet based filtering which we have used in the first step of our algorithm.

An appropriate approach to ascertain the visual quality is the method noise mechanism suggested by Buades et al. [19]. The method noise is defined as the difference between the given noisy image and its denoised version which depicts the noise removed by a specific denoising operator or algorithm. Buades et al. [19] emphasized that the method noise should be as closer to random additive

noise as possible to justify the efficacy of certain denoising algorithm. It provides a way to visualize fine geometrical features or details such as texture and edges which may not be preserved by certain denoising algorithm and may be removed along with the noise. Unfortunately, due to complex nature of several state-of-the-art algorithms compared here, the closed form of mathematical description is difficult to be obtained for the method noise comparison. Therefore the method noise will be estimated using the difference images only. It is worth noticing that the visual comparison of the method noise mechanism relies on the suitable selection of noise level of the additive white Gaussian noise. A reasonable value of standard deviation of the noise is 5 or 10 as very

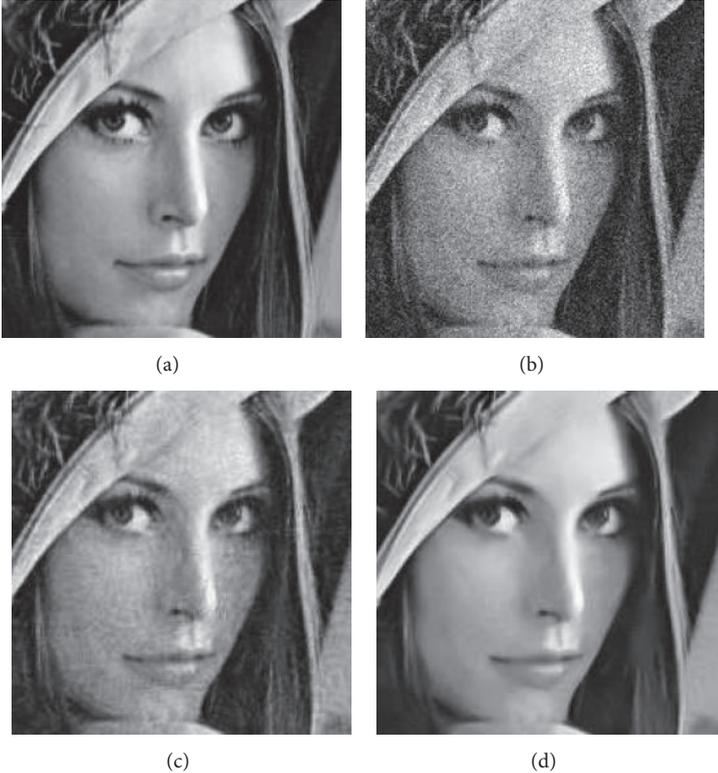


FIGURE 5: (a), (b), (c), and (d) represent the magnified original, noisy ( $\sigma = 20$ ), predenoised, and denoised subimages from Lena image, respectively.

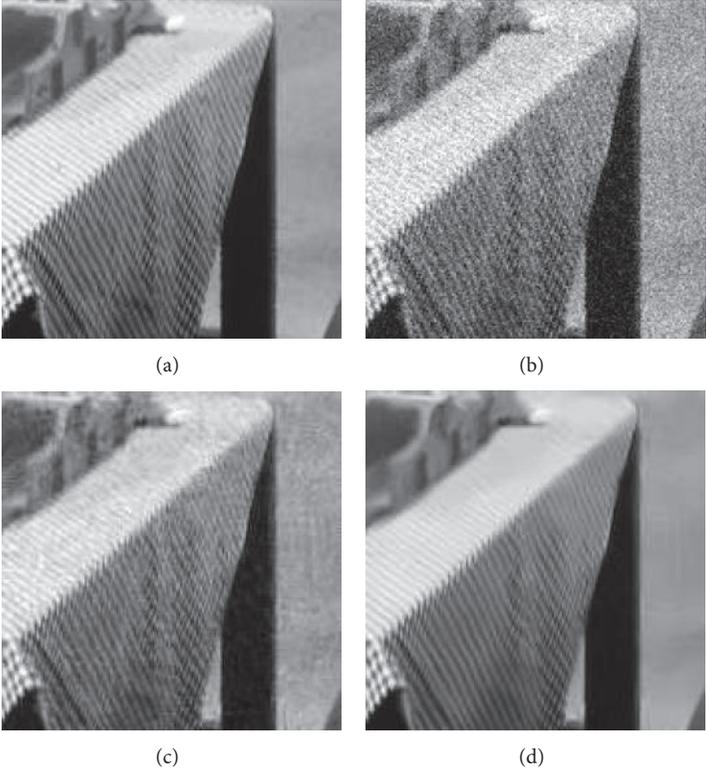
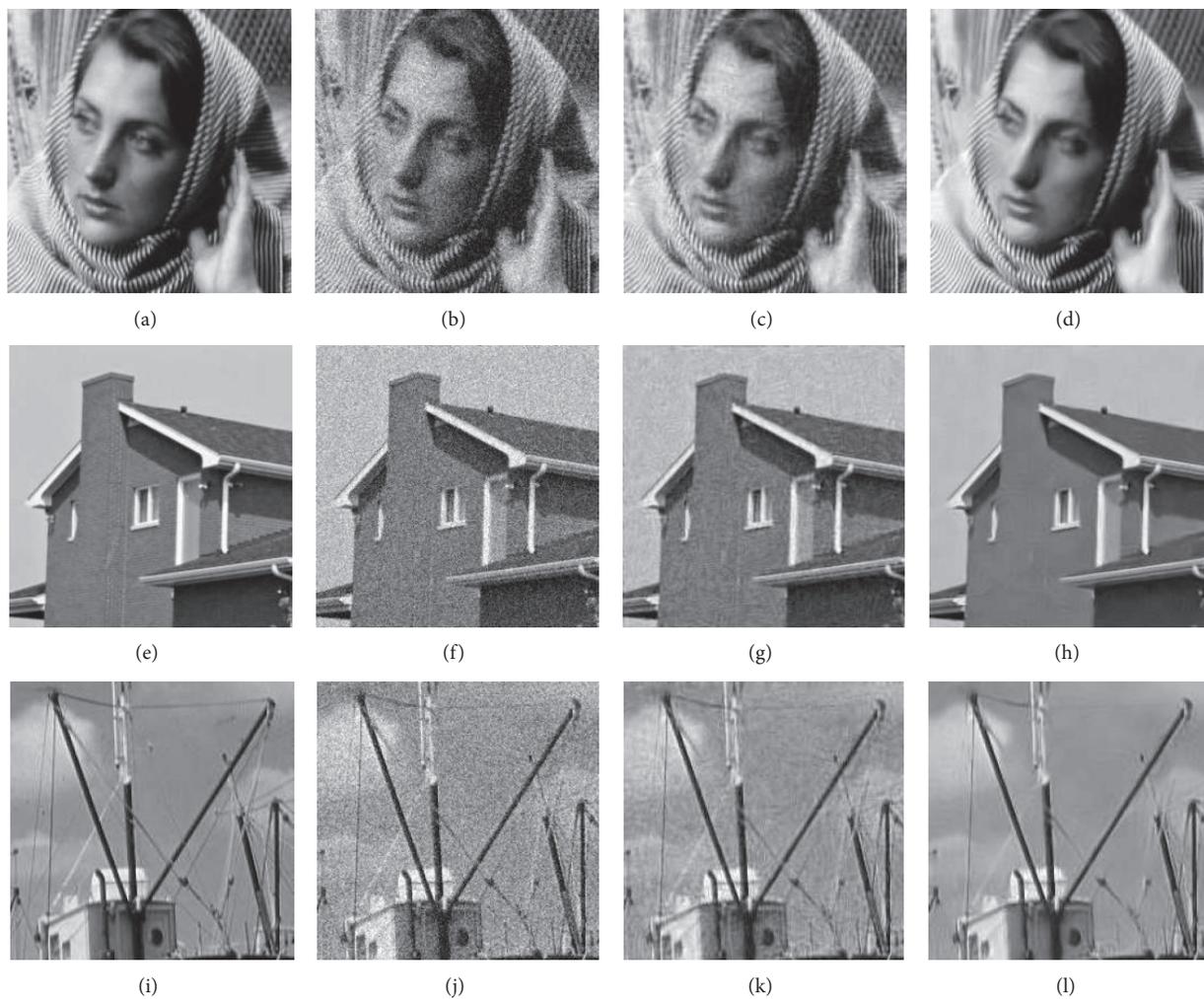


FIGURE 6: (a), (b), (c), and (d) represent the magnified original, noisy ( $\sigma = 20$ ), predenoised, and denoised subimages from Barbara image, respectively.

TABLE 3: PSNR comparison of the proposed and various state-of-the-art algorithms for color images.

Image	$\sigma$	BM3D	NL-Bayes	NLM	MHNLM
Lena	10	35.22	35.26	34.77	34.52
	15	33.94	33.75	33.15	33.15
	20	33.02	32.95	31.92	32.18
	25	32.27	32.08	30.96	31.48
	50	29.88	29.89	27.89	28.41
House	10	36.23	35.82	35.22	35.02
	15	34.85	34.90	33.75	33.55
	20	33.84	33.28	32.55	32.55
	25	33.03	32.42	31.44	31.84
	50	30.47	30.20	27.50	28.33
Peppers	10	33.78	33.92	33.74	33.20
	15	32.60	32.60	32.28	32.27
	20	31.83	31.84	31.24	31.27
	25	31.20	31.10	30.35	30.66
	50	28.93	28.98	27.14	28.02

FIGURE 7: Columns from left to right represent the magnified original, noisy ( $\sigma = 20$ ), predenoised, and denoised subimages from Barbara, house, and boat images, respectively.

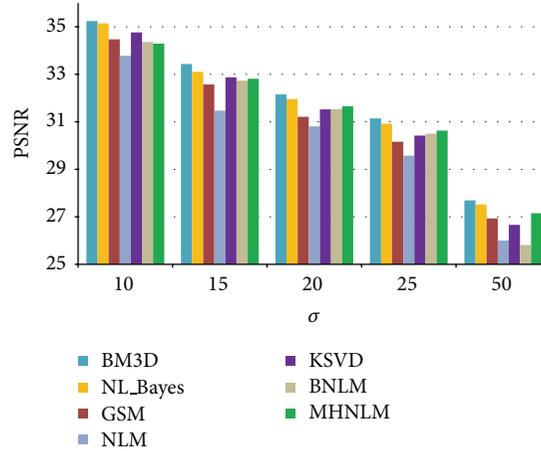


FIGURE 8: Average PSNR versus noise level  $\sigma$  comparison of BM3D, NL-Bayes, GSM, NLM, KSVD, BNLM, and MHNLM.

nically explained in [37]. This is due to the fact that when the standard deviation of the additive noise is higher than the image contrast, features even removed during denoising process will not be perceivable as they may be buried in the severe noise removed from the image. In such situation, the empirical visual assessment of the method noise may not be reliable.

The method noise of the BM3D [25], NL-Bayes [35], and classical NL-means [19] are compared with that of the proposed algorithm. Each of the gray-scale images of Barbara, Lena, boats, and house is contaminated with additive white Gaussian noise of standard deviation  $\sigma = 10$ . In order to obtain proper visualization, the difference images are rescaled as described in [37]. The method noise results of the compared algorithms, including the proposed one, are similar to the white Gaussian noise as shown in Figure 9. In case of classical nonlocal means, the amplitude of the noise removed is uniform all over the image regardless of the structures like textures or edges present in the image as can be seen in Figures 9(c), 9(g), 9(k), and 9(o). For the rest of the algorithms, the amplitude of the removed noise varies with the geometrical features of the underlying image. Moreover the magnitudes of those differences obtained through classical nonlocal means are comparatively greater than the respective results for the rest of the approaches. The method noise result of the proposed algorithm is similar to those of the BM3D and NL-Bayes as shown in Figure 9. It is worth noticing that BM3D performs thresholding in 3D transform domain and NL-Bayes exploits the Bayesian framework for shrinkage of 3D groups of similar patches. In contrast to those two approaches, the proposed algorithm does not perform any thresholding. Instead, it applies the accelerated version of the classical nonlocal means filtering on the 2D transform domain. Even then, the noise method is much similar to BM3D and NL-Bayes. This similarity of the method noise indicates that multiscale nonlocal means filtering in transform domain is empirically equivalent to thresholding or shrinkage in transform domain. Further, like BM3D and NL-Bayes, magnitude of the noise removed by the proposed

algorithm depends upon the variation of geometry of the underlying image. Due to this fact, the proposed method also removes very fine and sharp discontinuities like thin ropes in boat image as shown in Figure 9(l).

**4.4. Color Image Denoising.** To evaluate the results of the proposed method for *RGB* color images, we consider the color images of Lena, House, and Peppers. In the first step of the algorithm, we use  $9 \times 9$  patches,  $15 \times 15$  search windows, and  $h = 5\sigma$  as the filtering parameter. In the second step, we select  $7 \times 7$  patches with filtering parameter  $h = 4\sigma$ . The search window size is selected as  $15 \times 15$  for noise levels with  $\sigma \leq 25$  and is changed to  $19 \times 19$  otherwise. The proposed algorithm (MHNLM) is compared with various state-of-the-art algorithms: those by Dabov et al. [25] (BM3D), Lebrun et al. [35] (NL-Bayes), and Buades et al. [19] (NLM). The denoising results for NL-Bayes and standard NLM methods are obtained using source codes [46, 47], respectively (Figure 10). The source codes are available at the public domain IPOL. The PSNR for *RGB* color images is defined as

$$\begin{aligned} \text{PSNR}(\bar{u}) &= 10 \log \left( \frac{(255)^2}{(3|X|)^{-1} \sum_{c=1}^3 \sum_{x \in X} [u_o^{(c)}(x) - \bar{u}^{(c)}(x)]^2} \right), \end{aligned} \quad (31)$$

where  $c = 1, 2, 3$  corresponds to the color channel *R*, *G*, *B*, respectively. Also,  $|X|$  denotes the size of 2D image in the individual channel and  $u_o$  and  $\bar{u}$  represent the original and denoised *RGB* color images, respectively.

The results of BM3D and NL-Bayes are almost similar and clearly better than the proposed algorithm. Both the algorithms exploit the 3D grouping strategy, which is not employed in the proposed algorithm. Therefore, it would be interesting to examine the combined effects of 3D collaborative filtering and normal vectors patch comparison on denoising capability. For moderate and severe levels ( $\sigma \geq 20$ )

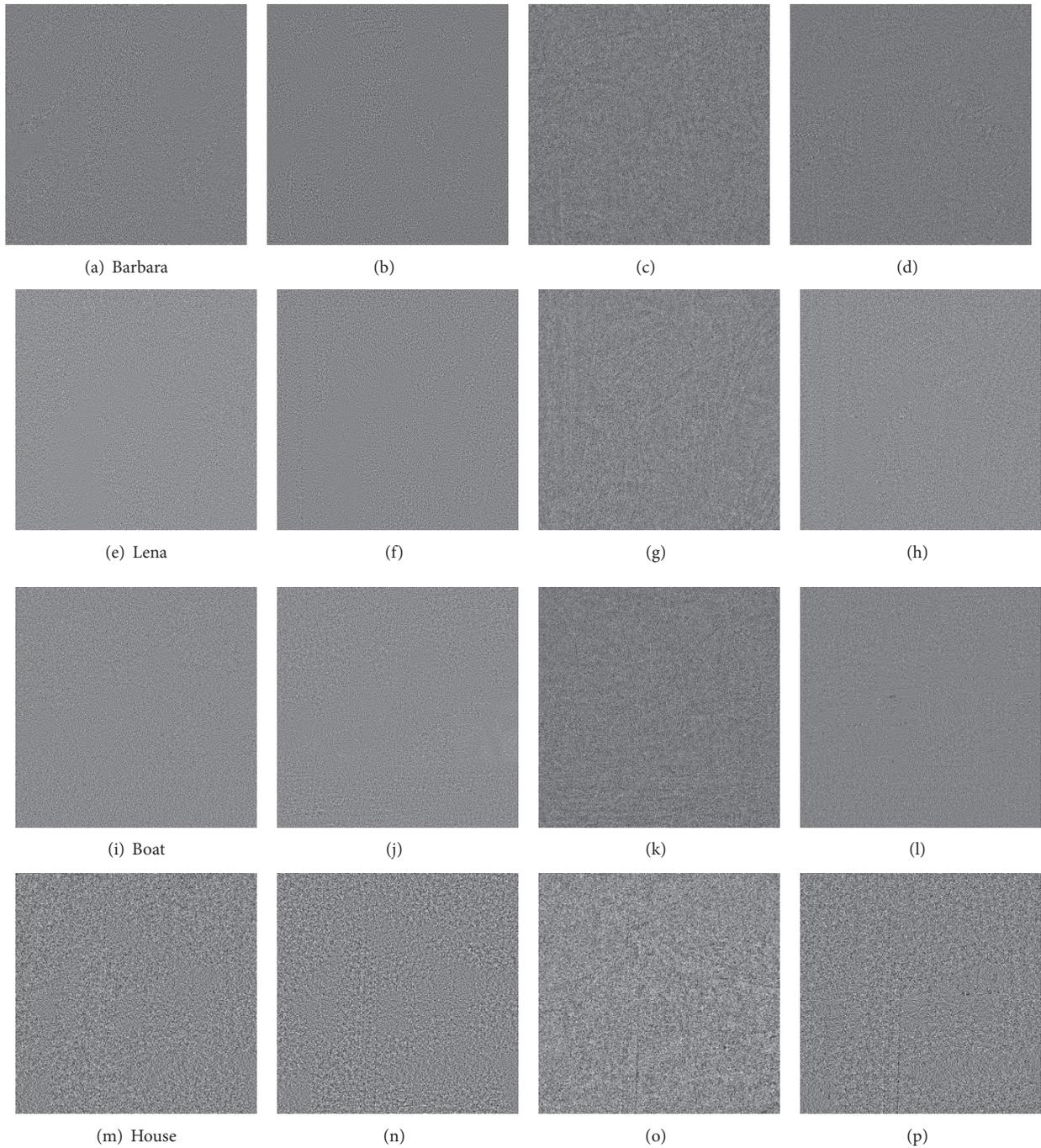


FIGURE 9: Columns from left to right represent difference of the noisy ( $\sigma = 10$ ) and the denoised images using BM3D [25], NL-Bayes [35], NLM [19], and the proposed MHNLM algorithms, respectively. The rows from top to bottom indicate the results on the gray-scale images of Barbara, Lena, boat, and house, respectively.

of noise, the proposed algorithm yields better results than the classical nonlocal means in terms of PSNR. However, in the presence of low noise ( $\sigma \leq 15$ ) the performance of the classical nonlocal means provides better results over the proposed algorithm. The performance of the proposed algorithm may be enhanced by certain modifications. For instance, 3D grouping used in BM3D or NL-Bayes can be combined with the proposed similarity measure.

## 5. Conclusions

A new definition of normal vectors patch is introduced to acquire more information about the similarity of intensity patches combined with a multiscale implementation of the accelerated nonlocal means filtering. The experimental results demonstrate the effectiveness of the proposed algorithm using the notion of normal vectors patch. In future,

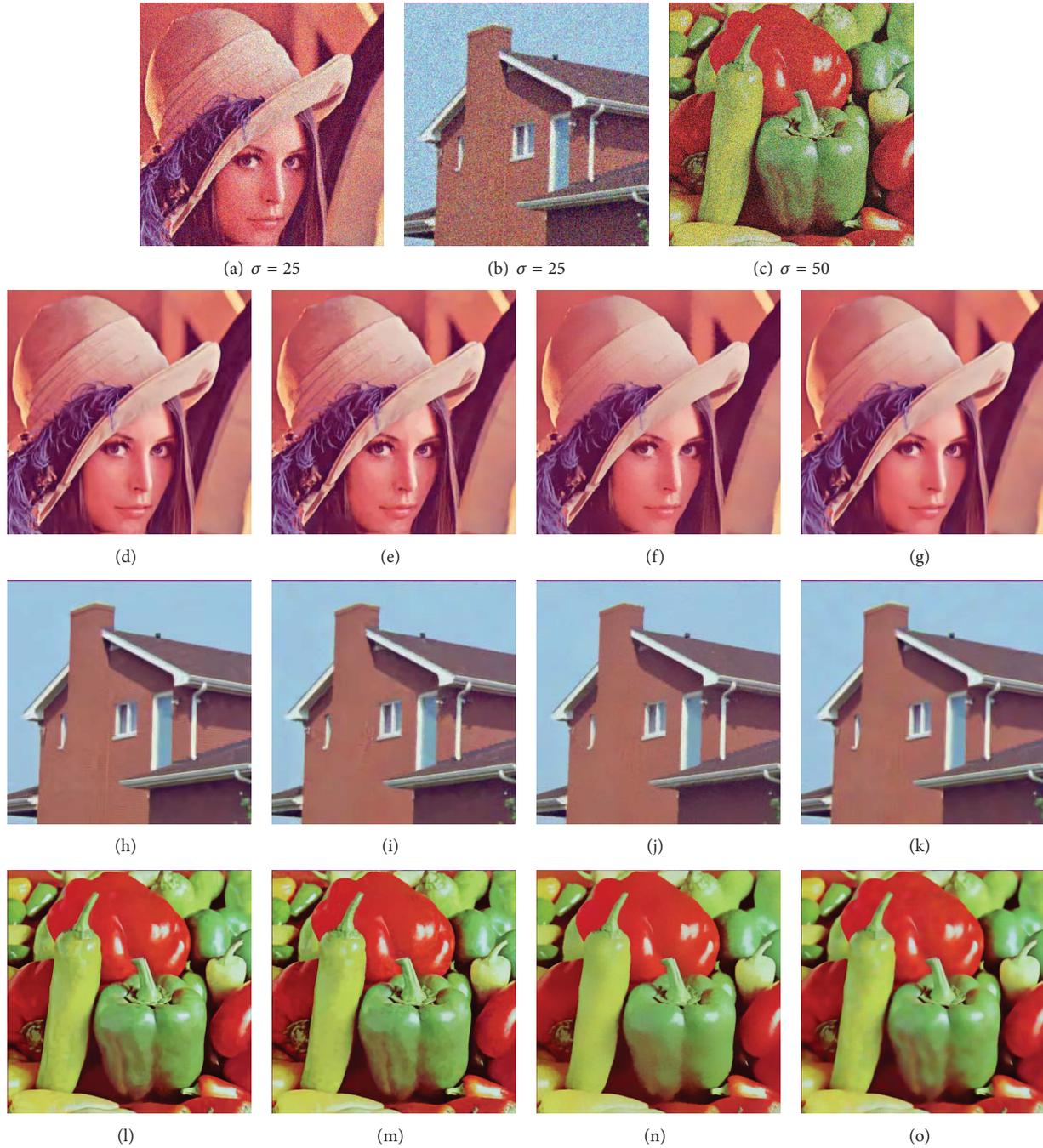


FIGURE 10: In the first row, the noisy *RGB* color images of Lena, house, and peppers are shown from left to right with standard deviations  $\sigma = 25$ ,  $\sigma = 25$ , and  $\sigma = 50$ , respectively. Columns from left to right in the remaining rows show the denoising results of BM3D [25], NL-Bayes [35], NLM [19], and the proposed MHNLM algorithms, respectively.

we plan to extend this approach in 3D transform domain that may achieve enhanced sparsity level to further improve the image denoising capability.

**Conflict of Interests**

The authors declare that there is no conflict of interests regarding the publication of this paper.

**References**

- [1] L. I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D: Nonlinear Phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.
- [2] S. Osher, A. Sol, and L. Vese, “Image decomposition and restoration using total variation minimization and the  $H^1$ ,” *Multiscale Modeling Simulation*, vol. 1, no. 3, pp. 349–370, 2003.

- [3] G. Gilboa, N. Sochen, and Y. Y. Zeevi, "Variational denoising of partly textured images by spatially varying constraints," *IEEE Transactions on Image Processing*, vol. 15, no. 8, pp. 2281–2289, 2006.
- [4] T. Chan, A. Marquina, and P. Mulet, "High-order total variation-based image restoration," *SIAM Journal on Scientific Computing*, vol. 22, no. 2, pp. 503–516, 2000.
- [5] Q. Chen, P. Montesinos, Q. S. Sun, P. A. Heng, and D. S. Xia, "Adaptive total variation denoising based on difference curvature," *Image and Vision Computing*, vol. 28, no. 3, pp. 298–306, 2010.
- [6] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, vol. 20, no. 1-2, pp. 89–97, 2004.
- [7] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.
- [8] Y.-L. You, W. Xu, A. Tannenbaum, and M. Kaveh, "Behavioral analysis of anisotropic diffusion in image processing," *IEEE Transactions on Image Processing*, vol. 5, no. 11, pp. 1539–1553, 1996.
- [9] G. Steidl, J. Weickert, T. Brox, P. Mrázek, and M. Welk, "On the equivalence of soft wavelet shrinkage, total variation diffusion, total variation regularization, and sides," *SIAM Journal on Numerical Analysis*, vol. 42, no. 2, pp. 686–713, 2004.
- [10] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 421–432, 1998.
- [11] M. Lysaker, A. Lundervold, and X.-C. Tai, "Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time," *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1579–1589, 2003.
- [12] T. Brox, J. Weickert, B. Burgeth, and P. Mrázek, "Nonlinear structure tensors," *Image and Vision Computing*, vol. 24, no. 1, pp. 41–55, 2006.
- [13] S. Mallat and W. L. Hwang, "Singularity detection and processing with wavelets," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 617–643, 1992.
- [14] J. Zhong and H. Sun, "Wavelet-based multiscale anisotropic diffusion with adaptive statistical analysis for image restoration," *IEEE Transactions on Circuits and Systems. I. Regular Papers*, vol. 55, no. 9, pp. 2716–2725, 2008.
- [15] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.
- [16] M. K. Mihçak, I. Kozintsev, K. Ramchandran, and P. Moulin, "Low-complexity image denoising based on statistical modeling of wavelet coefficients," *IEEE Signal Processing Letters*, vol. 6, no. 12, pp. 300–303, 1999.
- [17] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the 6th International Conference on Computer Vision*, pp. 839–846, January 1998.
- [18] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, 2003.
- [19] A. Buades, B. Coll, and J. M. Morel, "A non-local algorithm for image denoising," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 2, pp. 60–65, 2005.
- [20] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on Image Processing*, vol. 13, no. 9, pp. 1200–1212, 2004.
- [21] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proceedings of the 7th IEEE International Conference on Computer Vision (ICCV '99)*, vol. 2, pp. 1033–1038, September 1999.
- [22] C. Kervrann, J. Boulanger, and P. Coup, "Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal," in *Scale Space and Variational Methods in Computer Vision*, F. Sgallari, A. Murli, and N. Paragios, Eds., vol. 4485 of *Lecture Notes in Computer Science*, pp. 520–532, Springer, Berlin, Germany, 2007.
- [23] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.
- [24] T. Tasdizen, "Principal neighborhood dictionaries for nonlocal means image denoising," *IEEE Transactions on Image Processing*, vol. 18, no. 12, pp. 2649–2660, 2009.
- [25] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.
- [26] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proceedings of the 12th International Conference on Computer Vision (ICCV '09)*, pp. 2272–2279, Kyoto, Japan, October 2009.
- [27] G. Gilboa and S. Osher, "Nonlocal operators with applications to image processing," *Multiscale Modeling & Simulation*, vol. 7, no. 3, pp. 1005–1028, 2009.
- [28] T. Brox, O. Kleinschmidt, and D. Cremers, "Efficient nonlocal means for denoising of textural patterns," *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1083–1092, 2008.
- [29] L. Pizarro, P. Mrázek, S. Didas, S. Grewenig, and J. Weickert, "Generalised nonlocal image smoothing," *International Journal of Computer Vision*, vol. 90, no. 1, pp. 62–87, 2010.
- [30] Z. Yang and M. Jacob, "Nonlocal regularization of inverse problems: a unified variational framework," *IEEE Transactions on Image Processing*, vol. 22, no. 8, pp. 3192–3203, 2013.
- [31] D. Tschumperlé and L. Brun, "Non-local image smoothing by applying anisotropic diffusion PDE's in the space of patches," in *Proceedings of the IEEE 16th International Conference on Image Processing (ICIP '09)*, pp. 2957–2960, November 2009.
- [32] J. Salmon, "On two parameters for denoising with non-local means," *IEEE Signal Processing Letters*, vol. 17, no. 3, pp. 269–272, 2010.
- [33] M. Mahmoudi and G. Sapiro, "Fast image and video denoising via nonlocal means of similar neighborhoods," *IEEE Signal Processing Letters*, vol. 12, no. 12, pp. 839–842, 2005.
- [34] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 490–530, 2005.
- [35] M. Lebrun, A. Buades, and J. M. Morel, "A nonlocal Bayesian image denoising algorithm," *SIAM Journal on Imaging Sciences*, vol. 6, no. 3, pp. 1665–1688, 2013.
- [36] Y.-L. Liu, J. Wang, X. Chen, Y.-W. Guo, and Q.-S. Peng, "A robust and fast non-local means algorithm for image denoising," *Journal of Computer Science and Technology*, vol. 23, no. 2, pp. 270–279, 2008.

- [37] M. Lebrun, M. Colom, A. Buades, and J. M. Morel, "Secrets of image denoising cuisine," *Acta Numerica*, vol. 21, pp. 475–576, 2012.
- [38] P. Milanfar, "A tour of modern image filtering," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 106–123, 2013.
- [39] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "BM3D image denoising with shape-adaptive principal component analysis," in *Proceedings of the Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS'09)*, 2009.
- [40] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognition*, vol. 43, no. 4, pp. 1531–1549, 2010.
- [41] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE Transactions on Image Processing*, vol. 16, no. 5, pp. 1395–1411, 2007.
- [42] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, Academic Press, New York, NY, USA, 3rd edition, 2008.
- [43] G. Nason and B. Silverman, "The stationary wavelet transform and some statistical applications," in *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim, Eds., vol. 103 of *Lecture Notes in Statistics*, pp. 281–299, Springer, New York, NY, USA, 1995.
- [44] H. Zheng, A. Bouzerdoum, and S. L. Phung, "Wavelet based nonlocal-means super-resolution for video sequences," in *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP '10)*, pp. 2817–2820, Hong Kong, September 2010.
- [45] M. Z. Iqbal, A. Ghafoor, and A. M. Siddiqui, "Satellite image resolution enhancement using dual-tree complex wavelet transform and nonlocal means," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 3, pp. 451–455, 2013.
- [46] M. Lebrun, A. Buades, and J.-M. Morel, "Implementation of the "Non-Local Bayes" (NL-Bayes) image denoising algorithm," *Image Processing On Line*, vol. 3, pp. 1–42, 2013.
- [47] A. Buades, B. Coll, and J.-M. Morel, "Non-local means denoising," *Image Processing On Line*, vol. 1, 2011.
- [48] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: learning sparse dictionaries for sparse signal approximation," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, part 2, pp. 1553–1564, 2010.

## Research Article

# Applying BAT Evolutionary Optimization to Image-Based Visual Servoing

Marco Perez-Cisneros,<sup>1</sup> Gerardo Garcia-Gil,<sup>1</sup> Sabrina Vega-Maldonado,<sup>2</sup>  
J. Arámburo-Lizárraga,<sup>2</sup> Erik Cuevas,<sup>1</sup> and Daniel Zaldivar<sup>1</sup>

<sup>1</sup>Departamento de Ingenierías CUTONALA, Universidad de Guadalajara, Avenida N. Periférico No. 555 Ejido San José Tateposco, 48525 Tonalá, JAL, Mexico

<sup>2</sup>Departamento de Sistemas de Información CUCEA, Universidad de Guadalajara, Periférico Norte 799, Los Belenes, 45100 Zapopan, JAL, Mexico

Correspondence should be addressed to Marco Perez-Cisneros; marco.perez@cutonala.udg.mx

Received 11 October 2014; Accepted 14 May 2015

Academic Editor: Sabri Arik

Copyright © 2015 Marco Perez-Cisneros et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a predictive control strategy for an image-based visual servoing scheme that employs evolutionary optimization. The visual control task is approached as a nonlinear optimization problem that naturally handles relevant visual servoing constraints such as workspace limitations and visibility restrictions. As the predictive scheme requires a reliable model, this paper uses a local model that is based on the visual interaction matrix and a global model that employs 3D trajectory data extracted from a quaternion-based interpolator. The work assumes a free-flying camera with 6-DOF simulation whose results support the discussion on the constraint handling and the image prediction scheme.

## 1. Introduction

Past decades have witnessed the extensive development of the visual servoing (VS) control. Three fundamental schemes have practically represented most of VS implementations [1, 2]. First, the image-based visual control (IBVS), also known as 2DVS, employs an error computation between the visual features belonging to a target object in a given image and its corresponding features in a target image. Such error is subsequently employed as guidance for the visual control algorithm just as it is carefully detailed in the following. Second, the position-based visual servoing (PBVS), also named as 3DVS, works entirely on the visual computation of geometric poses, whose values are subsequently used to regulate the camera movement. Likewise, a third group is represented by a wide number of hybrid VS approaches that generally profit over a mindful combination between 2DVS and 3DVS advantages.

In particular, the classic IBVS control problem is defined as an exponential minimization of the aforementioned image plane error between the current and target image

$\mathbf{e}(t) = \mathbf{s}(t) - \mathbf{s}^*$ . In turn, such error can be subjected to a classic minimization procedure assuming a gradient-like approach such that  $\dot{\mathbf{e}} = -\lambda\mathbf{e}(t)$ . A well-known relationship between the object's velocity and its corresponding image plane velocity can thus be defined by stocking each point velocity relationship into a single matrix  $\mathbf{L}_s$  known as the interaction matrix or the visual Jacobian. Mathematically, the overall velocity relationship can thus be defined (see [1]) as  $\dot{\mathbf{s}} = \mathbf{L}_s\mathbf{v}(t)$  with  $\mathbf{L}_s$  being the interaction matrix and  $\mathbf{v}(t)$  representing the velocity screw vector over time. A classical feedback control law can thus be defined as

$$\boldsymbol{\tau}(t) = -\lambda\widehat{\mathbf{L}}_s^+\mathbf{e}(t). \quad (1)$$

In this case,  $\widehat{\mathbf{L}}_s^+$  is the pseudoinverse of the image Jacobian matrix and  $\lambda$  a negative constant, with  $\boldsymbol{\tau}(t)$  being the resulting control signal. Despite the implementation of such VS scheme being fairly simple, some important drawbacks have been highlighted by Chaumette in [3], with unstable behavior arising from the tracking of large displacements and complex rotations, or from the generation of nonfeasible motions.

Therefore, the handling of either the 2D constraints or the 3D limitations, as well as the generation of feasible trajectories for a given visual task, must be all appropriately addressed.

Two constraints must be appropriately handled in order to assure an appropriate visual control behavior: first, the well-known visibility constraint that refers to the adequate handling of the control problem in order to assure that visual features always remain within the camera field of view and second the 3D constraint that challenges the generation of convenient visual servoing schemes that yield admissible camera motions within a valid workspace.

The use of optimal control fundamentals for visual servoing has been defined as an appropriate and convenient tool to build visual servoing schemes that carefully considered the aforementioned visual constraints. Actually, several applications have been reported in the literature over the last two decades. First, the seminal works of Hashimoto and Kimura in [4] and Schramm and Morel in [5] that incorporated an LQ-based optimal control scheme and a Kalman filter-based algorithm, respectively, in order to guide the movements of a robotic manipulator.

Other approaches have capitalized the advantages of the LMI approach to build predictive control schemes for visual servoing [6, 7]. Despite the fact that such works have focused over the designing of an appropriate control law for the visual servoing scheme, other proposals have also included optimal schemes for the combination of path planning and trajectory tracking in order to assure the fulfilment of the visibility constraint and the generation of an optimal trajectory for the camera. Excellent examples of such combination can be found in the works of Schramm and Morel in [8] and the use of LMI structures made by Chesi in [9]. In the particular case of path planning, it is important to consider the work of Mezouar and Chaumette [10] and the robust approach proposed later by Kazemi et al. in [11]. In this case, an LMI based algorithm is used to define an optimal path planning solution assuming that not a unique solution for the problem may exist and also that it may not be unique, while the required camera tracking is supplied through a classic image-based visual controller [12].

Other optimal VS control implementations include the use of predictive control to compensate for errors in the tracking task of a visual feedback scheme in case of no prior information about the 3D model being supplied to the visual controller [13] or in the case of using active filtering through predictive control for biomedical applications that support robotized surgery [14].

Recently, the strategy to incorporate the handling of both visual constraints, that is, the visibility and the feasible motion constraint, within the visual control structure has been focused on expressing the overall visual task from a nonlinear optimization perspective. Therefore, this paper presents a novel optimization scheme that employs an evolutionary optimization method to handle both constraints through a visual predictive control scheme. Under such circumstances, 3D constraints can be considered as state variables while the visibility constraint can be assumed just as a constraint within the output space, just as it has been done in [12]. In order to provide an appropriate model prediction agent, two options

are to be considered following the proposal of Allibert and Courtial in [15]. First, a local model uses the classic image Jacobian matrix while a second test uses a quaternion-based 3D trajectory generator. As it will be carefully discussed, the optimization algorithm uses prediction to improve the overall visual servoing performance by means of a predictive control structure that has been specifically designed to fit within the visual control scheme.

Just as it has been widely demonstrated, the use of optimization within the visual servoing control scheme has delivered some relevant contributions in particular for the image-based schemes that naturally handle the most important visual constraints at the same time control signals are generated with remarkable examples being found in [12, 15–17]. However all these solutions use classic optimization methods in order to minimize an objective function since the goal of an optimization scheme is to find an acceptable solution of a given objective function that is defined over a given search space; novel methods that are known as Evolutionary Methods have been proposed as a handy alternative.

In particular, evolutionary algorithms (EA), which are considered as stochastic optimization methods, have been developed by a combination of rules and randomness that mimics several natural phenomena that include some evolutionary processes such as the evolutionary algorithm (EA) proposed by Fogel et al. [18], De Jong [19], and Koza [20], the Genetic Algorithm (GA) proposed by Holland [21] and Goldberg [22], the Artificial Immune System proposed by de Castro and bon Zuben [23], and the Differential Evolution Algorithm (DE) proposed by Storn and Price [24]. Some other methods which are based on physical processes include the Simulated Annealing proposed by Kirkpatrick et al. [25], the Electromagnetism-Like Algorithm proposed by Birbil and Fang [26], and the Gravitational Search Algorithm proposed by Rashedi et al. [27]. Also, there are other methods based on the animal-behavior phenomena such as the Particle Swarm Optimization (PSO) algorithm proposed by Kennedy & Eberhart [28], the Ant Colony Optimization (ACO) algorithm proposed by Dorigo et al. [29], and the BAT algorithm proposed by Yang [30], which is of special importance for this paper.

In particular, this paper approaches the IBVS from an optimization-like perspective that naturally supports the inclusion of visual constraints in the implementation of the vision-based control scheme. As a result, the overall performance of the visual servoing scheme is improved at the same time that the aforementioned constraints are carefully taken into consideration.

The paper has been developed as follows. Section 2 presents an overview of the overall optimization strategy, the control scheme, and its mathematical formulation, as well as the management of image-based constraints that support the optimal IBVS approach. Section 3 focuses on the principles of the BAT optimization algorithm and its basic operational principles. Section 4 discusses the local and global models that are required in the image-prediction scheme, which in turn are represented by the classic IBVS control algorithm and the quaternion-based guidance. Section 5 presents some simulation of the free flying 6-DOF camera in order to

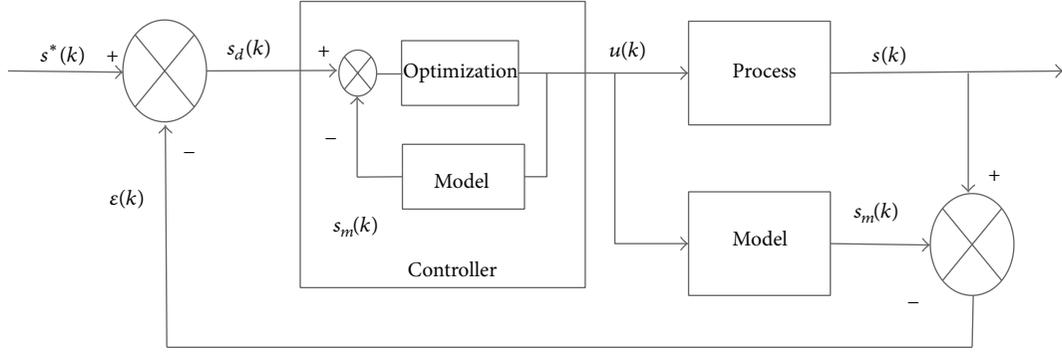


FIGURE 1: Predictive visual control scheme.

demonstrate the active contribution of the the algorithm's tracking performance and discuss the differences between using the local or the global model for prediction. The last section draws some final conclusions.

## 2. An Optimization Approach to IBVS

**2.1. Structure of the Control Scheme.** One of the most successful strategies to incorporate optimization into a feedback control scheme is beyond any doubt of the predictive control. In turn, one of the most well-known structures for predictive control is the internal model control approach [31], whose basic structure has been customized for the image-based visual servoing in the work of Allibert [12]. The basic structure is reproduced in Figure 1 where the robot and its attached camera are modelled inside the plant block. The control input to the system is represented by  $u$  while the output has been marked as  $s$  which represents the image plane coordinates of four selected features to track in the image of the object of interest. However, as it is typical in IBVS, the scheme requires the definition of desired (target) locations for the object features in the image, typically represented by  $s^*$ . By making use of the error model for IBVS from (1), the predictive control is based upon a generalized error that is defined by the difference between the current plant output at time  $k$  and the corresponding model output. Define such generalized error  $\varepsilon(k)$  as the difference of the system's output  $s(k)$  and the predicted model output  $s_m(k)$ , yielding  $\varepsilon(k) = s(k) - s_m(k)$ , at time  $k$ . The algorithm should assure that a desired trajectory of visual features on the image plane follows an adequate sequence of points in order to guarantee the fulfillment of both visual constraint that have been mentioned earlier. Therefore, an easy definition for the required trajectory  $s_d(k)$  can be defined as the difference between the target feature locations  $s^*(k)$  and the preregistered plant-model error  $\varepsilon(k)$ , which in turn generates the following expression:

$$s_d(k) = s^*(k) - \varepsilon(k). \quad (2)$$

The overall error  $\varepsilon(k)$  that includes the plant-model difference at time  $k$  can be included yielding:

$$s_d(k) = s^*(k) - [s(k) - s_m(k)]. \quad (3)$$

A very interesting fact emerges as the overall equation is rewritten as follows:

$$s_d(k) - s_m(k) = s^*(k) - s(k). \quad (4)$$

This last expression holds a key issue for the optimization approach of IBVS schemes. The minimization of the difference between the desired visual features location  $s^*(k)$  and the system's output  $s(k)$  corresponds to minimizing the difference between the required visual trajectory  $s_d(k)$  and the model output  $s_m(k)$ . Actually, the last fact supports the operation of the optimization algorithm that is to be completed if an objective function and some operative rules are defined as it is discussed below.

**2.2. Building the Mathematical Framework.** As explained above, the definition for the predictive control structure depends on drawing an appropriate objective function of the form:  $\min_{\tilde{u} \in K} J(u)$ , which will yield a control sequence of the form:

$$\tilde{u} = \{u(k), u(k+1), \dots, u(k+N_c), \dots, u(k+N_p-1)\} \quad (5)$$

with  $N_p$  and  $N_c$  representing the prediction and control horizon, respectively. The prediction horizon represents the amount of forecast terms to be calculated in advance from the model while the control horizon holds the number of calculated terms that are actually applied to control the plant [32]. In the particular IBVS implementation, only the first term of the control horizon is actually applied to the system [12].

Considering that the overall problem is managed over the image plane and that the visibility constraint is referred to the image plane, the objective function can be initially defined as follows:

$$J(u) = \sum_{j=k+1}^{k+N_p} [s_d(j) - s_m(j)]^T Q(j) [s_d(j) - s_m(j)] \quad (6)$$

with  $Q(j)$  being a weighting symmetric definite-positive matrix with the dynamics of the system being described by the nonlinear system:  $x(j) = f[x(j-1), u(j-1)]$  and

$s_m = h(x(j))$  with  $s_d(j) = s^*(j) - \varepsilon(j)$  and  $x(j)$  representing the predicted state at time  $j$  with  $\forall j \in [k + 1, k + N_p]$ . The variables for the state and the control signal are defined according to  $x, u \in \mathbb{R}^n$  and the model output as  $s_m \in \mathbb{R}^p$ . It is important to note that the state computation can vary depending on the particular prediction model that is employed. This issue is carefully addressed in the following.

**2.3. Constraints of the Predictive IBVS.** Since one of the immediate advantages of the optimization-like approach of the IBVS control is the natural handling of inner constraints of the visual challenge, it is important to denote how such constraints are to be managed by the proposed structure.

The most important constraints have been previously identified as the visibility constraint and the 3D motion constraint. The case for the visibility constraint is also known as the 2D condition. It aims to assure that the location of object's features of interest for the visual algorithm always remains within a valid location in the image plane. On the contrary, this property can be used to denote inconvenient areas within the image. In terms of the optimization algorithm, the constraint is simply introduced under the limit  $s_{\text{low}} \leq s_m(k) \leq s_{\text{up}}$  that includes both the lowest and the highest accepted location within the image space.

On the other hand, the generation of valid 3D trajectories also can be easily included in the optimization process. Since each robotic device must comply with mechanical and dynamic limitation due to workspace limits or actuator saturation, each kinematic pose can be defined in terms of the corresponding instantaneous generalized coordinates  $\mathbf{q}(k)$ . Likewise, the overall pose  $\mathbf{T}(q)$  can also be geometrically constraint under well-known properties such as the full-rank in the instantaneous Jacobian matrix [33]. Therefore, again both considerations can be introduced under the following expressions:

$$\begin{aligned} {}^0\mathbf{T}_{n_{\text{low}}} &\leq {}^0\mathbf{T}_n(k) \leq {}^0\mathbf{T}_{n_{\text{up}}}, \\ \mathbf{q}_{\text{low}} &\leq \mathbf{q}(k) \leq \mathbf{q}_{\text{up}} \end{aligned} \quad (7)$$

with  ${}^0\mathbf{T}_{n_{\text{low}}}$  and  ${}^0\mathbf{T}_{n_{\text{up}}}$  being minimum and maximum allowed pose while  $\mathbf{q}_{\text{low}}$  and  $\mathbf{q}_{\text{up}}$  represent the lowest and the highest generalized coordinate limits. In a similar fashion, it is even feasible to include other mechanical constraints such as actuator limits or torque or force constraints [12]. The optimization procedure commonly includes all aforementioned constraints in the form of nonlinear expression that can be evaluated as the overall predictive algorithm evolves.

**2.4. Optimal Approach to IBVS.** This section discusses the step by step implementation of the control structure presented by Figure 1. Two parameters are of vital importance in the implementation of the prediction horizon and the control horizon value. Both will in turn coordinate the extent of the optimization influence inside the predictive control scheme. The visual predictive controller and its corresponding optimization cycle are initially computed starting from the error calculation. Such error value is subsequently used to draw the

desired trajectory over the number of steps that are defined by the prediction horizon  $N_p$ . A step-by-step description is presented below.

- (1) Current location of visual features  $s(k)$  is registered.
- (2) Calculate the value of error  $\varepsilon(k) = s(k) - s_m(k)$ , assuming it is kept constant during the number of steps included in the prediction horizon  $N_p$ ; that is,  $\forall j \in [k + 1, k + N_p]$ .
- (3) Compute the desired trajectory according to  $s_d(j) = s^*(j) - \varepsilon(j)$ , also  $\forall j \in [k + 1, k + N_p]$ .
- (4) The measured current feature location in the image plane  $s(k)$  is employed to initialize the model output  $s_m(k)$ , which in turn constitutes the feedback loop that is required by the internal model control structure [31].
- (5) The optimal control signal  $u(k)$  is defined by the optimization algorithm according to (5); its value is kept constant over the interval  $u(k + N_c + 1)$  to  $u(k + N_p - 1)$ , with  $N_c$  and  $N_p$  being the control and the prediction horizon, respectively.

Evidently, the two most important parameters in the optimization process are the  $N_c$  and  $N_p$ . The value of  $N_p$  is vital to guarantee an adequate equilibrium between the system stability and the computational feasibility of the overall implementation. A high value of  $N_p$  implies the generation of softer control signals while a small value allows a wider exploration of novel control values at the cost of weakening the overall system stability. On the other hand, the control horizon value  $N_c$  regulates how many steps forward are required to reach the objective. A high value of  $N_c$  accounts for a slower control behavior that is not feasible for visual control implementations. In practical terms, the value of  $N_c$  is commonly assigned to 1, which corresponds to keeping the control signal constant over the number of steps previously defined by the prediction horizon  $N_p$ .

Finally, it is important to discuss about weighting matrix  $Q(j)$  that is a third participant of the optimization configuration set. Its value is commonly assigned to an identity matrix of dimension  $(p, p)$ , despite some successful examples of using a time-varying matrix values in order to increase the sensibility to the error value when the steps are close to reach the horizon value  $N_p$  [15].

Since the overall mathematical design of the visual predictive control has been envisioned, the study should turn to discuss over the feasibility of employing an evolutionary optimization algorithm to increase the performance of a predictive visual control structure.

### 3. The BAT Evolutionary Algorithm

Approaching the IBVS from an optimization-like perspective naturally supports the inclusion of visual constraints in the implementation of the vision-based control scheme. In particular, the use of the BAT evolutionary algorithm as the main optimization procedure provides an easy implementation while still delivering an acceptable performance.

The analogy supporting the BAT algorithm is based on the echolocation ability that is exhibited by microbats in their quest for food. Bats generate an ultrasonic beam that can vary its pulse frequency or its intensity which is commonly known as the loudness in the algorithm. The ultrasonic signal is delivered in advance to their movements. By using loudness variations, intensity variations between both ears and the time delay in receiving the signals back, bats are able to reconstruct an overall scenario despite the fact that they may move through a varying context.

The BAT algorithm is therefore built over the assumption that they fly at random while looking for food. Such movement is registered at position  $\mathbf{x}_i$  with velocity  $\mathbf{v}_i$  assuming the bat emits a fixed frequency  $f_{\min}$  and a variable with loudness  $A_0$ . During the search, the pulse emission rate  $r \in [0, 1]$  can vary in accordance to the proximity of the target. For simplicity the analogy considers frequency to fall in the interval  $f \in [0, f_{\max}]$  assuming that higher frequencies imply a shorter travelling distance. Under the same simple assumption, the rate pulse is computed as  $r \in [0, 1]$ , with 1 representing the maximum rate of pulse emission [30].

The heart of the BAT algorithm is centered over the location computation for virtual bats. The movement is administered over the  $d$ -dimensional search space by updating positions  $\mathbf{x}_i$  and velocities  $\mathbf{v}_i$  as follows:

$$\mathbf{x}_i(k) = \mathbf{x}_i(k-1) + \mathbf{v}_i(k), \quad (8)$$

$$\mathbf{v}_i(k) = \mathbf{v}_i(k-1) + [\mathbf{x}_i(k) - \mathbf{x}^*] \cdot f_i, \quad (9)$$

$$f_i = f_{\min} + (f_{\max} - f_{\min}) \cdot \beta. \quad (10)$$

In this case,  $\mathbf{x}^*$  represents the current global best solution after assessing all current available solutions. Frequency  $f_i$  is computed through the difference between  $f_{\max}$  and  $f_i$ , with  $\beta$  being a uniformly distributed random value. In practice, the value of  $f_i$  is of vital importance because it controls the movement scope for each particle.

In a similar fashion, loudness and pulse rate are defined with regard of the bat's analogy. Loudness  $A$  should decrease as the bat is approaching its prey. A range between 0 and 1 is used to implement this feature, with  $A = 0$  being used when one bat (searching particle) has found a prey (minimum) and therefore is not emitting a sound signal; otherwise  $A = 1$  signals when the bat is searching through the space and therefore producing its maximum sound. An easy implementation considers a variable  $0 < \alpha < 1$ , yielding

$$A_i(k+1) = \alpha \cdot A_i(k). \quad (11)$$

The pulse emission rate  $r$  is defined under a similar scheme but assuring an exponential decay on its influence as time is evolving as follows:

$$r_i(k+1) = r_i(0) [1 - e^{-\gamma t}]. \quad (12)$$

With  $t$  representing sampling time,  $t \rightarrow \infty$ , and  $\gamma$  being a tuning constant. Avoiding any loss of generality,  $r_i(0)$  accounts for the last updated value of pulse emission rate. It must be noticed that according to the original BAT

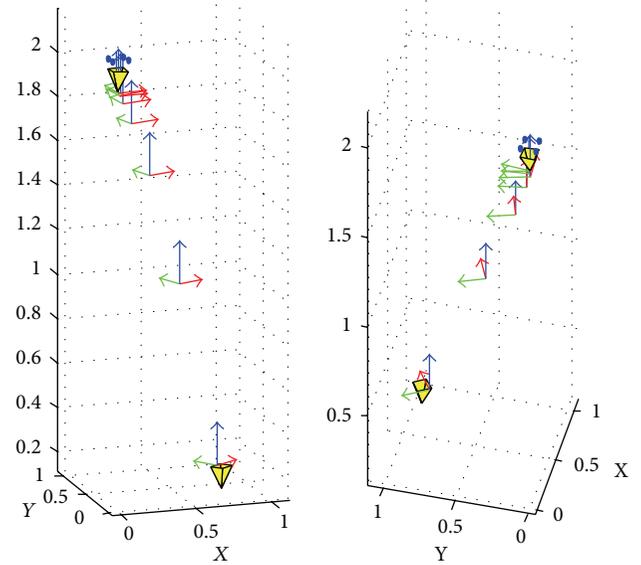
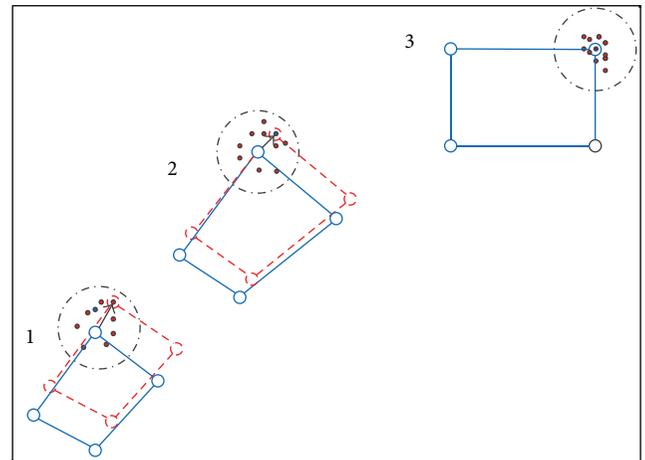


FIGURE 2: Classic visual servoing scheme.



- Particles generated by the heuristic algorithm BAT
- Particle generated by the quaternion spherical linear interpolation.
- Coordinates of the desired position  $f(x, y)$

FIGURE 3: Contribution of the slerp quaternion-based interpolation within the optimal visual servoing scheme.

implementation [30], loudness and pulse emission rate will only be updated if new solutions are improved which means that searching particles are moving closer to an optimal solution.

The overall BAT algorithm can be summarized over the following pseudocode:

- (1) Define the initial population  $\mathbf{x}_i$  ( $i = 1, 2, \dots, n$ ) and initial velocities in vector  $\mathbf{v}_i$ .
- (2) Select a pulse frequency  $f_i$  at  $\mathbf{x}_i$  and pulse rates  $r_i$  and loudness values  $A_i$ .
- (3) Do until get to  $k$  number of iterations:

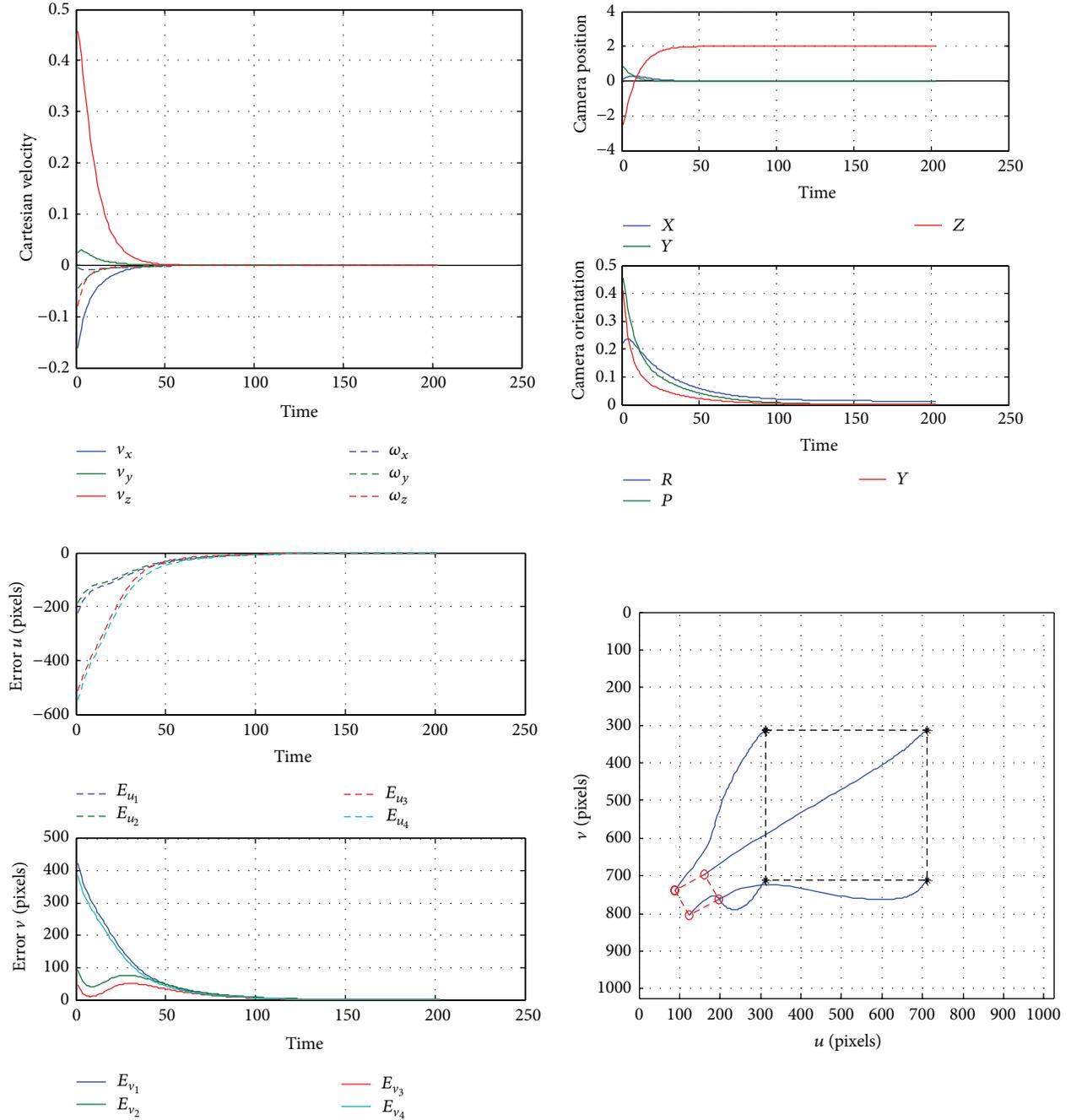


FIGURE 4: Evolution of the visual servoing control scheme, showing (clockwise from upper left) the screw vector evolution, pose errors, feature trajectories in the image plane, and feature trajectories in the horizontal and vertical directions.

- (3.1) Calculate new solutions through frequency  $f_i$ .
- (3.2) Update velocity and location for each bat (particle), using (8) and (9).
- (3.3) Generate a random value  $r$  and compare to  $r_i$ .
- (3.4) If  $r > r_i$ , one solution among best solutions must be chosen. A new local solution must be generated around that selected best solution.
- (3.5) Using a random bat's fly (particle movement), generate a new solution.
- (3.6) If  $f(x_i) < f(x^*)$  and  $r < A_i$ , then accept new solutions, increase  $r_i$ , and decrease  $A_i$ .
- (3.7) Reevaluate all particles to find the current new best  $x^*$ .
- (4) Publish results.

In practical grounds, the use of a similar value for  $\alpha$  and  $\gamma$  yields a similar treatment for decreasing the loudness and for increasing the emitted pulses rate at the time

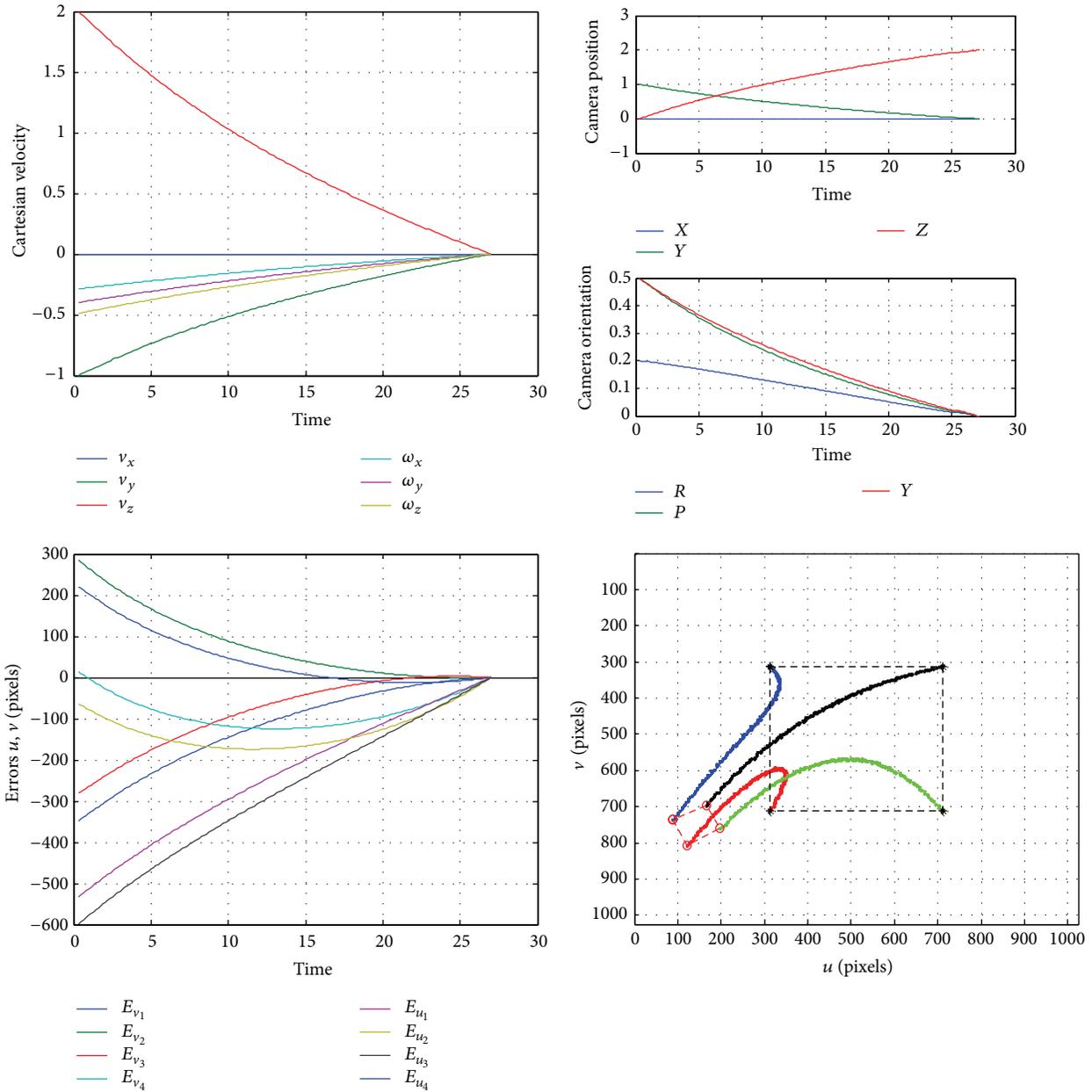


FIGURE 5: Predictive visual servoing control scheme using the BAT evolutionary algorithm with (clockwise from upper left) the screw vector evolution, pose errors, feature trajectories in the image plane, and feature trajectories in the horizontal and vertical directions.

a prey (minimum) is being located. Appropriate values for both values must be experimentally determined. In our implementation a simple selection of  $\alpha = \gamma = 0.9$  has been used with good results. Several other BAT implementations with different parameters settings are reported in [32].

#### 4. The Local and Global Mathematical Models for Optimization

Once the BAT optimization algorithm has been carefully developed, the discussion should turn into the selection of

the plant model that is to guide the predictive control strategy. The model is used to predict the movement of visual features with respect to the camera velocity over a finite prediction horizon [12]. The classic visual servoing scheme can be easily implemented to provide a simple model, denoted as SM in the following. Such model uses fundamental equations to define the movements on the image plane as a result of the camera movement. This paper has also explored the use of a quaternion-based interpolation to generate the required location of the visual features over the image plane. Fundamentals of the classic visual servoing model, denoted as SM, are sketched below.

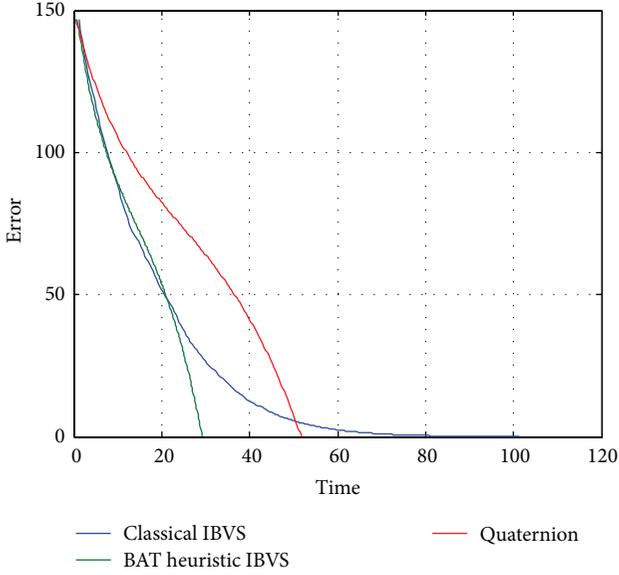


FIGURE 6: Time performance of the three algorithms: the predictive visual servoing control scheme, the classic IBVS scheme, and a pure-quaternion guidance of the visual servoing task.

**4.1. Local Model: The Classic Visual Servoing Scheme.** The most well-known visual servoing scheme employs point-like image features as main reference. The location of such features in the image plane is denoted as  $s_m = (u, v)$ , with  $m$  referring to the feature number and  $u$  and  $v$  representing the vertical and horizontal image plane coordinates, respectively. For a given 3D point in the space, that is,  $p = [x, y, z]^T$ , that is defined with respect to the camera frame, its projection into the image plane is easily defined in normalized coordinates assuming  $u = x/z$  and  $v = y/z$ . Likewise, the camera velocity is defined through the classic screw vector  $w = [d_x, d_y, d_z, w_x, w_y, w_z]^T$ . A careful review of the fundamentals of visual servoing in [1] explains the velocity relationship from the camera velocity  $w$  to the image plane feature's velocity through the following relationship:

$$\dot{s}_m(t) = L_s(t) w(t), \quad (13)$$

where  $L_s(t)$  represents the image Jacobian or interaction matrix which holds the required velocity relationship. The overall matrix is built by iteratively piling up the following matrix for each feature that is being characterized, as follows:

$$L_s = \begin{bmatrix} -\frac{1}{z} & 0 & \frac{u}{z} & uv & -(1+u^2) & v \\ 0 & -\frac{1}{z} & \frac{v}{z} & 1+v^2 & -uv & -u \end{bmatrix}. \quad (14)$$

The location of each feature can thus be defined through a simple integration method as follows:

$$s_m(k+1) = s_m(k) + \hat{L}_s(t) \cdot T_e \cdot w(t) \quad (15)$$

with  $T_e$  representing the sample time,  $w$  defining the screw vector, and  $\hat{L}_s(t)$  accounting for the best estimation of  $L_s(t)$ ,

considering that depth information is required and must be either calculated through a 3D model of the object of interest or through a careful assessment of its approximated value. In this paper, we use the advantage of holding a full 3D model of the image formation and therefore its computation can be exactly defined. Following the procedure in [12], the optimization process can be acutely described by considering each feature  $s_m$  as a state  $x$  which allows expressing the overall optimization function as follows:

$$\begin{aligned} x(k+1) &= x(k) + \hat{L}_s(t) \cdot T_e \cdot w(t) \\ &\rightarrow f(x(k), w(t)), \\ s_m(k) &= x(k) \rightarrow h(x(k)). \end{aligned} \quad (16)$$

By using such optimization function, 2D constraints are naturally handled within the SM visual servoing model.

**4.2. Global Model: Spherical Interpolation.** A global model is required to generate an alternative option in order to support the optimization contribution within the visual servoing scheme. In this case, the quaternion-based interpolation, also known as slerp [34], is a very useful tool considering its intrinsic advantages such as the smooth interpolation, fast concatenation and simple inversion of angular displacements, and a quick conversion to homogeneous transforms.

The classic visual servoing problem considers both a start and a target pose which, in turn, can be easily expressed in quaternion grounds. Once both are converted, the interpolation is easily computed by the following expression:

$$\begin{aligned} \text{slerp}(q_0, q_1, t) &= (q_1 q_0^{-1})^t q_0 \rightarrow q_0 = T(0); \\ q_1 &= T_d. \end{aligned} \quad (17)$$

With  $t$  being the step index,  $0 < t < 1$  whose value defines the interpolation step, with  $t = 1$  signaling for the last step in the sequence. The spherical interpolation is incorporated into the predictive control strategy as the model guiding the BAT optimization algorithm. It compares the fitness of the proposed interpolated pose through its corresponding features and those features that are generated by each proposed particle. The comparison variables can be easily explained by the following three cases that are illustrated by Figure 3, as follows:

- (1) Particles generated by the BAT evolutionary algorithm are depicted in bold circles while the particle generated by the spherical linear interpolation is represented in a void circle. The algorithm compares the fitness value for each candidate solution. In this case, one of the BAT generated particles is selected.
- (2) In the second case, the particle generated by the slerp interpolator is chosen as it has obtained the best fitness value.
- (3) The third case shows when the algorithm has reached the desired location and the visual servoing task is finished.

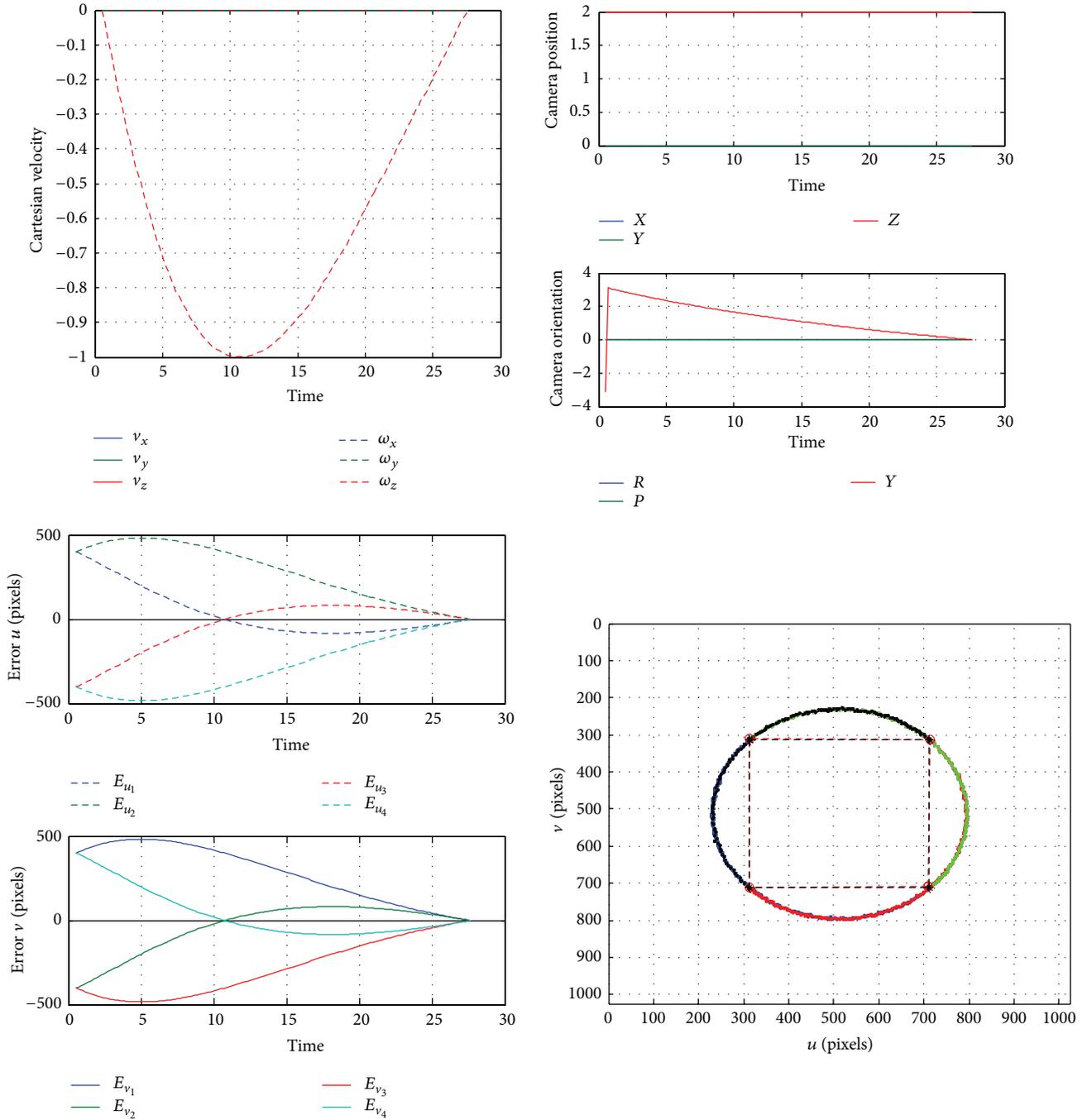


FIGURE 7: Predictive visual servoing control scheme using the BAT evolutionary algorithm to solve the half turn problem, with (clockwise from upper left) the screw vector evolution, pose errors, feature trajectories in the image plane, and each feature trajectory in the horizontal and vertical directions.

### 5. Simulations

For all the simulations in this paper, the sample time is constraint by the number of frames per second that is provided by the camera. Since a 30 frames per second device is used, the sampling period is  $t = 33 \text{ ms}$ . A free-flying camera is located in an initial position defined by vector  $\mathbf{x}(0) = [0, 1, -3, 0.2, 0.5, 0.5]^T$ . First three values are defined

in meters while last three components, the roll, pitch, and yaw angles, are referred in radians. Figure 2 shows the initial and final pose for the camera. It also illustrates the four features that represent the object of interest. The simulation uses the classic SM model with the classic image Jacobian matrix.

Figure 4 shows the evolution of the visual servoing control, with the graphs showing, in clockwise order from the upper left corner, the resulting screw vector evolution, the

camera pose error, considering location and orientation, and the image plane feature trajectories and the image errors in the horizontal and vertical direction, respectively.

The predictive visual servoing scheme is tested over the same initial conditions. 20 bat particles are initialized to random positions; the pulse emission rate and the loudness value are defined to  $\alpha = \gamma = 0.7$ , with the decay variable  $\alpha$  falling within the range of 0 to 1.

Figure 5 presents the results of the predictive visual servoing scheme with a BAT optimization algorithm in the control feedback. The graphs show the evolution of the visual servoing control featuring, in clockwise order from the upper left corner, the resulting screw vector evolution, the camera pose error, considering location and orientation, and the image plane feature trajectories and the image errors in the horizontal and vertical direction, respectively.

In practical grounds, the classic visual servoing scheme required 39.62 seconds to complete the full servoing task. On the other hand, the BAT-based predictive visual controller required only 13.28 seconds in order to accomplish the same assignment.

A very handy comparison of the BAT visual predictive controller is related to the execution time of such algorithms and their contrast to the use of a quaternion-based simple interpolator. Figure 6 shows the time improvement of the BAT-based predictive visual control scheme.

Figure 6 shows a time comparison between the classic IBVS method, the pure-quaternion solution, and the BAT-based predictive visual control scheme. The BAT-based scheme shows a sharper trajectory in the image plane as a result of the improvement related to the optimal use of the solution holding the best fitness value. It is important to remind that the scheme is also capable of naturally handling both of the required visual constraints.

A challenging second experiment is performed in order to demonstrate the handling of both aforementioned visual constraints. A typical problem in the classic visual servoing scheme emerges when the required movement is a rotation of  $\pi$  radians over the vertical axis of a given object; its visual feature tracking typically tends to fail in the control law, because such movement implies a sudden backwards movement of the camera, which in turn yields a tracking failure for the object of interest features.

The use of the predictive visual control scheme with the BAT optimization algorithm in the feedback loop yields a solution for the aforementioned problem. Thanks to the particle delivery and the fitness evaluation for each particle during the control phase, the controller manages to generate appropriate spherical trajectories that avoid any instability due to the feature trajectory crossing in the image plane [3]. Figure 7 shows an illustration of the optimal controller response to such circumstance.

Notice in Figure 7 that the generated screw vector requires velocity only in the direction of  $\omega_x$  as the controller aims to turn the overall problem 180 degrees. A schematic view is illustrated by Figure 8. It is evident how the required movement is simply a rotation around the vertical axis. The required time to complete the turning of image features reaches the 27 seconds. The feature trajectories in the image

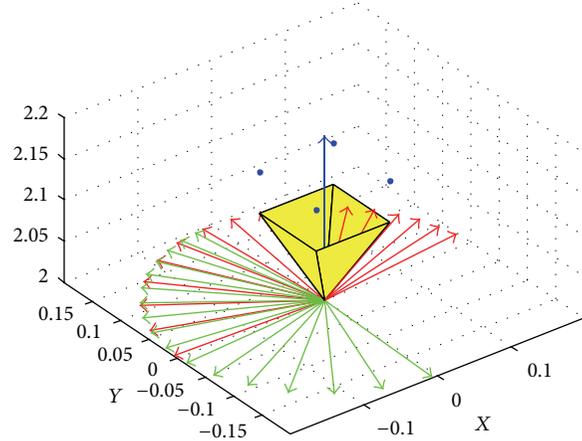


FIGURE 8: 3D view of the visual servoing control for the half turn problem.

plane seem to describe a circular movement as a result of the contribution from the quaternion-based interpolator.

## 6. Conclusions

This research has demonstrated the usefulness of a predictive control strategy for an image-based visual servoing scheme that employs an evolutionary optimization algorithm to improve the performance of the servoing task. The visual control task is approached as an optimization problem that naturally handles relevant visual servoing constraints such as workspace limitations and visibility restrictions. Two models have been used in the implementation: a simple model based on the classic visual servoing scheme and a quaternion-based slerp interpolator. The mindful contribution of both models is controlled through a particular objective function that evaluates the fitness of the proposed interpolated pose through its corresponding features and those features that are generated by each proposed particle. In practical grounds, the spherical interpolation is incorporated into the predictive control strategy as the model guiding the BAT optimization algorithm. The simulation results support the contribution of the proposed scheme regarding the handling of required visual constraints.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] F. Chaumette and S. Hutchinson, "Visual servo control. I. Basic approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.
- [2] F. Chaumette and S. Hutchinson, "Visual servo control. II. Advanced approaches," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 109–118, 2007.

- [3] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The Confluence of Vision and Control*, vol. 237 of *Lecture Notes in Control and Information Sciences*, pp. 66–78, Springer, London, UK, 1998.
- [4] K. Hashimoto and H. Kimura, "LQ optimal and nonlinear approaches to visual servoing," in *Visual Servoing*, K. Hashimoto, Ed., vol. 7 of *World Scientific Series in Robotics and Intelligent Systems*, pp. 165–198, World Scientific, Singapore, 1993.
- [5] F. Schramm and G. Morel, "Ensuring visibility in calibration-free path planning for image-based visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 22, no. 4, pp. 848–854, 2006.
- [6] S. Durola, P. Dan, D. F. Coutinho, and M. Courdresses, "Rational systems and matrix inequalities to the multicriteria analysis of visual servos," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, pp. 1504–1509, IEEE, Kobe, Japan, May 2009.
- [7] P. Danés and D. Bellot, "Towards an LMI approach to multicriteria visual servoing in robotics," *European Journal of Control*, vol. 12, no. 1, pp. 86–110, 2006.
- [8] F. Schramm and G. Morel, "Ensuring visibility in calibration-free path planning for image-based visual servoing," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 848–854, 2006.
- [9] G. Chesi, "Visual servoing path planning via homogeneous forms and LMI optimizations," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 281–291, 2009.
- [10] Y. Mezouar and F. Chaumette, "Optimal camera trajectory with image-based control," *International Journal of Robotics Research*, vol. 22, no. 10–11, pp. 781–803, 2003.
- [11] M. Kazemi, K. Gupta, and M. Mehrandezh, "Global path planning for robust visual servoing in complex environments," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, pp. 326–332, Kobe, Japan, May 2009.
- [12] G. Allibert, E. Courtial, and F. Chaumette, "Predictive control for constrained image-based visual servoing," *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 933–939, 2010.
- [13] J. A. Gangloff and M. F. De Mathelin, "Visual servoing of a 6-DOF manipulator for unknown 3-D profile following," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 4, pp. 511–520, 2002.
- [14] R. Ginhoux, J. Gangloff, M. de Mathelin, L. Soler, M. M. A. Sanchez, and J. Marescaux, "Active filtering of physiological motion in robotized surgery using predictive control," *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 67–79, 2005.
- [15] G. Allibert and E. Courtial, "What can prediction bring to image-based visual servoing?" in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '09)*, pp. 5210–5215, St. Louis, Mo, USA, October 2009.
- [16] G. Allibert, E. Courtial, and Y. Touré, "Visual predictive control for manipulators with catadioptric camera," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA' 08)*, pp. 510–515, Pasadena, Calif, USA, May 2008.
- [17] G. Allibert, E. Courtial, and Y. Touré, "Real-time visual predictive controller for image-based trajectory tracking of mobile robot," in *Proceedings of the 17th IFAC World Congress*, Seoul, Republic of Korea, July 2008.
- [18] L. J. Fogel, A. J. Owens, and M. J. Walsh, *Artificial Intelligence through Simulated Evolution*, John Wiley, Chichester, UK, 1966.
- [19] K. De Jong, *Analysis of the behavior of a class of genetic adaptive systems [Ph.D. thesis]*, University of Michigan, Ann Arbor, Mich, USA, 1975.
- [20] J. R. Koza, "Genetic programming: a paradigm for genetically breeding populations of computer programs to solve problems," Report No. STAN-CS-90-1314, Stanford University, Stanford, Calif, USA, 1990.
- [21] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Mich, USA, 1975.
- [22] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, Boston, Mass, USA, 1989.
- [23] L. N. de Castro and F. J. bon Zuben, "Artificial immune systems: part I—basic theory and applications," Tech. Rep. TR-DCA 01/99, 1999.
- [24] R. Storn and K. Price, "Differential evolution—a simple and efficient adaptive scheme for global optimisation over continuous spaces," Tech. Rep. TR-95–012, ICSI, Berkeley, Calif, USA, 1995.
- [25] S. Kirkpatrick, J. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [26] Ş. İ. Birbil and S.-C. Fang, "An electromagnetism-like mechanism for global optimization," *Journal of Global Optimization*, vol. 25, no. 3, pp. 263–282, 2003.
- [27] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "Filter modeling using gravitational search algorithm," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 1, pp. 117–122, 2011.
- [28] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, December 1995.
- [29] M. Dorigo, V. Maniezzo, and A. Colnari, "Positive feedback as a search strategy," Tech. Rep. 91-016, Politecnico di Milano, 1991.
- [30] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, J. R. González, D. A. Pelta, C. Cruz, G. Terrazas, and N. Krasnogor, Eds., vol. 284 of *Studies in Computational Intelligence*, pp. 65–74, Springer, Berlin, Germany, 2010.
- [31] M. Morari and E. Zafriou, *Robust Process Control*, Prentice-Hall, 1989.
- [32] L. Wang, *Model Predictive Control System Design and Implementation using MATLAB*, Springer, 2009.
- [33] P. Corke, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, Springer Tracts in Advanced Robotics, 1st edition, 2011.
- [34] F. Dunn and I. Parberry, *3D Math Primer for Graphics and Game Development*, A K Peters, CRC Press, 2nd edition, 2011.

## Research Article

# Multithreshold Segmentation by Using an Algorithm Based on the Behavior of Locust Swarms

Erik Cuevas,<sup>1,2</sup> Adrián González,<sup>1</sup> Fernando Fausto,<sup>1</sup>  
Daniel Zaldívar,<sup>1,2</sup> and Marco Pérez-Cisneros<sup>3</sup>

<sup>1</sup>Departamento de Electrónica, Universidad de Guadalajara, CUCEI, Avenida Revolución 1500,  
44430 Guadalajara, JAL, Mexico

<sup>2</sup>Centro Tapatío Educativo AC, Vicente Guerrero 232, Colonia Zapopan Centro, 45100 Zapopan, JAL, Mexico

<sup>3</sup>Centro Universitario de Tonalá, Avenida Nuevo Periférico No. 555 Ejido San José Tatepozco, 48525 Tonalá, JAL, Mexico

Correspondence should be addressed to Erik Cuevas; erik.cuevas@cucei.udg.mx

Received 7 October 2014; Revised 25 May 2015; Accepted 10 June 2015

Academic Editor: Bogdan Dumitrescu

Copyright © 2015 Erik Cuevas et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As an alternative to classical techniques, the problem of image segmentation has also been handled through evolutionary methods. Recently, several algorithms based on evolutionary principles have been successfully applied to image segmentation with interesting performances. However, most of them maintain two important limitations: (1) they frequently obtain suboptimal results (misclassifications) as a consequence of an inappropriate balance between exploration and exploitation in their search strategies; (2) the number of classes is fixed and known in advance. This paper presents an algorithm for the automatic selection of pixel classes for image segmentation. The proposed method combines a novel evolutionary method with the definition of a new objective function that appropriately evaluates the segmentation quality with respect to the number of classes. The new evolutionary algorithm, called Locust Search (LS), is based on the behavior of swarms of locusts. Different to the most of existent evolutionary algorithms, it explicitly avoids the concentration of individuals in the best positions, avoiding critical flaws such as the premature convergence to suboptimal solutions and the limited exploration-exploitation balance. Experimental tests over several benchmark functions and images validate the efficiency of the proposed technique with regard to accuracy and robustness.

## 1. Introduction

Image segmentation [1] consists in grouping image pixels based on some criteria such as intensity, color, and texture and still represents a challenging problem within the field of image processing. Edge detection [2], region-based segmentation [3], and thresholding methods [4] are the most popular solutions for image segmentation problems.

Among such algorithms, thresholding is the simplest method. It works by considering threshold (points) values to adequately separate distinct pixels regions within the image being processed. In general, thresholding methods are divided into two types depending on the number of threshold values, namely, bilevel and multilevel. In bilevel thresholding, only a threshold value is required to separate the two objects of an image (e.g., foreground and background). On the other hand, multilevel thresholding divides pixels into more

than two homogeneous classes that require several threshold values.

The thresholding methods use a parametric or non-parametric approach [5]. In parametric approaches [6, 7], it is necessary to estimate the parameters of a probability density function that is capable of modelling each class. A nonparametric technique [8–11] employs a given criteria such as the between-class variance or the entropy and error rate, in order to determine optimal threshold values.

A common method to accomplish parametric thresholding is the modeling of the image histogram through a Gaussian mixture model [12] whose parameters define a set of pixel classes (threshold points). Therefore, each pixel that belongs to a determined class is labeled according to its corresponding threshold points with several pixel groups gathering those pixels that share a homogeneous grayscale level.

The problem of estimating the parameters of a Gaussian mixture that better model an image histogram has been commonly solved through the Expectation Maximization (EM) algorithm [13, 14] or gradient-based methods such as Levenberg-Marquardt, LM [15]. Unfortunately, EM algorithms are very sensitive to the choice of the initial values [16], meanwhile gradient-based methods are computationally expensive and may easily get stuck within local minima [17].

As an alternative to classical techniques, the problem of Gaussian mixture identification has also been handled through evolutionary methods. In general, they have demonstrated to deliver better results than those based on classical approaches in terms of accuracy and robustness [18]. Under these methods, an individual is represented by a candidate Gaussian mixture model. Just as the evolution process unfolds, a set of evolutionary operators are applied in order to produce better individuals. The quality of each candidate solution is evaluated through an objective function whose final result represents the similarity between the mixture model and the histogram. Some examples of these approaches involve optimization methods such as Artificial Bee Colony (ABC) [19], Artificial Immune Systems (AIS) [20], Differential Evolution (DE) [21], Electromagnetism Optimization (EO) [22], Harmony Search (HS) [23], and Learning Automata (LA) [24]. Although these algorithms own interesting results, they present two important limitations. (1) They frequently obtain suboptimal approximations as a consequence of a limited balance between exploration and exploitation in their search strategies. (2) They are based on the assumption that the number of Gaussians (classes) in the mixture is preknown and fixed; otherwise, they cannot work. The cause of the first limitation is associated with their evolutionary operators employed to modify the individual positions. In such algorithms, during their evolution, the position of each agent for the next iteration is updated yielding an attraction towards the position of the best particle seen so far or towards other promising individuals. Therefore, as the algorithm evolves, these behaviors cause that the entire population rapidly concentrates around the best particles, favoring the premature convergence and damaging the appropriate exploration of the search space [25, 26]. The second limitation is produced as a consequence of the objective function that evaluates the similarity between the mixture model and the histogram. Under such an objective function, the number of Gaussians functions in the mixture is fixed. Since the number of threshold values (Gaussian functions) used for image segmentation varies depending on the image, the best threshold number and values are obtained by an exhaustive trial and error procedure.

On the other hand, bioinspired algorithms represent a field of research that is concerned with the use of biology as a metaphor for producing optimization algorithms. Such approaches use our scientific understanding of biological systems as an inspiration that, at some level of abstraction, can be represented as optimization processes.

In the last decade, several optimization algorithms have been developed by a combination of deterministic rules and randomness, mimicking the behavior of natural phenomena. Such methods include the social behavior of bird flocking

and fish schooling such as the Particle Swarm Optimization (PSO) algorithm [27] and the emulation of the differential evolution in species such as the Differential Evolution (DE) [28]. Although PSO and DE are the most popular algorithms for solving complex optimization problems, they present serious flaws such as premature convergence and difficulty to overcome local minima [29, 30]. The cause for such problems is associated with the operators that modify individual positions. In such algorithms, during the evolution, the position of each agent for the next iteration is updated yielding an attraction towards the position of the best particle seen so far (in case of PSO) or towards other promising individuals (in case of DE). As the algorithm evolves, these behaviors cause that the entire population rapidly concentrates around the best particles, favoring the premature convergence and damaging the appropriate exploration of the search space [31, 32].

Recently, the collective intelligent behavior of insect or animal groups in nature has attracted the attention of researchers. The intelligent behavior observed in these groups provides survival advantages, where insect aggregations of relatively simple and “unintelligent” individuals can accomplish very complex tasks using only limited local information and simple rules of behavior [33]. Locusts (*Schistocerca gregaria*) are a representative example of such collaborative insects [34]. Locust is a kind of grasshopper that can change reversibly between a solitary and a social phase, with clear behavioral differences among both phases [35]. The two phases show many differences regarding the overall level of activity and the degree to which locusts are attracted or repulsed among them [36]. In the solitary phase, locusts avoid contact to each other (locust concentrations). As consequence, they distribute throughout the space, exploring sufficiently over the plantation [36]. On the other hand, in the social phase, locusts frantically concentrate around those elements that have already found good food sources [37]. Under such a behavior, locusts attempt to efficiently find better nutrients by devastating promising areas within the plantation.

This paper presents an algorithm for the automatic selection of pixel classes for image segmentation. The proposed method combines a novel evolutionary method with the definition of a new objective function that appropriately evaluates the segmentation quality with regard to the number of classes. The new evolutionary algorithm, called Locust Search (LS), is based on the behavior presented in swarms of locusts. In the proposed algorithm, individuals emulate a group of locusts which interact to each other based on the biological laws of the cooperative swarm. The algorithm considers two different behaviors: solitary and social. Depending on the behavior, each individual is conducted by a set of evolutionary operators which mimics different cooperative conducts that are typically found in the swarm. Different to most of existent evolutionary algorithms, the behavioral model in the proposed approach explicitly avoids the concentration of individuals in the current best positions. Such fact allows avoiding critical flaws such as the premature convergence to suboptimal solutions and the incorrect exploration-exploitation balance. In order to automatically define the optimal number of pixel classes

(Gaussian functions in the mixture), a new objective function has been also incorporated. The new objective function is divided into two parts. The first part evaluates the quality of each candidate solution in terms of its similarity with regard to the image histogram. The second part penalizes the overlapped area among Gaussian functions (classes). Under these circumstances, Gaussian functions that do not “positively” participate in the histogram approximation could be easily eliminated in the final Gaussian mixture model.

In order to illustrate the proficiency and robustness of the proposed approach, several numerical experiments have been conducted. Such experiments are divided into two sections. In the first part, the proposed LS method is compared to other well-known evolutionary techniques over a set of benchmark functions. In the second part, the performance of the proposed segmentation algorithm is compared to other segmentation methods which are also based on evolutionary principles. The results in both cases validate the efficiency of the proposed technique with regard to accuracy and robustness.

This paper is organized as follows: in Section 2 basic biological issues of the algorithm analogy are introduced and explained. Section 3 describes the novel LS algorithm and its characteristics. A numerical study on different benchmark function is presented in Section 4 while Section 5 presents the modelling of an image histogram through a Gaussian mixture. Section 6 exposes the LS segmentation algorithm and Section 7 the performance of the proposed segmentation algorithm. Finally, Section 8 draws some conclusions.

## 2. Biological Fundamentals and Mathematical Models

Social insect societies are complex cooperative systems that self-organize within a set of constraints. Cooperative groups are good at manipulating and exploiting their environment, defending resources and breeding, yet allowing task specialization among group members [38, 39]. A social insect colony functions as an integrated unit that not only possesses the ability to operate at a distributed manner but also undertakes a huge construction of global projects [40]. It is important to acknowledge that global order for insects can arise as a result of internal interactions among members.

Locusts are a kind of grasshoppers that exhibit two opposite behavioral phases: solitary and social (gregarious). Individuals in the solitary phase avoid contact to each other (locust concentrations). As consequence, they distribute throughout the space while sufficiently exploring the plantation [36]. In contrast, locusts in the gregarious phase gather into several concentrations. Such congregations may contain up to  $10^{10}$  members, cover cross-sectional areas of up to  $10 \text{ km}^2$ , and a travelling capacity up to  $10 \text{ km}$  per day for a period of days or weeks as they feed causing a devastating crop loss [41]. The mechanism to switch from the solitary phase to the gregarious phase is complex and has been a subject of significant biological inquiry. Recently, a set of factors has been implicated to include geometry of the vegetation landscape and the olfactory stimulus [42].

Only few works [36, 37] that mathematically model the locust behavior have been published. Both approaches develop two different minimal models with the goal of reproducing the macroscopic structure and motion of a group of locusts. Considering that the method in [36] focuses on modelling the behavior for each locust in the group, its fundamentals have been employed to develop the algorithm that is proposed in this paper.

*2.1. Solitary Phase.* This section describes how each locust's position is modified as a result of its behavior under the solitary phase. Considering that  $\mathbf{x}_i^k$  represents the current position of the  $i$ th locust in a group of  $N$  different elements, the new position  $\mathbf{x}_i^{k+1}$  is calculated by using the following model:

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \Delta \mathbf{x}_i, \quad (1)$$

with  $\Delta \mathbf{x}_i$  corresponding to the change of position that is experimented by  $\mathbf{x}_i^k$  as a consequence of its social interaction with all other elements in the group.

Two locusts in the solitary phase exert forces on each other according to basic biological principles of attraction and repulsion (see, e.g., [36]). Repulsion operates quite strongly over a short length scale in order to avoid concentrations. Attraction is weaker and operates over a longer length scale, providing the social force that is required to maintain the group's cohesion. Therefore, the strength of such social forces can be modeled by the following function:

$$s(r) = F \cdot e^{-r/L} - e^{-r}. \quad (2)$$

Here,  $r$  is a distance,  $F$  describes the strength of attraction, and  $L$  is the typical attractive length scale. We have scaled the time and the space coordinates so that the repulsive strength and length scale are both represented by the unity. We assume that  $F < 1$  and  $L > 1$  so that repulsion is stronger and features in a shorter-scale, while attraction is applied in a weaker and longer-scale; both facts are typical for social organisms [21]. The social force exerted by locust  $j$  over locust  $i$  is

$$\mathbf{s}_{ij} = s(r_{ij}) \cdot \mathbf{d}_{ij}, \quad (3)$$

where  $r_{ij} = \|\mathbf{x}_j - \mathbf{x}_i\|$  is the distance between the two locusts and  $\mathbf{d}_{ij} = (\mathbf{x}_j - \mathbf{x}_i)/r_{ij}$  is the unit vector pointing from  $\mathbf{x}_i$  to  $\mathbf{x}_j$ . The total social force on each locust can be modeled as the superposition of all of the pairwise interactions:

$$\mathbf{S}_i = \sum_{\substack{j=1 \\ j \neq i}}^N \mathbf{s}_{ij}. \quad (4)$$

The change of position  $\Delta \mathbf{x}_i$  is modeled as the total social force experimented by  $\mathbf{x}_i^k$  as the superposition of all of the pairwise interactions. Therefore,  $\Delta \mathbf{x}_i$  is defined as follows:

$$\Delta \mathbf{x}_i = \mathbf{S}_i. \quad (5)$$

In order to illustrate the behavioral model under the solitary phase, Figure 1 presents an example, assuming a population of

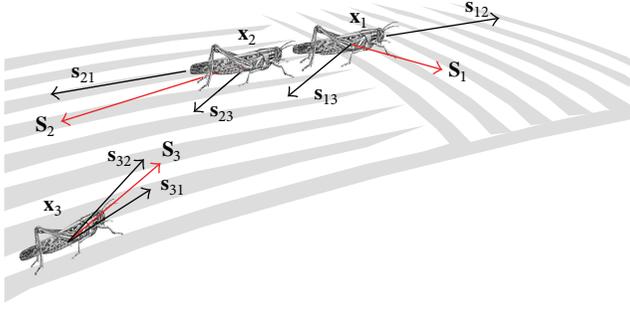


FIGURE 1: Behavioral model under the solitary phase.

three different members ( $N = 3$ ) which adopt a determined configuration in the current iteration  $k$ . As a consequence of the social forces, each element suffers an attraction or repulsion to other elements depending on the distance among them. Such forces are represented by  $s_{12}$ ,  $s_{13}$ ,  $s_{21}$ ,  $s_{23}$ ,  $s_{31}$ , and  $s_{32}$ . Since  $x_1$  and  $x_2$  are too close, the social forces  $s_{12}$  and  $s_{13}$  present a repulsive nature. On the other hand, as the distances  $\|x_1 - x_3\|$  and  $\|x_2 - x_3\|$  are quite long, the social forces  $s_{13}$ ,  $s_{23}$ ,  $s_{31}$ , and  $s_{32}$  between  $x_1 \leftrightarrow x_3$  and  $x_2 \leftrightarrow x_3$  all belong to the attractive nature. Therefore, the change of position  $\Delta x_1$  is computed as the vector resultant between  $s_{12}$  and  $s_{13}$  ( $\Delta x_1 = s_{12} + s_{13}$ ) is  $S_1$ . The values  $\Delta x_2$  and  $\Delta x_3$  are also calculated accordingly.

In addition to the presented model [36], some studies [43–45] suggest that the social force  $s_{ij}$  is also affected by the dominance of the involved individuals  $x_i$  and  $x_j$  in the pairwise process. Dominance is a property that relatively qualifies the capacity of an individual to survive, in relation to other elements in a group. The locust's dominance is determined by several characteristics such as size, chemical emissions, and location with regard to food sources. Under such circumstances, the social force is magnified or weakened depending on the most dominant individual that is involved in the repulsion-attraction process.

**2.2. Social Phase.** In this phase, locusts frantically concentrate around the elements that have already found good food sources. They attempt to efficiently find better nutrients by devastating promising areas within the plantation. In order to simulate the social phase, the food quality index  $Fq_i$  is assigned to each locust  $x_i$  of the group as such index reflects the quality of the food source where  $x_i$  is located.

Under this behavioral model, each of the  $N$  elements of the group is ranked according to its corresponding food quality index. Afterwards, the  $b$  elements featuring the best food quality indexes are selected ( $b \ll N$ ). Considering a concentration radius  $R_c$  that is created around each selected element, a set of  $c$  new locusts is randomly generated inside  $R_c$ . As a result, most of the locusts will be concentrated around the best  $b$  elements. Figure 2 shows a simple example of the behavioral model under the social phase. In the example, the configuration includes eight locusts ( $N = 8$ ), just as it is illustrated by Figure 2(a) that also presents the food quality index for each locust. A food quality index near to one indicates a better food source. Therefore, Figure 2(b) presents the final configuration after the social phase, assuming  $b = 2$ .

### 3. The Locust Search (LS) Algorithm

In this paper, some behavioral principles drawn from a swarm of locusts have been used as guidelines for developing a new swarm optimization algorithm. The LS assumes that entire search space is a plantation, where all the locusts interact to each other. In the proposed approach, each solution within the search space represents a locust position inside the plantation. Every locust receives a food quality index according to the fitness value of the solution that is symbolized by the locust's position. As it has been previously discussed, the algorithm implements two different behavioral schemes: solitary and social. Depending on each behavioral phase, each individual is conducted by a set of evolutionary operators which mimics the different cooperative operations that are typically found in the swarm. The proposed method formulates the optimization problem in the following form:

$$\begin{aligned} & \text{maximize/minimize} && f(\mathbf{I}), && \mathbf{I} = (I_1, \dots, I_n) \in \mathbb{R}^n \\ & \text{subject to} && \mathbf{I} \in \mathbf{S}, \end{aligned} \quad (6)$$

where  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  is a nonlinear function whereas  $\mathbf{S} = \{\mathbf{I} \in \mathbb{R}^n \mid lb_d \leq I_d \leq ub_d, d = 1, \dots, n\}$  is a bounded feasible search space, which is constrained by the lower ( $lb_d$ ) and upper ( $ub_d$ ) limits.

In order to solve the problem formulated in (6), a population  $\mathbf{L}^k (\{\mathbf{I}_1^k, \mathbf{I}_2^k, \dots, \mathbf{I}_N^k\})$  of  $N$  locusts (individuals) is evolved inside the LS operation from the initial point ( $k = 0$ ) to a total *gen* number of iterations ( $k = \text{gen}$ ). Each locust  $\mathbf{I}_i^k$  ( $i \in [1, \dots, N]$ ) represents an  $n$ -dimensional vector  $\{I_{i,1}^k, I_{i,2}^k, \dots, I_{i,n}^k\}$  where each dimension corresponds to a decision variable of the optimization problem to be solved. The set of decision variables constitutes the feasible search space  $\mathbf{S} = \{\mathbf{I}_i^k \in \mathbb{R}^n \mid lb_d \leq I_{i,d}^k \leq ub_d\}$ , where  $lb_d$  and  $ub_d$  correspond to the lower and upper bounds for the dimension  $d$ , respectively. The food quality index that is associated with each locust  $\mathbf{I}_i^k$  (candidate solution) is evaluated through an objective function  $f(\mathbf{I}_i^k)$  whose final result represents the fitness value of  $\mathbf{I}_i^k$ . In the LS algorithm, each iteration of the evolution process consists of two operators: (A) solitary and (B) social. Beginning by the solitary stage, the set of locusts is operated in order to sufficiently explore the search space. On the other hand, the social operation refines existent solutions within a determined neighborhood (exploitation) subspace.

**3.1. Solitary Operation (A).** One of the most interesting features of the proposed method is the use of the solitary operator to modify the current locust positions. Under this approach, locusts are displaced as a consequence of the social forces produced by the positional relations among the elements of the swarm. Therefore, near individuals tend to repel each other, avoiding the concentration of elements in regions. On the other hand, distant individuals tend to attract to each other, maintaining the cohesion of the swarm. A clear difference to the original model in [20] considers that social forces are also magnified or weakened depending on the most dominant (best fitness values) individuals that are involved in the repulsion-attraction process.

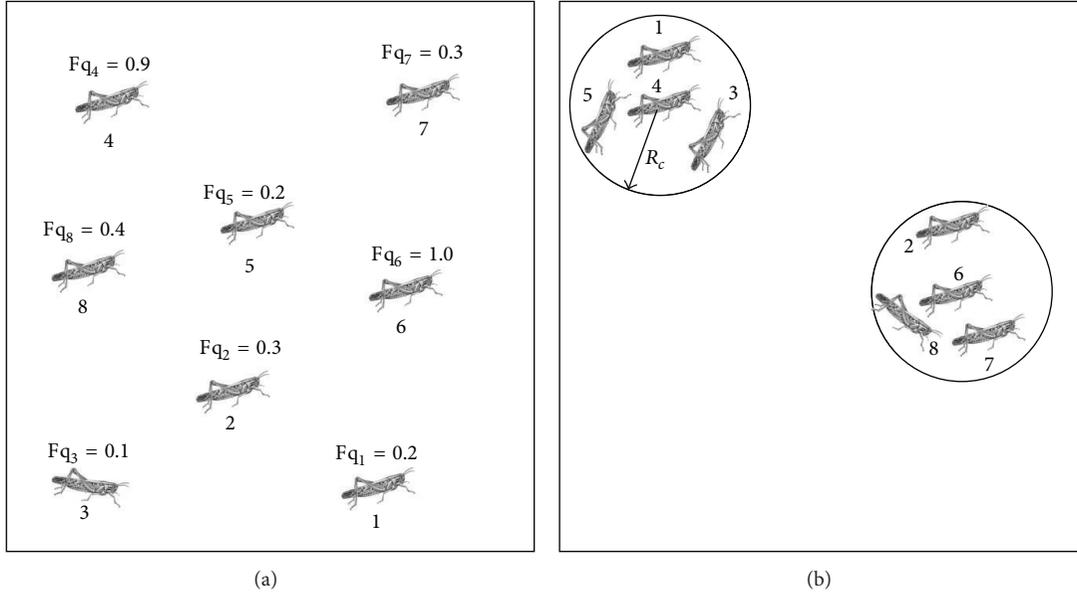


FIGURE 2: Behavioral model under the social phase. (a) Initial configuration and food quality indexes, (b) final configuration after the operation of the social phase.

In the solitary phase, a new position  $\mathbf{p}_i (i \in [1, \dots, N])$  is produced by perturbing the current locust position  $\mathbf{l}_i^k$  with a change of position  $\Delta \mathbf{l}_i (\mathbf{p}_i = \mathbf{l}_i^k + \Delta \mathbf{l}_i)$ . The change of position  $\Delta \mathbf{l}_i$  is the result of the social interactions experimented by  $\mathbf{l}_i^k$  as a consequence of its repulsion-attraction behavioral model. Such social interactions are pairwise computed among  $\mathbf{l}_i^k$  and the other  $N - 1$  individuals in the swarm. In the original model, social forces are calculated by using (3). However, in the proposed method, it is modified to include the best fitness value (the most dominant) of the individuals involved in the repulsion-attraction process. Therefore, the social force, that is exerted between  $\mathbf{l}_j^k$  and  $\mathbf{l}_i^k$ , is calculated by using the following new model:

$$\mathbf{s}_{ij}^m = \rho(\mathbf{l}_i^k, \mathbf{l}_j^k) \cdot s(r_{ij}) \cdot \mathbf{d}_{ij} + \text{rand}(1, -1), \quad (7)$$

where  $s(r_{ij})$  is the social force strength defined in (2) and  $\mathbf{d}_{ij} = (\mathbf{l}_j^k - \mathbf{l}_i^k)/r_{ij}$  is the unit vector pointing from  $\mathbf{l}_i^k$  to  $\mathbf{l}_j^k$ . Besides,  $\text{rand}(1, -1)$  is a randomly generated number between 1 and  $-1$ . Likewise,  $\rho(\mathbf{l}_i^k, \mathbf{l}_j^k)$  is the dominance function that calculates the dominance value of the most dominant individual from  $\mathbf{l}_j^k$  and  $\mathbf{l}_i^k$ . In order to operate  $\rho(\mathbf{l}_i^k, \mathbf{l}_j^k)$ , all the individuals from  $\mathbf{L}^k (\{\mathbf{l}_1^k, \mathbf{l}_2^k, \dots, \mathbf{l}_N^k\})$  are ranked according to their fitness values. The ranks are assigned so that the best individual receives the rank 0 (zero) whereas the worst individual obtains the rank  $N - 1$ . Therefore, the function  $\rho(\mathbf{l}_i^k, \mathbf{l}_j^k)$  is defined as follows:

$$\rho(\mathbf{l}_i^k, \mathbf{l}_j^k) = \begin{cases} e^{-(5 \cdot \text{rank}(\mathbf{l}_i^k)/N)} & \text{if } \text{rank}(\mathbf{l}_i^k) < \text{rank}(\mathbf{l}_j^k) \\ e^{-(5 \cdot \text{rank}(\mathbf{l}_j^k)/N)} & \text{if } \text{rank}(\mathbf{l}_i^k) > \text{rank}(\mathbf{l}_j^k), \end{cases} \quad (8)$$

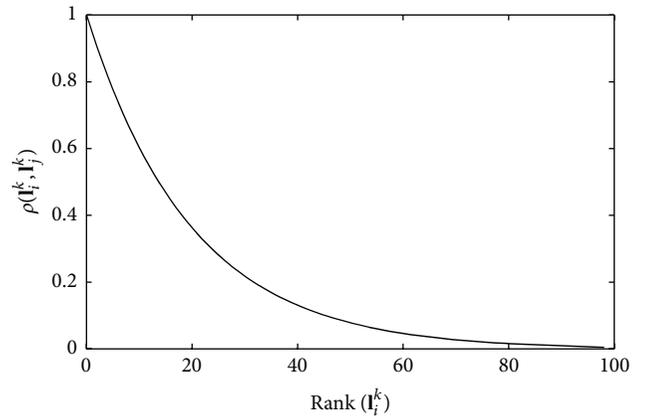


FIGURE 3: Behavior of  $\rho(\mathbf{l}_i^k, \mathbf{l}_j^k)$  considering 100 individuals.

where the function  $\text{rank}(\alpha)$  delivers the rank of the  $\alpha$ -individual. According to (8),  $\rho(\mathbf{l}_i^k, \mathbf{l}_j^k)$  yields a value within the interval  $(1, 0)$ . Its maximum value of one in  $\rho(\mathbf{l}_i^k, \mathbf{l}_j^k)$  is reached when either individual  $\mathbf{l}_j^k$  or  $\mathbf{l}_i^k$  is the best element of the population  $\mathbf{L}^k$  regarding their fitness values. On the other hand, a value close to zero is obtained when both individuals  $\mathbf{l}_j^k$  and  $\mathbf{l}_i^k$  possess quite bad fitness values. Figure 3 shows the behavior of  $\rho(\mathbf{l}_i^k, \mathbf{l}_j^k)$  considering 100 individuals. In the figure, it is assumed that  $\mathbf{l}_i^k$  represents one of the 99 individuals with ranks between 0 and 98 whereas  $\mathbf{l}_j^k$  is fixed to the element with the worst fitness value (rank 99).

Under the incorporation of  $\rho(\mathbf{l}_i^k, \mathbf{l}_j^k)$  in (7), social forces are magnified or weakened depending on the best fitness value (the most dominant) of the individuals involved in the repulsion-attraction process.

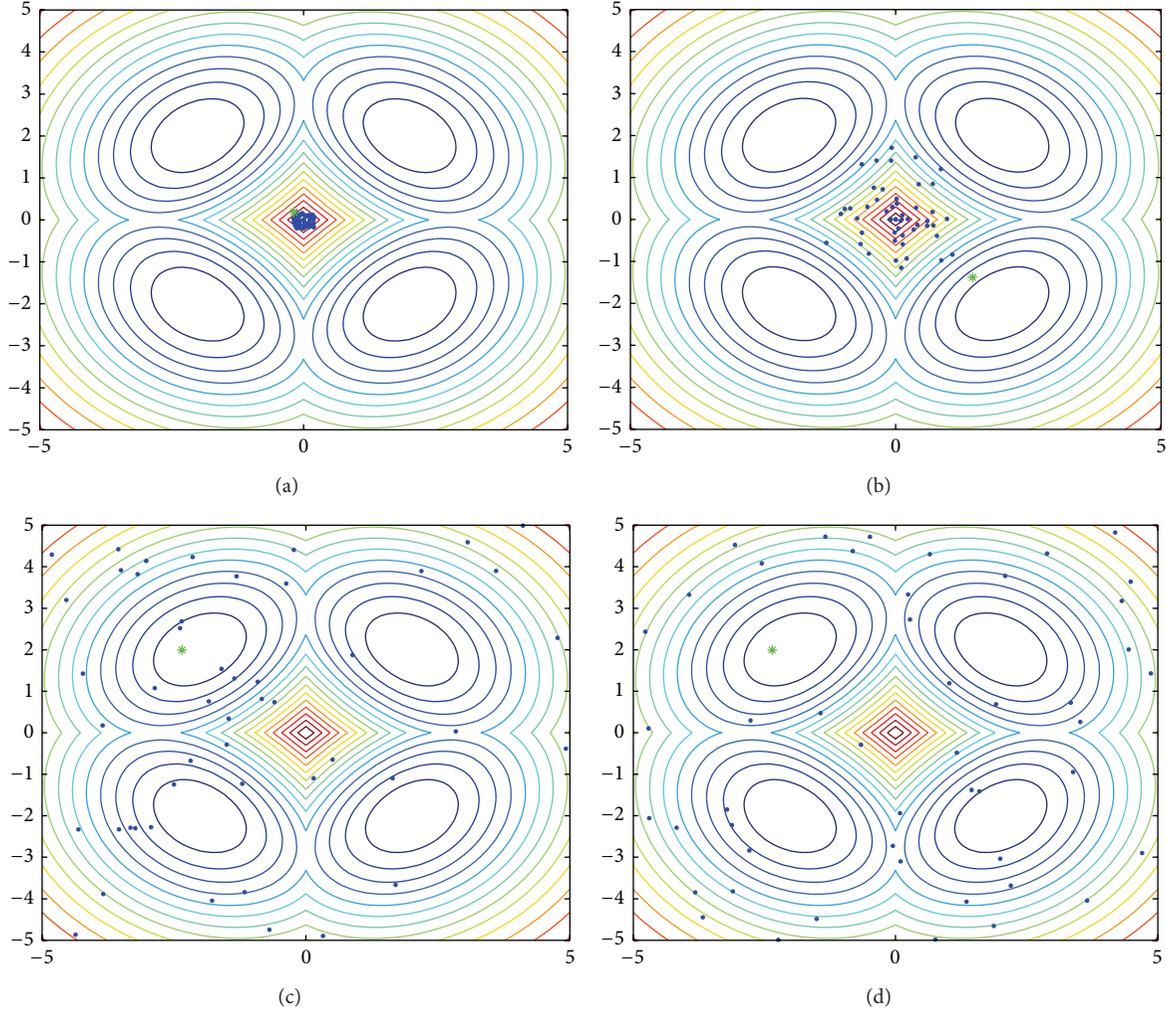


FIGURE 4: Examples of different distributions. (a) Initial condition, (b) distribution after applying 25, (c) 50 and (d) 100 operations. The green asterisk represents the minimum value so far.

Finally, the total social force on each individual  $\mathbf{l}_i^k$  is modeled as the superposition of all of the pairwise interactions exerted over it:

$$\mathbf{S}_i^m = \sum_{\substack{j=1 \\ j \neq i}}^N \mathbf{s}_{ij}^m. \quad (9)$$

Therefore, the change of position  $\Delta \mathbf{l}_i$  is considered as the total social force experimented by  $\mathbf{l}_i^k$  as the superposition of all of the pairwise interactions. Thus,  $\Delta \mathbf{l}_i$  is defined as follows:

$$\Delta \mathbf{l}_i = \mathbf{S}_i^m. \quad (10)$$

After calculating the new positions  $\mathbf{P}$  ( $\{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N\}$ ) of the population  $\mathbf{L}^k$  ( $\{\mathbf{l}_1^k, \mathbf{l}_2^k, \dots, \mathbf{l}_N^k\}$ ), the final positions  $\mathbf{F}$  ( $\{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_N\}$ ) must be calculated. The idea is to admit only the changes that guarantee an improvement in the search strategy. If the fitness value of  $\mathbf{p}_i$  ( $f(\mathbf{p}_i)$ ) is better than  $\mathbf{l}_i^k$  ( $f(\mathbf{l}_i^k)$ ), then  $\mathbf{p}_i$  is accepted as the final solution. Otherwise,

$\mathbf{l}_i^k$  is retained. This procedure can be resumed by the following statement (considering a minimization problem):

$$\mathbf{f}_i = \begin{cases} \mathbf{p}_i & \text{if } f(\mathbf{p}_i) < f(\mathbf{l}_i^k) \\ \mathbf{l}_i^k & \text{otherwise.} \end{cases} \quad (11)$$

In order to illustrate the performance of the solitary operator, Figure 4 presents a simple example with the solitary operator being iteratively applied. A population of 50 different members ( $N = 50$ ) is assumed which adopt a concentrated configuration as initial condition (Figure 4(a)). As a consequence of the social forces, the set of elements tends to distribute throughout the search space. Examples of different distributions are shown in Figures 4(b), 4(c), and 4(d) after applying 25, 50, and 100 different solitary operations, respectively.

3.2. *Social Operation (B)*. The social procedure represents the exploitation phase of the LS algorithm. Exploitation is

the process of refining existent individuals within a small neighborhood in order to improve their solution quality.

The social procedure is a selective operation which is applied only to a subset  $E$  of the final positions  $F$  (where  $E \subseteq F$ ). The operation starts by sorting  $F$  with respect to fitness values, storing the elements in a temporary population  $B = \{b_1, b_2, \dots, b_N\}$ . The elements in  $B$  are sorted so that the best individual receives the position  $b_1$  whereas the worst individual obtains the location  $b_N$ . Therefore, the subset  $E$  is integrated by only the first  $g$  locations of  $B$  (promising solutions). Under this operation, a subspace  $C_j$  is created around each selected particle  $f_j \in E$ . The size of  $C_j$  depends on the distance  $e_d$  which is defined as follows:

$$e_d = \frac{\sum_{q=1}^n (ub_q - lb_q)}{n} \cdot \beta, \quad (12)$$

where  $ub_q$  and  $lb_q$  are the upper and lower bounds in the  $q$ th dimension and  $n$  is the number of dimensions of the optimization problem, whereas  $\beta \in [0, 1]$  is a tuning factor. Therefore, the limits of  $C_j$  can be modeled as follows:

$$\begin{aligned} uss_j^q &= b_{j,q} + e_d, \\ lss_j^q &= b_{j,q} - e_d, \end{aligned} \quad (13)$$

where  $uss_j^q$  and  $lss_j^q$  are the upper and lower bounds of the  $q$ th dimension for the subspace  $C_j$ , respectively.

Considering the subspace  $C_j$  around each element  $f_j \in E$ , a set of  $h$  new particles ( $M_j^h = \{m_j^1, m_j^2, \dots, m_j^h\}$ ) are randomly generated inside bounds fixed by (13). Once the  $h$  samples are generated, the individual  $I_j^{k+1}$  of the next population  $L^{k+1}$  must be created. In order to calculate  $I_j^{k+1}$ , the best particle  $m_j^{best}$ , in terms of its fitness value from the  $h$  samples (where  $m_j^{best} \in [m_j^1, m_j^2, \dots, m_j^h]$ ), is compared to  $f_j$ . If  $m_j^{best}$  is better than  $f_j$  according to their fitness values,  $I_j^{k+1}$  is updated with  $m_j^{best}$ ; otherwise,  $f_j$  is selected. The elements of  $F$  that have not been processed by the procedure ( $f_w \notin E$ ) transfer their corresponding values to  $L^{k+1}$  with no change.

The social operation is used to exploit only prominent solutions. According to the proposed method, inside each subspace  $C_j$ ,  $h$  random samples are selected. Since the number of selected samples at each subspace is very small (typically  $h < 4$ ), the use of this operator substantially reduces the number of fitness function evaluations.

In order to demonstrate the social operation, a numerical example has been set by applying the proposed process to a simple function. Such function considers the interval of  $-3 \leq d_1, d_2 \leq 3$  whereas the function possesses one global maxima of value 8.1 at (0, 1.6). Notice that  $d_1$  and  $d_2$  correspond to the axis coordinates (commonly  $x$  and  $y$ ). For this example, a final position population  $F$  of six 2-dimensional members ( $N = 6$ ) is assumed. Figure 5 shows the initial configuration of the proposed example, with the black points representing half of the particles with the best fitness values (the first three elements of  $B$ ,  $g = 3$ ) whereas the grey points ( $f_2, f_4, f_6 \notin E$ ) correspond to the remaining individuals. From Figure 5, it

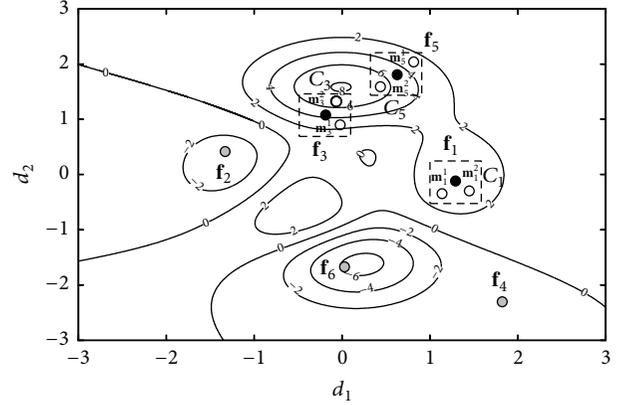


FIGURE 5: Operation of the social procedure.

can be seen that the social procedure is applied to all black particles ( $f_5 = b_1$ ,  $f_3 = b_2$ , and  $f_1 = b_3$ ,  $f_5, f_3, f_1 \in E$ ) yielding two new random particles ( $h = 2$ ), which are characterized by white points  $m_1^1, m_1^2, m_1^3, m_2^1, m_2^2, m_2^3$ , for each black point inside of their corresponding subspaces  $C_1, C_3$ , and  $C_5$ . Considering the particle  $f_3$  in Figure 7, the particle  $m_3^2$  corresponds to the best particle ( $m_3^{best}$ ) from the two randomly generated particles (according to their fitness values) within  $C_3$ . Therefore, the particle  $m_3^{best}$  will substitute  $f_3$  in the individual  $I_3^{k+1}$  for the next generation, since it holds a better fitness value than  $f_3$  ( $f(f_3) < f(m_3^{best})$ ).

The LS optimization procedure is defined over a bounded search space  $S$ . Search points that do not belong to such area are considered to be infeasible. However, during the evolution process, some candidate solutions could fall outside the search space. In the proposed approach, such infeasible solutions are arbitrarily placed with a random position inside the search space  $S$ .

**3.3. Complete LS Algorithm.** LS is a simple algorithm with only seven adjustable parameters: the strength of attraction  $F$ , the attractive length  $L$ , number of promising solutions  $g$ , the population size  $N$ , the tuning factor  $\beta$ , the number of random samples  $h$ , and the number of generations  $gen$ . The operation of LS is divided into three parts: initialization of the solitary and social operations. In the initialization ( $k = 0$ ), the first population  $L^0$  ( $\{I_1^0, I_2^0, \dots, I_N^0\}$ ) is produced. The values  $\{l_{i,1}^0, l_{i,2}^0, \dots, l_{i,n}^0\}$  of each individual  $I_i^k$  and each dimension  $d$  are randomly and uniformly distributed between the prespecified lower initial parameter bound  $lb_d$  and the upper initial parameter bound  $ub_d$ :

$$\begin{aligned} l_{i,j}^0 &= lb_d + \text{rand} \cdot (ub_d - lb_d); \\ i &= 1, 2, \dots, N; \quad d = 1, 2, \dots, n. \end{aligned} \quad (14)$$

In the evolution process, the solitary (A) and social (B) operations are iteratively applied until the number of iterations  $k = gen$  has been reached. The complete LS procedure is illustrated in Algorithm 1.

(1) <b>Input:</b> $F, L, g, N, gen, h$ and $\beta$ .	
(2) Initialize $\mathbf{L}^0$ ( $k = 0$ )	
(3) <b>until</b> ( $k = gen$ )	
(4) $\mathbf{F} \leftarrow \text{SolitaryOperation}(\mathbf{L}^k)$	Solitary operator (Section 3.1)
(5) $\mathbf{L}^{k+1} \leftarrow \text{SocialOperation}(\mathbf{L}^k, \mathbf{F})$	Social operator (Section 3.2)
(6) $k = k + 1$	
(7) <b>end until</b>	

ALGORITHM 1: Locust Search (LS) algorithm.

*3.4. Discussion about the LS Algorithm.* Evolutionary algorithms (EA) have been widely employed for solving complex optimization problems. These methods are found to be more powerful than conventional methods that are based on formal logics or mathematical programming [46]. In the EA algorithm, search agents have to decide whether to explore unknown search positions or to exploit already tested positions in order to improve their solution quality. Pure exploration degrades the precision of the evolutionary process but increases its capacity to find new potentially solutions. On the other hand, pure exploitation allows refining existent solutions but adversely drives the process to local optimal solutions. Therefore, the ability of an EA to find a global optimal solution depends on its capacity to find a good balance between the exploitation of found-so-far elements and the exploration of the search space [47].

Most of swarm algorithms and other evolutionary algorithms tend to exclusively concentrate the individuals in the current best positions. Under such circumstances, such algorithms seriously limit their exploration-exploitation capacities.

Different to most of existent evolutionary algorithms, in the proposed approach, the modeled behavior explicitly avoids the concentration of individuals in the current best positions. Such fact allows not only to emulate the cooperative behavior of the locust colony in a good realistic way but also to incorporate a computational mechanism to avoid critical flaws that are commonly present in the popular evolutionary algorithms, such as the premature convergence and the incorrect exploration-exploitation balance.

It is important to emphasize that the proposed approach conducts two operators (solitary and social) within a single iteration. Such operators are similar to those that are used by other evolutionary methods such as ABC (employed bees, onlooker bees, and scout bees), AIS (clonal proliferation operator, affinity maturation operator, and clonal selection operator), and DE (mutation, crossover, and selection), which are all executed in a single iteration.

## 4. Numerical Experiments over Benchmark Functions

A comprehensive set of 13 functions, all collected from [48–50], has been used to test the performance of the LS approach as an optimization method. Tables 11 and 12 present the benchmark functions used in our experimental study. Such

functions are classified into two different categories: unimodal test functions (Table 11) and multimodal test functions (Table 12). In these tables,  $n$  is the function dimension;  $f_{\text{opt}}$  is the minimum value of the function, with  $\mathbf{S}$  being a subset of  $R^n$ . The optimum locations ( $\mathbf{x}_{\text{opt}}$ ) for functions in Tables 11 and 12 are in  $[0]^n$ , except for  $f_5$ ,  $f_{12}$ , and  $f_{13}$  with  $\mathbf{x}_{\text{opt}}$  in  $[1]^n$  and  $f_8$  in  $[420.96]^n$ . A detailed description of optimum locations is given in Tables 11 and 12.

We have applied the LS algorithm to 13 functions whose results have been compared to those produced by the Particle Swarm Optimization (PSO) method [27] and the Differential Evolution (DE) algorithm [28], both considered as the most popular algorithms for many optimization applications. In all comparisons, the population has been set to 40 ( $N = 40$ ) individuals. The maximum iteration number for all functions has been set to 1000. Such stop criterion has been selected to maintain compatibility to similar works reported in the literature [48, 49].

The parameter setting for each of the algorithms in the comparison is described as follows:

- (1) PSO: in the algorithm,  $c_1 = c_2 = 2$  while the inertia factor ( $\omega$ ) is decreased linearly from 0.9 to 0.2.
- (2) DE: the DE/Rand/1 scheme is employed. The parameter settings follow the instructions in [28, 51]. The crossover probability is  $\text{CR} = 0.9$  and the weighting factor is  $F = 0.8$ .
- (3) In LS,  $F$  and  $L$  are set to 0.6 and  $L$ , respectively. Besides,  $g$  is fixed to 20 ( $N/2$ ),  $h = 2$ ,  $\beta = 0.6$  whereas  $gen$  and  $N$  are configured to 1000 and 40, respectively. Once such parameters have been experimentally determined, they are kept for all experiments in this section.

In the comparison, three indexes are considered: the average best-so-far solution (ABS), the standard deviation (SD), and the number of function evaluations (NFE). The first two indexes assess the accuracy of the solution whereas the last one measures the computational cost. The average best-so-far solution (ABS) expresses the average value of the best function evaluations that have been obtained from 30 independent executions. The standard deviation (SD) indicates the dispersion of the ABS values. Evolutionary methods are, in general, complex pieces of software with several operators and stochastic branches. Therefore, it is difficult to conduct a complexity analysis from a deterministic perspective. Under such circumstances, it is more appropriate

TABLE 1: Minimization results from the benchmark functions test in Table II with  $n = 30$ . Maximum number of iterations = 1000.

		PSO	DE	LS
$f_1$	ABS	$1.66 \times 10^{-1}$	$6.27 \times 10^{-3}$	$4.55 \times 10^{-4}$
	SD	$3.79 \times 10^{-1}$	$1.68 \times 10^{-1}$	$6.98 \times 10^{-4}$
	NFE	28,610	20,534	16,780
$f_2$	ABS	$4.83 \times 10^{-1}$	$2.02 \times 10^{-1}$	$5.41 \times 10^{-3}$
	SD	$1.59 \times 10^{-1}$	0.66	$1.45 \times 10^{-2}$
	NFE	28,745	21,112	16,324
$f_3$	ABS	2.75	$5.72 \times 10^{-1}$	$1.61 \times 10^{-3}$
	SD	1.01	0.15	$1.32 \times 10^{-3}$
	NFE	38,320	36,894	20,462
$f_4$	ABS	1.84	0.11	$1.05 \times 10^{-2}$
	SD	0.87	0.05	$6.63 \times 10^{-3}$
	NFE	37,028	36,450	21,158
$f_5$	ABS	3.07	2.39	$4.11 \times 10^{-2}$
	SD	0.42	0.36	$2.74 \times 10^{-3}$
	NFE	39,432	37,264	21,678
$f_6$	ABS	6.36	6.51	$5.88 \times 10^{-2}$
	SD	0.74	0.87	$1.67 \times 10^{-2}$
	NFE	38,490	36,564	22,238
$f_7$	ABS	6.14	0.12	$2.71 \times 10^{-2}$
	SD	0.73	0.02	$1.18 \times 10^{-2}$
	NFE	37,274	35,486	21,842

to use the number of function evaluations (NFE), just as it is used in the literature [52, 53], to evaluate and assess the computational effort (time) and the complexity among optimizers. It represents how many times an algorithm uses the objective function to evaluate the objective (fitness) function until the best solution of a determined execution has been found. Since the experiments require 30 different executions, the NFE index corresponds to the averaged value obtained from these executions. A small NFE value indicates that less time is needed to reach the global optimum.

**4.1. Unimodal Test Functions.** Functions  $f_1$  to  $f_7$  are unimodal functions. The results for unimodal functions over 30 runs are reported in Table 1 considering the average best-so-far solution (ABS), the standard deviation (SD), and the number of function evaluations (NFE). According to this table, LS provides better results than PSO and DE for all functions in terms of ABS and SD. In particular, the test yields the largest performance difference in functions  $f_4$ – $f_7$ . Such functions maintain a narrow curving valley that is hard to optimize, in case the search space cannot be explored properly and the direction changes cannot be kept up with [54]. For this reason, the performance differences are directly related to a better trade-off between exploration and exploitation that is produced by LS operators. In the practice, a main goal of an optimization algorithm is to find a solution as good as possible within a small time window. The computational cost for the optimizer is represented by its NFE values. According to Table 1, the NFE values that are obtained by the proposed method are smaller than its

TABLE 2:  $p$  values produced by Wilcoxon’s test that compares LS versus PSO and DE over the “average best-so-far” values from Table 1.

LS versus	PSO	DE
$f_1$	$1.83 \times 10^{-4}$	$1.73 \times 10^{-2}$
$f_2$	$3.85 \times 10^{-3}$	$1.83 \times 10^{-4}$
$f_3$	$1.73 \times 10^{-4}$	$6.23 \times 10^{-3}$
$f_4$	$2.57 \times 10^{-4}$	$5.21 \times 10^{-3}$
$f_5$	$4.73 \times 10^{-4}$	$1.83 \times 10^{-3}$
$f_6$	$6.39 \times 10^{-5}$	$2.15 \times 10^{-3}$
$f_7$	$1.83 \times 10^{-4}$	$2.21 \times 10^{-3}$

counterparts. Lower NFE values are more desirable since they correspond to less computational overload and, therefore, faster results. In the results perspective, it is clear that PSO and DE need more than 1000 generations in order to produce better solutions. However, this number of generations is considered in the experiments aiming for producing a visible contrast among the approaches. If the number of generations has been set to an exaggerated value, then all methods would converge to the best solution with no significant troubles.

A nonparametric statistical significance proof known as Wilcoxon’s rank sum test for independent samples [55, 56] has been conducted with an 5% significance level, over the “average best-so-far” data of Table 1. Table 2 reports the  $p$  values produced by Wilcoxon’s test for the pairwise comparison of the “average best-so-far” of two groups. Such groups are formed by LS versus PSO and LS versus DE. As a null hypothesis, it is assumed that there is no significant difference between mean values of the two algorithms. The alternative hypothesis considers a significant difference between the “average best-so-far” values of both approaches. All  $p$  values reported in the table are less than 0.05 (5% significance level) which is a strong evidence against the null hypothesis, indicating that the LS results are statistically significant and that it has not occurred by coincidence (i.e., due to the normal noise contained in the process).

**4.2. Multimodal Test Functions.** Multimodal functions possess many local minima which make the optimization a difficult task to be accomplished. In multimodal functions, the results reflect the algorithm’s ability to escape from local optima. We have applied the algorithms over functions  $f_8$  to  $f_{13}$  where the number of local minima increases exponentially as the dimension of the function increases. The dimension of such functions is set to 30. The results are averaged over 30 runs, with performance indexes being reported in Table 3 as follows: the average best-so-far solution (ABS), the standard deviation (SD), and the number of function evaluations (NFE). Likewise,  $p$  values of the Wilcoxon signed-rank test of 30 independent runs are listed in Table 4. From the results, it is clear that LS yields better solutions than others algorithms for functions  $f_9$ ,  $f_{10}$ ,  $f_{11}$ , and  $f_{12}$ , in terms of the indexes ABS and SD. However, for functions  $f_8$  and  $f_{13}$ , LS produces similar results to DE. The Wilcoxon rank test results, that are presented in Table 6, confirm that

TABLE 3: Minimization results from the benchmark functions test in Table 12 with  $n = 30$ . Maximum number of iterations = 1000.

		PSO	DE	LS
$f_8$	ABS	$-6.7 \times 10^3$	$-1.26 \times 10^4$	$-1.26 \times 10^4$
	SD	$6.3 \times 10^2$	$3.7 \times 10^2$	$1.1 \times 10^2$
	NFE	38,452	35,240	21,846
$f_9$	ABS	14.8	$4.01 \times 10^{-1}$	$2.49 \times 10^{-3}$
	SD	1.39	$5.1 \times 10^{-2}$	$4.8 \times 10^{-4}$
	NFE	37,672	34,576	20,784
$f_{10}$	ABS	14.7	$4.66 \times 10^{-2}$	$2.15 \times 10^{-3}$
	SD	1.44	$1.27 \times 10^{-2}$	$3.18 \times 10^{-4}$
	NFE	39,475	37,080	21,235
$f_{11}$	ABS	12.01	1.15	$1.47 \times 10^{-4}$
	SD	3.12	0.06	$1.48 \times 10^{-5}$
	NFE	38,542	34,875	22,126
$f_{12}$	ABS	$6.87 \times 10^{-1}$	$3.74 \times 10^{-1}$	$5.58 \times 10^{-3}$
	SD	$7.07 \times 10^{-1}$	$1.55 \times 10^{-1}$	$4.18 \times 10^{-4}$
	NFE	35,248	30,540	16,984
$f_{13}$	ABS	$1.87 \times 10^{-1}$	$1.81 \times 10^{-2}$	$1.78 \times 10^{-2}$
	SD	$5.74 \times 10^{-1}$	$1.66 \times 10^{-2}$	$1.64 \times 10^{-3}$
	NFE	36,022	31,968	18,802

TABLE 4:  $p$  values produced by Wilcoxon's test comparing LS versus PSO and DE over the "average best-so-far" values from Table 3.

LS versus	PSO	DE
$f_8$	$1.83 \times 10^{-4}$	0.061
$f_9$	$1.17 \times 10^{-4}$	$2.41 \times 10^{-4}$
$f_{10}$	$1.43 \times 10^{-4}$	$3.12 \times 10^{-3}$
$f_{11}$	$6.25 \times 10^{-4}$	$1.14 \times 10^{-3}$
$f_{12}$	$2.34 \times 10^{-5}$	$7.15 \times 10^{-4}$
$f_{13}$	$4.73 \times 10^{-4}$	0.071

LS performed better than PSO and DE considering four problems  $f_9$ – $f_{12}$ , whereas, from a statistical viewpoint, there is no difference between results from LS and DE for  $f_8$  and  $f_{13}$ . According to Table 3, the NFE values obtained by the proposed method are smaller than those produced by other optimizers. The reason of this remarkable performance is associated with its two operators: (i) the solitary operator allows a better particle distribution in the search space, increasing the algorithm's ability to find the global optima and (ii) the use of the social operation provides a simple exploitation operator that intensifies the capacity of finding better solutions during the evolution process.

## 5. Gaussian Mixture Modelling

In this section, the modeling of image histograms through Gaussian mixture models is presented. Let one consider an image holding  $L$  gray levels  $[0, \dots, L - 1]$  whose distribution

is defined by a histogram  $h(g)$  represented by the following formulation:

$$h(g) = \frac{n_g}{Np}, \quad h(g) > 0,$$

$$Np = \sum_{g=0}^{L-1} n_g, \quad (15)$$

$$\sum_{g=0}^{L-1} h(g) = 1,$$

where  $n_g$  denotes the number of pixels with gray level  $g$  and  $Np$  the total number of pixels in the image. Under such circumstances,  $h(g)$  can be modeled by using a mixture  $p(x)$  of Gaussian functions of the form:

$$p(x) = \sum_{i=1}^K \frac{P_i}{\sqrt{2\pi}\sigma_i} \exp\left[-\frac{(x - \mu_i)^2}{2\sigma_i^2}\right], \quad (16)$$

where  $K$  symbolizes the number of Gaussian functions of the model whereas  $P_i$  is the a priori probability of function  $i$ .  $\mu_i$  and  $\sigma_i$  represent the mean and standard deviation of the  $i$ th Gaussian function, respectively. Furthermore, the constraint  $\sum_{i=1}^K P_i = 1$  must be satisfied by the model. In order to evaluate the similarity between the image histogram and a candidate mixture model, the mean square error can be used as follows:

$$J = \frac{1}{n} \sum_{j=1}^L [p(x_j) - h(x_j)]^2 + \omega \cdot \left| \left( \sum_{i=1}^K P_i \right) - 1 \right|, \quad (17)$$

where  $\omega$  represents the penalty associated with the constrain  $\sum_{i=1}^K P_i = 1$ . Therefore,  $J$  is considered as the objective function which must be minimized in the estimation problem. In order to illustrate the histogram modeling through a Gaussian mixture, Figure 6 presents an example, assuming three classes, that is,  $K = 3$ . Considering Figure 6(a) as the image histogram  $h(x)$ , the Gaussian mixture  $p(x)$ , that is shown in Figure 6(c), is produced by adding the Gaussian functions  $p_1(x)$ ,  $p_2(x)$ , and  $p_3(x)$  in the configuration presented in Figure 6(b). Once the model parameters that better model the image histogram have been determined, the final step is to define the threshold values  $T_i$  ( $i \in [1, \dots, K]$ ) which can be calculated by simple standard methods, just as it is presented in [19–21].

## 6. Segmentation Algorithm Based on LS

In the proposed method, the segmentation process is approached as an optimization problem. Computational optimization generally involves two distinct elements: (1) a search strategy to produce candidate solutions (individuals, particles, insects, locust, etc.) and (2) an objective function that evaluates the quality of each selected candidate solution. Several computational algorithms are available to perform the first element. The second element, where the objective function is designed, is unquestionably the most critical. In the objective function, it is expected to embody the full

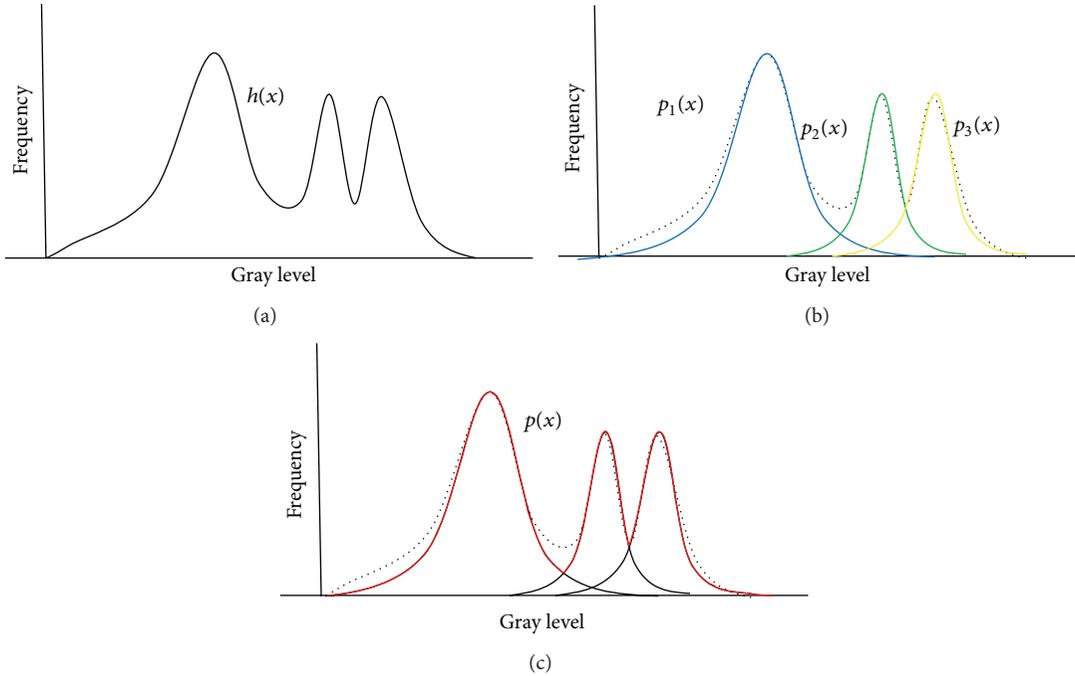


FIGURE 6: Histogram approximation through a Gaussian mixture. (a) Original histogram, (b) configuration of the Gaussian components  $p_1(x)$ ,  $p_2(x)$ , and  $p_3(x)$ , and (c) final Gaussian mixture  $p(x)$ .

complexity of the performance, biases, and restrictions of the problem to be solved. In the segmentation problem, candidate solutions represent Gaussian mixtures. The objective function  $J$  (17) is used as a fitness value to evaluate the similarity presented between the Gaussian mixture and the image histogram. Therefore, guided by the fitness values ( $J$  values), a set of encoded candidate solutions are evolved using the evolutionary operators until the best possible resemblance can be found.

Over the last decade, several algorithms based on evolutionary and swarm principles [19–22] have been proposed to solve the problem of segmentation through a Gaussian mixture model. Although these algorithms own certain good performance indexes, they present two important limitations. (1) They frequently obtain suboptimal approximations as a consequence of an inappropriate balance between exploration and exploitation in their search strategies. (2) They are based on the assumption that the number of Gaussians (classes) in the mixture is preknown and fixed; otherwise, they cannot work.

In order to eliminate such flaws, the proposed approach includes (A) a new search strategy and (B) the definition of a new objective function. For the search strategy, the LS method (Section 4) is adopted. Under LS, the concentration of individuals in the current best positions is explicitly avoided. Such fact allows reducing critical problems such as the premature convergence to suboptimal solutions and the incorrect exploration-exploitation balance.

**6.1. New Objective Function  $J^{\text{new}}$ .** Previous segmentation algorithms based on evolutionary and swarm methods use (17) as objective function. Under these circumstances, each

candidate solution (Gaussian mixture) is only evaluated in terms of its approximation with the image histogram.

Since the proposed approach aims to automatically select the number of Gaussian functions  $K$  in the final mixture  $p(x)$ , the objective function must be modified. The new objective function  $J^{\text{new}}$  is defined as follows:

$$J^{\text{new}} = J + \lambda \cdot Q, \quad (18)$$

where  $\lambda$  is a scaling constant. The new objective function is divided into two parts. The first part  $J$  evaluates the quality of each candidate solution in terms of its similarity with regard to the image histogram (17). The second part  $Q$  penalizes the overlapped area among Gaussian functions (classes), with  $Q$  being defined as follows:

$$Q = \sum_{i=1}^K \sum_{\substack{j=1 \\ j \neq i}}^K \sum_{l=1}^L \min(P_i \cdot p_i(l), P_j \cdot p_j(l)), \quad (19)$$

where  $K$  and  $L$  represent the number of classes and the gray levels, respectively. The parameters  $p_i(l)$  and  $p_j(l)$  symbolize the Gaussian functions  $i$  and  $j$ , respectively, that are to be evaluated on the point  $l$  (gray level) whereas the elements  $P_i$  and  $P_j$  represent the a priori probabilities of the Gaussian functions  $i$  and  $j$ , respectively. Under such circumstances, mixtures with Gaussian functions that do not “positively” participate in the histogram approximation are severely penalized.

Figure 7 illustrates the effect of the new objective function  $J^{\text{new}}$  in the evaluation of Gaussian mixtures (candidate solutions). From the image histogram (Figure 7(a)), it is evident that two Gaussian functions are enough to accurately

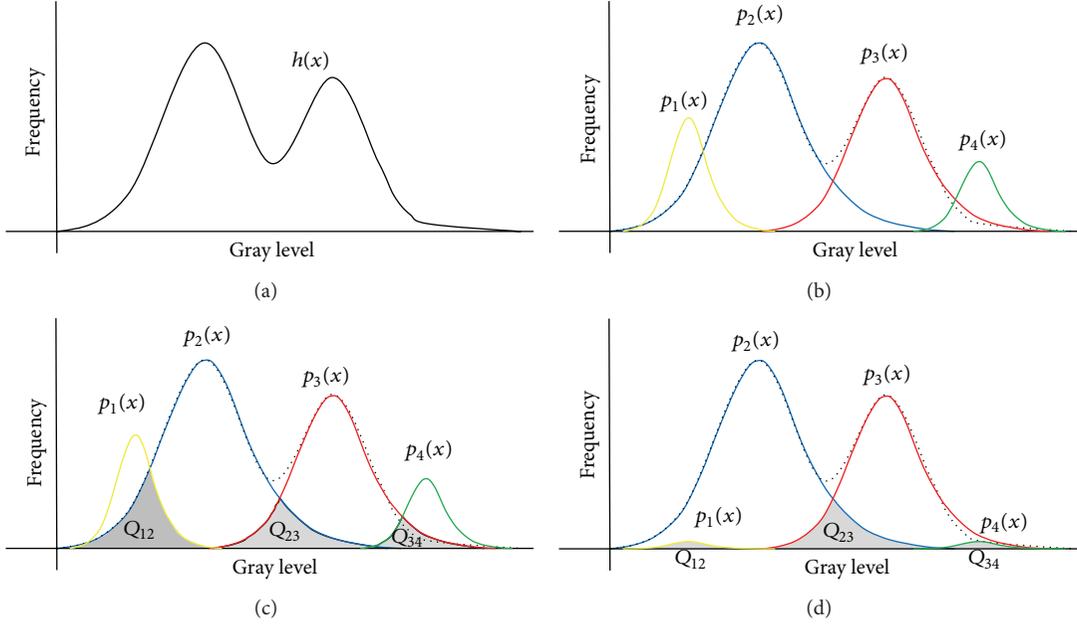


FIGURE 7: Effect of the new objective function  $J^{\text{new}}$  in the evaluation of Gaussian mixtures (candidate solutions). (a) Original histogram, (b) Gaussian mixture considering four classes, (c) penalization areas, and (d) Gaussian mixture of better quality solution.

approximate the original histogram. However, if the Gaussian mixture is modeled by using a greater number of functions (e.g., four as it is shown in Figure 7(b)), the original objective function  $J$  is unable to obtain a reasonable approximation. Under the new objective function  $J^{\text{new}}$ , the overlapped area among Gaussian functions (classes) is penalized. Such areas, in Figure 7(c), correspond to  $Q_{12}$ ,  $Q_{23}$ , and  $Q_{34}$ , where  $Q_{12}$  represents the penalization value produced between the Gaussian function  $p_1(x)$  and  $p_2(x)$ . Therefore, due to the penalization, the Gaussian mixture shown in Figures 7(b) and 7(c) provides a solution of low quality. On the other hand, the Gaussian mixture presented in Figure 7(d) maintains a low penalty; thus, it represents a solution of high quality. From Figure 7(d), it is easy to see that functions  $p_1(x)$  and  $p_4(x)$  can be removed from the final mixture. This elimination could be performed by a simple comparison with a threshold value  $\theta$ , since  $p_1(x)$  and  $p_4(x)$  have a reduced amplitude ( $p_1(x) \approx p_2(x) \approx 0$ ). Therefore, under  $J^{\text{new}}$ , it is possible to find the reduced Gaussian mixture model starting from a considerable number of functions.

Since the proposed segmentation method is conceived as an optimization problem, the overall operation can be reduced to solve the formulation of (20) by using the LS algorithm:

$$\begin{aligned}
 &\text{minimize } J^{\text{new}}(\mathbf{x}) = J(\mathbf{x}) + \lambda \cdot Q(\mathbf{x}), \\
 &\quad \mathbf{x} = (P_1, \mu_1, \sigma_1, \dots, P_K, \mu_K, \sigma_K) \in \mathbb{R}^{3 \cdot K} \\
 &\text{subject to } 0 \leq P_d \leq 1, \quad d \in (1, \dots, K) \quad (20) \\
 &\quad 0 \leq \mu_d \leq 255 \\
 &\quad 0 \leq \sigma_d \leq 25,
 \end{aligned}$$

where  $P_d$ ,  $\mu_d$ , and  $\sigma_d$  represent the probability, mean, and standard deviation of the class  $d$ . It is important to remark that the new objective function  $J^{\text{new}}$  allows the evaluation of a candidate solution in terms of the necessary number of Gaussian functions and its approximation quality. Under such circumstances, it can be used in combination with any other evolutionary method and not only with the proposed LS algorithm.

**6.2. Complete Segmentation Algorithm.** Once the new search strategy (LS) and objective function ( $J^{\text{new}}$ ) have been defined, the proposed segmentation algorithm can be summarized by Algorithm 2. The new algorithm combines operators defined by LS and operations for calculating the threshold values.

(Line 1) The algorithm sets the operative parameters  $F$ ,  $L$ ,  $g$ ,  $gen$ ,  $N$ ,  $K$ ,  $\lambda$ , and  $\theta$ . They rule the behavior of the segmentation algorithm. (Line 2) Afterwards, the population  $\mathbf{L}^0$  is initialized considering  $N$  different random Gaussian mixtures of  $K$  functions. The idea is to generate an  $N$ -random Gaussian mixture subjected to the constraints formulated in (20). The parameter  $K$  must hold a high value in order to correctly segment all images (recall that the algorithm is able to reduce the Gaussian mixture to its minimum expression). (Line 3) Then, the Gaussian mixtures are evolved by using the LS operators and the new objective function  $J^{\text{new}}$ . This process is repeated during  $gen$  cycles. (Line 8) After this procedure, the best Gaussian mixture  $\mathbf{I}_{\text{best}}^{\text{gen}}$  is selected according to its objective function  $J^{\text{new}}$ . (Line 9) Then, the Gaussian mixture  $\mathbf{I}_{\text{best}}^{\text{gen}}$  is reduced by eliminating those functions whose amplitudes are lower than  $\theta$  ( $p_i(x) \leq \theta$ ). (Line 10) Then, the threshold values  $T_i$  from the reduced model are calculated. (Line 11) Finally, the calculated  $T_i$  values are employed to segment the image. Figure 8 shows a flowchart of the complete segmentation procedure.

(1) <b>Input:</b> $F, L, g, gen, N, K, \lambda$ and $\theta$ .	
(2) Initialize $\mathbf{L}^0$ ( $k = 0$ )	
(3) <b>until</b> ( $k = gen$ )	
(4) $\mathbf{F} \leftarrow \text{SolitaryOperation}(\mathbf{L}^k)$	Solitary operator (Section 3.1)
(5) $\mathbf{L}^{k+1} \leftarrow \text{SocialOperation}(\mathbf{L}^k, \mathbf{F})$	Social operator (Section 3.2)
(6) $k = k + 1$	
(7) <b>end until</b>	
(8) Obtain $\mathbf{I}_{best}^{gen}$	
(9) Reduce $\mathbf{I}_{best}^{gen}$	
(10) Calculate the threshold values $T_i$ from the reduced model	
(11) Use $T_i$ to segment the image	

ALGORITHM 2: Segmentation LS algorithm.

The proposed segmentation algorithm permits to automatically detect the number of segmentation partitions (classes). Furthermore, due to its remarkable search capacities, the LS method maintains a better accuracy than previous algorithms based on evolutionary principles. However, the proposed method presents two disadvantages: first, it is related to its implementation which in general is more complex than most of the other segmentators based on evolutionary basics. The second refers to the segmentation procedure of the proposed approach which does not consider any spatial pixel characteristics. As a consequence, pixels that may belong to a determined region due to its position are labeled as a part of another region due to its gray level intensity. Such a fact adversely affects the segmentation performance of the method.

## 7. Segmentation Results

This section analyses the performance of the proposed segmentation algorithm. The discussion is divided into three parts: the first one shows the performance of the proposed LS segmentation algorithm while the second presents a comparison between the proposed approach and others segmentation algorithms that are based on evolutionary and swam methods. The comparison mainly considers the capacities of each algorithm to accurately and robustly approximate the image histogram. Finally, the third part presents an objective evaluation of segmentation results produced by all algorithms that have been employed in the comparisons.

### 7.1. Performance of LS Algorithm in Image Segmentation.

This section presents two experiments that analyze the performance of the proposed approach. Table 5 presents the algorithm's parameters that have been experimentally determined and kept for all the test images through all experiments.

*First Image.* The first test considers the histogram shown by Figure 9(b) while Figure 9(a) presents the original image. After applying the proposed algorithm, just as it has been configured according to parameters in Table 5, a minimum model of four classes emerges after the start from Gaussian

TABLE 5: Parameter setup for the LS segmentation algorithm.

$F$	$L$	$g$	$gen$	$N$	$K$	$\lambda$	$\theta$
0.6	0.2	20	1000	40	10	0.01	0.0001

TABLE 6: Results of the reduced Gaussian mixture for the first and the second image.

Parameters	First image	Second image
$\overline{P_1}$	0.004	0.032
$\overline{\mu_1}$	18.1	12.1
$\overline{\sigma_1}$	8.2	2.9
$\overline{P_2}$	0.0035	0.0015
$\overline{\mu_2}$	69.9	45.1
$\overline{\sigma_2}$	18.4	24.9
$\overline{P_3}$	0.01	0.006
$\overline{\mu_3}$	94.9	130.1
$\overline{\sigma_3}$	8.9	17.9
$\overline{P_4}$	0.007	0.02
$\overline{\mu_4}$	163.1	167.2
$\overline{\sigma_4}$	29.9	10.1

mixtures of 10 classes. Considering 30 independent executions, the averaged parameters of the resultant Gaussian mixture are presented in Table 6. One final Gaussian mixture (ten classes), which has been obtained by LS, is presented in Figure 10. Furthermore, the approximation of the reduced Gaussian mixture is also visually compared with the original histogram in Figure 10. On the other hand, Figure 11 presents the segmented image after calculating the threshold points.

*Second Image.* For the second experiment, the image in Figure 12 is tested. The method aims to segment the image by using a reduced Gaussian mixture model obtained by the LS approach. After executing the algorithm according to parameters from Table 5, the resulting averaged parameters of the resultant Gaussian mixture are presented in Table 6. In order to assure consistency, the results are averaged considering 30 independent executions. Figure 13 shows the approximation quality that is obtained by the reduced Gaussian mixture model in (a) and the segmented image in (b).

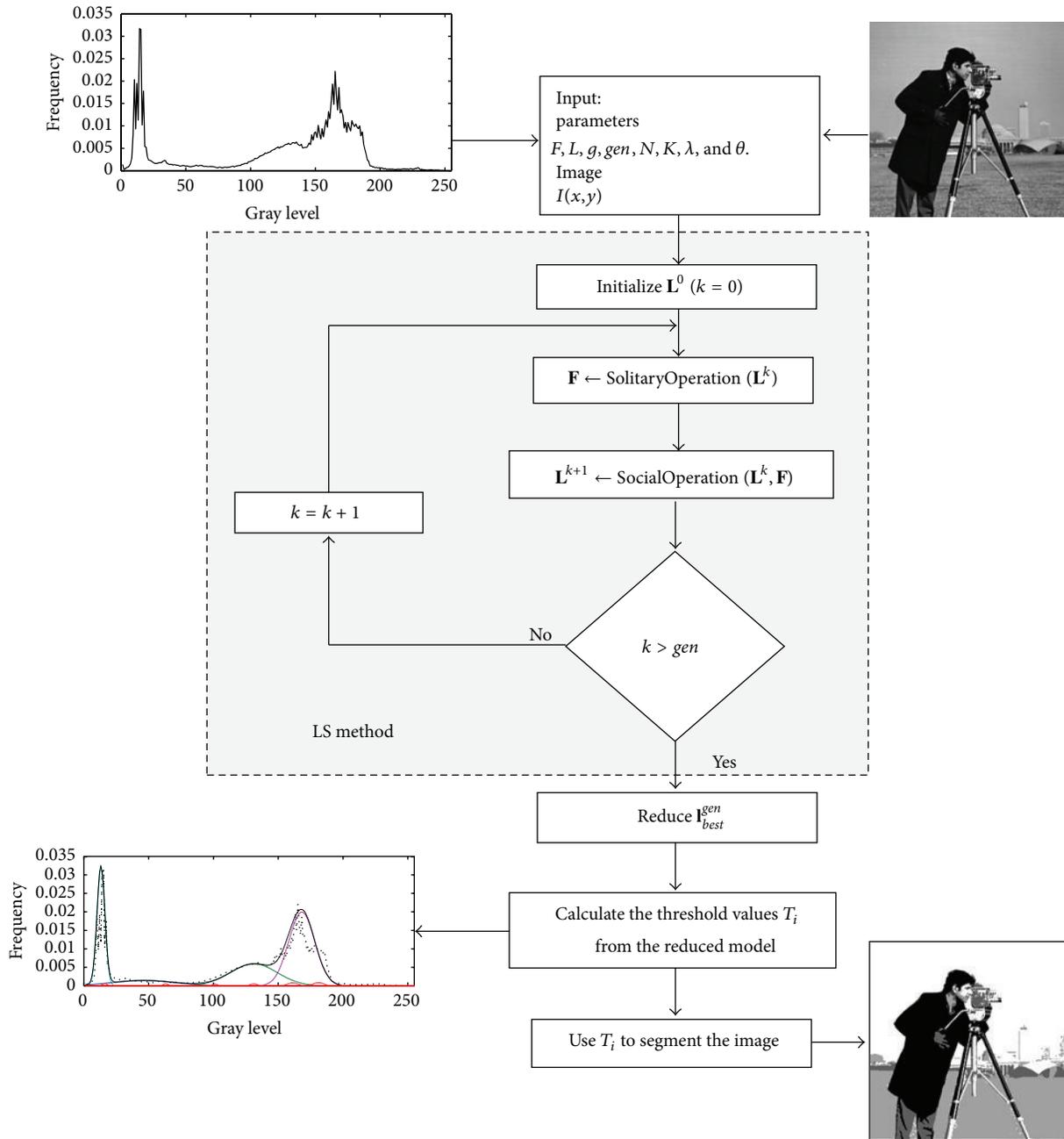


FIGURE 8: Flowchart of the complete segmentation procedure.

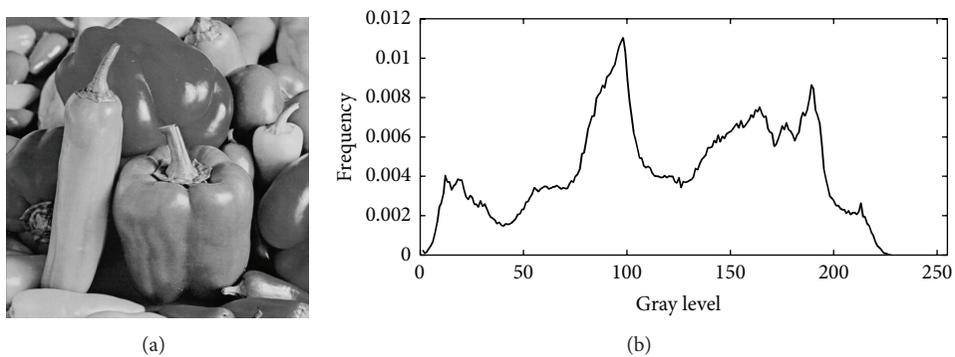


FIGURE 9: (a) Original first image used on the first experiment and (b) its histogram.

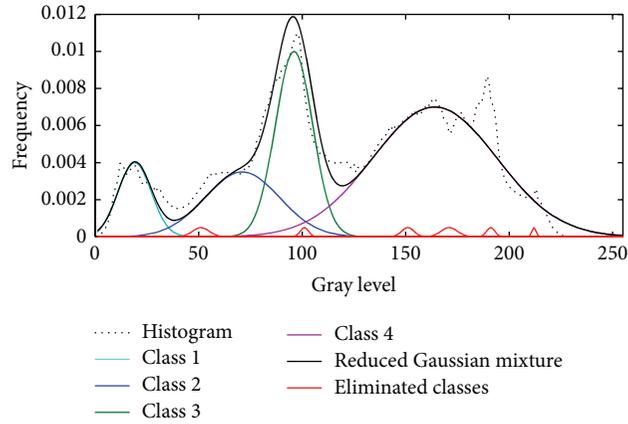


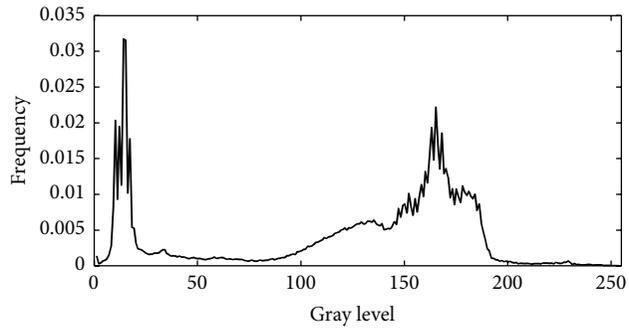
FIGURE 10: Gaussian mixture obtained by LS for the first image.



FIGURE 11: Image segmented with the reduced Gaussian mixture.



(a)



(b)

FIGURE 12: (a) Original second image used on the first experiment and (b) its histogram.

7.2. *Histogram Approximation Comparisons.* This section discusses the comparison between the proposed segmentation algorithm and other evolutionary-segmentation methods that have been proposed in the literature. The set of methods used in the experiments includes the  $J + ABC$  [19],  $J + AIS$  [20], and  $J + DE$  [21]. These algorithms consider the combination between the original objective function  $J$  (17) and an evolutionary technique such as Artificial Bee Colony (ABC), the Artificial Immune Systems (AIS), and the Differential Evolution (DE) [21], respectively.

Since the proposed segmentation approach considers the combination of the new objective function  $J^{new}$  (18) and the LS algorithm, it will be referred to as  $J^{new} + LS$ . The comparison focuses mainly on the capacities of each algorithm to accurately and robustly approximate the image histogram.

In the experiments, the populations have been set to 40 ( $N = 40$ ) individuals. The maximum iteration number for all functions has been set to 1000. Such stop criterion has been considered to maintain compatibility to similar experiments

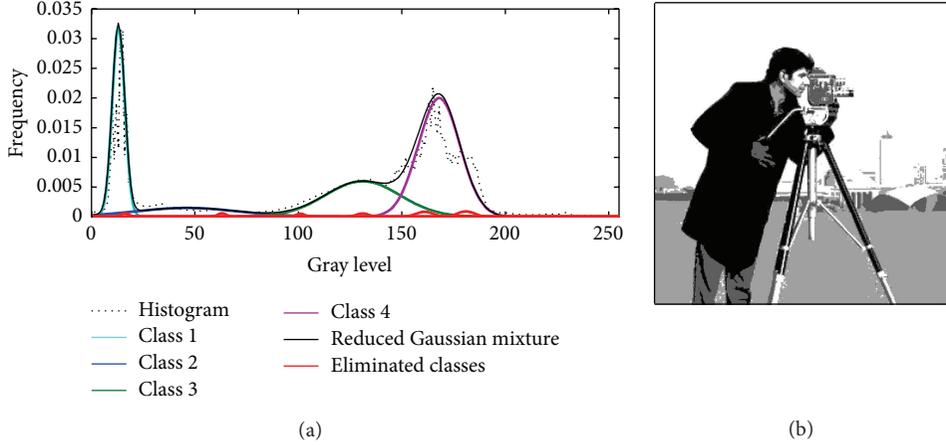


FIGURE 13: (a) Segmentation result obtained by LS for the first image and (b) the segmented image.

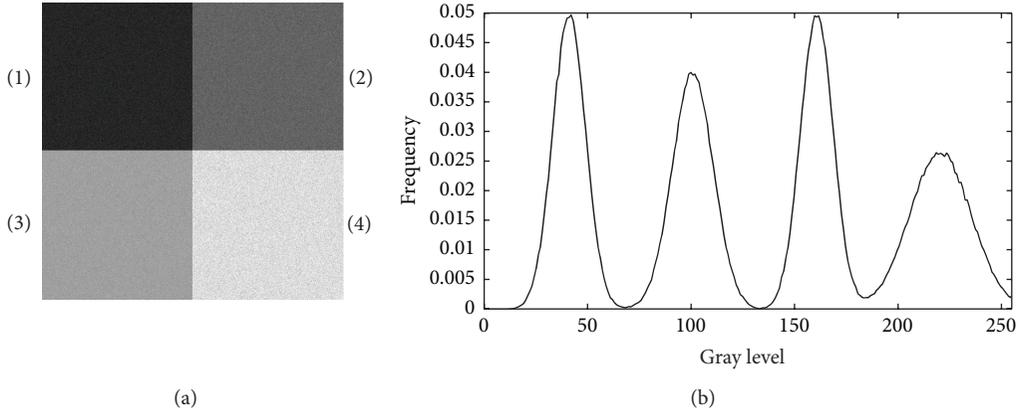


FIGURE 14: (a) Synthetic image used in the comparison and (b) its histogram.

that are reported in the literature [18]. The parameter setting for each of the segmentation algorithms in the comparison is described as follows:

- (1)  $J + ABC$  [19]: in the algorithm, its parameters are configured as follows: the abandonment limit = 100,  $\alpha = 0.05$  and limit = 30.
- (2)  $J + AIS$  [20]: it presents the following parameters,  $h = 120$ ,  $N_c = 80$ ,  $\rho = 10$ ,  $P_r = 20$ ,  $L = 22$ , and  $T_e = 0.01$ .
- (3)  $J + DE$  [21]: the DE/Rand/1 scheme is employed. The parameter settings follow the instructions in [21]. The crossover probability is  $CR = 0.9$  and the weighting factor is  $F = 0.8$ .
- (4) In  $J^{\text{new}} + LS$ , the method is set according to the values described in Table 5.

In order to conduct the experiments, a synthetic image is designed to be used as a reference in the comparisons. The main idea is to know in advance the exact number of classes (and their parameters) that are contained in the image so that the histogram can be considered as a ground truth. The

TABLE 7: Employed parameters for the design of the reference image.

	$P_i$	$\mu_i$	$\sigma_i$
(1)	0.05	40	8
(2)	0.04	100	10
(3)	0.05	160	8
(4)	0.027	220	15

synthetic image is divided into four sections. Each section corresponds to a different class which is produced by setting each gray level pixel  $Pv^j$  to a value that is determined by the following model:

$$Pv_i = e^{-((x-\mu_i)^2/2\sigma_i^2)}, \quad (21)$$

where  $i$  represents the section, whereas  $\mu_i$  and  $\sigma_i$  are the mean and the dispersion of the gray level pixels, respectively. The comparative study employs the image of  $512 \times 512$  that is shown in Figure 14(a) and the algorithm's parameters that have been presented in Table 7. Figure 14(b) illustrates the resultant histogram.

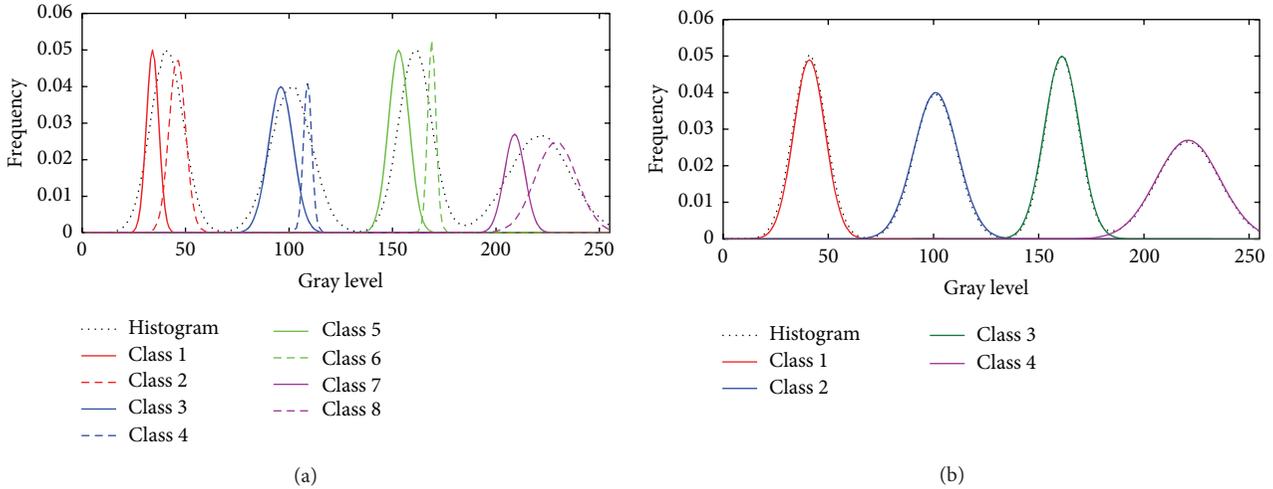


FIGURE 15: Convergence results. (a) Convergence of the following methods:  $J + ABC$ ,  $J + AIS$ , and  $J + DE$  considering Gaussian mixtures of 8 classes. (b) Convergence of the proposed method (reduced Gaussian mixture).

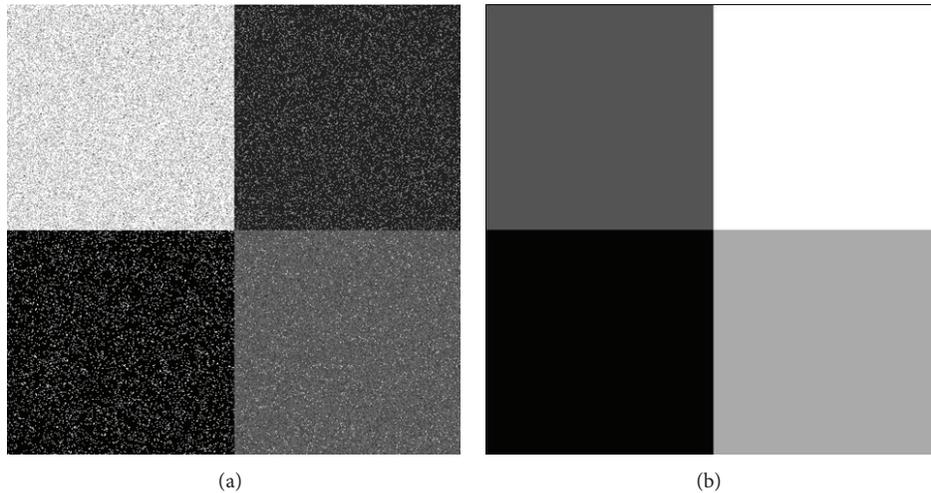


FIGURE 16: Segmentation results obtained by (a) several methods including  $J + ABC$ ,  $J + AIS$ , and  $J + DE$  considering Gaussian mixtures of 8 classes and (b) the proposed method (reduced Gaussian mixture).

In the comparison, the discussion focuses on the following issues: first of all, accuracy; second, convergence; and third, computational cost.

**Convergence.** This section analyzes the approximation convergence when the number of classes that are used by the algorithm during the evolution process is different to the actual number of classes in the image. Recall that the proposed approach automatically finds the reduced Gaussian mixture which better adapts to the image histogram.

In the experiment, the methods,  $J + ABC$ ,  $J + AIS$ , and  $J + DE$ , are executed considering Gaussian mixtures composed of 8 functions. Under such circumstances, the number of classes to be detected is higher than the actual number of classes in the image. On the other hand, the

proposed algorithm maintains the same configuration of Table 5. Therefore, it can detect and calculate up to ten classes ( $K = 10$ ).

As a result, the techniques  $J + ABC$ ,  $J + AIS$ , and  $J + DE$  tend to overestimate the image histogram. This effect can be seen in Figure 15(a), where the resulting Gaussian functions are concentrated within actual classes. Such a behavior is a consequence of the evaluation that is considered by the original objective function  $J$ , which privileges only the approximation between the Gaussian mixture and the image histogram. This effect can be graphically illustrated by Figure 16(a) that shows the pixel misclassification produced by the wrong segmentation of Figure 14(a). On the other hand, the proposed approach obtains a reduced Gaussian mixture model which allows the detection of each class from

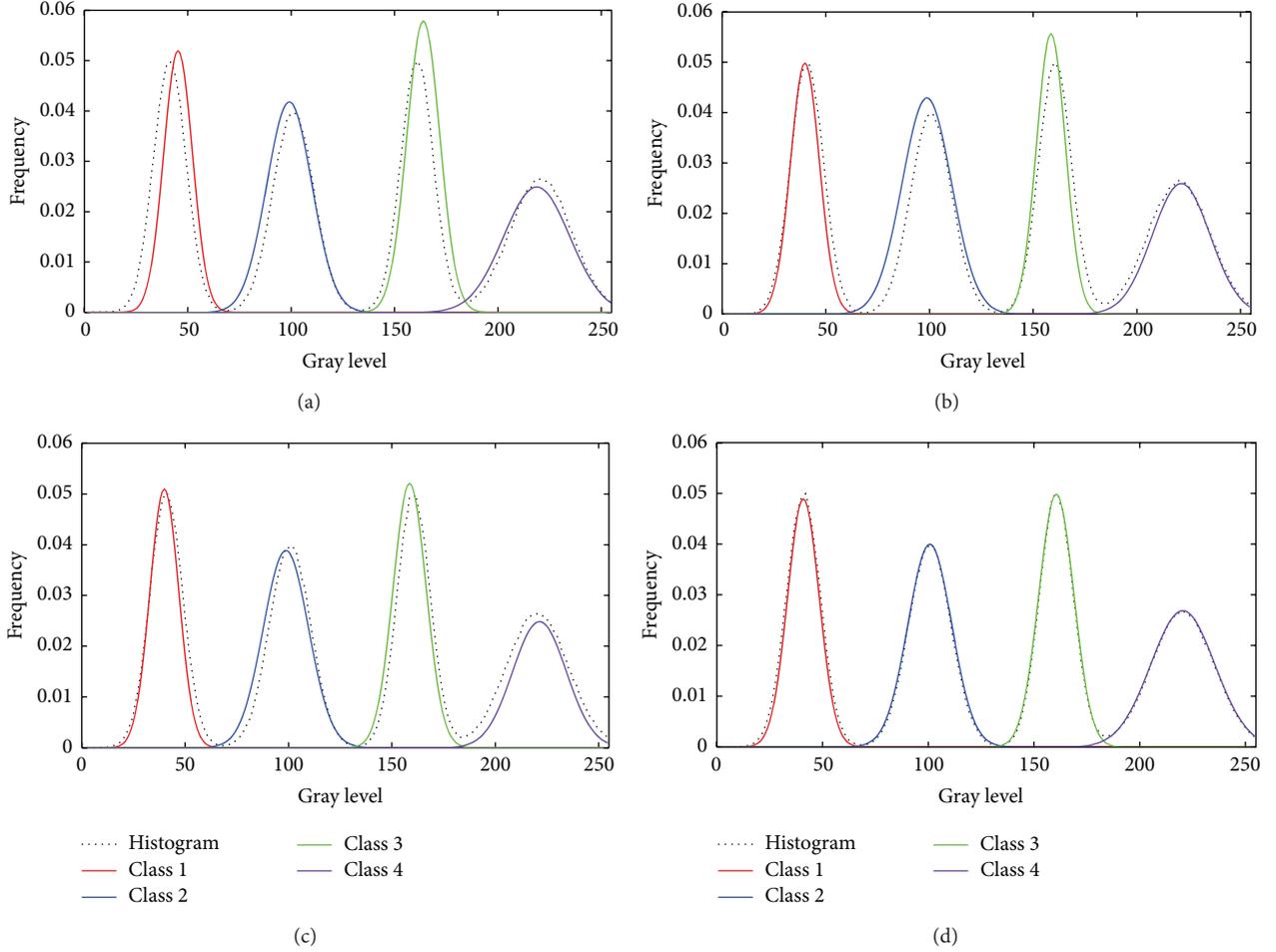


FIGURE 17: Approximation results in terms of accuracy. (a)  $J + ABC$ , (b)  $J + AIS$ , (c)  $J + DE$ , and (d) the proposed  $J^{\text{new}} + LS$  approach.

the actual histogram (see Figure 15(b)). As a consequence, the segmentation is significantly improved by eliminating the pixel misclassification, as it is shown by Figure 16(b).

It is evident from Figure 15 that the techniques,  $J + ABC$ ,  $J + AIS$ , and  $J + DE$ , all need an a priori knowledge of the number of classes that are contained in the actual histogram in order to obtain a satisfactory result. On the other hand, the proposed algorithm is able to find a reduced Gaussian mixture whose classes coincide with the actual number of classes that are contained in the image histogram.

**Accuracy.** In this section, the comparison among the algorithms in terms of accuracy is reported. Most of the reported comparisons [19–26] are concerned about comparing the parameters of the resultant Gaussian mixtures by using real images. Under such circumstances, it is difficult to consider a clear reference in order to define a meaningful error. Therefore, the image defined in Figure 14 has been used in the experiments because its construction parameters are clearly defined in Table 7.

Since the parameter values from Table 7 act as ground truth, a simple averaged difference between them and the values that are computed by each algorithm could be used as

comparison error. However, as each parameter maintains different active intervals, it is necessary to express the differences in terms of percentage. Therefore, if  $\Delta\beta$  is the parametric difference and  $\beta$  the ground truth parameter, the percentage error  $\Delta\beta\%$  can be defined as follows:

$$\Delta\beta\% = \frac{\Delta\beta}{\beta} \cdot 100\%. \quad (22)$$

In the segmentation problem, each Gaussian mixture represents a  $K$ -dimensional model where each dimension corresponds to a Gaussian function of the optimization problem to be solved. Since each Gaussian function possesses three parameters  $P_i$ ,  $\mu_i$ , and  $\sigma_i$ , the complete number of parameters is  $3 \cdot K$  dimensions. Therefore, the final error  $E$  produced by the final Gaussian mixture is

$$E = \frac{1}{K \cdot 3} \sum_{v=1}^{K \cdot 3} \Delta\beta_v\%, \quad (23)$$

where  $\beta_v \in (P_i, \mu_i, \sigma_i)$ .

In order to compare accuracy, the algorithms,  $J + ABC$ ,  $J + AIS$ ,  $J + DE$ , and the proposed approach are all executed over the image shown by Figure 14(a). The experiment aims

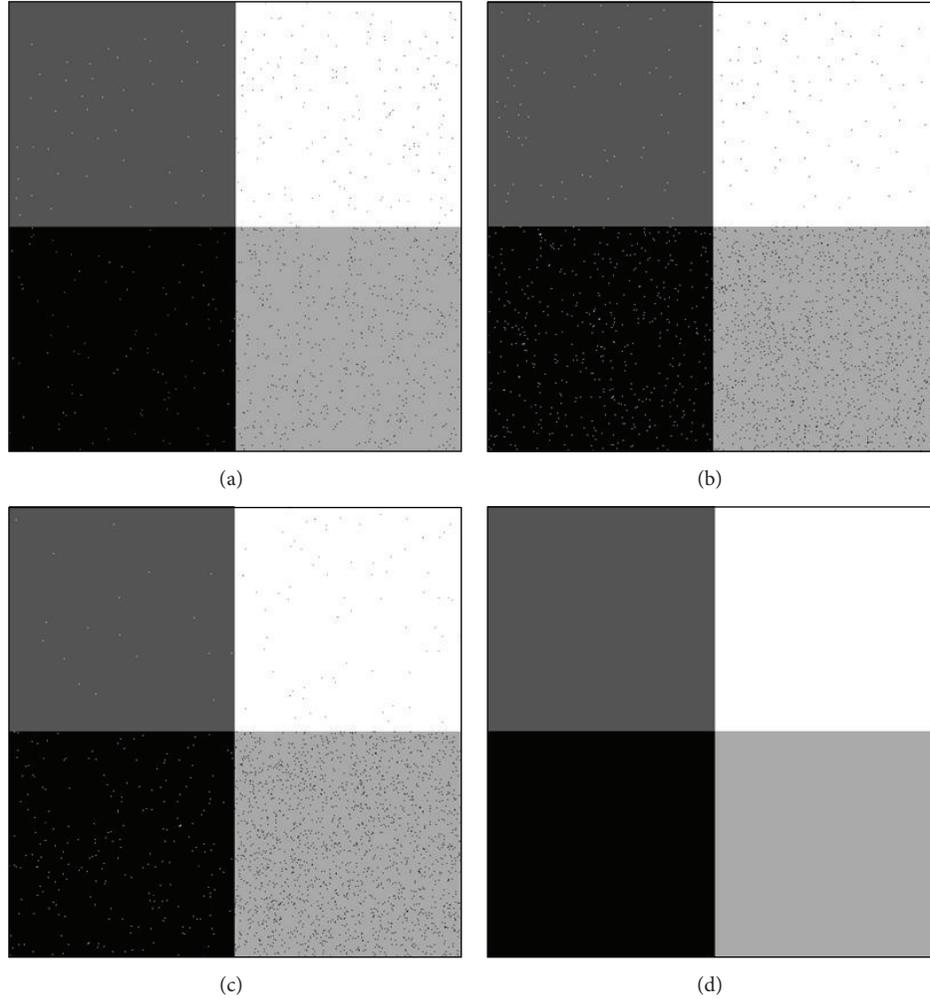


FIGURE 18: Segmentation results in terms of accuracy. (a)  $J + ABC$ , (b)  $J + AIS$ , (c)  $J + DE$ , and (d) the proposed  $J^{new} + LS$  approach.

to estimate the Gaussian mixture that better approximates the actual image histogram. Methods  $J + ABC$ ,  $J + AIS$ , and  $J + DE$  consider Gaussian mixtures composed of 4 functions ( $K = 4$ ). In case of the  $J^{new} + LS$  method, although the algorithm finds a reduced Gaussian mixture of four functions, it is initially set with ten functions ( $K = 10$ ). Table 8 presents the final Gaussian mixture parameters and the final error  $E$ . The final Gaussian mixture parameters have been averaged over 30 independent executions in order to assure consistency. A close inspection of Table 8 reveals that the proposed method is able to achieve the smallest error ( $E$ ) in comparison to the other algorithms. Figure 16 presents the histogram approximations that are produced by each algorithm whereas Figure 17 shows their correspondent segmented images. Both illustrations present the median case obtained throughout 30 runs. Figure 18 exhibits that  $J + ABC$ ,  $J + AIS$ , and  $J + DE$  present different levels of misclassifications which are nearly absent in the proposed approach case.

*Computational Cost.* The experiment aims to measure the complexity and the computing time spent by the  $J + ABC$ ,

TABLE 8: Results of the reduced Gaussian mixture in terms of accuracy.

Algorithm	Gaussian function	$\bar{P}_i$	$\bar{\mu}_i$	$\bar{\sigma}_i$	$E$
$J + ABC$	(1)	0.052	44.5	6.4	11.79%
	(2)	0.084	98.12	12.87	
	(3)	0.058	163.50	8.94	
	(4)	0.025	218.84	175	
$J + AIS$	(1)	0.072	31.01	6.14	22.01%
	(2)	0.054	88.52	12.21	
	(3)	0.039	149.21	9.14	
	(4)	0.034	248.41	13.84	
$J + DE$	(1)	0.041	35.74	7.86	13.57%
	(2)	0.036	90.57	11.97	
	(3)	0.059	148.47	9.01	
	(4)	0.020	201.34	13.02	
$J^{new} + LS$	(1)	0.049	40.12	7.5	3.98%
	(2)	0.041	102.04	10.4	
	(3)	0.052	168.66	8.3	
	(4)	0.025	110.92	15.2	



FIGURE 19: Images employed in the computational cost analysis.

the  $J + AIS$ , the  $J + DE$ , and the  $J^{\text{new}} + LS$  algorithm while calculating the parameters of the Gaussian mixture in benchmark images (see Figures 19(a)–19(d)).  $J + ABC$ ,  $J + AIS$ , and  $J + DE$  consider Gaussian mixtures that are composed of 4 functions ( $K = 4$ ). In case of the  $J^{\text{new}} + LS$  method, although the algorithm finds a reduced Gaussian mixture of four functions despite being initialized with ten functions ( $K = 10$ ), Table 9 shows the averaged measurements after 30 experiments. It is evident that the  $J + ABC$  and  $J +$

$DE$  are the slowest to converge (iterations) and the  $J + AIS$  shows the highest computational cost (time elapsed) because it requires operators which demand long times. On the other hand, the  $J^{\text{new}} + LS$  shows an acceptable trade-off between its convergence time and its computational cost. Therefore, although the implementation of  $J^{\text{new}} + LS$  in general requires more code than most of other evolution-based segmentators, such a fact is not reflected in the execution time. Finally, Figure 19 below shows the segmented images as they are

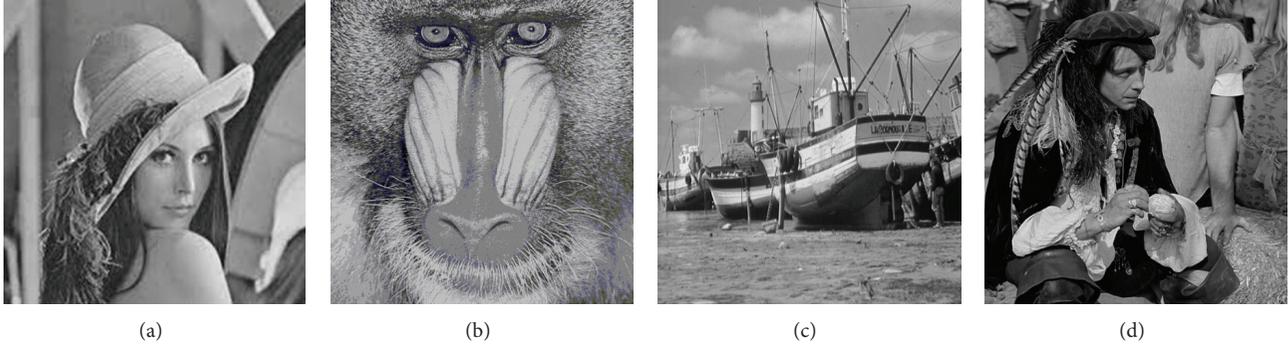


FIGURE 20: Experimental set used in the evaluation of the segmentation results.

 TABLE 9: Iterations and time requirements of the  $J + ABC$ , the  $J + AIS$ , the  $J + DE$ , and the  $J^{new} + LS$  algorithm as they are applied to segment benchmark images (see Figure 17).

Iterations	(a)	(b)	(c)	(d)
Time elapsed				
$J + ABC$	855 2.72 s	833 2.70 s	870 2.73 s	997 3.1 s
$J + AIS$	725 1.78 s	704 1.61 s	754 1.41 s	812 2.01 s
$J + DE$	657 1.25 s	627 1.12 s	694 1.45 s	742 1.88 s
$J^{new} + LS$	314 0.98 s	298 0.84 s	307 0.72 s	402 1.02 s

TABLE 10: Evaluation of the segmentation results in terms of the ROS index.

Number of classes	$N_R = 4$	$N_R = 3$	$N_R = 4$	$N_R = 4$
Image	(a)	(b)	(c)	(d)
$J + ABC$	0.534	0.328	0.411	0.457
$J + AIS$	0.522	0.321	0.427	0.437
$J + DE$	0.512	0.312	0.408	0.418
$J^{new} + LS$	0.674	0.401	0.514	0.527

generated by each algorithm. It can be seen that the proposed approach generate more homogeneous regions whereas  $J + ABC$ ,  $J + AIS$ , and  $J + DE$  present several artifacts that are produced by an incorrect pixel classification.

**7.3. Performance Evaluation of the Segmentation Results.** This section presents an objective evaluation of segmentation results that are produced by all algorithms in the comparisons. The ill-defined nature of the segmentation problem makes the evaluation of a candidate algorithm difficult [57]. Traditionally, the evaluation has been conducted by using some supervised criteria [58] which are based on the computation of a dissimilarity measure between a segmentation result and a ground truth image. Recently, the use of unsupervised measures has substituted supervised indexes for the objective evaluation of segmentation results [59]. They

TABLE II: Unimodal test functions.

Test function	$S$	$f_{opt}$
$f_1(\mathbf{x}) = \sum_{i=1}^n x_i^2$	$[-100, 100]^n$	0
$f_2(\mathbf{x}) = \sum_{i=1}^n  x_i  + \prod_{i=1}^n  x_i $	$[-10, 10]^n$	0
$f_3(\mathbf{x}) = \sum_{i=1}^n \left( \sum_{j=1}^i x_j \right)^2$	$[-100, 100]^n$	0
$f_4(\mathbf{x}) = \max_i \{ x_i , 1 \leq i \leq n\}$	$[-100, 100]^n$	0
$f_5(\mathbf{x}) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-30, 30]^n$	0
$f_6(\mathbf{x}) = \sum_{i=1}^n (x_i + 0.5)^2$	$[-100, 100]^n$	0
$f_7(\mathbf{x}) = \sum_{i=1}^n ix_i^4 + \text{rand}(0, 1)$	$[-1.28, 1.28]^n$	0

enable the quantification of the quality of a segmentation result without a priori knowledge (ground truth image).

**Evaluation Criteria.** In this paper, the unsupervised index ROS proposed by Chabrier et al. [60] has been used to objectively evaluate the performance of each candidate algorithm. This index evaluates the segmentation quality in terms of the homogeneity within segmented regions and the heterogeneity among the different regions. ROS can be computed as follows:

$$\text{ROS} = \frac{\overline{D} - \underline{D}}{2}, \quad (24)$$

where  $\underline{D}$  quantifies the homogeneity within segmented regions. Similarly,  $\overline{D}$  measures the disparity among the regions. A segmentation result  $S_1$  is considered better than  $S_2$ , if  $\text{ROS}_{S_1} > \text{ROS}_{S_2}$ . The interregion homogeneity characterized by  $\underline{D}$  is calculated considering the following formulation:

$$\underline{D} = \frac{1}{N_R} \sum_{c=1}^{N_R} \frac{R_c}{I} \cdot \frac{\sigma_c}{\left( \sum_{l=1}^{N_R} \sigma_l \right)}, \quad (25)$$

where  $N_R$  represents the number of partitions in which the image has been segmented.  $R_c$  symbolizes the number of

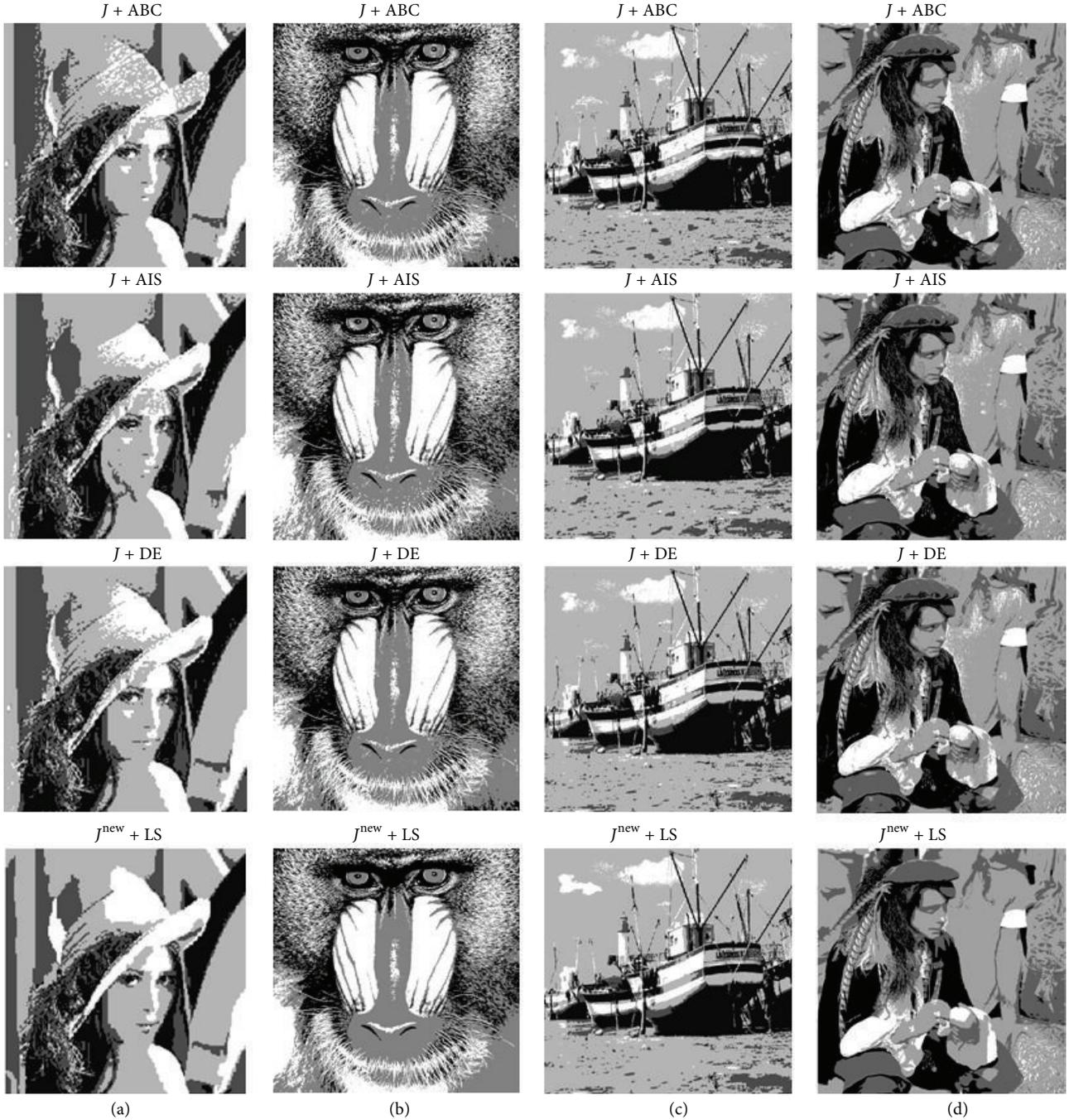


FIGURE 21: Segmentation results using in the evaluation.

pixels contained in the partition  $c$  whereas  $I$  considers the number of pixels that integrate the complete image. Similarly,  $\sigma_c$  represents the standard deviation from the partition  $c$ . On the other hand, disparity among the regions  $\bar{D}$  is computed as follows:

$$\bar{D} = \frac{1}{N_R} \sum_{c=1}^{N_R} \frac{R_c}{I} \cdot \left[ \frac{1}{(N_R - 1)} \sum_{l=1}^{N_R} \frac{|\mu_c - \mu_l|}{255} \right], \quad (26)$$

where  $\mu_c$  is the average gray level in the partition  $c$ .

*Experimental Protocol.* In the comparison of segmentation results, a set of four classical images has been chosen to integrate the experimental set (Figure 20). The segmentation methods used in the comparison are  $J + ABC$  [19],  $J + AIS$  [20], and  $J + DE$  [21].

From all segmentation methods used in the comparison, the proposed  $J^{\text{new}} + LS$  algorithm is the only one that has the capacity to automatically detect the number of segmentation partitions (classes). In order to conduct a fair comparison, all algorithms have been proved by using the same number

TABLE 12: Multimodal test functions.

Test function	$\mathbf{S}$	$f_{\text{opt}}$
$f_8(\mathbf{x}) = \sum_{i=1}^n -x_i \sin(\sqrt{ x_i })$	$[-500, 500]^n$	$-418.98 * n$
$f_9(\mathbf{x}) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-5.12, 5.12]^n$	0
$f_{10}(\mathbf{x}) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20$	$[-32, 32]^n$	0
$f_{11}(\mathbf{x}) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]^n$	0
$f_{12}(\mathbf{x}) = \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\} + \sum_{i=1}^n u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i + 1}{4} \quad u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	$[-50, 50]^n$	0
$f_{13}(\mathbf{x}) = 0.1 \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \right\} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	$[-50, 50]^n$	0

of partitions. Therefore, in the experiments, the  $J^{\text{new}} + \text{LS}$  segmentation algorithm is firstly applied to detect the best possible number of partitions  $N_R$ . Once we obtained the number of partitions  $N_R$ , the rest of the algorithms were configured to approximate the image histogram with this number of classes.

Figure 21 presents the segmentation results obtained by each algorithm considering the experimental set from Figure 20. On the other hand, Table 10 shows the evaluation of the segmentation results in terms of the ROS index. Such values represent the averaged measurements after 30 executions. From them, it can be seen that the proposed  $J^{\text{new}} + \text{LS}$  method obtains the best ROS indexes. Such values indicate that the proposed algorithm maintains the best balance between the homogeneity within segmented regions and the heterogeneity among the different regions. From Figure 21, it can be seen that the proposed approach generates more homogeneous regions whereas  $J + \text{ABC}$ ,  $J + \text{AIS}$ , and  $J + \text{DE}$  present several artifacts that are produced by an incorrect pixel classification.

## 8. Conclusions

Despite the fact that several evolutionary methods have been successfully applied to image segmentation with interesting results, most of them have exhibited two important limitations: (1) they frequently obtain suboptimal results (misclassifications) as a consequence of an inappropriate balance between exploration and exploitation in their search strategies; (2) the number of classes is fixed and known in advance.

In this paper, a new swarm algorithm for the automatic image segmentation, called the Locust Search (LS), has been presented. The proposed method eliminates the typical flaws presented by previous evolutionary approaches by combining a novel evolutionary method with the definition

of a new objective function that appropriately evaluates the segmentation quality with respect to the number of classes. In order to illustrate the proficiency and robustness of the proposed approach, several numerical experiments have been conducted. Such experiments have been divided into two parts. First, the proposed LS method has been compared to other well-known evolutionary techniques on a set of benchmark functions. In the second part, the performance of the proposed segmentation algorithm has been compared to other segmentation methods based on evolutionary principles. The results in both cases validate the efficiency of the proposed technique with regard to accuracy and robustness.

Several research directions will be considered for future work such as the inclusion of other indexes to evaluate similarity between a candidate solution and the image histogram, the consideration of spatial pixel characteristics in the objective function, the modification of the evolutionary LS operators to control the exploration-exploitation balance, and the conversion of the segmentation procedure into a multiobjective problem.

## Appendix

### List of Benchmark Functions

In Table 11,  $n$  is the dimension of function,  $f_{\text{opt}}$  is the minimum value of the function, and  $\mathbf{S}$  is a subset of  $R^n$ . The optimum location ( $\mathbf{x}_{\text{opt}}$ ) for functions in Table 11 is in  $[0]^n$ , except for  $f_5$  with  $\mathbf{x}_{\text{opt}}$  in  $[1]^n$ .

The optimum locations ( $\mathbf{x}_{\text{opt}}$ ) for functions in Table 12 are in  $[0]^n$ , except for  $f_8$  in  $[420.96]^n$  and  $f_{12}$ - $f_{13}$  in  $[1]^n$ .

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgment

The present research has been supported under the Grant CONACYT CB 181053.

## References

- [1] H. Zhang, J. E. Fritts, and S. A. Goldman, "Image segmentation evaluation: a survey of unsupervised methods," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 260–280, 2008.
- [2] T. Uemura, G. Koutaki, and K. Uchimura, "Image segmentation based on edge detection using boundary code," *International Journal of Innovative Computing*, vol. 7, pp. 6073–6083, 2011.
- [3] L. Wang, H. Wu, and C. Pan, "Region-based image segmentation with local signed difference energy," *Pattern Recognition Letters*, vol. 34, no. 6, pp. 637–645, 2013.
- [4] N. Otsu, "A thresholding selection method from gray-level histogram," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [5] R. Peng and P. K. Varshney, "On performance limits of image segmentation algorithms," *Computer Vision and Image Understanding*, vol. 132, pp. 24–38, 2015.
- [6] M. A. Balafar, "Gaussian mixture model based segmentation methods for brain MRI images," *Artificial Intelligence Review*, vol. 41, no. 3, pp. 429–439, 2014.
- [7] G. J. McLachlan and S. Rathnayake, "On the number of components in a Gaussian mixture model," *Data Mining and Knowledge Discovery*, vol. 4, no. 5, pp. 341–355, 2014.
- [8] D. Oliva, V. Osuna-Enciso, E. Cuevas, G. Pajares, M. Pérez-Cisneros, and D. Zaldivar, "Improving segmentation velocity using an evolutionary method," *Expert Systems with Applications*, vol. 42, no. 14, pp. 5874–5886, 2015.
- [9] Z.-W. Ye, M.-W. Wang, W. Liu, and S.-B. Chen, "Fuzzy entropy based optimal thresholding using bat algorithm," *Applied Soft Computing*, vol. 31, pp. 381–395, 2015.
- [10] S. Sarkar, S. Das, and S. Sinha Chaudhuri, "A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution," *Pattern Recognition Letters*, vol. 54, pp. 27–35, 2015.
- [11] A. K. Bhandari, A. Kumar, and G. K. Singh, "Modified artificial bee colony based computationally efficient multilevel thresholding for satellite image segmentation using Kapur's, Otsu and Tsallis functions," *Expert Systems with Applications*, vol. 42, no. 3, pp. 1573–1601, 2015.
- [12] H. Permuter, J. Francos, and I. Jermyn, "A study of Gaussian mixture models of color and texture features for image classification and segmentation," *Pattern Recognition*, vol. 39, no. 4, pp. 695–706, 2006.
- [13] A. P. Dempster, A. P. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society Series B*, vol. 39, no. 1, pp. 1–38, 1977.
- [14] Z. Zhang, C. Chen, J. Sun, and K. L. Chan, "EM algorithms for Gaussian mixtures with split-and-merge operation," *Pattern Recognition*, vol. 36, no. 9, pp. 1973–1983, 2003.
- [15] H. Park, S.-I. Amari, and K. Fukumizu, "Adaptive natural gradient learning algorithms for various stochastic models," *Neural Networks*, vol. 13, no. 7, pp. 755–764, 2000.
- [16] H. Park and T. Ozeki, "Singularity and slow convergence of the em algorithm for gaussian mixtures," *Neural Processing Letters*, vol. 29, no. 1, pp. 45–59, 2009.
- [17] L. Gupta and T. Sortrakul, "A Gaussian-mixture-based image segmentation algorithm," *Pattern Recognition*, vol. 31, no. 3, pp. 315–325, 1998.
- [18] V. Osuna-Enciso, E. Cuevas, and H. Sossa, "A comparison of nature inspired algorithms for multi-threshold image segmentation," *Expert Systems with Applications*, vol. 40, no. 4, pp. 1213–1219, 2013.
- [19] E. Cuevas, F. Senci3n, D. Zaldivar, M. P3rez-Cisneros, and H. Sossa, "A multi-threshold segmentation approach based on artificial bee colony optimization," *Applied Intelligence*, vol. 37, no. 3, pp. 321–336, 2012.
- [20] E. Cuevas, V. Osuna-Enciso, D. Zaldivar, M. P3rez-Cisneros, and H. Sossa, "Multithreshold segmentation based on artificial immune systems," *Mathematical Problems in Engineering*, vol. 2012, Article ID 874761, 20 pages, 2012.
- [21] E. Cuevas, D. Zaldivar, and M. P3rez-Cisneros, "A novel multi-threshold segmentation approach based on differential evolution optimization," *Expert Systems with Applications*, vol. 37, no. 7, pp. 5265–5271, 2010.
- [22] D. Oliva, E. Cuevas, G. Pajares, D. Zaldivar, and V. Osuna, "A multilevel thresholding algorithm using electromagnetism optimization," *Neurocomputing*, vol. 139, pp. 357–381, 2014.
- [23] D. Oliva, E. Cuevas, G. Pajares, D. Zaldivar, and M. P3rez-Cisneros, "Multilevel thresholding segmentation based on harmony search optimization," *Journal of Applied Mathematics*, vol. 2013, Article ID 575414, 24 pages, 2013.
- [24] E. Cuevas, D. Zaldivar, and M. P3rez-Cisneros, "Seeking multi-thresholds for image segmentation with Learning Automata," *Machine Vision and Applications*, vol. 22, no. 5, pp. 805–818, 2011.
- [25] K. C. Tan, S. C. Chiam, A. A. Mamun, and C. K. Goh, "Balancing exploration and exploitation with adaptive variation for evolutionary multi-objective optimization," *European Journal of Operational Research*, vol. 197, no. 2, pp. 701–713, 2009.
- [26] G. Chen, C. P. Low, and Z. Yang, "Preserving and exploiting genetic diversity in evolutionary programming algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 3, pp. 661–673, 2009.
- [27] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [28] R. Storn and K. Price, "Differential Evolution—a simple and efficient adaptive scheme for global optimisation over continuous spaces," Tech. Rep. TR-95-012, ICSI, Berkeley, Calif, USA, 1995.
- [29] Y. Wang, B. Li, T. Weise, J. Wang, B. Yuan, and Q. Tian, "Self-adaptive learning based particle swarm optimization," *Information Sciences*, vol. 181, no. 20, pp. 4515–4538, 2011.
- [30] J. Tvrdík, "Adaptation in differential evolution: a numerical comparison," *Applied Soft Computing Journal*, vol. 9, no. 3, pp. 1149–1155, 2009.
- [31] H. Wang, H. Sun, C. Li, S. Rahnamayan, and J.-s. Pan, "Diversity enhanced particle swarm optimization with neighborhood search," *Information Sciences*, vol. 223, pp. 119–135, 2013.
- [32] W. Gong, Á. Fialho, Z. Cai, and H. Li, "Adaptive strategy selection in differential evolution for numerical optimization: an empirical study," *Information Sciences*, vol. 181, no. 24, pp. 5364–5386, 2011.
- [33] D. M. Gordon, "The organization of work in social insect colonies," *Complexity*, vol. 8, no. 1, pp. 43–46, 2002.
- [34] S. Kizaki and M. Katori, "Stochastic lattice model for locust outbreak," *Physica A: Statistical Mechanics and its Applications*, vol. 266, no. 1-4, pp. 339–342, 1999.

- [35] S. M. Rogers, D. A. Cullen, M. L. Anstey et al., "Rapid behavioural gregarization in the desert locust, *Schistocerca gregaria* entails synchronous changes in both activity and attraction to conspecifics," *Journal of Insect Physiology*, vol. 65, pp. 9–26, 2014.
- [36] C. M. Topaz, A. J. Bernoff, S. Logan, and W. Toolson, "A model for rolling swarms of locusts," *European Physical Journal: Special Topics*, vol. 157, no. 1, pp. 93–109, 2008.
- [37] C. M. Topaz, M. R. D'Orsogna, L. Edelstein-Keshet, and A. J. Bernoff, "Locust dynamics: behavioral phase change and swarming," *PLoS Computational Biology*, vol. 8, no. 8, Article ID e1002642, 2012.
- [38] G. Oster and E. Wilson, *Caste and Ecology in the Social Insects*, Princeton University Press, Princeton, NJ, USA, 1978.
- [39] B. Hölldobler and E. O. Wilson, *Journey to the Ants: A Story of Scientific Exploration*, 1994.
- [40] B. Hölldobler and E. O. Wilson, *The Ants*, Harvard University Press, Cambridge, Mass, USA, 1990.
- [41] S. Tanaka and Y. Nishide, "Behavioral phase shift in nymphs of the desert locust, *Schistocerca gregaria*: special attention to attraction/avoidance behaviors and the role of serotonin," *Journal of Insect Physiology*, vol. 59, no. 1, pp. 101–112, 2013.
- [42] E. Gaten, S. J. Huston, H. B. Dowse, and T. Matheson, "Solitary and gregarious locusts differ in circadian rhythmicity of a visual output neuron," *Journal of Biological Rhythms*, vol. 27, no. 3, pp. 196–205, 2012.
- [43] I. Benaragama and J. R. Gray, "Responses of a pair of flying locusts to lateral looming visual stimuli," *Journal of Comparative Physiology A*, vol. 200, no. 8, pp. 723–738, 2014.
- [44] M. G. Sergeev, "Distribution patterns of grasshoppers and their Kin in the boreal zone," *Psyche*, vol. 2011, Article ID 324130, 9 pages, 2011.
- [45] S. O. Ely, P. G. N. Njagi, M. O. Bashir, S. El-Tom El-Amin, and A. Hassanali, "Diel behavioral activity patterns in adult solitary desert locust, *Schistocerca gregaria* (Forskål)," *Psyche*, vol. 2011, Article ID 459315, 9 pages, 2011.
- [46] X.-S. Yang, *Nature-Inspired Metaheuristic Algorithms*, Luniver Press, Beckington, UK, 2008.
- [47] E. Cuevas, A. Echavarría, and M. A. Ramírez-Ortegón, "An optimization algorithm inspired by the States of Matter that improves the balance between exploration and exploitation," *Applied Intelligence*, vol. 40, no. 2, pp. 256–272, 2014.
- [48] M. M. Ali, C. Khompatraporn, and Z. B. Zabinsky, "A numerical evaluation of several stochastic algorithms on selected continuous global optimization test problems," *Journal of Global Optimization*, vol. 31, no. 4, pp. 635–672, 2005.
- [49] R. Chelouah and P. Siarry, "Continuous genetic algorithm designed for the global optimization of multimodal functions," *Journal of Heuristics*, vol. 6, no. 2, pp. 191–213, 2000.
- [50] F. Herrera, M. Lozano, and A. M. Sánchez, "A taxonomy for the crossover operator for real-coded genetic algorithms: an experimental study," *International Journal of Intelligent Systems*, vol. 18, no. 3, pp. 309–338, 2003.
- [51] E. Cuevas, D. Zaldivar, M. Pérez-Cisneros, and M. Ramírez-Ortegón, "Circle detection using discrete differential evolution optimization," *Pattern Analysis & Applications*, vol. 14, no. 1, pp. 93–107, 2011.
- [52] A. Sadollah, H. Eskandar, A. Bahreininejad, and J. H. Kim, "Water cycle algorithm with evaporation rate for solving constrained and unconstrained optimization problems," *Applied Soft Computing*, vol. 30, pp. 58–71, 2015.
- [53] M. S. Kiran, H. Hakli, M. Gunduz, and H. Uguz, "Artificial bee colony algorithm with variable search strategy for continuous optimization," *Information Sciences*, vol. 300, pp. 140–157, 2015.
- [54] F. Kang, J. Li, and Z. Ma, "Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions," *Information Sciences*, vol. 181, no. 16, pp. 3508–3531, 2011.
- [55] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.
- [56] S. García, D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC'2005 Special Session on Real Parameter Optimization," *Journal of Heuristics*, vol. 15, no. 6, pp. 617–644, 2009.
- [57] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward objective evaluation of image segmentation algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 929–944, 2007.
- [58] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "A measure for objective evaluation of image segmentation algorithms," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, p. 34, San Diego, Calif, USA, June 2005.
- [59] Y. J. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognition*, vol. 29, no. 8, pp. 1335–1346, 1996.
- [60] S. Chabrier, B. Emile, C. Rosenberger, and H. Laurent, "Unsupervised performance evaluation of image segmentation," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, Article ID 96306, 12 pages, 2006.

## Research Article

# Face Recognition Using MLP and RBF Neural Network with Gabor and Discrete Wavelet Transform Characterization: A Comparative Study

**Fatma Zohra Chelali and Amar Djeradi**

*Speech Communication and Signal Processing Laboratory, Faculty of Electronics and Computer Science, University of Science and Technology Houari Boumedienne (USTHB), P.O. Box 32, 16111 Algiers, Algeria*

Correspondence should be addressed to Fatma Zohra Chelali; [chelali\\_zohra@yahoo.fr](mailto:chelali_zohra@yahoo.fr)

Received 13 September 2014; Revised 1 December 2014; Accepted 16 December 2014

Academic Editor: Marco Pérez-Cisneros

Copyright © 2015 F. Z. Chelali and A. Djeradi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Face recognition has received a great attention from a lot of researchers in computer vision, pattern recognition, and human machine computer interfaces in recent years. Designing a face recognition system is a complex task due to the wide variety of illumination, pose, and facial expression. A lot of approaches have been developed to find the optimal space in which face feature descriptors are well distinguished and separated. Face representation using Gabor features and discrete wavelet has attracted considerable attention in computer vision and image processing. We describe in this paper a face recognition system using artificial neural networks like multilayer perceptron (MLP) and radial basis function (RBF) where Gabor and discrete wavelet based feature extraction methods are proposed for the extraction of features from facial images using two facial databases: the ORL and computer vision. Good recognition rate was obtained using Gabor and DWT parameterization with MLP classifier applied for computer vision dataset.

## 1. Introduction

Face recognition has received a great attention from a lot of scientists in computer vision, pattern recognition, and human machine computer interfaces in recent years.

Face is the most common biometric identifier used by humans; this domain is motivated by the increased interest in the commercial applications of automatic face recognition (AFR) as well as by the emergence of real-time processors. Automatic face recognition (FR) is also one of the most visible and challenging research topics in computer vision, machine learning, and biometrics [1].

Face recognition has become important because of the potential value for applications and its theoretical challenges. Today, face recognition technology is being used to combat passport fraud, support law enforcement, identify missing children, and identify fraud. There are so many challenges in face recognition; some of them are (1) illumination variances, (2) occlusions, and (3) different expressions and poses [2].

There are many approaches to evaluate face images. The methods used in face recognition can be generally classified into image feature based and geometry feature based methods. In feature geometry based approach, recognition is based on the relationship between human facial features such as eye(s), mouth, nose, and face boundary and subspace analysis approach attempts to capture and define the face as a whole. The subspace method is the most famous technique for face recognition. In this method the face is treated as two-dimensional pattern of intensity variation [3].

Although facial images have a high dimensionality, they usually lie on a lower dimensional subspace or submanifold. Therefore, subspace learning and manifold learning methods have been dominantly and successfully used in appearance based FR. The classical Eigenface and Fisherface algorithms consider only the global scatter of training samples and they fail to reveal the essential data structures nonlinearly embedded in high-dimensional space [1].

Face representation using Gabor features has attracted considerable attention in computer vision, image processing,

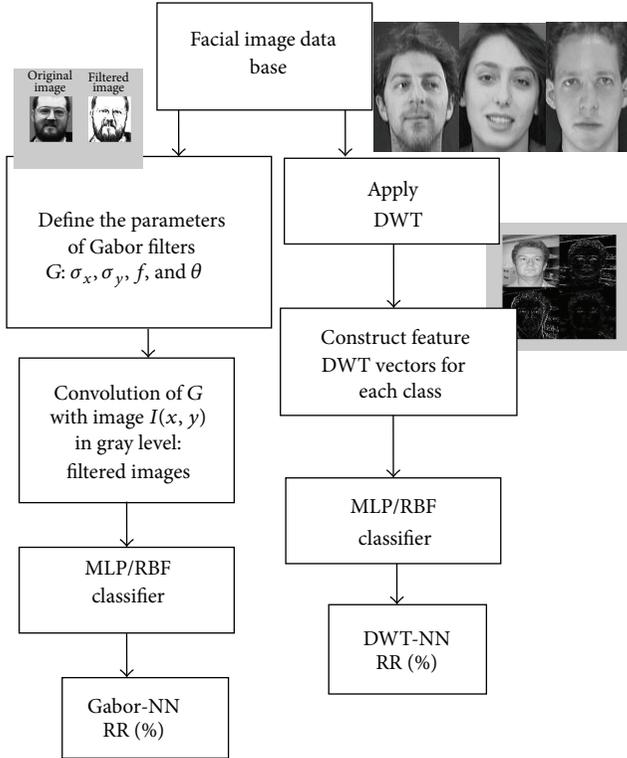


FIGURE 1: An overview of face recognition system.

pattern recognition, and so on. The principal motivation to use Gabor filters is biological relevance that the receptive field profiles of neurons in the primary visual cortex of mammals are oriented and have characteristic spatial frequencies.

Gabor filters can exploit salient visual properties such as spatial localization, orientation selectivity, and spatial frequency characteristics [4].

The Gabor filter was first introduced by David Gabor in 1946 and was later shown as models of simple cell receptive fields. Since the Gabor features are extracted in local regions, they are less sensitive to variations of illumination, expression, and pose than the holistic features such as Eigenface and Randomface [1].

Face recognition using Gabor filters was firstly introduced by Yang and Zhang and soon proved to be a very effective means in human facial features extraction [1].

We are interested in our work to develop a recognition system using Gabor filters and discrete wavelet transform as feature extraction. A multilayer perceptron (MLP) and radial basis function (RBF) neural network classifier are used to classify an individual from two face databases: the ORL and computer vision databases. Figure 1 describes our system.

We investigate then the best way to characterize the facial images; efficiency is evaluated by computing recognition rate (RR%) for each parameterization and classifier.

The paper is presented as follows: Section 2 presents the face characterization using Gabor filters and discrete wavelet transform DWT, Section 3 presents simulation results, and conclusion is drawn in Section 5.

## 2. Feature Face Extraction and Face Parameterization

The most important step in face recognition system is the feature extraction. It permits us to reduce dimensionality and characterize the face image using a vector descriptor which minimizes intraperson dissimilarities and maximizes the extraperson difference. This task is difficult because of variations of pose, age, expression and illumination, and so forth.

In our work, we use Gabor filters and discrete wavelet transform to describe our face and to build a face recognition system using the multilayer perceptron (MLP) and radial basis function (RBF) neural network classifier. Comparison is done for each classifier and facial database to decide the optimal characterization. We describe first the Gabor function and discrete wavelet transforms to characterize our facial images.

### 2.1. Gabor Filter Representation

**2.1.1. Gabor Wavelet: A Mathematical Overview.** Gabor filter, defined by Dennis Gabor, is widely used in image analysis, pattern recognition, and so forth. Gabor filters present two interesting properties: frequency localization and selectivity in orientation.

A lot of studies demonstrate that representation by Gabor wavelet for image analysis is robust to variation in illumination and facial expressions.

Gabor filter in 2D was introduced in biometric research by Daugman for iris recognition. Lades et al. used Gabor filter for face recognition using “Dynamic Link Architecture.”

The Gabor wavelet, which captures the properties of orientation selectivity, spatial localization and optimally localized in the space and frequency domains, has been extensively and successfully used in face recognition [5]. Daugman pioneered the using of the 2D Gabor wavelet representation in computer vision in 1980s [6, 7].

Gabor wavelets (filters) characteristics for frequency and orientation representations are quite similar to those of human visual system. These have been found appropriate for texture representation and discrimination. This Gabor wavelet based extraction of features directly from the gray-level images is successful and has widely been applied to texture segmentation and fingerprint recognition. The commonly used Gabor filters in face recognition area [3, 4] are defined as follows [6]:

$$\Psi_{u,v}(Z) = \frac{\|k_{u,v}\|^2}{\sigma^2} \exp\left(\frac{-\|k_{u,v}\|^2 \|Z\|^2}{2 * \sigma^2}\right) \left[ e^{ik_{u,v}Z} - e^{-\sigma^2/2} \right]. \quad (1)$$

$Z(x, y)$  is the point of coordinates  $(x, y)$  in image space,  $u$  and  $v$  define the orientation Gabor frequency, and  $\sigma$  is the standard deviation of the Gaussian.

$\mu$  and  $\nu$  define the orientation and the scale of the Gabor filters, and  $k_{\mu,\nu}$  is defined as the following form:

$$k_{\mu,\nu} = k_{\nu} e^{j\varphi_{\mu}}. \quad (2)$$

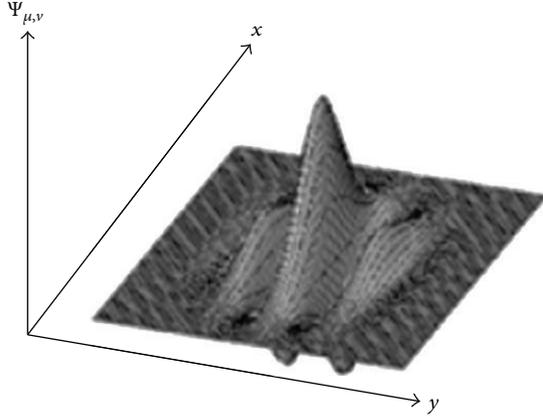


FIGURE 2: Gabor filter [2].

$k_v = k_{\max}/f^v$  and  $\varphi_u = \pi\mu/8$ .  $k_{\max}$  is the maximum frequency, and  $f$  is the spacing factor between kernels in the frequency domain. Usually  $\sigma = 2\pi$ ,  $k_{\max} = \pi/2$ , and  $f = \sqrt{2}$ .

The Gabor wavelet representation of a face image is obtained by doing a convolution between the image and a family of Gabor filters as described by (3). The convolution of image  $I(z)$  and a Gabor filter  $\Psi_{\mu,\nu}(z)$  can be defined as follows [6]:

$$F_{\mu,\nu}(z) = I(z) * \Psi_{\mu,\nu}(z), \quad (3)$$

where  $z = (x, y)$ ,  $*$  denotes the convolution operator, and  $F_{\mu,\nu}(z)$  is the Gabor filter response of the image with orientation  $u$  and scale  $\nu$  [6]. Figure 2 illustrates this principle.

The Gabor kernels in (1) are all self-similar since they can be generated from the same filter, the mother wavelet, by scaling and rotating via the wave vector  $k_{\mu,\nu}$ .

Each kernel is a product of a Gaussian envelope and a complex plane wave. These representation results display scale, locality, and orientation properties corresponding to those displayed by the Gabor wavelets [6].

Each kernel is a product of a Gaussian envelope and a complex plane wave and can be separated into real and imaginary parts. Hence, a band of Gabor filters is generated by a set of various scales and rotations. A detail of Gabor filters calculations or design can be described later.

Gabor wavelet of an image  $I$  is then a convolution product of  $I$  with a set of Gabor filters with different frequency and orientations.

The convolution of the image  $I$  with Gabor  $\Psi_{\mu,\nu}(z)$  is defined by

$$G_{\mu,\nu}(Z) = I(Z) * \Psi_{\mu,\nu}(z). \quad (4)$$

Figure 3 presents an example of facial representation in Gabor wavelet using 40 Gabor filters.

**2.1.2. Design of Gabor Filter.** Gabor filter works as a band-pass filter for the local spatial frequency distribution, achieving an optimal resolution in both spatial and frequency domains. The 2D Gabor filter  $\Psi_{f,\theta}(x, y)$  can be represented as a complex

sinusoidal signal modulated by a Gaussian kernel function as follows [4, 8]:

$$\Psi_{f,\theta}(x, y) = \exp\left[-\frac{1}{2}\left\{\frac{x_{\theta n}^2}{\sigma_x^2} + \frac{y_{\theta n}^2}{\sigma_y^2}\right\}\right] \exp(2\pi f x_{\theta n}), \quad (5)$$

where

$$\begin{bmatrix} x_{\theta n} \\ y_{\theta n} \end{bmatrix} = \begin{bmatrix} \sin \theta_n & \cos \theta_n \\ -\cos \theta_n & \sin \theta_n \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \quad (6)$$

$\sigma_x, \sigma_y$  are the standard deviations of the Gaussian envelope along the  $x$ - and  $y$ -dimensions;  $f$  is the central frequency of the sinusoidal plane wave and  $\theta_n$  the orientation. The rotation of the  $x$ - $y$  plane by an angle  $\theta_n$  will result in a Gabor filter at the orientation  $\theta_n$ . The angle  $\theta_n$  is defined by

$$\theta_n = \frac{\pi}{p}(n-1). \quad (7)$$

$n = 1, 2, \dots, p$  and  $p \in N$ , where  $p$  denotes the number of orientations.

Design of Gabor filters is accomplished by tuning the filter with a specific band of spatial frequency and orientation by appropriately selecting the filter parameters; the spread of the filters  $\sigma_x, \sigma_y$ , radial frequency  $f$ , and the orientation of the filter  $\theta_n$  [4].

The important issue in the design of Gabor filters for face recognition is the choice of filter parameters.

The Gabor representation of a face image is computed by convolving the face image with the Gabor filters. Let  $f(x, y)$  be the intensity at the coordinate  $(x, y)$  in a grayscale face image; its convolution with a Gabor filter  $\Psi_{f,\theta}(x, y)$  is defined as [4]

$$g_{f,\theta}(x, y) = f(x, y) * \Psi_{f,\theta}(x, y), \quad (8)$$

where  $*$  denotes the convolution operator. The response to each Gabor kernel filter representation is a complex function with a real part  $\text{Re}\{g_{f,\theta}(x, y)\}$  and an imaginary part  $\text{Im}\{g_{f,\theta}(x, y)\}$ . The magnitude response  $\|g_{f,\theta}(x, y)\|$  is expressed as

$$\|g_{f,\theta}(x, y)\| = \sqrt{\text{Re}^2\{g_{f,\theta}(x, y)\} + \text{Im}^2\{g_{f,\theta}(x, y)\}}. \quad (9)$$

We use the magnitude response  $\|g_{f,\theta}(x, y)\|$  to represent our facial images.

The Gabor filter is basically a Gaussian modulated by a complex sinusoid described by (5) and (6).

All filters could be produced by rotation and dilatation of a mother wavelet. A majority of face recognition approaches based on Gabor filters used only the magnitude of the response or a fusion of the magnitude and the real part because of the variation of the phase with local patterns and it can be considered irrelevant.

The response of the Gabor filter depends on the following parameters.

- (a) The orientation  $\theta$ : it describes the orientation of the wavelet and characterizes the angles of contour and line images.

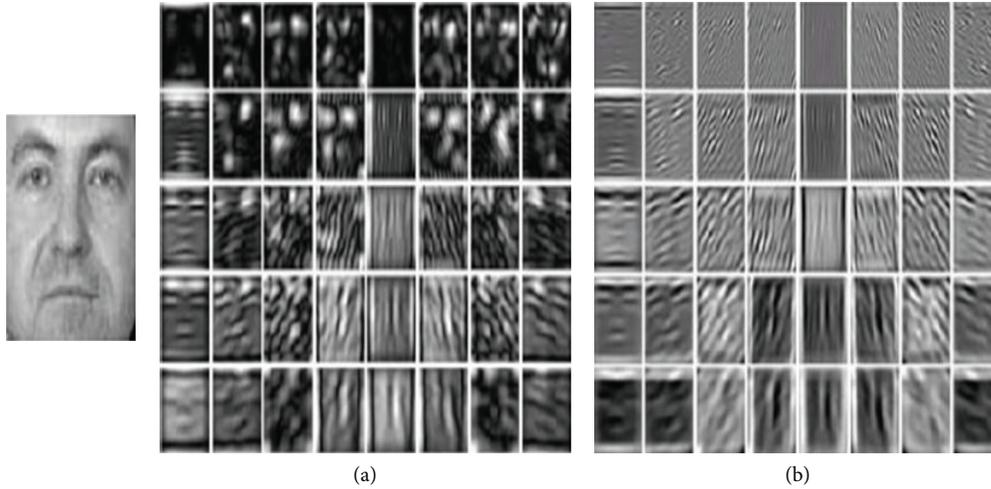


FIGURE 3: Example of facial representation in Gabor wavelet of face: (a) amplitude response, (b) phase response with 40 Gabor filters (5 scales and 8 orientations).

- (b) The frequency  $f$ : it specifies the wavelengths of the function; wavelets with large wavelength are sensitive to progressive illumination changes in the image, whereas low wavelengths are sensitive to contours.
- (c) The standard deviation  $\sigma$ : this parameter defines the radius of the Gaussian. The size of the Gaussian determines the number of pixels of image taken into account for the convolution operator.

The set of Gabor filters is characterized by a certain number of resolutions, orientations, and frequencies known as “characteristics.”

Figure 4 shows the response of the Gabor filter when varying the total parameters. After a lot of experiments, it has been proved that the following parameters,  $\sigma_x = 2$ ;  $\sigma_y = 4$ ;  $\theta = \pi/3$ ;  $f = 16$ , are relevant for face parameterization and build our face recognition system that will be described in future sections.

Gabor filters are used in contour detection; this parameter can be used to differentiate an individual from another. They represent also the characteristic points in the human face. They are widely used in texture characterization and in contour detection.

A simple Gabor filter can be expressed as the product of a Gaussian and a complex exponential. The resultant filtered image IG can be illustrated for the choice orientation and standard deviations by Figure 5.

We use only the magnitude information without the phase; the module of the filtered image shows local information for the details of the image (frequency decomposition) and for the orientation (directional information). Magnitude analysis permits us to detect local structures of interest. The filtered image presents the characteristic features of a person like lip, eyes, and so forth.

Face recognition using Gabor filters was firstly introduced by Lades et al. [9] and soon proved to be a very effective means in human facial features extraction. Wang et al. [10]

proposed a face recognition algorithm combined vector features consisting of the magnitude of Gabor, PCA, and classification SVM [6].

The Gabor wavelet is a continuous filter, making the filter process computationally demanding. A wavelet is a waveform of limited duration that has an average value of zero. The limited duration of the wavelet enables preservation of spatial information in the signal [11].

The following paragraph presents an overview of discrete wavelet analysis followed by the application for face parameterization.

**2.2. Wavelet Analysis.** The discrete wavelet transform is analogous to the Fourier transform with the exception that the DWT uses scaled and shifted versions of a wavelet. It decomposes a signal into a sum of shifted and scaled wavelets. The DWT kernels are very similar to Gabor kernels and exhibit properties of horizontal, vertical, and diagonal directionality. Also, the DWT possesses the additional advantages of sparse representation and nonredundant computation, which make it advantageous to employ the DWT over the Gabor transform for facial feature extraction [11]. DWT has been successfully used for denoising, compression, and feature detection applications in image processing to provide faster performance times and achieve results comparable to those employing Gabor wavelets [11].

As described in [11], we present a brief review of continuous and discrete wavelet relevant to our analysis; 2D discrete wavelet decomposition is then described and applied for face recognition.

The continuous wavelet transform (CWT) between a signal  $s(t)$  and a wavelet  $\Psi(t)$  is mathematically defined as [11]

$$C(a, b) = \frac{1}{\sqrt{a}} \int s(t) \Psi\left(\frac{t-b}{a}\right) dt, \quad (10)$$

where  $a$  is the scale,  $t$  is the time, and  $b$  is the shift.

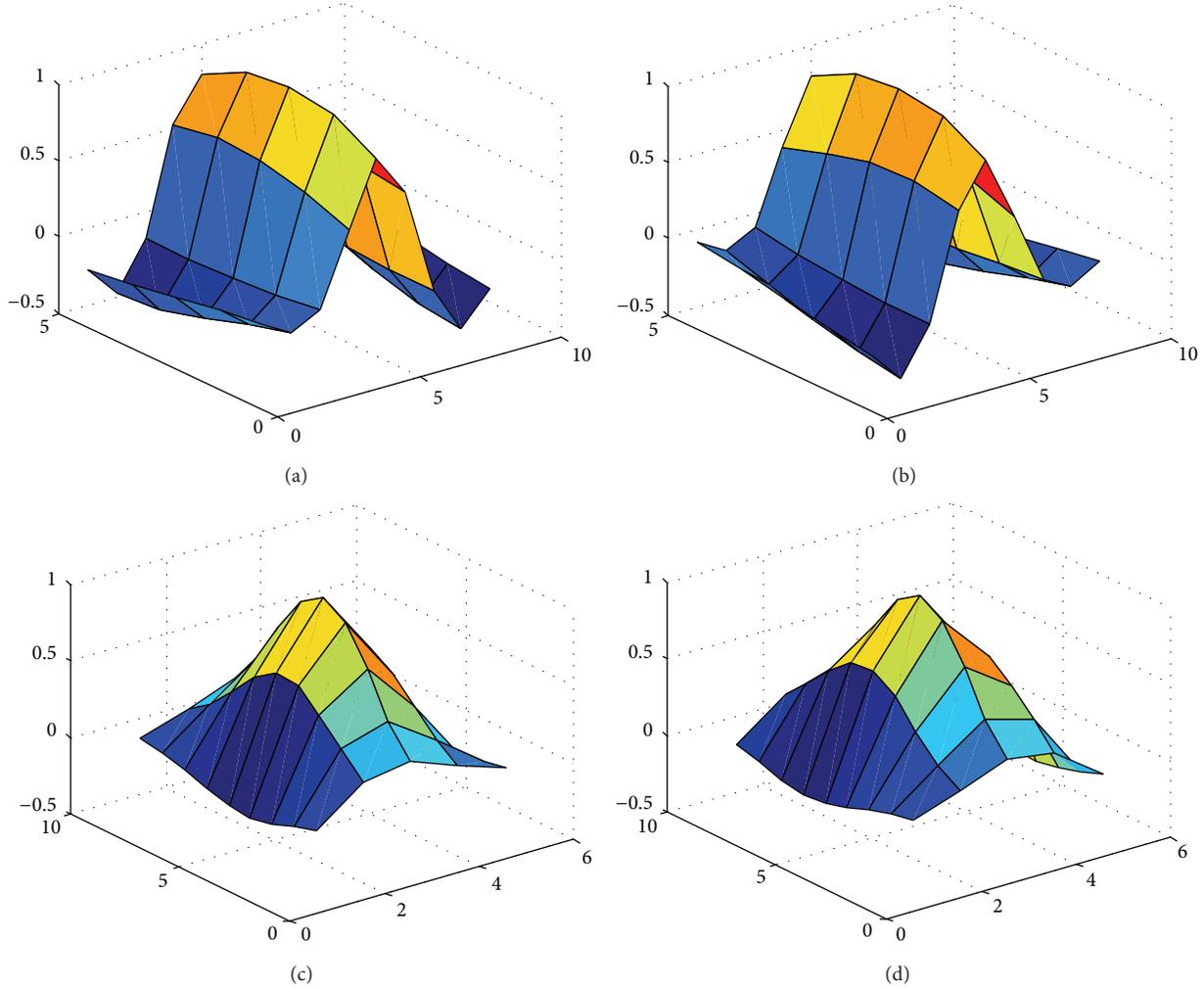


FIGURE 4: The response of the filter using the parameters: (a)  $\sigma_x = 2; \sigma_y = 4; \theta = \pi/3; f = 16$ , (b)  $\sigma_x = 2; \sigma_y = 4; \theta = 2\pi/3; f = 16$ , (c)  $\sigma_x = 4; \sigma_y = 2; \theta = \pi/3; f = 16$ , and (d)  $\sigma_x = 4; \sigma_y = 2; \theta = 2\pi/3; f = 16$ .

The DWT is obtained by restricting the scale,  $a$ , to powers of 2 and the position,  $b$ , to integer multiples of the scales and is given by

$$c_{j,k} = C \left( \frac{1}{2^j}, \frac{k}{2^j} \right) = 2^{j/2} \int_{-\infty}^{+\infty} x(t) \Psi(2^j t - k) dt, \quad (11)$$

where  $j$  and  $k$  are integers and  $\Psi_{j,k}$  are orthogonal baby wavelets, defined as

$$\Psi_{j,k} = 2^{j/2} \Psi(2^j t - k). \quad (12)$$

Substituting (12) into (11) yields

$$c_{j,k} = \int_{-\infty}^{+\infty} x(t) \Psi_{j,k} dt. \quad (13)$$

The restricted choice of scale and position results in a subsample of coefficients. Baby wavelets  $\Psi_{j,k}$  have an associated baby scaling function given by

$$\phi_{j,k}(t) = 2^{j/2} \phi(2^j t - k). \quad (14)$$

The scaling function or dilation equation can be expressed in terms of low-pass filter coefficients  $h_0(n)$  as

$$\phi(t) = \sum_n h_0(n) \sqrt{2} \phi(2t - n). \quad (15)$$

In addition, the wavelet function itself can be expressed in terms of high-pass filter coefficients  $h_1(n)$  as

$$\Psi(t) = \sum_n h_1(n) \sqrt{2} \phi(2t - n). \quad (16)$$

Also, a signal  $x(t)$  can be represented by the scaling and wavelet functions:

$$x(t) = \sum_k cA_1(k) \phi_{j-1,k}(t) + \sum_k cD_1(k) \Psi_{j-1,k}(t). \quad (17)$$

Here,  $cA_1(k)$  are the approximation coefficients at level 1, and  $cD_1(k)$  are the detail coefficients at level 1. Equation (17) represents the analysis of a signal that can be repeated numerous times to decompose the signal into lower subspaces as shown in Figure 6 [11].

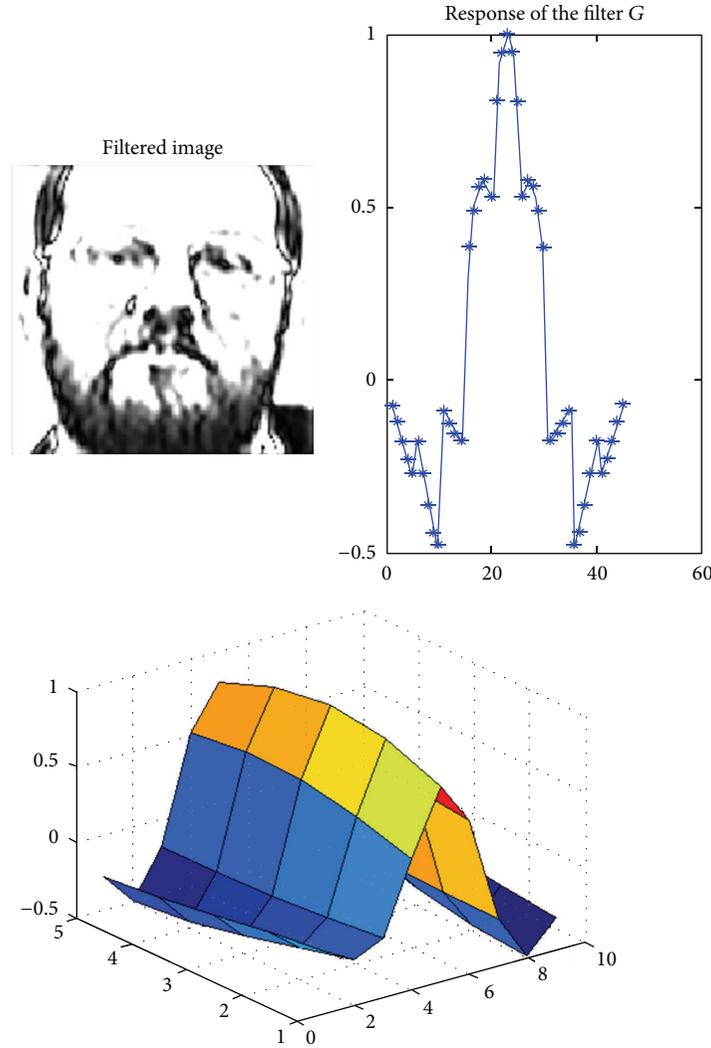


FIGURE 5: The response of the filter using the following parameters:  $\sigma_x = 2$ ;  $\sigma_y = 4$ ;  $\theta = \pi/3$ ;  $f = 16$ .

The detail and approximation coefficients are found using the filter coefficients  $h_0(n)$  and  $h_1(n)$ :

$$cA_1(k) = \sum_n cA_0(n) h_0(n - 2k). \tag{18}$$

Equation (18) shows that the level 1 approximation coefficients  $cA_1$  can be found by convolving  $cA_0$  with the low-pass filter  $h_0(n)$  and downsampling by 2 (indicated by  $\downarrow 2$ ), which simply means disregarding every second output coefficient. Similarly, the level 1 detailed coefficient,  $cD_1$ , can be found by convolving  $cD_0$  with the high-pass filter  $h_1(n)$  and downsampling by 2. Therefore, the DWT can be performed using the filter coefficients  $h_0(n)$  and  $h_1(n)$ . The process is applied repeatedly to produce higher levels of approximation and detail coefficients as shown in Figure 6. Each time the filter outputs are downsampled, the number of samples is halved. The original signal,  $x(t)$ , is assumed to start in the subspace  $V_0$ . Therefore, the level 0 approximation coefficients are the discrete values of the signal  $x(n)$  [11].

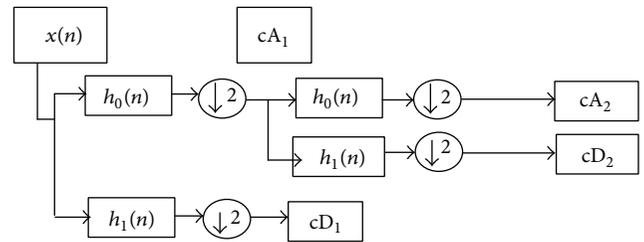


FIGURE 6: DWT using filter banks.

**2.3. Discrete Wavelet Transform (DWT) Applied for Face Recognition.** Discrete wavelet transform (DWT) is a suitable tool for extracting image features because it allows the analysis of images on various levels of resolution. Typically, low-pass and high-pass filters are used for decomposing the original image.

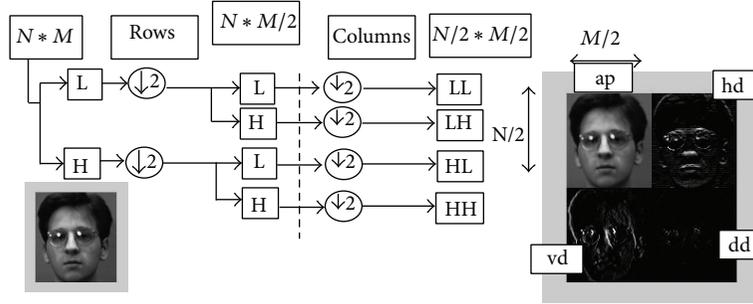


FIGURE 7: First level of decomposition of a facial image.

The low-pass filter results in an approximation image and the high-pass filter generates a detail image. The approximation image can be further split into a deeper level of approximation and detail according to different applications [12].

Suppose that the size of an input image is  $N \times M$ . At the first filtering in the horizontal direction of downsampling, the size of images will be reduced to  $N \times (M/2)$ . After further filtering and downsampling in the vertical direction, four subimages are obtained, each being of size  $(N/2) \times (M/2)$ . The outputs of these filters are given by [12]

$$a_{j+1}[p] = \sum_{n=-\infty}^{n=+\infty} l[n-2p] a_j[n], \tag{19}$$

$$d_{j+1}[p] = \sum_{n=-\infty}^{n=+\infty} h[n-2p] a_j[n],$$

where  $l[n]$  and  $h[n]$  are coefficients of low-pass and high-pass filters, respectively.

Accordingly, we can obtain four images denoted as LL, HL, LH, and HH. The LL image is generated by two continuous low-pass filters; HL is filtered by a high-pass filter first and a low-pass filter later; LH is created using a low-pass filter followed by a high-pass filter; HH is generated by two successive high-pass filters [12].

Figure 7 presents the first level of decomposition applied for a facial image of dimension  $N * M$  pixels, where

- LL describes the approximation (ap in Figure 7),
- LH and HL describe the horizontal and vertical details,
- HH describes the diagonal detail.

After first level wavelet decomposition, the output images become input images of second level decomposition.

Every subimage can be decomposed further into smaller images by repeating the above procedure. The main feature of DWT is the multiscale representation of a function. By using the wavelets, a given image can be analyzed at various levels of resolution. Since the LL part contains most important information and discards the effect of noises and irrelevant parts, we extract features from the LL part of the first level decomposition. The reasons are that the LL part keeps the necessary information and the dimensionality of



FIGURE 8: 1st level decomposition.

the image is reduced sufficiently for computation at the next stage [12].

One major advantage afforded by wavelets is the ability to perform local analysis, that is, to analyze a localized area of a larger signal. In wavelet analysis, we often speak about approximations and details. The approximations are the high-scale, low-frequency components of the signal. The details are the low-scale, high-frequency components.

A lot of tests have been carried; we adopt the Haar, db2, db4, db8, and bior2.2 wavelet with first level decomposition. The DWT feature vector is assigned for each individual and for the whole database containing facial images. The choice of wavelet function is very important task in features extraction. The effectiveness of recognition system depends on this selection.

Figure 8 shows the first level decomposition using Haar wavelet; a facial image size from computer vision database is reduced to  $(100 * 90)$  with 1st level decomposition. The dimension becomes  $(50 * 45)$  with 2nd level of decomposition.

The graph in Figure 9 describes the distribution of the resultant signal or images (approximation and detail) with

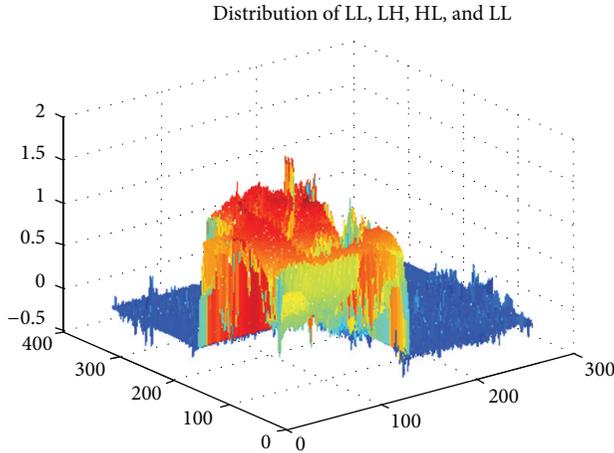


FIGURE 9: Distribution of 1st level with Haar DWT.

first level decomposition using Haar wavelet. It is clear that a major energy of the approximation information is concentrated in the region with red colour, whereas the horizontal and vertical details appear with a little energy in blue colour.

From our experiments and after a lot of tests, only first level decomposition is sufficient to describe correctly our face image and to build a face recognition system with good recognition rate.

The DWT is able to extract the horizontal and vertical details of a facial image. However, unlike the Gabor wavelets, which can be oriented to any angle, the 2D DWT is restricted in terms of orientation to horizontal, diagonal, and directionality ( $0^\circ$ ,  $45^\circ$ , and  $90^\circ$ ). Since features of the face in a 2D grayscale image are a union or combination of infinitely oriented lines, the ability to filter at several orientations is necessary to achieve acceptable face recognition performance [11].

The DWT analysis applied for face recognition has a lot of advantages compared to Gabor analysis; the first benefit yields with dimensionality reduction which permits us less computational complexity, whereas the second advantage concerns the image decomposition in frequency domain into subbands at different scales.

In this paper we examine the effects of different feature extraction methods like DWT and Gabor on the face recognition system and compare with other approaches like DCT feature extraction and PCA or LDA spatial feature reduction realized and discussed in a previous work [13].

### 3. Simulation Results

**3.1. Face Databases.** Our experiments were performed on two face databases, computer vision, and ORL database.

First, we used the computer vision dataset; it contains frontal images of 395 individuals, and each person has 20 frontal images. This dataset contains images of people of various racial origins, mainly of first year undergraduate students, so the majority of individuals are between 18 and 20 years old but some older individuals are also present.



FIGURE 10: Examples from computer vision database [14].



FIGURE 11: Some examples of acquisition with different face orientation and with different illuminations from ORL database [15].

Images for one person differ from each other in lighting and facial expression. The images are 256-colour level with size of  $200 \times 180$ . Some individuals are wearing glasses and beards. The total number of images is 7900. Samples of the database are shown in Figure 10.

Wavelet has got a great attention for face recognition; the wavelet transform has emerged as a cutting edge technology. The discrete wavelet transform (DWT) is a linear, invertible, and fast orthogonal operation. Wavelets are a good mathematical tool for hierarchically decomposing occupations. Without degrading the quality of the image to an unacceptable level image compression minimizes the size in bytes of a graphics file. The reduction in file size permits more images to be stored in a certain amount of disk or memory space [3].

We used also ORL database for analysing the performance of our classifier; this database contains a set of face images taken at the Olivetti Research Laboratory (ORL) in Cambridge University, UK. We use 200 images of 20 individuals. For some subjects, the images were taken at different times, which contain quite a high degree of variability in lighting, facial expression (open/closed eyes, smiling/nonsmiling, etc.), pose (upright, frontal position, etc.), and facial details (glasses/no glasses). All images are 8-bit grayscale of size  $112 \times 92$  pixels. Samples of the ORL database are shown in Figure 11.

**3.2. Face Characterization.** Our face recognition system is tested first on the computer vision database. We use 200 images of 10 subjects. In the following experiments, ten images are randomly selected as the training samples and another ten images as test images. Therefore, a total of 100 images (10 for each individual) are used for training and another 100 for testing and there are no overlaps between the training and testing sets.

For ORL database, we select randomly 100 samples (5 for each individual) for training. The remaining 100 samples are used as the test set.

In our face recognition experiments, we extract features of facial images from different wavelet transforms (Haar, Daubechies, and Biorthogonal) by decomposing face image in LL subbands at first level. The wavelets used are Haar, db2, db4, db8, and bior2.2. In addition, we apply Gabor wavelet in order to characterize our facial images. Feature vectors extracted from each method are fed into a MLP and RBF neural network classifier to perform efficiency of our face recognition.

Before applying our face recognition system based on MLP and RBF neural network, we have applied the correlation criterion between training and test features first for Haar DWT and Gabor features for primary analysis.

Figure 12 shows variation of correlation for some individuals with the 10 test images from the computer vision dataset. We noticed that we have a good separation (correlation coefficient is greater than a threshold) for the first, the second, and the fifth individual and bad separation for the third and the sixth individual. A threshold of 0.8 was chosen for the coefficient of correlation to decide the correct classification. The discrimination is clear for some individuals, whereas results obtained were not very satisfactory; a recognition rate of 72.5% was obtained.

We have computed correlation between training and test feature vectors on the computer vision dataset using ten classes and features from Gabor and Haar DWT. Figure 13 shows recognition rate obtained for each class; a global recognition rate of 72 to 73% was obtained. We noticed bad classification for the third, the sixth, and the eight classes.

In addition, the investigation of number of training images when varied from 5 to 10 for computer vision dataset permits us to perform better recognition rate for 10 facial images in training and testing phase. Figure 14 shows the correct classification for each class; a global recognition rate of 72% was obtained with 10 facial images, whereas only 67% was obtained with 5 training images for each class.

However, the correct classification rate depends on the number of variables in each class which will improve the correct classification rate for a large number of training images for each individual.

We conclude from this analysis that the choice of 10 images for training and test is more efficient to characterize the total database and to perform an appreciable correct classification only by applying the correlation criterion.

We investigate now the use of a neural network classifier to build a face recognition system based on discrete wavelet and Gabor characterization. The following paragraph describes a brief theory on the classifiers used in our work and discusses the simulation results obtained.

**3.3. Face Recognition System Using Artificial Neural Network.** Neural networks are feedforward and use the backpropagation algorithm. We imply feedforward networks and backpropagation algorithm (plus full connectivity). While inputs are fed to the ANN forwardly, the “back” in backpropagation algorithm refers to the direction to which the error is transmitted [16].

Face recognition is achieved by employing a multilayer perceptron with backpropagation algorithm and radial basis

function classifier. For the MLP classifier, the first layer receives the Gabor features or the DWT coefficients with first level decomposition. The number of nodes in this layer is equal to the dimension of the feature vector incorporating the DWT or Gabor features. The number of the output nodes is equal to the number of the registered individuals. Finally, the number of the hidden nodes is chosen by the user.

The features test matrix is defined with variables called target; the target matrix has the same dimension as the training matrix. The network is trained to output a 1 in the correct position of the output vector and to fill the rest of the output vector with 0's.

The MLP NN trained present fast convergence and the training process terminated within 4000 to 6000 epochs depending on the facial database and the size of feature descriptors, with the summed squared error (SSE) reaching the prespecified goal (10-3).

We used log-sigmoid and tang-sigmoid functions as a transfer function at all neurons (in hidden layer and output layer); log-sigmoid function is ideal for face recognition system using Gabor filters and wavelet decomposition.

In order to show the importance of processing elements, we trained our MLP classifier with variable hidden unit from 10 to 100. For a small number of neurons (10 to 20) in the hidden layer we observed large mean squared error (MSE), so low accuracy. The MLP generalizes poorly. After ~60 to 70 neurons, MSE came back to the levels of a system with only 10 neurons in the hidden layer. When there are too many neurons, poor performance is a direct effect of overfitting. The system overfits the training data and does not perform well on novel patterns.

In addition, we have used an RBF classifier to decide the correct classification. An RBF neural network, shown in Figure 15, can be considered a mapping:  $R^r \rightarrow R^s$ .

Let  $P \in R^r$  be the input vector and let  $C_i \in R^r$  ( $1 \leq i \leq u$ ) be the prototype of the input vectors [16]. The output of each RBF unit is as follows:

$$R_i(P) = R_i(\|P - C_i\|) \quad i = 1, \dots, u, \quad (20)$$

where  $\|\cdot\|$  indicates the Euclidean norm on the input space. Usually, the Gaussian function (Figure 15) is preferred among all possible radial basis functions due to the fact that it is factorizable:

$$R_i(P) = \exp\left[-\frac{\|P - C_i\|^2}{\sigma_i^2}\right], \quad (21)$$

where  $\sigma_i$  is the width of the  $i$ th RBF unit. The  $j$ th output  $y_j(P)$  of a neural network is

$$y_j(P) = \sum_{i=1}^u R_i(P) * w(j, i), \quad (22)$$

where  $R_0 = 1$ ,  $w(j, i)$  is the weight or strength of the  $i$ th receptive field to the  $j$ th output [16].

According to [16], (21) and (22) show that the outputs of an RBF neural classifier are characterized by a linear discriminant function. They generate linear decision

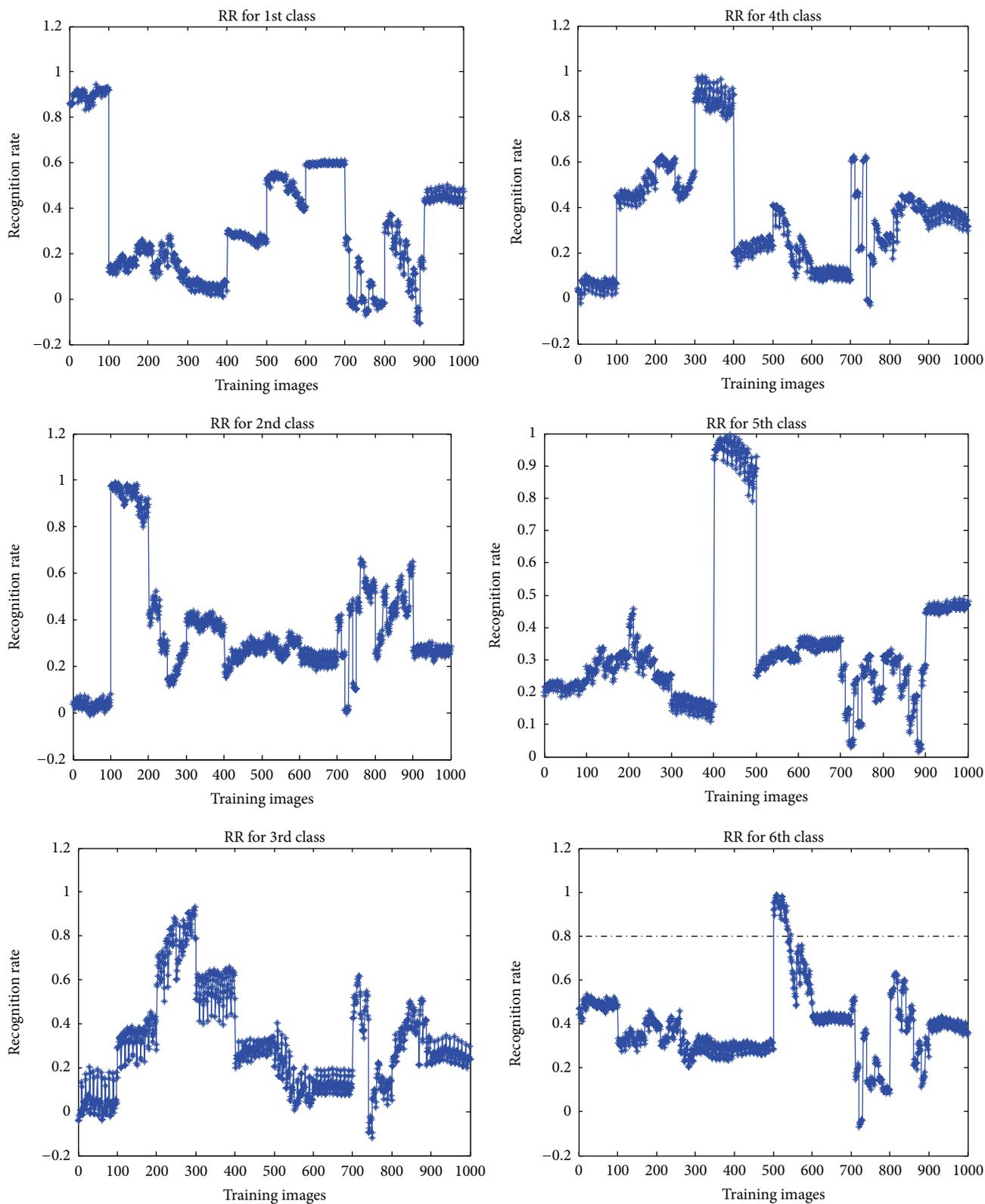


FIGURE 12: Variation of correlation for some individuals with the 10 test images from the computer vision dataset for Haar representation.

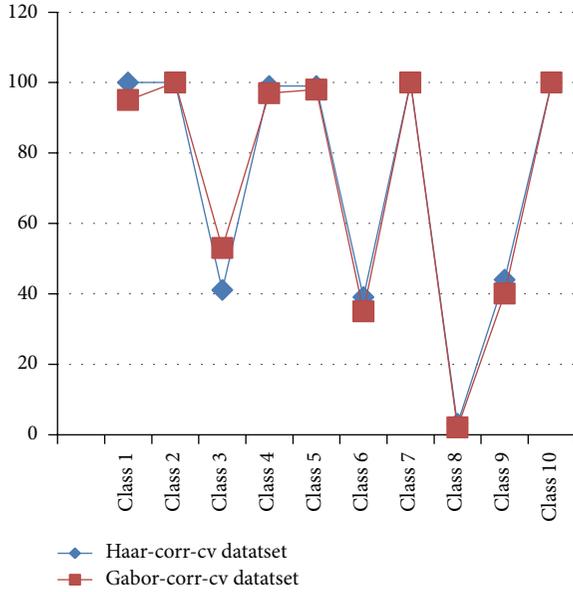


FIGURE 13: Recognition rate obtained for Gabor and Haar using correlation analysis applied for computer vision dataset.

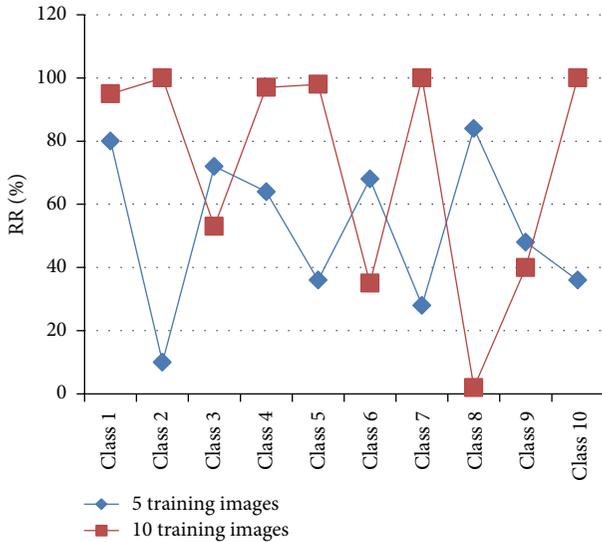


FIGURE 14: Recognition rate obtained for Gabor using correlation analysis applied for computer vision dataset when varying the number of training images from 5 to 10.

boundaries (hyperplanes) in the output space. Consequently, the performance of an RBF neural classifier strongly depends on the separability of classes in the  $u$ -dimensional space generated by the nonlinear transformation carried out by the  $u$  RBF units [16].

According to Cover’s theorem on the separability of patterns where in a complex pattern classification problem cast in a high-dimensional space nonlinearly is more likely to be linearly separable than in a low-dimensional space, the number of Gaussian nodes  $u \geq r$ , where  $r$  is the dimension of input space. On the other hand, the increase of Gaussian

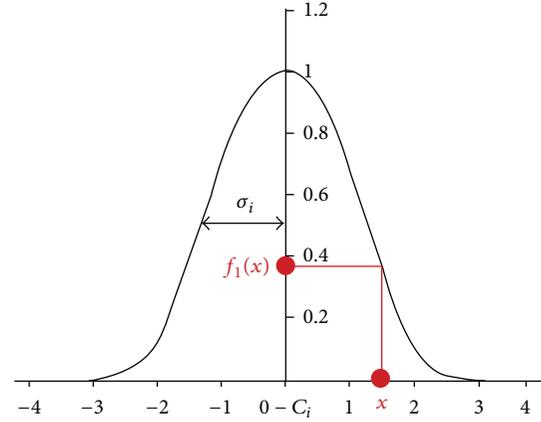


FIGURE 15: Radial basis function.

units may result in poor generalization because of overfitting, especially, in the case of small training sets [16]. It is important to analyze the training patterns for the appropriate choice of RBF hidden nodes.

Geometrically, the key idea of an RBF neural network is to partition the input space into a number of subspaces which are in the form of hyperspheres. Accordingly, clustering algorithms ( $k$ -means clustering, fuzzy-means clustering, and hierarchical clustering) which are widely used in RBF neural networks are a logical approach to solve the problems [16].

However, it should be noted that these clustering approaches are inherently unsupervised learning algorithms as no category information about patterns is used. While considering the category information of training patterns, it should be emphasized that the class memberships not only depended on the distance of patterns but also depended on the Gaussian widths [16].

To perform the efficiency of our proposed method applied for ORL and computer vision CV database, we compared recognition rate (RR) obtained for each database and for each parameterization. Section 3.4 will describe the face identification experiments using the MLP classifier whereas Section 3.5 presents the simulation results for the RBF classifier.

3.4. MLP Neural Network Classifier for Face Recognition. Once feature extraction is studied, the novel vectors are fed into a MLP and RBF classifier to decide the correct classification rate.

After Gabor filtering, we construct our training matrix consisting of the module of filtered image. The characteristic features vectors are fed into a MLP neural classifier with its architecture. We use only the magnitude information for Gabor characterization.

Using Haar discrete wavelet and the ORL database, for example, the images of size  $112 * 92$  pixels, we retain the approximation vector of size  $(56 * 46)$ ; the resultant vector descriptor using the reshape function by Matlab permits us to have a novel vector of size 2576 of all the images in ORL database. The approximation vector of db2, db4, db8, and bior2.2 is 2679, 2891, 3339, and 2784, respectively. Table 1 summarizes the total vector size of feature vectors

TABLE 1: Size of feature vector with DWT approach.

Method (wavelet)	Haar	db2	db4	db8	bior2.2
Size of feature vector (ORL: 112 * 92)	2576	2679	2891	3339	2784
Size of feature vector (CV: 200 * 180)	9000	9090	9579	10379	9384

obtained with each wavelet and for the two databases used in our work.

After calculating the DWT and Gabor transform, these feature vectors are calculated for the training set and then used to train the neural network; this architecture is called DWT-MLP (DWT-RBF) and Gabor-MLP (Gabor-RBF).

The performance of Gabor-MLP or DWT-MLP approach is evaluated by using the recognition rate (RR) standard defined by

$$R(i) = \frac{N_i}{N}, \quad (23)$$

where  $N$  is the total number of test sample tests and  $N_i$  is the number of test images recognized correctly.

The choice of feature extraction is a very important step, and then the application of a classifier such as MLP or RBF permits us to obtain good results for a development of an individual identification process.

Various transfer functions were tested for training the network and average minimum MSE on training is measured; log-sigmoid function is the most suitable transfer function. The MLP neural network is trained using learning rules, namely, conjugate gradient (CG). TRAINSCG is a network training function that updates weight and bias values according to the scaled conjugate gradient method. Finally network is tested on training and testing dataset.

### 3.4.1. Searching for the Optimal NN Topology

*Number of Hidden Neurons.* In our experiments we demonstrate how we decided on the optimal number of neurons for the hidden layer.

We attempted to find the optimal NN topology or architecture by training and testing MLPs, whose number of neurons in hidden layer varies from 10 to 100; we keep the number of maximum iterations constant (from 3000 to 6000) and the MSE training as presented in Table 2. We used sigmoid functions as transfer functions.

For a small number of neurons (10 to 20) in the hidden layer we observed large MSE, low accuracy, and generally large partiality. The MLPs were not flexible enough to capture the underlying process and due to large bias they generalize poorly. We faced similar generalization performance, even when we used too many neurons in the hidden layer. After ~90 neurons, MSE came back to the levels of a system with only 10 neurons in the hidden layer.

By adding more and more units in the hidden layer the training error can be made as small as desired but generally each additional unit will produce less and less benefits. With too many neurons, poor performance is a direct effect of overfitting. The system overfits the training data and does not perform well on novel patterns.

TABLE 2: The optimal neural network topology.

Parameter	Value	Fixed/variable
Dataset	Feature vectors (DWT or Gabor)	Fixed
DWT coefficients	Haar, db2, db4, bior2.2, Gabor	Variable
Training algorithm	Gradient descent	Fixed
Transfer function	Both layers use log-sigmoid	Fixed
Number of neurons in hidden layer	10, 20, 30, 40, ..., 90, 100	Variable
Minimum training MSE	0.001	Fixed
Maximum epochs	3000 to 6000	Variable

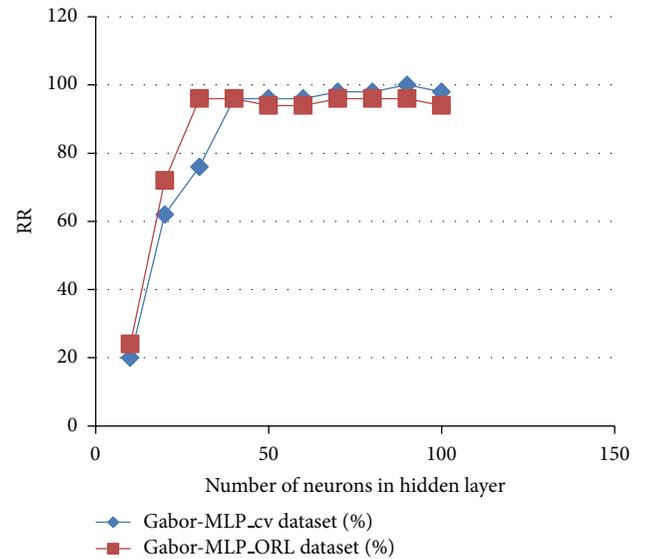


FIGURE 16: Recognition rate with Gabor-MLP for computer vision and ORL database.

Table 2 summarizes the optimal neural network topology or architecture found in our experiments to achieve best recognition rate.

Figure 16 shows recognition efficiency obtained for both computer vision and ORL database with Gabor wavelet characterization and MLP neural network classifier. Good recognition rate about 96% to 97% was obtained for a number of neurons superior to 60 for the two databases.

Figure 17 shows the variation of recognition rate when varying the number of neurons in hidden layer with db2, db4, db8, and bior2.2 DWT with MLP classifier.

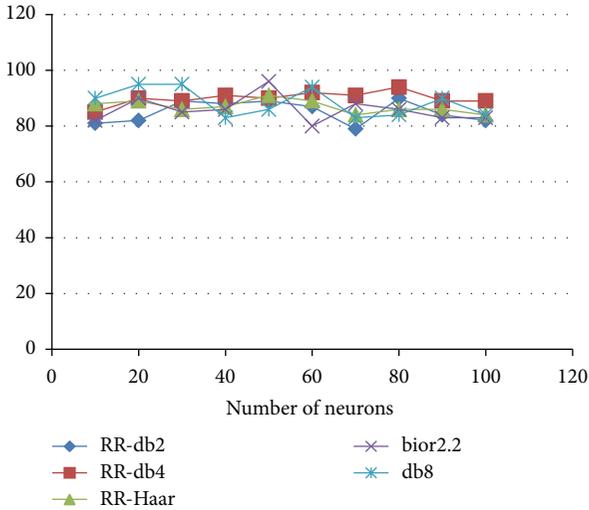


FIGURE 17: Recognition rate with DWT-MLP for computer vision database when varying number of neurons in hidden layer.

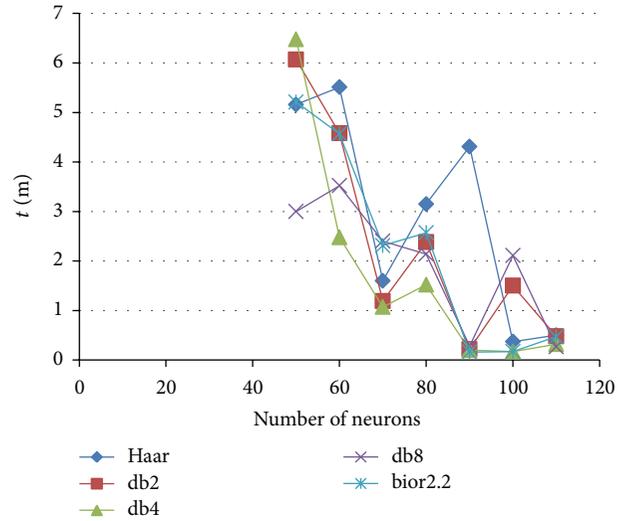


FIGURE 19: Recognition time obtained for ORL database using variable DWT with MLP classifier.

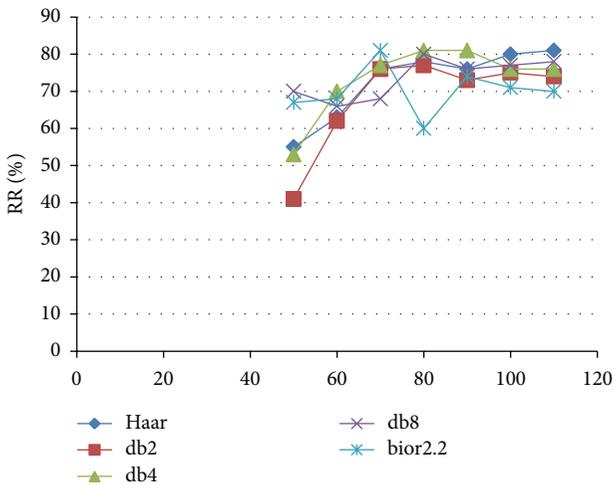


FIGURE 18: Recognition rate with DWT-MLP for ORL database when varying number of neurons in hidden layer.

The best recognition rate about 96% was obtained for db8 and bior2.2 wavelet. We conclude for this preliminary result that db8 and Biorthogonal 2.2 wavelet perform better than the Haar wavelet; good parametrization can be achieved with the two discrete wavelets only with 20 to 30 neurons in hidden layer.

In addition, results obtained for computer vision dataset are superior to those obtained for ORL dataset where the best recognition rate obtained is about 81%. Figure 18 shows efficiency of our recognition system for the different discrete wavelet obtained with ORL dataset studied in our work.

We also noticed that, for 20 to 40 neurons in hidden layer, we have no convergence with the MLP algorithm for ORL dataset; this can be explained by the fact that the number of neurons is not enough to train the MLP classifier. In addition,

facial images in ORL dataset present a lot of variabilities in orientation and in facial expressions.

All these parameters complicate the task of recognition and classification, whereas the advantage of DWT parametrization is essentially the reduction of feature extraction and time computation compared to Gabor wavelet. The classification time required for achieving recognition face system is presented in Figure 19 in order to provide speed comparisons of the different discrete wavelet transform characterization. Simulation results are computed on a PC intel core i3 with a 1.2 GHz processor.

It is clear that time recognition is faster for the different DWT transform when a number of neurons are superior than 80; we noticed from Figure 19 that corresponds to a fast convergence of the MLP classifier. We conclude the relationship between the optimal convergence of the neural network classifier and time required for this task.

### 3.5. RBF Neural Network Classifier for Face Recognition.

Simulations were carried with MATLAB 7.6 and using some predefined functions to train our RBF NN classifier. For RBF neural network simulation, a lot of tests have been carried. First we create an RBF function classifier using newrbe function to design a radial basis network with zero error on the design vectors. This function creates a two-layer network; the first layer has radbas neurons and calculates its weighted input with dist and its net input with netprod. The second layer has purlin neurons and calculates its weighted input with dotprod and its net inputs with netsum. Both layers have biases.

Newrbe sets the first layer weights to  $X'$ , where  $X$  is our training vector descriptors (Gabor filtered images and discrete wavelet decomposition) and the first layer biases are all set to  $0.8326/\text{spread}$  resulting in radial basis functions that cross 0.5 at weighted input of  $\pm \text{spread}$ . Spread is the smoother function approximation (neural network toolbox).

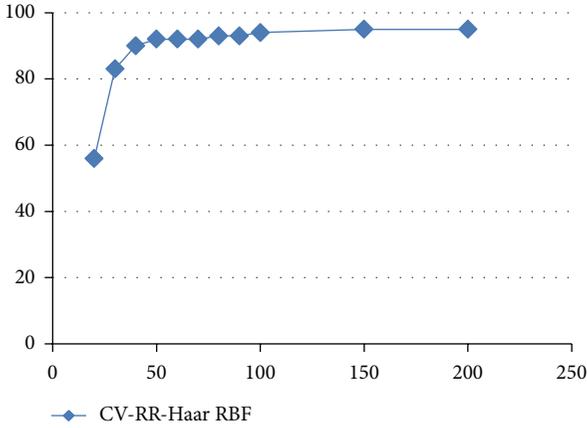


FIGURE 20: Recognition rate with Haar-RBF for computer vision database.

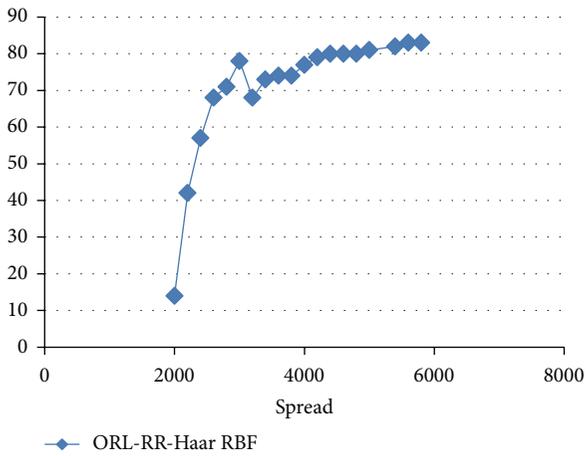


FIGURE 21: Recognition rate with Haar-RBF for ORL database.

For our simulation, we calculate recognition rate when varying the spread parameter from 10 to 250 for computer vision database, where variation is from 200 to 5600 for ORL database. We note that spread parameter depends on the quality of analyzed image and the database. Figures 20 and 21 show the variation of recognition rate when varying the spread parameter.

It is clear that recognition rate is about 95% for computer vision database obtained with a spread parameter equal to 90, whereas it is about 83% obtained for spread equal to 5200.

We conclude the following.

- (1) The use of wavelets is a good means to characterize the data with high speed compared to Gabor filters.
- (2) The Gabor wavelet can extract the object features in different orientations and different scales; on the contrary, the two-dimensional wavelets subband coefficients extract information in only two directions.
- (3) The MLP neural network is a good classifier for face identification using the conjugate gradient.

TABLE 3: Recognition rate for proposed method.

RR (%) classifier	Wavelet		Gabor wavelet	
	bior2.2 MLP	Haar-RBF	MLP	RBF
CV database	96%	95%	95%	94%
ORL database	81%	83%	83%	85%

Table 3 summarizes efficiency of the characterized method and classifier used in our work. Recognition performance is highly dependent on the facial dataset.

## 4. Discussion

Simulation results showed the efficiency of the proposed method; about 94 to 96% was obtained with Gabor filters and MLP/RBF classifier for computer vision database, whereas about 85% was obtained for ORL dataset.

We conclude that the best results are obtained for computer vision database; in addition, MLP can perform well with Gabor parameterization.

We also conclude that MLP classifier performs better than RBF where 94% of correct classification was obtained for Gabor parameterization, whereas for bior2.2 and db8 wavelet decomposition, it can achieve 96% with the two-classifier neural network.

We can explain these results that computer vision database was obtained in controlled environment with a little variation in illumination and pose, whereas ORL database presents a lot of variation in pose, illumination, and facial expression. All these parameters decrease the efficiency of our method.

The results obtained show the importance of designing neural network for face recognition. In addition, feature extractions like Gabor filters and wavelet decomposition perform well for the total database. We conclude an improvement in face recognition when using the Gabor parameterization for ORL dataset but time required for classification is long compared with discrete wavelet. This can be explained by the following: the Gabor wavelet can extract the object features in different orientations and different scales, whereas the two-dimensional wavelets subband analysis extracts information in only two directions. Both Gabor filters and wavelet decomposition characterize the facial images where the best recognition rate was obtained for Biorthogonal 2.2 wavelet and MLP classifier.

*4.1. Comparisons with Other Approaches.* Many face recognition approaches have been performed on the computer vision database. In order to compare the recognition performance, we choose some recent approaches tested under similar conditions for comparison. Approaches are evaluated on recognition rate.

The performance of the Gabor-MLP and DWT-RBF algorithm in comparison with some existing algorithms for face recognition is listed in Table 4. It is evident from the table

TABLE 4: Comparison between DCT-MLP algorithm and other statistic methods (computer vision database).

Method	PCA	LDA	DCT + MLP	Gabor-MLP	DWT-MLP
Recognition accuracy (CV)	60% [13]	90% [13]	96% [13]	95	96

that our proposed method achieves high recognition rate as well as high training and recognition speed.

Table 4 presents results in terms of correct classification rate obtained in our previous work [13] and compared with our approach for computer vision dataset.

It is clear from Table 4 that our proposed approach achieves high recognition rate. It can be seen that the DWT-MLP algorithm exhibits better performance than other linear classifiers like PCA and LDA.

We conclude that the MLP neural network with only one hidden layer performs well compared to the RBF classifier for the two databases. We also find the methods extracting the feature DCT are good for the recognition system, whereas the time required is long (about 5 mn for 100 training images) compared to DWT extraction and classification.

A lot of studies have investigated the use of discrete wavelet and other statistical reduction methods like PCA for ORL, computer vision, and FERET database. Results obtained are contradictory; Zhang et al. present in their work [17] a face recognition system by combining two recently proposed neural network models, namely, Gabor wavelet network (GWN) and kernel associative memory (KAM), into a unified structure called Gabor wavelet associative memory (GWAM) using three popular face databases, that is, FERET database, Olivetti-Oracle Research Lab (ORL) database, and AR face database. A recognition rate about 98% was obtained with ORL database and 96% for AR dataset [17].

AlEnzi et al. apply different levels of discrete wavelet transform (DWT) in different levels and the two-dimensional principal component (2DPCA) to compute the face recognition accuracy processing it through the ORL image database [18]. From their experiments and combining 2-level DWL technique with 2DPCA method, the highest recognition accuracy (94.5%) was obtained with a time rate of 4.28 sec.

Authors in [19] propose a face recognition scheme that combined wavelet transform and 2-DPCA applied for ORL dataset. They compared the recognition performances of various wavelets at 8 levels, 1 to 8. Accuracy with only 2-DPCA is 85% whereas the average accuracy of 2-DPCA with wavelet in subbands 2 and 3 is 94.5%.

Jain and Bhati [20] develop a face recognition system using neural network with PCA to reduce the dimensionality in wavelet domain. For face detection Haar wavelet is used to form the coefficient matrix and PCA is used for extracting features. These features were used to train the artificial neural networks classifier applied for ORL dataset. The best recognition rate obtained was about 81.11%.

From our overall analysis and comparison, we agree with Jain and Bhati where a similar correct classification rate about

81% to 83% was obtained for ORL dataset. The facial images varied in facial expression and illumination. PCA reduction can be applied in future work to investigate the effect of spatial reduction.

## 5. Conclusion

In this paper, Gabor and discrete wavelet based feature extraction methods are proposed for the extraction of features from facial images. Face recognition experiments were carried out by using MLP and RBF classifier. We have used two facial databases the ORL and computer vision. We have implemented and studied two neural networks architectures trained by different characteristic feature vectors like discrete wavelet transform (Haar, db2, db4, db8, and bior2.2) and Gabor wavelet.

Simulation results showed the efficiency of the proposed method; about 94 to 96% was obtained with Gabor filters and MLP classifier for computer vision database, whereas about 83 to 85% was obtained with discrete wavelet decomposition using the approximation feature descriptor for ORL database. Results obtained depend first on the quality of stored images, on the classifier, and also on the characterization method.

The presented MLP model for face recognition improves the face recognition performance by reducing the input features and performs well compared to the RBF classifier. Simulation results on computer vision face database show that the proposed method achieves high training and recognition speed, as well as high recognition rate about 96%. Another classifier like SVM can be applied for comparison with our method and other parameterizations like contourlets or other discrete wavelets could be investigated to perform better recognition rate. In future work, we plan to apply curvelets analysis which is better at handling curve discontinuities.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] M. Yang and L. Zhang, "Gabor feature based sparse representation for face recognition with gabor occlusion dictionary," in *Proceedings of the 11th European Conference on Computer Vision (ECCV '10)*, pp. 448–461, Springer, Crete, Greece, 2010.
- [2] S. Kalaimagal and M. Edwin Jaya Singh, "Face recognition using gabor volume based local binary pattern," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 2, no. 6, 2012.
- [3] A. Venus, M. Alfiras, and F. Alsaqre, "Face recognition algorithm using two dimensional principal component analysis based on discrete wavelet transform," in *Digital Information Processing and Communications*, V. Snaes, J. Platos, and E. El-Qawasmeh, Eds., vol. 188 of *Communications in Computer and Information Science*, pp. 426–438, Springer, Berlin, Germany, 2011.
- [4] A. Bhuiyan and C. H. Liu, "On face recognition using gabor filters," *International Journal of Computer, Information, Systems and Control Engineering*, vol. 1, no. 4, pp. 837–842, 2007.

- [5] J. Wang and J. Cheng, "Face recognition based on fusion of Gabor and 2DPCA features," in *Proceedings of the 18th International Symposium on Intelligent Signal Processing and Communication Systems*, pp. 1–4, IEEE, Chengdu, China, December 2010.
- [6] F. Bellakhddhar, K. Loukil, and M. Abid, "Face recognition approach using Gabor Wavelets, PCA and SVM," *International Journal of Computer Science Issues*, vol. 10, no. 2, p. 201, 2013.
- [7] L. Zhao, C. Yang, F. Pan, and J. Wang, "Face recognition based on Gabor with 2DPCA and PCA," in *Proceedings of the 24th Chinese Control and Decision Conference (CCDC '12)*, pp. 2632–2635, May 2012.
- [8] T. Andrysiak and M. Choras, "Image retrieval based on hierarchical Gabor filters," *International Journal of Applied Mathematics and Computer Science*, vol. 15, no. 4, pp. 471–480, 2005.
- [9] M. Lades, J. C. Vorbrueggen, J. Buhmann et al., "Distortion invariant object recognition in the dynamic link architecture," *IEEE Transactions on Computers*, vol. 42, no. 3, pp. 300–311, 1993.
- [10] X.-M. Wang, C. Huang, G.-Y. Ni, and J.-G. Liu, "Face recognition based on face Gabor image and SVM," in *Proceedings of the 2nd International Congress on Image and Signal Processing (CISP '09)*, pp. 1–4, IEEE, Tianjin, China, October 2009.
- [11] M. Meade, S. C. Sivakumar, and W. J. Phillips, "Comparative performance of principal component analysis, Gabor wavelets and discrete wavelet transforms for face recognition," *Canadian Journal of Electrical and Computer Engineering*, vol. 30, no. 2, pp. 93–102, 2005.
- [12] F. Y. Shih, C.-F. Chuang, and P. S. P. Wang, "Performance comparisons of facial expression recognition in JAFFE database," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 22, no. 3, pp. 445–459, 2008.
- [13] F. Z. Chelali and A. Djeradi, "Face recognition system using discrete cosine transform combined with MLP and RBF neural networks," *International Journal of Mobile Computing and Multimedia Communications*, vol. 4, no. 4, pp. 1–35, 2012.
- [14] "Computer Vision Science Research Projects data set," <http://cmp.felk.cvut.cz/~spacelib/faces/>.
- [15] ORL face databases, <http://www.face-rec.org/databases/>.
- [16] M. J. Er, W. Chen, and S. Wu, "High-speed face recognition based on discrete cosine transform and RBF neural networks," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 679–691, 2005.
- [17] H. Zhang, B. Zhang, W. Huang, and Q. Tian, "Gabor wavelet associative memory for face recognition," *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 275–278, 2005.
- [18] V. AlEnzi, M. Alfiras, and F. Alsaqre, "Face recognition algorithm using two dimensional principal component analysis based on discrete wavelet transform," in *Digital Information Processing and Communications*, V. Snasel, J. Platos, and E. El-Qawasmeh, Eds., vol. 188 of *Communications in Computer and Information Science*, pp. 426–438, Springer, Berlin, Germany, 2011.
- [19] K. S. Kinage and S. G. Bhirud, "Face recognition based on two-dimensional PCA on wavelet subband," *International Journal of Recent Trends in Engineering*, vol. 2, no. 2, pp. 51–54, 2009.
- [20] S. Jain and D. Bhati, "Face recognition using ANN with reduce feature by PCA," *International Journal of Scientific Engineering and Technology*, vol. 2, no. 2, pp. 595–599, 2013.

## Research Article

# Inferring Visual Perceptual Object by Adaptive Fusion of Image Salient Features

Xin Xu,<sup>1,2</sup> Nan Mu,<sup>1,2</sup> and Hong Zhang<sup>1,2</sup>

<sup>1</sup>*School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430081, China*

<sup>2</sup>*Hubei Province Key Laboratory of Intelligent Information Processing and Real-Time Industrial System, Wuhan University of Science and Technology, Wuhan 430081, China*

Correspondence should be addressed to Xin Xu; [xuxin0336@163.com](mailto:xuxin0336@163.com)

Received 19 September 2014; Revised 25 January 2015; Accepted 7 February 2015

Academic Editor: Marco Pérez-Cisneros

Copyright © 2015 Xin Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Saliency computational model with active environment perception can be useful for many applications including image retrieval, object recognition, and image segmentation. Previous work on bottom-up saliency computation typically relies on hand-crafted low-level image features. However, the adaptation of saliency computational model towards different kinds of scenes remains a challenge. For a low-level image feature, it can contribute greatly on some images but may be detrimental for saliency computation on other images. In this work, a novel data driven approach is proposed to adaptively select proper features for different kinds of images. This method exploits low-level features containing the most distinguishable salient information per image. Then the image saliency can be calculated based on the adaptive weight selection scheme. A large number of experiments are conducted on the MSRA database to compare the performance of the proposed method with the state-of-the-art saliency computational models.

## 1. Introduction

Saliency computational model with active environment perception can be useful for many applications including image retrieval, object recognition, and image segmentation. Generally, visual saliency can be defined as what captures human perceptual attention. Saliency detection plays an important role in image analysis and processing, which is capable of allocating the limited resources effectively. For example, the detecting of visual saliency can be effectively used to automatically zoom the “interesting” areas [1] or automatically crop the “important” areas in an image [2]. Object recognition algorithms can use the results of saliency detection to quickly locate the position of visual salient objects. Salient object detection can also reduce the interference of cluttered background to further improve the performance of image segmentation algorithm and image retrieval system [3].

Most of the existing saliency computational models are based on the bottom-up mechanism because the visual attention is generally driven by the low-level stimulus such as edge [4], color [5, 6], orientation [7], and symmetry [8].

These models typically contain two main procedures in saliency computation. The first step extracts low-level features from input image. Then the saliency map can be computed by fusing the extracted low-level features. In the past, low-level image features have been extensively studied. However, the selecting of proper features containing saliency information per-image is still complex and difficult to determine. The reason is mainly due to the lack of a well-defined feature which can exhaustively interpret saliency information in different images. Most of the existing saliency computational models face great difficulties in adaptively selecting low-level image features towards different images.

Aiming to address this problem, this paper puts forward an adaptive fusion scheme towards low-level image features for saliency detection. This method firstly extracts various low-level features to largely reflect saliency information of image. Then, the visual perceptual object can be detected by the adaptive weight selection towards these low-level image features. This method can retain the most significant low-level image features for saliency computation. The flowchart of the proposed method is illustrated in Figure 1.

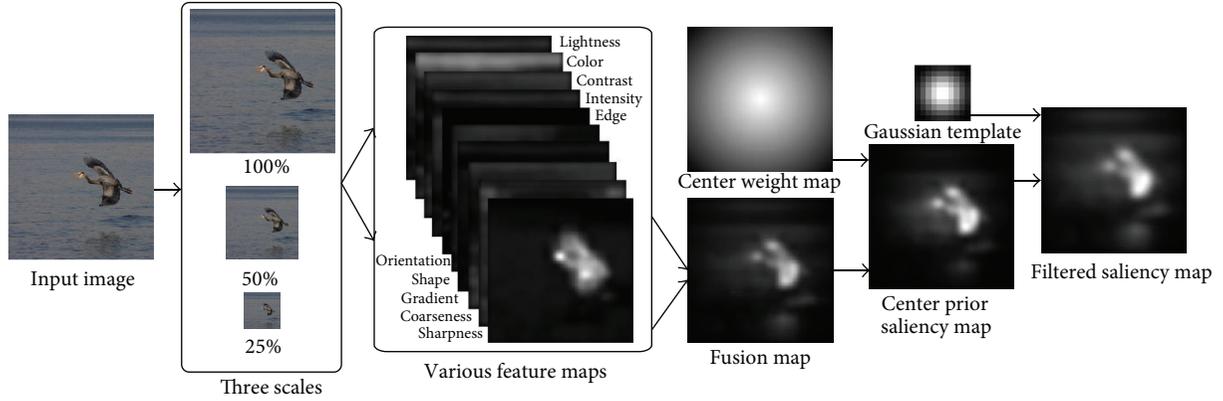


FIGURE 1: Flowchart of the proposed saliency computational model.

## 2. Related Works

Visual saliency reflects how much an image region or object stands out from its surrounding. Saliency computational model aims to provide a numerical measure indicating the degree of visual saliency, and is useful for a wide range of applications. Consequently, a number of computational approaches for salient object detection have been developed in recent years. Based on the biologically plausible visual attention architecture [9] and the feature integration theory [10], Itti et al. [11] proposed the bottom-up saliency model which depends only on low-level image features. The salient object in image can be determined using dynamic neural networks for feature fusion. Following this model, many state-of-the-art saliency computational models focus on low-level features, such as color and contrast [12, 13].

For example, Achanta et al. [14, 15] used the luminance and color features to detect salient object in image. This method calculated the contrast between local image region and its surrounding region and utilized the average color vector difference to obtain the saliency value of image. Aziz and Mertsching [8], Liu et al. [16], and Cheng et al. [17] proposed a contrast computational model based on region segmentation. These methods took use of the image segmentation algorithm to divide the image into different regions according to the homogeneity of different low-level image features such as luminance, texture, and color.

Most of the existing saliency computational models are based on low-level image features. Consequently, there are various formulations for exploiting well-defined salient features. Gao and Vasconcelos [18] studied the statistics of nature images and constructed an optimal detector based on the discriminate saliency. This method can effectively integrate the saliency features. Klein and Frinrop [19] detected the salient object by reconstituting the cognitive visual attention model. Their approach computed the saliency of all feature channels in an information-theoretic way. The optimal features can be determined by using Kullback-Leibler divergence (KLD) to fuse different features. Lu et al. [20] put forward the diffusion-based salient object detection model. This model can learn the optimal saliency seeds by combining two kinds

of features: the bottom-up saliency maps and midlevel vision cues.

However, most of the existing saliency computational models fail to consider the adaptability of different low-level features towards different images. Some low-level features which contribute greatly on some images may actually be detrimental to the detection of saliency on other images. In this paper, a novel data driven approach is proposed to adaptively select proper features for different images. This method exploits the most distinguishable low-level features containing salient information per image. Then the image saliency can be calculated based on the adaptive weight selection scheme.

The rest of this paper is organized as follows. Section 3 details the extraction of different low-level image features. Section 4 puts forward the adaptive feature fusion scheme. The experimental results are given in Section 5. And finally, the conclusions are drawn in Section 6.

## 3. Low-Level Image Feature Extraction

According to physiological experiments, it can be found that human attention towards image is mainly driven by low-level image features [21]. In order to fully and accurately describe the saliency information in image, ten low-level image features are chosen according to their visual properties. These ten low-level features are lightness, color, contrast, intensity, edge, orientation, shape, gradient, coarseness, and sharpness. In this section, we describe the extraction of these features and analyze the contributions of each feature in saliency computation.

The low-level features are extracted using the block-based method from input image (denoted by  $I$ ), which is divided into  $8 \times 8$  blocks (denoted by  $b$ ) with 50% overlap. Let  $F_n(I)$ ,  $n = 1, \dots, 10$ , denote the saliency map corresponding to each low-level feature; then  $F_n(b)$  can be used to denote the saliency map of block  $b$ .

**3.1. Lightness Feature Extraction.** Lightness is the perception attribute of human vision system for visible objects radiation

or glow amount. The lightness feature may affect the performance of other features to a certain extent.

The lightness feature measures how the brightness of each block is different from the average brightness of the image. Therefore, the object having a high brightness can be treated as a salient object by the human eye gaze. Let  $F_1(b)$  denote the Euclidean distance of the average lightness value between each image block  $b$  and the image  $I$ . The larger distance stands for the greater lightness value difference. Specific methods are shown as follows.

Firstly, the input image is converted from the RGB color space to a uniform LAB color space. Then, the lightness feature  $F_1(b)$  can be obtained by

$$F_1(b) = \left| \bar{L}(b) - \bar{L}(I) \right|, \quad (1)$$

where  $\bar{L}$  represents the average lightness component  $L$  in LAB color space. The lightness feature can well distinguish the illumination differences of object in image.

**3.2. Color Feature Extraction.** As the salient object has a strong contrast or a strong variation, the color value of the image object is relatively far from the average color value of the whole image. The background region can be seen as the smooth area in which the color value is closed to the average color value. Therefore, the color feature can be extracted by calculating the Euclidean distance of the average color value between each image block and the background region. The color feature  $F_2(b)$  of the image block  $b$  can be computed via

$$F_2(b) = \sqrt{(\bar{A}(b) - \bar{A}(I))^2 + (\bar{B}(b) - \bar{B}(I))^2}, \quad (2)$$

where  $\bar{A}$  and  $\bar{B}$  represent the average  $A$  and  $B$  color components in LAB color space, respectively.

Most of the existing saliency computational models are conducted in the RGB or LAB color feature space. RGB is mainly used for the color representation of the image, while CIELab provides a representation of color that corresponds to how observers perceive chromatic differences. Thus, this method extracts the color feature in the LAB color space.

Color feature can simply describe the global distribution of colors in an image. Thus, it can measure the proportion of different colors accounting for the whole image. The color feature is especially suitable for the image which does not take the spatial position of the object into account.

**3.3. Contrast Feature Extraction.** The saliency of the image block  $b$  in image  $I$  depends on the difference between the image block and its surrounding, while the visual characteristic of the image block itself does not determine its saliency. Thus, the greater the difference between the image block and its surrounding, the more likely it is a salient region. The calculation method to extract the contrast feature  $F_3(b)$  of the image block  $b$  is shown as follows.

First, convert each block to luminance image block via

$$l(b) = (i + jb)^\varepsilon, \quad (3)$$

where  $i = 0.7656$ ,  $j = 0.0364$ , and  $\varepsilon = 2.2$  denote the display conditions of the Adobe RGB color space. The contrast feature of the image block  $b$  can be expressed as

$$F_3(b) = \begin{cases} \frac{\sigma(b)}{\mu(b)}, & \mu(b) > 0, \\ 0, & \mu(b) = 0, \end{cases} \quad (4)$$

where  $\sigma(b)$  and  $\mu(b)$  represent the standard deviation and the average value of  $l(b)$ , respectively. The contrast is critically important on the visual attention. Generally, the greater the contrast is, the sharper the image can be.

**3.4. Intensity Feature Extraction.** The intensity feature (denoted by  $S(I)$ ) can also be treated as a lightness feature and can be obtained by averaging the three RGB color channels (denoted by  $R(I)$ ,  $G(I)$ , and  $B(I)$ ). Besides, the intensity feature can also reflect the brightness information and the color variations of image, which can represent human subjective feeling. The intensity feature  $F_4(b)$  of image block  $b$  can be generated by calculating the Euclidean distance between the intensity value of the image block  $S(b)$  and the whole image  $S(I)$ :

$$F_4(b) = \left| \bar{S}(b) - \bar{S}(I) \right|, \quad (5)$$

where  $\bar{S}(b)$  and  $\bar{S}(I)$  represent the average intensity values of  $S(b)$  and  $S(I)$ , respectively. The human eye perception towards the brightness is mainly associated with the luminous intensity of observed objects.

**3.5. Edge Feature Extraction.** The edge refers to a collection of image pixel in which the grayscale intensity has a strong contrast change. Thus, we use the edge feature to capture the image region with dramatic brightness variations. Let  $E(b)$  denote the binary map obtained through the Roberts edge detection. The edge feature  $F_5(b)$  of the image block  $b$  is then achieved by computing the average value of  $E(b)$  as

$$F_5(b) = \mu(E(b)). \quad (6)$$

The edge in image is often associated with the discontinuity of image grayscale.

**3.6. Orientation Feature Extraction.** Orientation feature is similar to the color feature, which is also an integrity feature. Based on the Itti model, the proposed method calculates the orientation feature by running the Gabor filter on grayscale image (denoted by  $g(x, y)$ ) of input image  $I$ . The Gabor wavelet can extract the direction feature effectively and can eliminate the redundant information at the same time. The two-dimensional Gabor function  $G(x, y)$  is shown as follows:

$$G_{\lambda, \theta, \varphi, \sigma, \gamma}(x, y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \varphi\right),$$

$$x' = x \cos \theta + y \sin \theta,$$

$$y' = -x \sin \theta + y \cos \theta, \quad (7)$$

where  $\gamma = 0.5$  represents the spatial aspect ratio and  $\sigma = 0.56\lambda$  represents the standard deviation of the Gaussian factor. The parameters  $\lambda$ ,  $\varphi$ , and  $\theta$  denote the wavelength, the phase offset, and the angle, respectively.

The 2D Gabor operator based orientation image feature extraction (denoted by  $O_\theta(I)$ ,  $\theta \in \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$ ) can be obtained using the following form of convolution:

$$O_\theta(I) = I(x, y) * G_\theta(x, y). \quad (8)$$

Let  $F_6(b)$  denote the orientation feature of image block  $b$ , which can be obtained by computing the Euclidean distance between the orientation image block  $O_\theta(b)$  and the orientation image  $O_\theta(I)$  via

$$F_6(b) = \sum_{\theta=\{0^\circ, 45^\circ, 90^\circ, 135^\circ\}} |\overline{O}_\theta(b) - \overline{O}_\theta(I)|, \quad (9)$$

where  $\overline{O}(b)$  and  $\overline{O}(I)$  represent the average orientation values of  $O(b)$  and  $O(I)$ , respectively. The orientation feature has global properties and thus can enable the saliency information with good feasibility and stability even in complex scenes.

**3.7. Shape Feature Extraction.** The shape feature can be extracted using Hu's moment invariants, which has the invariant properties of rotation, translation, and scale. The Hu moment invariants are commonly used to identify the large objects in an image and can better describe the shape feature of an object. The proposed method calculates the 7 moments (denoted by  $M_i$ ,  $i = 1, \dots, 7$ ) by Hu's method, which is obtained by normalizing the central moments through orders two and three.

Let  $M_i(b)$  and  $M_i(I)$ ,  $i = 1, \dots, 7$ , denote the 7 moments of the image block  $b$  and the input image  $I$ , respectively. Thus, the shape feature (denoted by  $F_7(b)$ ) of  $b$  can be expressed as the Euclidean distance between  $M_i(b)$  and  $M_i(I)$ :

$$F_7(b) = \sum_{i=\{1,2,\dots,7\}} |M_i(b) - M_i(I)|. \quad (10)$$

The shape feature is not sensitive to the lightness and the contrast changes and thus can effectively reduce the influence of lightness.

**3.8. Gradient Feature Extraction.** The gradient feature is sensitive to the gradient variation; however, it is not sensitive to the grayscale of the image. The image gradient can not only be able to capture the contour, silhouette, and some texture information, but also further weaken the influence of illumination. Let  $g(x, y)$  be the grayscale at pixel  $(x, y)$  in the image region, with the size  $M \times N$ ; the gradient feature (denoted by  $F_8(b)$ ) of image block  $b$  can be calculated by averaging the abscissa squared gradient (denoted by

$g_x(b)$ ) and the ordinate squared gradient (denoted by  $g_y(b)$ ) through

$$\begin{aligned} g_x(b) &= \sum_{x=0}^{M-2} \sum_{y=0}^{N-1} (g(x+1, y) - g(x, y))^2, \\ g_y(b) &= \sum_{x=0}^{M-1} \sum_{y=0}^{N-2} (g(x, y+1) - g(x, y))^2, \\ F_8(b) &= \mu \left( \frac{g_x(b) + g_y(b)}{2} \right). \end{aligned} \quad (11)$$

Gradient value can describe the magnitude of the dramatic changes of the pixel values. Thus, the gradient map constituted by the pixel gradient values can reflect the local grayscale changes in image.

**3.9. Coarseness Feature Extraction.** Coarseness is the fundamental perceptual texture feature. It can measure the particle size of the texture pattern. The larger particle size means the coarser image texture. The coarseness feature can be calculated as follows.

Firstly, the average gray value (denoted by  $A_k(x, y)$ ,  $i = 1, 2, \dots, 5$ ) of the neighborhood with size  $2^k \times 2^k$  in image  $I$  is calculated as

$$A_k(x, y) = \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y-2^{k-1}-1} \frac{g(x, y)}{2^{2k}}, \quad (12)$$

where  $g(x, y)$  is the gray value at pixel  $(x, y)$  in the active window.

Then, for each pixel, the average intensity difference (denoted by  $E_{k,h}(x, y)$  and  $E_{k,v}(x, y)$ ) can be calculated between the nonoverlapping neighborhoods in the horizontal and vertical directions, respectively:

$$\begin{aligned} E_{k,h}(x, y) &= |A_k(x + 2^{k-1}, y) - A_k(x - 2^{k-1}, y)|, \\ E_{k,v}(x, y) &= |A_k(x, y + 2^{k-1}) - A_k(x, y - 2^{k-1})|. \end{aligned} \quad (13)$$

Finally, the optimal size can be set by  $k$  which gives the highest value of  $E$ . The coarseness (denoted by  $C$ ) is the average of  $S_{\text{best}}(x, y) = 2^k$ :

$$C = \frac{1}{m \times n} \sum_{x=1}^m \sum_{y=1}^n S_{\text{best}}(x, y). \quad (14)$$

Let  $C(b)$  and  $C(I)$  denote the coarseness of the image block  $b$  and image  $I$ , respectively. Thus, the coarseness feature (denoted by  $F_9(I)$ ) of  $b$  can be expressed as the Euclidean distance between  $C(b)$  and  $C(I)$ :

$$F_9(b) = |C(b) - C(I)|. \quad (15)$$

Coarseness feature represents the surface properties of the whole image and can well describe the integrity of the salient object. Meanwhile, the coarseness feature has good rotation invariance; it can effectively resist the interference of noise.

**3.10. Sharpness Feature Extraction.** The sharpness feature measures how the acutance of each region is different from its surrounding, which can indicate the contrast of the adjacent region. The proposed method extracts the sharpness value  $\phi(p)$  at position  $p$  by computing the convolution between the input image  $I$  and the first-order derivatives of the Gaussian via

$$\phi(p) = \sum_x \sum_y [g(x, y) * G_x^\sigma(x, y)]^2 + [g(x, y) * G_y^\sigma(x, y)]^2, \quad (16)$$

where  $g(x, y)$  is the grayscale at pixel  $(x, y)$  in the image region.  $G_x^\sigma(x, y)$  and  $G_y^\sigma(x, y)$  represent the first-order derivatives of the Gaussian in the vertical and the horizontal directions, respectively.  $\sigma$  is the scale of the Gaussian filter.

The Gaussian derivative method can measure the acutance variation and suppress the influence of noise and illumination. Let  $\bar{\phi}(b)$  and  $\bar{\phi}(I)$  denote the average sharpness value of the image block  $b$  and the input image  $I$ , respectively. Thus, the sharpness feature (denoted by  $F_{10}(b)$ ) of  $b$  can be expressed as the Euclidean distance between  $\bar{\phi}(b)$  and  $\bar{\phi}(I)$ :

$$F_{10}(b) = |\bar{\phi}(b) - \bar{\phi}(I)|. \quad (17)$$

Sharpness represents the image definition and the edge acuteness. The higher the sharpness is, the higher the image contrast can be. The sharpness feature can be less susceptible to local variations.

#### 4. Adaptive Fusion of Low-Level Image Features

According to the discrete degree and clarity of the saliency map of each low-level feature, different weights are assigned to different features. Thus, low-level features that contribute greatly on some images will assign a large weight; other low-level features that may be detrimental to the detection of saliency will assign a low weight or completely ignore during the saliency computation. Let  $\nu_n$  denote the statistical validity and  $\omega_n$  denote the weights of different feature maps  $F_n(I)$ ,  $n = 1, 2, \dots, 10$ . The  $\nu_n$  can be defined as

$$\nu_n = \sigma n^2 + \kappa n, \quad (18)$$

where  $\sigma n^2$  and  $\kappa n$  represent the variance and the kurtosis of  $F_n(I)$ , respectively.

The weights  $\omega_n$  of different feature maps  $F_n(I)$  are determined by the numerical magnitudes of the statistical validity  $\nu_n$  via

$$\omega_n = \begin{cases} 1, & \text{if } \nu_n = \nu_1^* \\ \frac{4}{5}, & \text{if } \nu_n = \nu_2^* \\ \frac{3}{5}, & \text{if } \nu_n = \nu_3^* \\ \frac{2}{5}, & \text{if } \nu_n = \nu_4^* \\ \frac{1}{5}, & \text{if } \nu_n = \nu_5^* \\ 0, & \text{otherwise,} \end{cases} \quad (19)$$

where  $\nu_i^* = \text{sort}\{\nu_i\}$  using the descending order. The proposed method assigns different weights according to numerical sort of  $\nu_n$ .

The final fusion map (denoted by  $F_I$ ) is calculated by the weighted sum of the ten feature maps  $F_n(I)$ ,  $n = 1, 2, \dots, 10$ :

$$F_I = \frac{\sum_n \omega_n F_n(I)}{\sum_n \omega_n}. \quad (20)$$

To enhance the robustness of detection process and achieve a preferable visual effect, the proposed method is performed in three scales {100%, 50%, 25%} to better restrain the background information.

The obtained saliency map is then refined using center prior principle to enhance visual effect. When humans watch a picture, they will naturally gaze on the objects next to the center of image [22]. Thus, in order to obtain the saliency objects closer to the human visual fixations, more weight is needed to add to the center of image. Therefore, a feature (denoted by  $f_c(b)$ ) is included to indicate the distance between each image block and the center of image. For each image block feature  $F_n(b)$ ,  $n = 1, \dots, 10$ , it can be recalculated via

$$F_n^*(b) = F_n(b) f_c(b), \quad (21)$$

$$f_c(b) = 1 - \frac{\sqrt{(r(b) - C^*)^2 + (c(b) - R^*)^2}}{\sqrt{(M/2)^2 + (N/2)^2}},$$

where  $(r(b), c(b))$  denote the upper-left coordinate of image block  $b$ ,  $(C^*, R^*)$  represent the center coordinate of the estimated image region, and  $M$  and  $N$  denote the width and height of the image region, respectively.

The feature map  $F_n(I)$  of the whole image  $I$  can be generated by combining the image feature  $F_n^*(b)$  of all the blocks. The normalized feature map is calculated by

$$F_n(I) = \frac{F_n(I) - \min(F_n(I))}{\max(F_n(I)) - \min(F_n(I))}. \quad (22)$$

Finally, the generated saliency map is smoothed by a Gaussian filter (the template size is  $10 \times 10$ , and  $\sigma$  is 2.5).

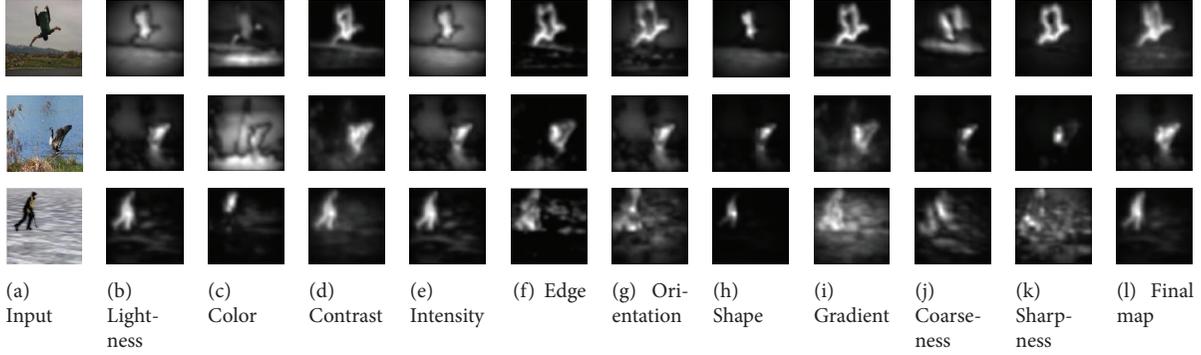


FIGURE 2: Saliency maps of each low-level features. (a) Input image, ((b)–(k)) corresponding saliency maps of low-level image features respectively, (l) the final saliency map.

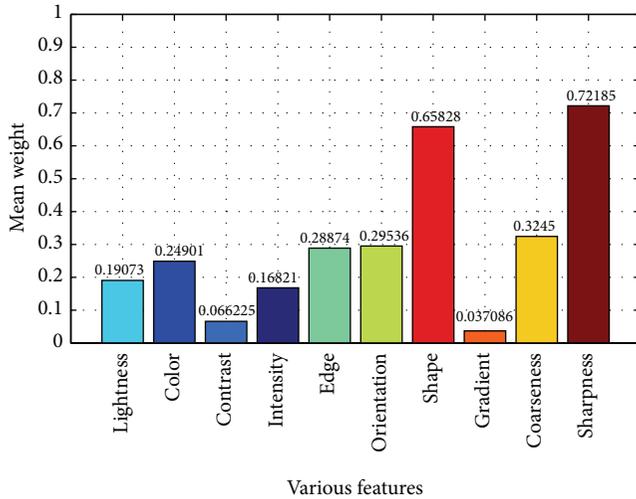


FIGURE 3: The average weight distribution of the ten low-level image features.

The saliency maps of each low-level feature and the final saliency maps are shown in Figure 2.

As illustrated in Figure 2, the ten low-level image features have their own advantages and disadvantages towards different images. On the contrary, the final saliency map can adaptively fuse the optimal features to achieve better performance.

In addition, we also provide the statistical of the ten low-level image features in different images. The result from different images is shown in Figure 3. As illustrated in Figure 3, the sharpness feature and the shape feature have good stability in our testing. Their weights have little change due to the deviation caused by local contrast transformation. On the contrary, the gradient feature can only reflect the local differences.

## 5. Experimental Results

The performance evaluation is conducted using the MSRA salient object database [16]. This database contains over

20 000 images, which includes two parts: (i) image set A, containing 20,000 images, and the principle salient objects are labeled by three users, and (ii) image set B, containing 5,000 images, and the principle salient objects are labeled by nine users. The proposed multifeature fusion (MF) approach is compared with the other seven state-of-the-art methods: *Itti's* (IT) method [11], *spectral residual* (SR) method [23], *saliency using natural statistics* (SUN) method [24], *frequency-tuned* (FT) method [25], *S3* method [26], *nonparametric* (NP) method [27], and *context-aware* (CA) method [28].

Figure 4 illustrates the performance comparison of these various salient region detection methods. As can be seen from Figure 4, the saliency maps extracted using our proposed method are more consistent with the ground-truth rectangle, and the detected saliency objects are more similar to the ground-truth binary masks. The approaches developed in [23, 24, 26] fail to detect and clearly identify the location of salient object from complex background. The saliency maps generated using methods [11, 27] look rather blurry and are difficult to clearly distinguish the salient region. The saliency map of method [25] retains a lot of background information. The CA method [28] can achieve good detection performance on some images; however this method is unable to highlight the entire salient object itself.

The objective assessment is implemented by computing the *true positive rate* (TPR) and the *false positive rate* (FPR). Given the ground-truth binary masks  $G(x, y)$  and the obtained saliency map  $F_t(x, y)$  ( $0 \leq F_t(x, y) \leq 1$ ), a threshold  $t$  ( $0 \leq t \leq 1$ ) is used to obtain the binary masks  $B_t(x, y)$ , in which 0 denotes the background and 1 denotes the salient objects. The TPR and FPR can be computed via

$$\begin{aligned} \text{TPR} &= E \left( \prod_t B_t(x, y) \cdot G(x, y) \right), \\ \text{FPR} &= E \left( \prod_t (1 - B_t(x, y)) \cdot G(x, y) \right). \end{aligned} \quad (23)$$

Figure 5 shows the TPR and FPR results of the seven methods and the proposed method. As seen in Figure 5, the overall performance of the proposed method is more excellent than the other seven methods.



FIGURE 4: Saliency maps obtained from the proposed method (MF) and state-of-the-art saliency computational models.

TABLE 1: The precision, recall, and  $F$ -measure of the saliency computational models.

	IT [11]	SR [23]	SUN [24]	FT [25]	S3 [26]	NP [27]	CA [28]	MF
Precision	0.3548	0.5662	0.2769	0.6845	0.2159	0.2132	0.4894	0.7155
Recall	0.9477	0.4177	0.7491	0.6519	0.6991	0.9522	0.8770	0.8150
$F$ -measure	0.4391	0.4951	0.3291	0.6443	0.2715	0.2781	0.5580	0.7370

TABLE 2: Average execution time (in seconds) for the saliency computational models.

	IT [11]	SR [23]	SUN [24]	FT [25]	S3 [26]	NP [27]	CA [28]	MF
Run time	1.0047	0.1810	1.7783	0.1710	8.4211	1.4895	68.3333	10.3720

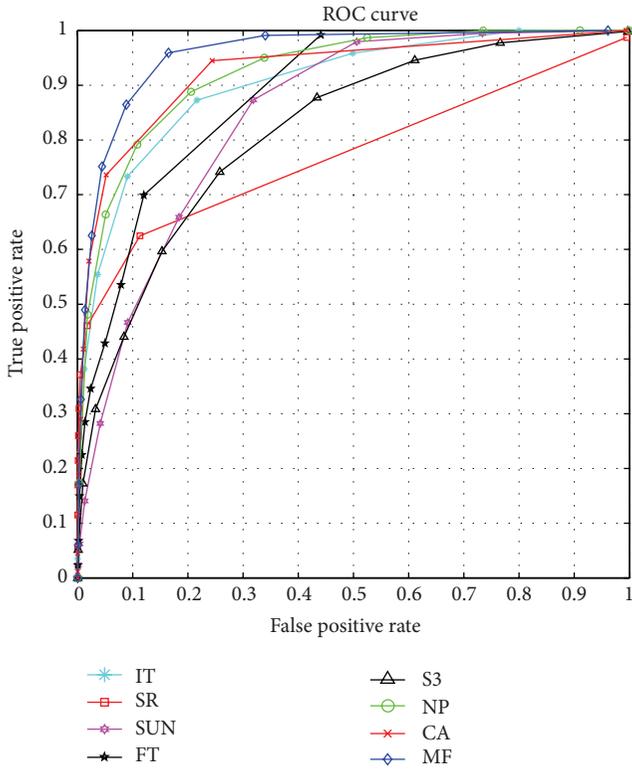


FIGURE 5: ROC curves obtained from the proposed method and state-of-the-art saliency computational models.

Given the generated saliency map  $F_I(x, y)$ , we set a threshold  $T$  (computed by Otsu's method) to segment the saliency objects. The binary mask is denoted by  $B_I(x, y)$ .

Let  $G$  and  $B_I$  denote the ground-truth binary masks and binary masks of the proposed approach, respectively. The Precision =  $R(B_I \cap G)/R(B_I)$  and Recall =  $R(B_I \cap G)/R(G)$ , where  $R(\cdot)$  represent the salient region. The evaluation criterion  $F$ -measure can be computed via:

$$F_\beta = \frac{(1 + \beta^2) \text{Precision} \times \text{Recall}}{\beta^2 \times \text{Precision} + \text{Recall}}. \quad (24)$$

The proposed method uses  $\beta^2 = 0.5$  to weigh the precision and recall. The precision, recall, and  $F$ -measure of these methods are shown in Table 1.

Finally, we compare the computational complexity of the different saliency computational models discussed. These models are implemented using the MATLAB programming language and run on a PC with a Pentium G2020 CPU and a 4 GB RAM. Table 2 shows the results of the proposed method and the other methods. The proposed method yields slightly higher computational load than the conventional approach; however, the proposed method can achieve more accurate saliency detection in various images.

## 6. Conclusion

In this paper, a novel feature selection scheme is proposed to adaptively select the proper features for different images. This method exploits the most distinguishable salient information in ten low-level features per image. The generated saliency map can highlight the salient object in different images even containing complex background. A large number of experiments are conducted on the MSRA database to compare the performance of the proposed method with the state-of-the-art saliency computational models. And the experimental results indicate that the proposed method outperforms the state-of-the-art saliency computational models to achieve better performance.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61440016 and 61373109), the Natural Science Foundation of Hubei Provincial of China (no. 2014CFB247), and the Project of the Key Laboratory for Metallurgical Equipment and Control of Ministry of Education (no. 2013B08).

## References

- [1] L.-Q. Chen, X. Xie, X. Fan, W.-Y. Ma, H.-J. Zhang, and H.-Q. Zhou, "A visual attention model for adapting images on small displays," *Multimedia Systems*, vol. 9, no. 4, pp. 353–364, 2003.

- [2] F. Stentiford, "Attention based auto image cropping," in *Proceedings of the 5th International Conference on Computer Vision Systems*, Bielefeld, Germany, March 2007.
- [3] V. Gopalakrishnan, Y. Hu, and D. Rajan, "Random walks on graphs to model saliency in images," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR '09)*, pp. 1698–1705, June 2009.
- [4] Q. Deng and Y. Luo, "Edge-based method for detecting salient objects," *Optical Engineering*, vol. 50, no. 5, Article ID 057007, 2011.
- [5] K. Fu, C. Gong, J. Yang, Y. Zhou, and I. Yu-Hua Gu, "Superpixel based color contrast and color distribution driven salient object detection," *Signal Processing: Image Communication*, vol. 28, no. 10, pp. 1448–1463, 2013.
- [6] J. Kim, D. Han, Y.-W. Tai, and J. Kim, "Salient region detection via high-dimensional color transform," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 883–890, Columbus, Ohio, USA, June 2014.
- [7] V. Gopalakrishnan, Y. Hu, and D. Rajan, "Salient region detection by modeling distributions of color and orientation," *IEEE Transactions on Multimedia*, vol. 11, no. 5, pp. 892–905, 2009.
- [8] M. Z. Aziz and B. Mertsching, "Fast and robust generation of feature maps for region-based visual attention," *IEEE Transactions on Image Processing*, vol. 17, no. 5, pp. 633–644, 2008.
- [9] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," *Human Neurobiology*, vol. 4, no. 4, pp. 219–227, 1985.
- [10] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97–136, 1980.
- [11] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.
- [12] Y. Lu, W. Zhang, H. Lu, and X. Xue, "Salient object detection using concavity context," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 233–240, November 2011.
- [13] K. Shi, K. Wang, J. Lu, and L. Lin, "PISA: pixelwise image saliency by aggregating complementary appearance contrast measures with spatial priors," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 2115–2122, June 2013.
- [14] R. Achanta, F. Estrada, P. Wils, and S. Susstrunk, "Salient region detection and segmentation," in *Computer Vision Systems: 6th International Conference, ICVS 2008 Santorini, Greece, May 12–15, 2008 Proceedings*, vol. 5008 of *Lecture Notes in Computer Science*, pp. 66–75, Springer, Berlin, Germany, 2008.
- [15] R. Achanta and S. Susstrunk, "Saliency detection using maximum symmetric surround," in *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP '10)*, pp. 2653–2656, September 2010.
- [16] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2007.
- [17] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 409–416, June 2011.
- [18] D. Gao and N. Vasconcelos, "Bottom-up saliency is a discriminant process," in *Proceedings of the IEEE 11th International Conference on Computer Vision (ICCV '07)*, pp. 1–6, Rio de Janeiro, Brazil, October 2007.
- [19] D. A. Klein and S. Frintrop, "Center-surround divergence of feature statistics for salient object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 2214–2219, November 2011.
- [20] S. Lu, V. Mahadevan, and N. Vasconcelos, "Learning optimal seeds for diffusion-based salient object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 2790–2797, IEEE, Columbus, Ohio, USA, June 2014.
- [21] C. T. Vu and D. M. Chandler, "Main subject detection via adaptive feature selection," in *Proceedings of the 16th IEEE International Conference on Image Processing (ICIP '09)*, pp. 3101–3104, Cairo, Egypt, November 2009.
- [22] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *Proceedings of the 12th International Conference on Computer Vision (ICCV '09)*, pp. 2106–2113, September–October 2009.
- [23] X. Hou and L. Zhang, "Saliency detection: a spectral residual approach," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pp. 1–8, June 2007.
- [24] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, "SUN: a Bayesian framework for saliency using natural statistics," *Journal of Vision*, vol. 8, no. 7, article 32, 2008.
- [25] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pp. 1597–1604, Miami, Fla, USA, June 2009.
- [26] C. T. Vu, T. D. Phan, and D. M. Chandler, "S<sub>3</sub>: a spectral and spatial measure of local perceived sharpness in natural images," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 934–945, 2012.
- [27] N. Murray, M. Vanrell, X. Otazu, and C. A. Parraga, "Saliency estimation using a non-parametric low-level vision model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 433–440, June 2011.
- [28] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 1915–1926, 2012.

## Research Article

# Multiframe Superresolution Reconstruction Based on Self-Learning Method

Shao-Shuo Mu,<sup>1,2</sup> Ye Zhang,<sup>1</sup> Ping Jia,<sup>1</sup> Xun Yang,<sup>1</sup> and Xiao-Feng Qiu<sup>3</sup>

<sup>1</sup>Key Laboratory of Airborne Optical Imaging and Measurement, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China

<sup>2</sup>Graduate University of Chinese Academy of Sciences, Beijing 100039, China

<sup>3</sup>Northeast Normal University, Changchun 130033, China

Correspondence should be addressed to Shao-Shuo Mu; 860600894@qq.com

Received 28 August 2014; Revised 11 November 2014; Accepted 11 November 2014

Academic Editor: Daniel Zaldivar

Copyright © 2015 Shao-Shuo Mu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

One category of the superresolution algorithms widely used in practical applications is dictionary-based superresolution algorithms, which constructs a single high-resolution (HR) and high-clarity image from multiple low-resolution (LR) images. Despite the fact that general dictionary-based superresolution algorithms obtain redundant dictionaries from numerous HR-LR images, HR image distortion is unavoidable. To solve this problem, this paper proposes a multiframe superresolution reconstruction based on self-learning methods. First, multiple images from the same scene are selected to be both input and training images, and larger-scale images, which are also involved in the training set, are constructed from the learning dictionary. Then, different larger-scale images are constructed via repetition of the first step and the initial HR sets whose scale closely approximates that of the target HR image are finally obtained. Lastly, initial HR images are fused into one target HR image under the NLM idea, while the IBP idea is adopted to meet the global constraint. The simulation results demonstrate that the proposed algorithm produces more accurate reconstructions than those produced by other general superresolution algorithms, while, in real scene experiments, the proposed algorithm can run well and create clearer HR images from input images captured by cameras.

## 1. Introduction

In modern engineering, high-resolution (HR) images are always desirable which provides important information for making important decisions in various practical applications, for example, biometrics, airborne detection system, medical diagnosis, high definition television (HDTV), and remote surveillance. One way to obtain HR images is to increase sensor pixel resolution, which, however, leads to increased cost in sensors. A promising alternative is the superresolution technique, which reconstructs one HR image from one or more low-resolution (LR) images shot from the same scene, providing sufficient relevant information.

In recent years, superresolution techniques have been extensively studied and put into practice. CEVA Company firstly combined superresolution techniques with CEVA-MM3101 low-power imaging and vision platform successfully, which used LR sensors to construct HR image, and have been

applied to camera-enabled mobile devices. French company SPOT applied superresolution techniques to SPOT-5 satellite which can obtain clearer ground scene images and identify targets more accurately, compared with SPOT-4 which did not adopt superresolution techniques. So superresolution technique has been a hot spot to improve images' resolution without changing the optical system of the camera.

Superresolution techniques have three categories in general, namely, interpolation-based, reconstruction-based, and dictionary-based algorithms. Interpolation-based algorithms are simple but ineffective, in which the unknown image pixel values are calculated from the known pixels using interpolation formula [1–4], such as bilinear, bicubic, and Lanczos interpolations. Reconstruction-based algorithms have process inverse to imaging system solving. These algorithms use prior image knowledge and recover high-frequency information via established imaging models [5–10].

Most recently, dictionary-based algorithms have been more widely applied due to the development in compressed sensing [11–15]. This category of algorithms requires a number of external training images, using similarity redundancy structure of natural images to restore missing high-frequency information in input images. Roweis and Saul [16] proposed a nonlinear dimension reduction method, namely, locally linear embedding (LLE), which has been widely applied to image processing. Based on [16], Chang et al. [17] applied LLE technology to superresolution reconstruction. In this method, the single input LR image is divided into blocks with image shift; then, each image block searches for  $k$  nearest blocks in all LR training image blocks whereby the optimal weights are obtained; finally, a HR image is constructed from a linear combination of the  $k$  HR training blocks corresponding to the  $k$  LR training blocks. References [18–20] improved the algorithm in [17] and achieved enhanced reconstruction results. Yang et al. [21] proposed a sparse coding method for dictionary learning. Compared with previous algorithms, their algorithm relies on a larger number of databases, while obtaining missing high-frequency information more effectively. In [22], their sparse coding is improved to strengthen the sparse similarity between LR and HR image blocks. This method is simpler and more effective compared with [21]. He et al. [23] adopted a beta process dictionary learning approach to make the HR and LR dictionaries more accurate and consistent, gaining better results in comparison with [22]. Jeong et al. [24] proposed a superresolution method robust to noise, which involves two phases: the learning phase and the reconstruction phase. In the learning phase, the training images are classified and different dictionaries are constructed according to noise quantity; and, in the reconstruction phase, the input images with various noise quantities adopt appropriate dictionaries to reconstruct HR images.

All the above reconstruction methods require a large number of external training images. When external images cannot be accessed, it is feasible to start from the point of similarity redundancy structure of input images. In [25–28], different self-learning methods are proposed. Glasner et al. [25] realized superresolution reconstruction without external images by adopting the gradual magnification scheme. Chen et al. [26] used high-frequency and low-frequency components of input images to train dictionary image blocks. Then, they used the Approximate Nearest Neighbor Search to reconstruct images and recovered the missing high-frequency information. Zhang et al. [27] proposed a self-similarity redundant algorithm, which selects original LR images and corresponding degraded images as the training images and uses LLE algorithm to obtain one HR image. In [28], Ramakanth and Babu proposed single image learning dictionary and adopted the approximate nearest neighbor fields (ANNF) method to find similarities between input LR blocks and LR dictionary. This method reconstructs HR images more effectively and more efficiently compared with other dictionary-based methods.

Other studies [25–28] have shown that natural images have self-similarity information on different scales and high-frequency information in small-scale images can be obtained

from large-scale images. So this paper proposes a multiple-frame superresolution algorithm based on self-learning, and this algorithm requires no external training image. First, self-similarity information of multiple input images is used to create larger-scale training images through a step by step process. Then, an over-complete dictionary is studied by sparse representations. Finally, the reconstructed images are further improved by exerting the global constraint. Compared with [25–27], the proposed algorithm possesses several unique features as follows.

- (1) Larger-scale images are created from input images. Both input images and larger-scale images are involved in the training set. Sparse representation is adopted to learn the mapping relation of the dictionary.
- (2) One larger-scale image is reconstructed from one input image; apply this reconstruction to all  $n$  input images. Through gradual magnification,  $n$  initial estimates are obtained, each corresponding to one of the input images, with a scale closely approximating that of the target HR image. Then, the NLM method is employed to fuse all estimates into one target HR image.
- (3) In order to restore more high-frequency parts, degraded images of input images are used as LR training set and high-frequency parts of input images are used as HR training set.

Our theoretical analysis demonstrates that the proposed algorithm is feasible, and the simulation results show that the proposed algorithm performs better than general dictionary-based superresolution algorithms. In real scene experiments carried out on a camera system, the proposed algorithm obtains clearer real HR images.

This paper is organized as follows. Section 2 gives a discussion on image sparse models. Section 3 presents the framework of our proposed algorithm as well as its implementation, followed by experimental results in Section 4. Finally, Section 5 gives a conclusion of this paper.

## 2. Image Sparse Model

Superresolution reconstruction is a reverse evaluation problem. Dictionary-based superresolution reconstruction trains the corresponding LR dictionary and HR dictionary by mapping the relationship between LR images and HR images. Then, the relationship between LR dictionary and HR dictionary is solved, and a HR image is reconstructed with recovery of the missing high-frequency information. Imaging model between LR image and HR image is as shown in (1): when the real scene is captured through the optical system, we obtain a group of LR image scenes due to the impact of optical blur and downsampling. Consider

$$Y = SHX. \quad (1)$$

$Y$  is a LR image captured by the camera.  $X$  is the real scene image.  $H$  is the optical blur matrix.  $S$  is the downsampling matrix.

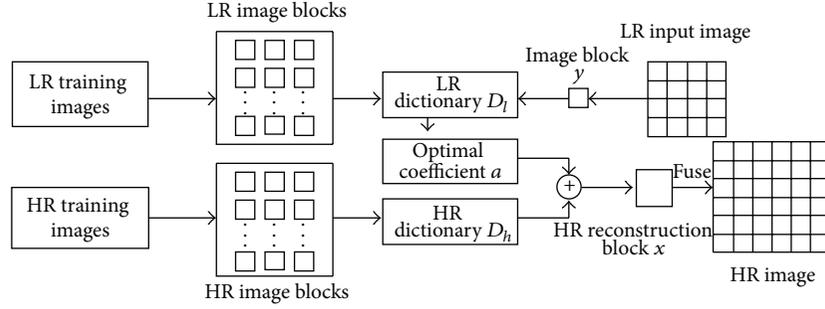


FIGURE 1: Simple diagram of general dictionary-based reconstruction algorithm.

Superresolution reconstruction is a highly ill-condition problem. Given one LR input image  $Y$ , many HR images  $X$  are suitable for formula (1). With the development of compressed sensing, dictionary-based superresolution methods have been applied more widely in image restoration areas. We can divide image into blocks  $y$  of the same dimension, every image block is represented by both sparse representation “ $a$ ” and sparse matrix  $D$ , such as  $y = D \times a$ , where  $D$  is a dictionary matrix and “ $a$ ” is sparse coefficients. The basic process of dictionary-based reconstruction is shown in Figure 1.

First, we require lots of training images and train the corresponding LR dictionary and HR dictionary by learning LR training images and HR training images. Second, the optimal sparse coefficients “ $a$ ” of LR input image block are solved according to the equation  $\min(y - D_l \times a)$ . Then, we get the corresponding HR image block  $x$  via solving the formula  $D_h \times a$  and fuse all HR image blocks into one HR image  $X$ .

### 3. The Proposed Method

When one LR image is given to reconstruct one HR image, the main problem of dictionary-based methods is how to find lots of HR-LR training images and utilize self-similarity information of external images. But it will cause the reconstructed HR image distortion, given the absence of the input images’ frequency information in the selected HR-LR training set. In this paper, we propose a multiple superresolution reconstruction method based on self-learning dictionary. We collect multiple images of the same scene as both input images and training image set, and larger-scale images, which are constructed from last training set, are also involved in the training set constantly. After that, we get the initial reconstruction images whose scale closely approximates target HR image. Then, we use NLM method to fuse the initial reconstruction images into one HR image. Lastly, IBP idea is adopted to satisfy the global reconstruction constraint. Here, four major parts are detailed in the following. In Section 3.1, we mainly discuss how to use input images to gradually magnify image and create a large number of different larger-scale images as training set. In Section 3.2, sparse representation is adopted to jointly learn redundant dictionary. In Section 3.3, we employ NLM method to fuse initial reconstruction images

into one HR image and introduce the importance of global reconstruction constraint. In Section 3.4, the working process of the proposed algorithm is introduced. In Section 3.5, we discuss how to simplify the operation.

**3.1. Self-Learning Method.** As we have said above, dictionary-based reconstruction algorithms need lots of training images to learn the overcomplete dictionary for constructing HR image. Literatures [25–28] point out that input image has enough self-similarity information and the high-frequency information of small-scale image can be obtained from large-scale image. They can use multiframe input images to reconstruct images accurately. Therefore, when external images cannot be accessed, this paper proposes a self-learning pyramid method which constructs larger-scale images gradually to join in training set. And it is feasible. We can fully learn all the redundant information from different scale images. First, assuming that the target HR image resolution is  $P$  times that of the input LR image resolution, we will enlarge input images resolution  $p_i$  times step by step, as the following formula:

$$\begin{aligned} p_1 \times p_2 \times \cdots \times p_n \times s &= P, \\ s < p_i, \quad i &= 1, \dots, n, \\ p_1 \times p_2 \times \cdots \times p_n &\leq P. \end{aligned} \quad (2)$$

Assuming that the input images are  $N \times N$  dimension, there are two cases for the principle of  $p$  as follows:

- (a)  $p_1 = p_2 = \cdots = p_n = f/g$ ;  $g^n \times w = N$  ( $w < g$  and  $f, g$  all are integers.);
- (b)  $p_1 = f_1/g_1, \dots, p_n = f_n/g_n$ ;  $f_i/g_j > 1$  ( $f_i, g_j$  all are integers,  $s > p_i$ , and  $i = 1, 2, \dots, n$ ).

We discuss this paper based on condition (a).

Then, the degraded images  $Y_1^\wedge$  are solved via the formula  $Y_1 \downarrow_p \uparrow_p$  ( $\downarrow_p$  is  $p$  times downsampling;  $\uparrow_p$  is  $p$  times upsampling), which together constitute the HR-LR training set  $Q_1$  with high-frequency part of input images  $Y_1$ , as the formula (3). (The degraded images  $Y_1^\wedge$  are LR training set. High-frequency part ( $Y_1 - Y_1^\wedge$ ) of input images  $Y_1$  is HR training set.) This is the difference between this paper and [25–27], and our method can recover the edge high-frequency better. We train  $Q_1$  to get dictionary  $D_1$  and construct the corresponding

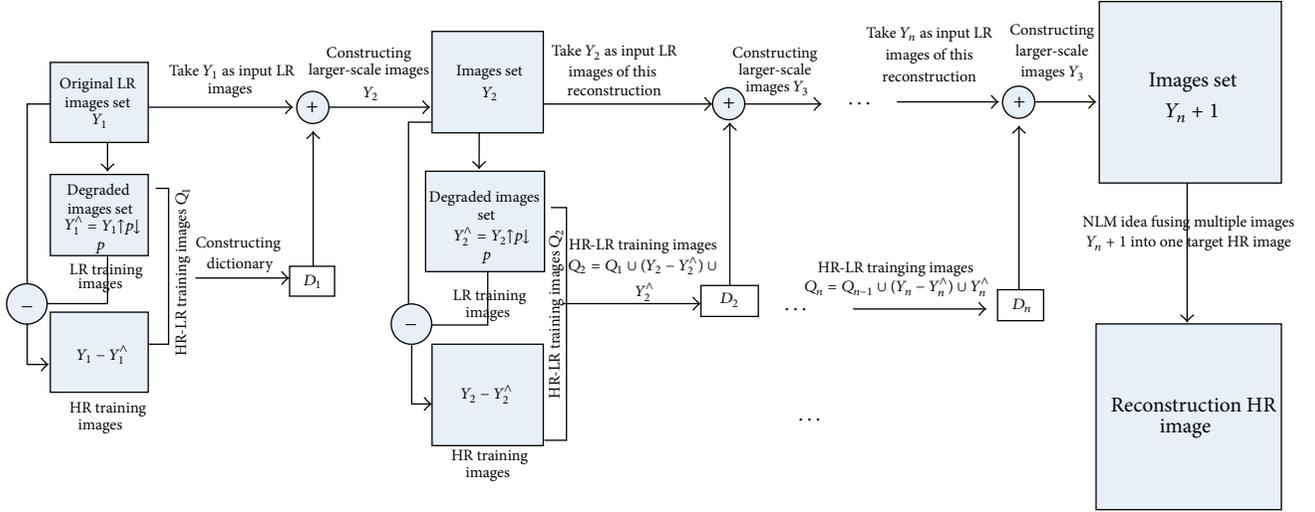


FIGURE 2: Process of training images.

larger-scale HR images  $Y_2$  whose resolution is  $p$  times that of the input images. Then, degraded images  $Y_2^\wedge$  are solved, which together constitute new HR-LR training set  $Q_2$  with high-frequency part of  $Y_2$  and  $Q_1$  again. It is as shown in (3) that we reconstruct final HR images whose resolution is closest to target HR images, followed by  $n$  repeated learning. Process of training images is as shown in Figure 2. We enlarge every input image step by step just like pyramid. Consider

$$\begin{aligned}
 Q_1 &= Y_1^\wedge \cup (Y_1 - Y_1^\wedge), \\
 Q_2 &= Y_2^\wedge \cup (Y_2 - Y_2^\wedge) \cup Q_1, \\
 &\vdots \\
 Q_m &= Y_m^\wedge \cup (Y_m - Y_m^\wedge) \cup Q_{m-1}.
 \end{aligned} \tag{3}$$

**3.2. Learning Dictionary.** In the previous section, we have discussed the self-learning pyramid method. Chen et al. [26] and Zhang et al. [27] use image blocks manifold to learn dictionary. They solve the optimal weight between input image block and several nearest local training blocks. But it cannot effectively utilize global image information. Yang et al. [22] point out that sparse representation can utilize global image information better and construct universal dictionary with a better reconstructed image. So we propose jointly learning dictionary method based on sparse representation in this section. As shown in Section 3.1 that  $Q_m$  is the training set, we use images  $Y_m$  of  $Q_m$  as input images of next reconstruction.  $x$  and  $y$  are HR training blocks and LR training blocks respectively, which are represented by atomic linear combination of matrix  $D$ :  $x = D_h \times \alpha$ ;  $y = D_l \times \beta$  ( $\alpha$  and  $\beta$  are sparse coefficients.  $D_h$  and  $D_l$  are HR dictionary and LR dictionary, resp.). In this section, the premise of learning dictionary is that sparse coefficients  $\alpha$  and  $\beta$  are the same. It is as shown in (4). Consider

$$\begin{aligned}
 y &= D_l \times \alpha, \\
 x &= D_h \times \alpha.
 \end{aligned} \tag{4}$$

By solving the sparse coefficients  $\alpha$  of LR dictionary  $D_l$ , we can get HR image blocks according to  $x = D_h \times \alpha$ . This is the basic idea of training dictionary. Before reconstructing HR image, we must learn dictionaries  $D_l$  and  $D_h$  which must satisfy constraints of (4) firstly. So we can convert (4) into the following optimization problem, as shown in (5). Consider

$$\begin{aligned}
 \min \|\alpha\|_0, \quad \text{s.t. } \|y - D_l \times \alpha\|_2^2 &\leq \varepsilon, \\
 \min \|\alpha\|_0, \quad \text{s.t. } \|y - D_h \times \alpha\|_2^2 &\leq \varepsilon.
 \end{aligned} \tag{5}$$

As long as the sparse coefficients  $\alpha$  are sparse enough, (5) is solved by solving  $L^1$  norm problem, as shown in (6) and (7). Consider

$$\operatorname{argmin} \frac{1}{2} \|x - D_h \times \alpha\|_2^2 + \mu \|\alpha\|_1, \tag{6}$$

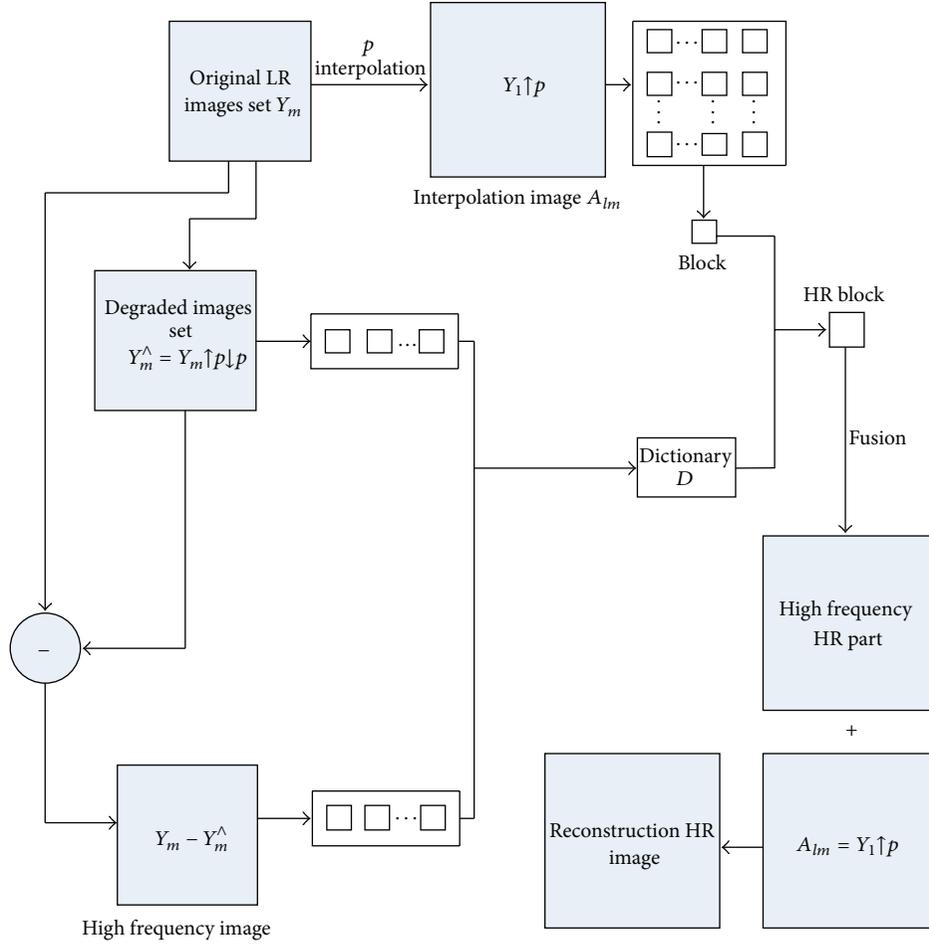
$$\operatorname{argmin} \frac{1}{2} \|y - D_l \times \alpha\|_2^2 + \mu \|\alpha\|_1. \tag{7}$$

In order to get HR dictionary and LR dictionary of the same coefficient  $\alpha$ , (6) and (7) can be converted into (8). Consider

$$\operatorname{argmin} \frac{1}{2} \|P - D \times \alpha\|_2^2 + \mu \|\alpha\|_1. \tag{8}$$

In (8),

$$P = \begin{pmatrix} \frac{x}{\sqrt{N}} \\ \frac{y}{\sqrt{M}} \end{pmatrix}, \quad D = \begin{pmatrix} \frac{D_h}{\sqrt{N}} \\ \frac{D_l}{\sqrt{M}} \end{pmatrix}, \tag{9}$$


 FIGURE 3: Process of constructing dictionary and reconstructing image (the upscale factor is  $p$ ).

where  $N$  is dimension of HR image blocks and  $M$  is dimension of LR image blocks. In order to solve (8), we set the initial value of matrix  $D$  and (8) can be converted into (10). Consider

$$\min \frac{1}{2} (P - D \times \alpha)^T (P - D \times \alpha) + \mu |\alpha|. \quad (10)$$

Then, the transformation of (10) is as shown in (11). Consider

$$\min \frac{1}{2} Q' \alpha Q + p' \alpha + \mu |\alpha|, \quad (11)$$

where  $Q = D^T D$  and  $p = D^T y$ . After obtaining  $\alpha$ , we can solve (12) to get the value of matrix  $D$ . Consider

$$\operatorname{argmin} \|P - D \times \alpha\|_2^2. \quad (12)$$

Therefore, dictionaries  $D_l$  and  $D_h$  are obtained. Then, we enlarge every input image of  $Y_m$  one by one, as shown in Figure 3. First, every input image is enlarged to  $A_{lm}$  by interpolation and the upscale factor is  $p$ . Second,  $A_{lm}$  is divided into blocks  $y$  of the same size. According to dictionaries  $D_l$  and  $D_h$ , we solve (7), (8), and (10) to get  $\alpha$ . Then, HR image

blocks are obtained according to the formula  $x = D_h \times \alpha$ . When all of the HR image blocks are gained, they are merged into one HR image  $A_{hm}$  according to the fixed order.  $A_{hm} + A_{lm}$  is the larger-scale reconstructed image and also is one of reconstruction images  $Y_{m+1}$ , as shown in Figure 3. Lastly, we obtain the reconstruction images set  $Y_{m+1}$  by using dictionary to construct every image in  $Y_m$ . Then, image sets  $Y_{m+1}$  and  $Q_{m+1}$  will become the input images and training images set of next reconstruction enlargement, respectively.  $Q_{m+1}$  is as shown in (13). Consider

$$Q_{m+1} = Y_{m+1}^{\wedge} \cup (Y_{m+1} - Y_{m+1}^{\wedge}) \cup Q_m. \quad (13)$$

**3.3. Reconstruction Fusion.** In Sections 3.1 and 3.2, we use self-learning pyramid method to construct training images and get reconstruction set  $Y_{n+1}$ . But the scale of reconstruction images is  $p^n$  times that of the original input images' scale which may not be the same scale with target HR image. So we use NLM idea to fuse all images of  $Y_{n+1}$  into one HR image which is the same scale with target HR image. Meanwhile, it can effectively eliminate fusion noise.

Nonlocal mean (NLM) filter is a very effective denoising filter, which is based on an assumption that each pixel is a repetition of surrounding pixels. It can be used to obtain the

target pixel value with a weighted sum of the surrounding pixels. Now, NLM is widely used in the field of super-resolution reconstruction. Just as [9], when we use NLM idea to reconstruct multiframe images, the smaller the magnification is, the smaller the reconstruction error is. So when image dimension of  $Y_{n+1}$  is closest to target HR image, using NLM idea to process  $Y_{n+1}$  is feasible. In this section, we adopt (14) to fuse images of  $Y_{n+1}$  into target HR image  $X_0$ . Consider

$$X = \operatorname{argmin} \sum_{(k,l) \in \omega} \sum_{t=1}^p \sum_{(i,j) \in N_{k,l}} \omega(k, l, i, j, t) \times \|D_{k,l} E_{k,l}^H X - E_{k,l}^L y_t\|, \quad (14)$$

where  $\omega(k, l, i, j, t)$  is the weight coefficient, usually calculated by the following formula. Consider

$$\omega(k, l, i, j, t) = \exp\left(-\frac{\|P_{k,l} y_t - P_{i,j} y_t\|^2}{2\sigma^2}\right). \quad (15)$$

The  $(k, l)$  is a pixel of the target HR image.  $y_t$  represents one of the image sets,  $Y_{n+1}$ .  $(i, j)$  is a pixel of image  $y_t$ .  $P_{k,l} y_t$  represents extracting image block from image  $y_t$  whose center is  $(k, l)$ . After amplification and denoising of reconstruction image, we get the prime estimation image  $X_0$ . Finally, we must add global reconstruction constraint to meet  $Y = SHX$ , as shown in (16). Consider

$$X = \operatorname{argmin} \|y - SHX\|_2^2 + \mu \|X - X_0\|_2^2. \quad (16)$$

Turn into the following formula (17). Consider

$$X = X_0 + SH^T (y - SHX) + \mu (X - X_0). \quad (17)$$

We can solve (17) to obtain the optimal estimated HR image  $X$  via IBP idea.

**3.4. The Proposed Algorithm.** The process of the proposed algorithm is as shown in the following writing based on the above sections. Suppose that we have four LR images of the same scene.

*Objective: Reconstruct HR Image from Multiple LR Images*

- (1) Initialization: input four LR images  $Y_1$  and solve the corresponding degrade images by formula  $Y_1 \downarrow_p \uparrow_p$ ; construct the first training image set  $Q_1$ ; set magnification  $p$  and iterations  $n$ .
- (2) For  $m = 1 : n$ , consider the following.
  - (a) HR image and LR image of the HR-LR training set  $Q_m$  are divided into image blocks  $x$  and  $y$ , respectively. Training dictionaries  $D_h$  and  $D_l$  are solved according to section (b) and the following formula. Consider

$$\begin{aligned} \operatorname{argmin} \frac{1}{2} \|x - D_h \times \alpha\|_2^2 + \mu \|\alpha\|_1, \\ \operatorname{argmin} \frac{1}{2} \|y - D_l \times \alpha\|_2^2 + \mu \|\alpha\|_1. \end{aligned} \quad (18)$$

- (b) Take  $Y_m$  as new input LR images to be reconstructed. And solve new HR images  $Y_{m+1}$  from dictionaries  $D_h$  and  $D_l$ . Then, we get four reconstruction images  $Y_{m+1}$ , in which resolution is  $p^m$  times of the original input images.
- (c) Put high-frequency of  $Y_{m+1}$  and corresponding degrade images into the HR training set and LR training set, respectively. So we get new training image set according to section (a) and (3). Take  $Y_{m+1}$  as new LR images to be reconstructed. Consider

$$\begin{aligned} Y_m^\wedge &= ((Y_m) \downarrow_p) \uparrow_p, \\ Q_{m+1} &= Y_{m+1}^\wedge \cup (Y_{m+1} - Y_{m+1}^\wedge) \cup Q_m. \end{aligned} \quad (19)$$

- (3) End
- (4) Adopt NLM idea to fuse images of  $Y_{n+1}$  into one the same scale HR image  $X_0$  with target image.
- (5) Add global reconstruction constraint to meet  $Y = SHX$ . Use IBP idea and gradient descent algorithm to solve (16) and obtain the optimal reconstruction HR image  $X$ . Consider

$$X = X_0 + SH^T (y - SHX) + \mu (X - X_0). \quad (20)$$

- (6) Output the final HR image  $X$ .

**3.5. The Discussion of Reconstruction Speed.** When we apply the NLM method to adjust reconstruction image in Section 3.3, it will spend much operation time by solving (14) directly. So, in this section, we will accelerate reconstruction speed by simplifying section (c) in this section. Set derivation of the (14) to be equal to zero. We can get (21). Consider

$$\begin{aligned} X &= \left( \sum_{(k,l) \in \omega} \sum_{t=1}^p \sum_{(i,j) \in N_{k,l}} \omega(k, l, i, j, t) \right. \\ &\quad \left. \times (E_{k,l}^H)^T D_{k,l}^T D_{k,l} E_{k,l}^H \right)^{-1} \\ &\quad \times \left[ \sum_{(k,l) \in \omega} \sum_{t=1}^p \sum_{(i,j) \in N_{k,l}} \omega(k, l, i, j, t) \times (E_{k,l}^H)^T D_{k,l}^T E_{k,l}^H y_t \right]. \end{aligned} \quad (21)$$

$E_{k,l}^L$  represents extracting one pixel from the HR image. Equation (21) becomes (22). Consider

$$X = \frac{\sum_{t=1}^p \sum_{(i,j) \in N_{k,l}} \omega(k, l, i, j, t) y_t(i, j)}{\sum_{t=1}^p \sum_{(i,j) \in N_{k,l}} \omega(k, l, i, j, t)}. \quad (22)$$



FIGURE 4: The input LR images. (a) is one of several Lena images. (b) is one of several building images. (c) is one of several flower images.

As described above, it will reduce operation time and avoid complicated matrix operations. Meanwhile, this algorithm can be parallelized to deal with such steps as 2(a), 2(b), and (4) of Section 3.4. Therefore, we can take advantage of GPU to further improve the reconstruction speed. As shown in Section 4.2, we use Gforce 780 M GPU to process the captured images in camera system experimentation.

Some blocks of the input LR image are smooth with few details, as shown in Figure 4 (the black box). These blocks contain little high-frequency information and have no effect for reconstruction HR image. So this section proposes interpolation algorithm to enlarge these blocks, and the algorithm can run faster. Smooth image block is defined by the following formula:

$$A = \sqrt{\frac{1}{p^2} (\bar{z} - z_i^l)^2}, \quad (23)$$

where  $z_i^l$  is the pixel value of image block and  $\bar{z}$  is the average value of block pixel. We set a fixed value  $A^0$ . If  $A < A^0$ , we use interpolation method to reconstruct the block.

## 4. Experimental Results and Analysis

**4.1. Simulation Experimentation.** In this section, we perform 2x magnification experiments on three test images from Figure 4 to validate the effectiveness of the proposed method. These images cover a lot of contents including Lena, flower, and building, as shown in Figure 4. The compared group includes five other methods: the interpolation-based method [4], the NLM method based on reconstruction [8], Glassner's method [25], Yang's method [23], and Zhang's method [27]. For example, the input LR Lena images are simulated from a  $256 \times 256$  normative HR image revolved, translated randomly, and downsampled by a factor of two. The image block size is set to  $5 \times 5$  with overlap of 4 pixels between adjacent blocks. And upscale factor  $P$  is 2. The gradual enlargement  $p$  is set to 1.25,  $n$  is 3, and  $s$  is 128/125. Although the proposed method has a certain similarity in the idea of dictionary-based reconstruction [23], it takes the following unique features.

- (1) This paper proposes self-learning pyramid method to construct larger-scale image step by step. And it does

not need to search lots of external HR-LR images as a training set.

- (2) The reconstructed result depends on the selected training images. It will cause to the reconstructed HR image distortion, given the absence of the input images' frequency information in the selected HR-LR training set. This method puts different scale reconstructed image into the training set, and it can provide much real similarity information to ensure the authenticity of the reconstruction HR image.
- (3) Because the reconstruction results have fusion noise and may not be the same scale with target HR image in Section 3.3, we use NLM idea to meet both cases and merge them into one HR image, which contains more target image information. Lastly, we use IBP idea to meet global reconstruction constraint.

Figures 5, 6, and 7 are the reconstruction results of the testing images Lena, flower, and building.

We prove the superiority of the proposed algorithm from the visual quality and objective quality of the reconstructed HR image. And peak signal to noise ratio (PSNR), structural similarity (SSIM), mean squared error (MSE), and mean absolute error (MAE) are employed to evaluate the objective quality of the reconstructed HR image. Definitions are as follows.

Mean squared error is as follows:

$$\text{MSE} = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (g(i, j) - f(i, j))^2. \quad (24)$$

Peak signal to noise ratio is as follows:

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}}. \quad (25)$$

Mean absolute error is as follows:

$$\text{MAE} = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N |g(i, j) - \bar{g}|. \quad (26)$$

The  $g(i, j)$  and  $f(i, j)$  are pixels of the reconstructed HR image and the original HR image, respectively. The  $M$  and  $N$

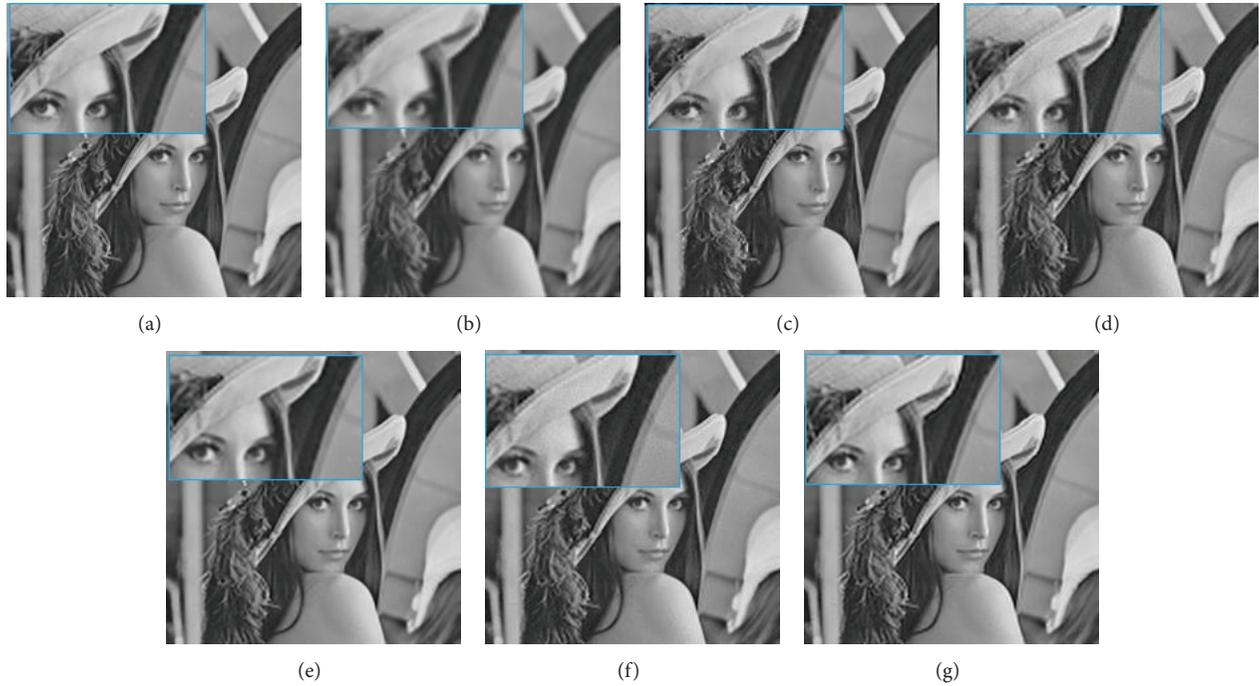


FIGURE 5: Comparison of reconstructed HR images of Lena. (a) Original HR image. (b) Interpolation-based method results [4]. (c) NLM results [8]. (d) Glassner's method [25]. (e) Yang's method [23]. (f) Zhang's method [27]. (g) The proposed method results.

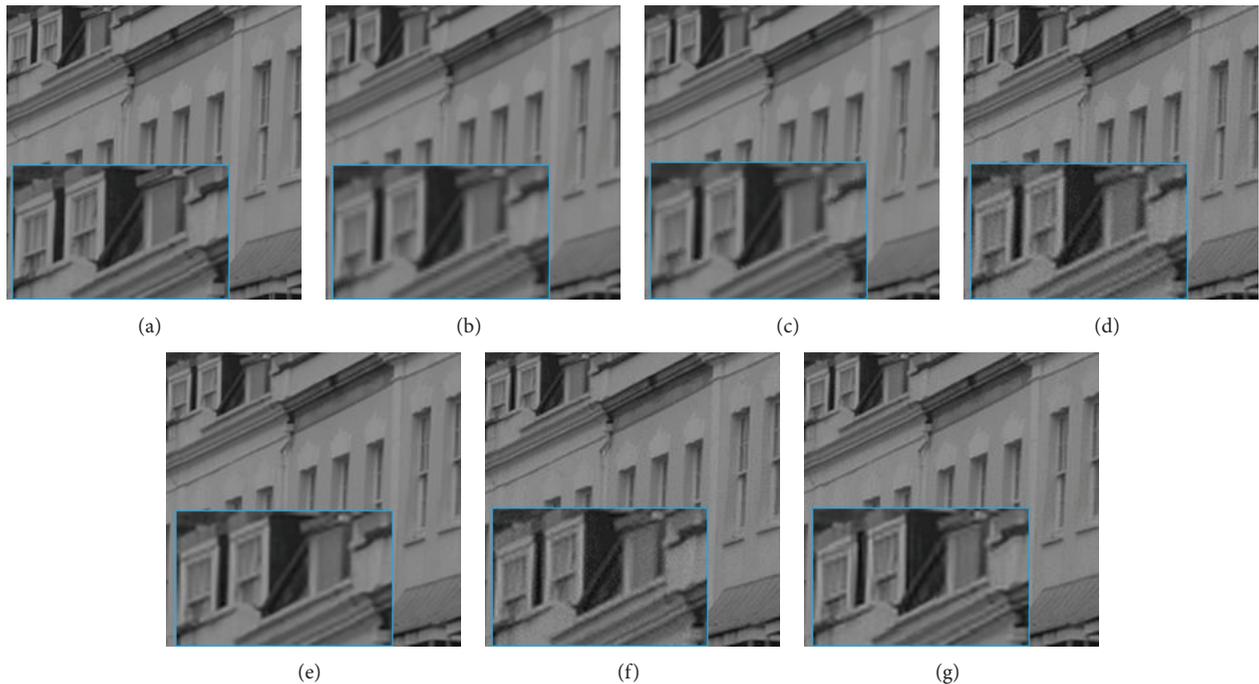


FIGURE 6: Comparison of reconstructed HR images of building. (a) Original HR image. (b) Interpolation-based method results [4]. (c) NLM results [8]. (d) Glassner's method [25]. (e) Yang's method [23]. (f) Zhang's method [27]. (g) The proposed method results.

are dimensions of reconstructed HR image. The  $\bar{g}$  is the mean value of all pixels of reconstructed HR image. When PSNR is larger and MAE is smaller, the quality of the reconstructed image is better. SSIM is to show the reconstructed quality by comparing image intensity, image contrast, and structural

similarity based on the visual system. The larger the SSIM is, the clearer the image is. Values of PSNR, SSIM, and MAE are shown in Table 1. The first, second, and third rows for each test image indicate PSNR, MAE, and SSIM values, respectively. Table 1 suggests that the proposed algorithm

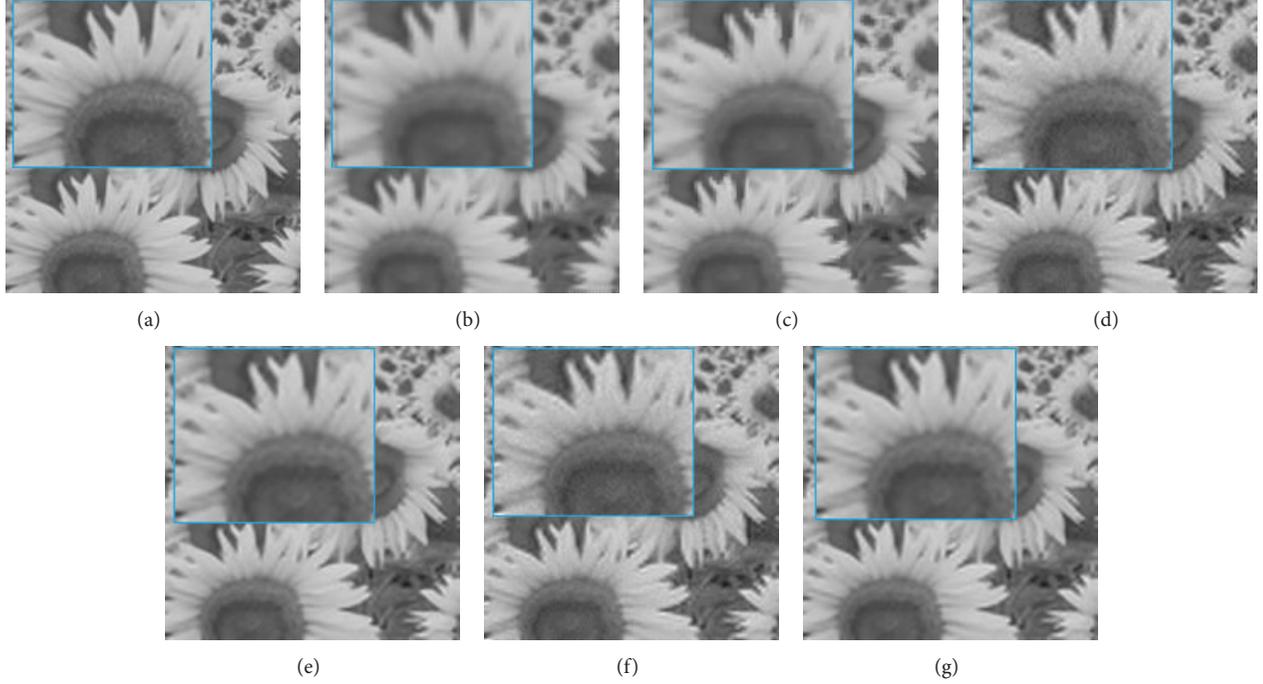


FIGURE 7: Comparison of reconstructed HR images of flower. (a) Original HR image. (b) Interpolation-based method results [4]. (c) NLM results [8]. (d) Glassner's method [25]. (e) Yang's method [23]. (f) Zhang's method [27]. (g) The proposed method results.

can achieve better reconstructed result and recover more missing high-frequency information and clearer edge details in comparison with other five methods.

To further assess the visual quality obtained by different methods, we compare reconstructed image details of different methods with original HR image, as shown in Figures 5, 6, and 7. When we contrast eye and the brim of a hat in Figure 5, the result of interpolation-based method is vaguer and NLM method's reconstructed result appears distortion and mosaic. On the contrary, the proposed method does not cause distortion and vague compared with interpolation-based and NLM cases. Moreover, reconstructed details of our method are more delicate than other algorithms. In Figures 6 and 7, the proposed method can also produce better reconstructed details, eaves and flower border, both accurately and visually in comparison with other superresolution algorithms.

**4.2. Camera System Experimentation.** In this section, we conduct research on real scene reconstruction to prove the superiority of the proposed algorithm. It is the camera system assembled by ourselves, which consists of a microdisplacement lens, PI turntable, and Gforce 780 M GPU. By controlling the PI turntable, we can continuously receive multiple real images, as shown in Figure 8(a) (one of multiple real images). The images are taken to be reconstructed by our method, and result is as shown in Figure 8(f). Figures 8(b)–8(e) are the reconstruction results of the NLM method based on reconstruction [9], Glassner's method [25], Yang's method, and Zhang's method [27]. Signal noise ratio (SNR), average gradient (AG), information entropy (IE), and standard deviation (SD), which do not require reference HR

image, are employed to evaluate the objective quality of the reconstructed real HR image in real scene reconstruction of camera system. Definitions are as follows. Consider

$$\text{SNR} = \frac{M}{\text{STD}_{\max}}. \quad (27)$$

$M$  represents image mean value. The  $\text{STD}_{\max}$  represents the max value of local standard deviation. Consider

$$\text{IE} = \sum_{i=0}^{L-1} P_i \log_2 P_i. \quad (28)$$

$L$  is the total image gray level, and  $P_i$  represents the ratio of pixels in gray value  $i$  to pixels in all image gray values. Consider

$$\begin{aligned} \text{AG} &= \sum_{i=1}^{M-1} \sum_{j=1}^{N-1} \sqrt{\left( \left( \frac{\partial g(x_i, y_j)}{\partial x_i} \right)^2 + \left( \frac{\partial g(x_i, y_j)}{\partial y_j} \right)^2 \right)} \times \frac{1}{2} \\ &\quad \times ((M-1)(N-1))^{-1}, \\ \text{SD} &= \sqrt{\frac{\sum_{i=1}^M \sum_{j=1}^N (g(x_i, y_j) - \bar{g})^2}{M \times N}}. \end{aligned} \quad (29)$$

The average gradient represents clarity. Information entropy can indicate image information abundance. Standard deviation represents discrete situation of image gray level. When SNR, SD, IE, and AG are larger, the reconstructed

TABLE 1: The results of PSNR, MAE, and SSIM. For every image, there are three rows. The first, second, and third rows for each test image indicate PSNR, MAE, and SSIM values.

	Interpolation [4]	NLM method [8]	Glassner's method [25]	Yang's method [23]	Zhang's method [27]	The proposed method
Lena	27.3993	27.4644	30.6150	31.0123	31.5623	<b>32.6016</b>
	6.0320	5.8437	4.2831	3.9561	3.8601	<b>3.7735</b>
	0.8631	0.8643	0.9051	0.9168	0.9250	<b>0.9538</b>
Building	31.2571	31.2962	37.1000	37.0549	36.9531	<b>37.6990</b>
	4.1998	4.0993	2.3300	2.3254	2.5365	<b>2.0510</b>
	0.8746	0.8753	0.9537	0.9524	0.9124	<b>0.9653</b>
Flower	26.3318	25.7424	30.5731	31.2439	31.3826	<b>32.1054</b>
	8.0847	8.3127	5.240	4.6710	4.6425	<b>4.4371</b>
	0.8176	0.7931	0.9031	0.9280	0.9315	<b>0.9501</b>
Average	28.3294	28.1676	32.76	33.1037	33.2993	<b>34.1354</b>
	6.1055	6.0852	3.9510	3.6508	3.6797	<b>3.4205</b>
	0.8517	0.8442	0.9206	0.9324	0.9229	<b>0.9564</b>

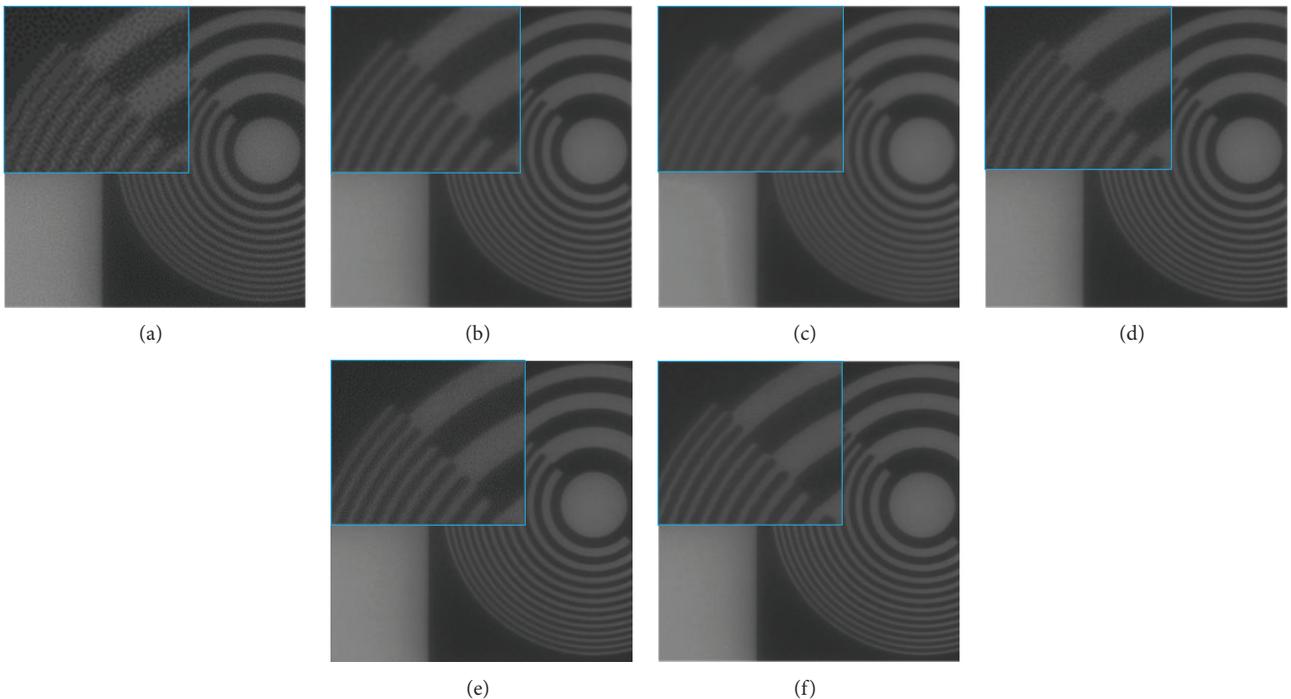


FIGURE 8: Real scene reconstruction. (a) One of the captured images. (b) NLM results [8]. (c) Glassner's method [25]. (d) Yang's method [23]. (e) Zhang's method [27]. (f) The proposed method results.

image is clearer and contains richer information. Therefore, the numeric results of proposed method are better than those of other algorithms as shown in Table 2. Visually, in comparison of contour outline of the reconstructed image detail, the reconstructed image in Figure 8(f) is significantly clearer than the other reconstructed image. Consequently, the proposed method can work out well in the camera system experimentation and be capable of reproducing plausible details and sharp edges with minimal artifacts.

## 5. Conclusion

It is known that, in superresolution reconstruction, distortion of reconstructed images occurs when frequency information

of target images is not contained in training images. To solve this problem, this paper proposes a multiple super-resolution reconstruction algorithm based on self-learning dictionary. First, images on a series of scales created from multiple input LR images are used as training images. Then, a learned overcomplete dictionary provides sufficient real image information that ensures the authenticity of the reconstructed HR image. In both simulation and real experiments, the proposed algorithm achieves better results than that achieved by other algorithms, recovering missing high-frequency information and edge details more effectively. Finally, several ideas towards higher computation speed are suggested.

TABLE 2: The results of SNR, AG (average gradient), IE (information entropy), and SD (standard deviation).

	Real scene	NLM method [8]	Glassner's method [25]	Yang's method [23]	Zhang's method [27]	The proposed method
SNR	5.3115	6.7257	6.6978	6.7899	6.8049	<b>7.1254</b>
AG	1.1059	1.1895	1.7399	1.7296	1.7327	<b>1.8138</b>
IE	6.3191	6.3151	6.3536	6.3411	6.3609	<b>6.4141</b>
SD	26.2563	26.2807	26.2999	26.3164	26.3043	<b>26.3733</b>

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work was supported by the project of National Science Fund for Distinguished Young Scholars of China (Grant no. 60902067) and the key Science-Technology Project of Jilin province (Grant no. 11ZDGG001).

## References

- [1] P. Thevenaz, T. Blu, and M. Unser, *Image Interpolation and Resampling*, Academic Press, New York, NY, USA, 2000.
- [2] H. S. Hou and H. C. Andrews, "Cubic spline for image interpolation and digital filtering," *IEEE Transactions on Signal Process*, vol. 26, no. 6, pp. 508–517, 1978.
- [3] N. A. Dodgson, "Quadratic interpolation for image resampling," *IEEE Transactions on Image Processing*, vol. 6, no. 9, pp. 1322–1326, 1997.
- [4] V. Patrick, S. Sabine, and V. Martin, "A frequency domain approach to registration of aliased images with application to super-resolution," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 71459, 14 pages, 2006.
- [5] M. Irani and S. Peleg, "Improving resolution by image registration," *Graphical Models and Image Processing*, vol. 53, no. 3, pp. 231–239, 1991.
- [6] P. Cheeseman, B. Kanefsky, R. Kraft, J. Stutz, and R. Hanson, "Super-resolved surface reconstruction from multiple images," in *Maximum Entropy and Bayesian Methods: Santa Barbara, California, U.S.A., 1993*, vol. 62 of *Fundamental Theories of Physics*, pp. 293–308, Springer, 1996.
- [7] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, "Joint MAP registration and high-resolution image estimation using a sequence of under-sampled images," *IEEE Transactions on Image Processing*, vol. 6, no. 12, pp. 1621–1633, 1997.
- [8] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1327–1344, 2004.
- [9] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the nonlocal-means to super-resolution reconstruction," *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 36–51, 2009.
- [10] J.-D. Li, Y. Zhang, and P. Jia, "Super-resolution reconstruction using nonlocal means," *Optics and Precision Engineering*, vol. 21, no. 6, pp. 1576–1585, 2013.
- [11] C. Yang, J. Huang, and M. Yang, "Exploiting self-similarities for single frame super-resolution," in *Proceedings of the 10th Asian Conference on Computer Vision (ACCV '10)*, vol. 3, pp. 497–510, 2010.
- [12] K. Zhang, X. Gao, X. Li, and D. Tao, "Partially supervised neighbor embedding for example-based image super-resolution," *IEEE Journal on Selected Topics in Signal Processing*, vol. 5, no. 2, pp. 230–239, 2011.
- [13] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Curves and Surfaces*, pp. 711–730, Springer, 2010.
- [14] Y.-W. Tai, S. Liu, M. S. Brown, and S. Lin, "Super resolution using edge prior and single image detail synthesis," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 2400–2407, San Francisco, Calif, USA, June 2010.
- [15] J. Sun, J. Zhu, and M. F. Tappen, "Context-constrained hallucination for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 231–238, IEEE, San Francisco, Calif, USA, June 2010.
- [16] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [17] H. Chang, D. Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 1, pp. 275–282, 2004.
- [18] Q. Liu, L. Liu, Y. Wang, and Z. Zhang, "Locally linear embedding based example learning for pan-sharpening," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR '12)*, pp. 1928–1931, Tsukuba, Japan, 2012.
- [19] Y. Tao, Z. L. Gan, and X. C. Zhang, "Novel neighbor embedding super resolution method for compressed images," in *Proceedings of the 4th International Conference on Image Analysis and Signal Processing (IASP '12)*, pp. 1–4, November 2012.
- [20] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. Alberi Morel, "Neighbor embedding based single-image super-resolution using semi-nonnegative matrix factorization," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '12)*, vol. 1, pp. 1289–1292, Kyoto, Japan, March 2012.
- [21] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008.
- [22] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.
- [23] L. He, H. Qi, and R. Zaretzki, "Beta process joint dictionary learning for coupled feature spaces with application to single image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 345–352, Portland, Ore, USA, June 2013.
- [24] S.-C. Jeong, Y.-W. Kang, and B. C. Song, "Super-resolution algorithm using noise level adaptive dictionary," in *Proceedings of the 14th IEEE International Symposium on Consumer Electronics (ISCE '10)*, June 2010.

- [25] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proceedings of the 12th International Conference on Computer Vision (ICCV '09)*, pp. 349–356, Kyoto, Japan, October 2009.
- [26] S. Chen, H. Gong, and C. Li, "Super resolution from a single image based on self similarity," in *Proceedings of the International Conference on Computational and Information Sciences (ICCIS '11)*, pp. 91–94, Chengdu, China, October 2011.
- [27] K. Zhang, X. Gao, D. Tao, and X. Li, "Single image super-resolution with multiscale similarity learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 10, pp. 1648–1659, 2013.
- [28] S. A. Ramakanth and R. V. Babu, "Super resolution using a single image dictionary," in *Proceedings of the IEEE International Conference on Electronics, Computing and Communication Technologies (IEEE CONECCT '14)*, pp. 1–6, Bangalore, India, January 2014.

## Research Article

# Automatic Segmentation of Nature Object Using Salient Edge Points Based Active Contour

Shangbing Gao,<sup>1,2,3</sup> Yunyang Yan,<sup>1</sup> Youdong Zhang,<sup>1</sup> Jingbo Zhou,<sup>1,3</sup>  
Suqun Cao,<sup>1,2</sup> and Jianxun Xue<sup>2</sup>

<sup>1</sup>Faculty of Computer Engineering, Huaiyin Institute of Technology, Huai'an 223003, China

<sup>2</sup>Jiangsu Provincial Key Laboratory for Advanced Manufacturing Technology, Huaiyin Institute of Technology, Huai'an 223003, China

<sup>3</sup>The School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China

Correspondence should be addressed to Shangbing Gao; [luxiaofen\\_2002@126.com](mailto:luxiaofen_2002@126.com)

Received 8 September 2014; Revised 14 December 2014; Accepted 15 December 2014

Academic Editor: Erik Cuevas

Copyright © 2015 Shangbing Gao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Natural image segmentation is often a crucial first step for high-level image understanding, significantly reducing the complexity of content analysis of images. LRAC may have some disadvantages. (1) Segmentation results heavily depend on the initial contour selection which is a very skillful task. (2) In some situations, manual interactions are infeasible. To overcome these shortcomings, we propose a novel model for unsupervised segmentation of viewer's attention object from natural images based on localizing region-based active model (LRAC). With aid of the color boosting Harris detector and the core saliency map, we get the salient object edge points. Then, these points are employed as the seeds of initial convex hull. Finally, this convex hull is improved by the edge-preserving filter to generate the initial contour for our automatic object segmentation system. In contrast with localizing region-based active contours that require considerable user interaction, the proposed method does not require it; that is, the segmentation task is fulfilled in a fully automatic manner. Extensive experiments results on a large variety of natural images demonstrate that our algorithm consistently outperforms the popular existing salient object segmentation methods, yielding higher precision and better recall rates. Our framework can reliably and automatically extract the object contour from the complex background.

## 1. Introduction

Object segmentation is one of the most important and challenging issues in image analysis and computer vision research. It facilitates a number of high-level applications, such as object recognition, image retrieval, image editing, and scene reconstruction [1, 2]. Most existing object segmentation systems adopt interaction-based paradigms [3, 4]; that is, users are asked to provide segmentation cues manually and carefully.

Although the interaction-based methods are promising, they all pose a critical problem in which they need the users' semantic intention. Such manual labeling is time consuming and often infeasible. Moreover, the segmentation performance heavily depends on the user-specified seed locations. Thus, additional interactions are necessary when the seeds are not accurately provided. Specially, localizing region-based active contour (called LRAC) [5] is exactly

one of the classic interaction-based methods. Segmentation results heavily depend on the initial contour selection. Thus, it needs the specified initial contour which should be close to the boundary of object.

For this reason, developing a sophisticated fully automatic object segmentation method has been strongly demanded. The human brain and visual system can effortlessly grasp certain salient regions in cluttered scenes. By observing the fact that, under most circumstances, the salient parts of an image are usually consistent with interesting objects to be segmented, salient regions have been attempted for estimation. In contrast with existing interaction-based approaches that specify the object and background seeds by manual labeling, some methods (e.g., Fu's method [6] and Achanta's method [7]) determine the seed locations based on the visual attention model. Since the accuracy of the visual attention model plays a crucial role in object segmentation, these algorithms also depend on the quality of the chosen saliency

map. Alternatively speaking, the worse the chosen saliency map is, the worse the corresponding final extraction result is.

To remedy such shortcoming, we pay close attention to salient object edge points rather than the saliency map itself. After the salient object edge points were detected, the region which is constrained by these corner points will be obtained. The boundary of this region is close to the object edge. Thereby, the boundary of this region is used as the initial contour of LRAC model.

In our method, the salient edge points are generated by the color boosting Harris detector for input image firstly. We then explore the salient object seeds by the core saliency map, and the salient object edge points are determined by these salient object seeds. Initial contour is then created by convex hull algorithm with salient object edge points automatically. Finally, the object will be extracted accurately by LRAC method with the initial contour in the previous step.

The remainder of this paper is organized as follows. Section 2 reviews some related work about saliency models and an interactive image segmentation method. Section 3 presents the proposed salient edge point based active contour for natural object segmentation algorithm. Section 4 demonstrates extensive experimental comparison results. Section 5 finally draws the conclusions.

## 2. Related Work

*2.1. The State-of-the-Art Automatic Image Segmentation Methods.* In [6], Fu et al. proposed an automatic object segmentation approach integrating saliency detection and graph cuts [8], namely, Fu's method, to overcome the disadvantages of interactive graph cuts. They also explored the effects of labels to graph based segmentation, and the so-called "Professional Labels" are introduced to evaluate labels and a multiresolution framework is designed to provide such "Professional Labels" automatically. This method obtains quite complete object segmentation comparable to interactive graph cuts with manual "Professional Labels."

Achanta's method [7] is also an automatic image segmentation method. It oversegments the input image using mean-shift algorithm and retains only those segments whose average saliency is greater than an adaptive threshold. The binary maps representing the salient object are thus obtained by assigning ones to pixels of chosen segments and zeroes to the rest of the pixels.

These two methods are absolutely automatic and involve none of manual interactions. Fu's method is based on either of the graph cuts while Achanta's method uses mean-shift algorithm. However, there are several desirable advantages of LRAC over graph cuts and mean-shift algorithm. First, LRAC can achieve subpixel accuracy of object boundaries [5]. Second, LRAC can be easily formulated under a principled energy minimization framework and allow incorporation of various prior knowledge for robust image segmentation. Third, LRAC can provide smooth and closed contours as segmentation results which are necessary and can be readily used for further applications, such as shape analysis and recognition.

*2.2. Localizing Region-Based Active Contour Model.* In [5], Lankton and Tannenbaum proposed a natural framework that allows any region-based segmentation energy to be reformulated in a local way.

In general, this algorithm could reliably extract the object contour if the user inputs appropriate markers. Namely, the interactive segmentation algorithm is more or less sensitive to the position and quality of the user-inputs (see an example in Figure 1).

Here, we choose a complex energy that looks past simple means and compares the full histograms of the foreground and background. Consider  $P_u(z)$  and  $P_v(z)$  to be two smoothed intensity histograms computed from the global interior and exterior regions of a partitioned image using intensity bins.

Here, we choose the global region-based energy that uses mean intensities which is the one proposed by Wen et al. [9] which we refer to as histogram separation energy:

$$E_{\text{HS}} = \text{BC} = \int_z \sqrt{P_u(z) P_v(z)} dz, \quad (1)$$

where BC is the Bhattacharyya coefficient used to compare probability density functions and  $P_u(z)$  and  $P_v(z)$  represent two smoothed intensity histograms computed from the global interior and exterior regions of a partitioned image using intensity bins. Optimizing this energy causes that the interior and exterior means have the largest difference possible.

In [5], Lankton and Tannenbaum introduced  $B(x, y)$  to mask local regions. Function  $B(x, y)$  will be 1 when the point  $y$  is within a ball of radius centered at  $x$  and 0 otherwise.

Accordingly, the corresponding internal energy function  $F$  is formed by localizing the histogram separation energy as shown in

$$F_{\text{HS}} = \int_z \sqrt{P_{u,x}(z) P_{v,x}(z)} dz, \quad (2)$$

where  $P_{u,x}(z)$  and  $P_{v,x}(z)$  represent the intensity histograms in the local image regions  $B(x, y) \cdot H\phi(y)$  and  $B(x, y) \cdot (1 - H\phi(y))$ , respectively.

We can get the following local region-based flow:

$$\begin{aligned} \frac{\partial \phi}{\partial t} = & \delta \phi(x) \int_{\Omega_y} \frac{B(x, y) \delta \phi(y)}{2} \\ & \times \left[ F_{\text{HS}} \left( \frac{1}{A_v} - \frac{1}{A_u} \right) \right. \\ & \left. + \int_z K(z - I(y)) \right. \\ & \left. \times \left( \frac{1}{A_u} \sqrt{\frac{P_{v,x}(z)}{P_{u,x}(z)}}} \right) \right] \end{aligned}$$

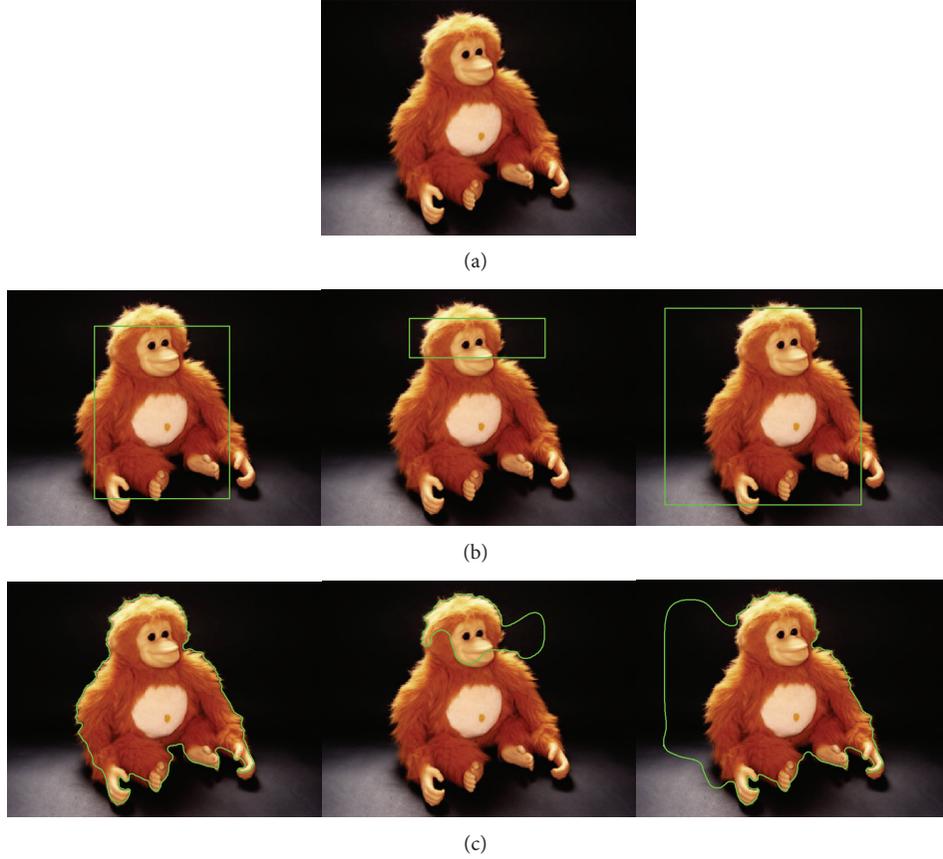


FIGURE 1: Interactive image segmentation by LRAC [5] with user-specified strokes of the object (green). First row: input image. Second row: three different user-specified inputs. Third row: the corresponding segmented objects with respect to different user-specified inputs.

$$\begin{aligned}
 & -\frac{1}{A_v} \sqrt{\frac{P_{u,x}(z)}{P_{v,x}(z)}} dz \Big] dy \\
 & + \lambda \delta\phi(x) \operatorname{div} \left( \frac{\nabla\phi(x)}{|\phi(x)|} \right),
 \end{aligned} \tag{3}$$

where  $K$  is a Gaussian kernel,  $\lambda$  is a parameter which weights the length of the curve,  $\Omega_y$  denotes a bounded open subset of  $\mathbb{R}^2$ , and  $A_u$  and  $A_v$  are the areas of the local interior and local exterior regions, respectively, given by

$$\begin{aligned}
 A_u &= \int_{\Omega_y} B(x, y) \cdot H\phi(y) dy, \\
 A_v &= \int_{\Omega_y} B(x, y) \cdot (1 - H\phi(y)) dy, \\
 \delta\phi(x) &= \begin{cases} 1 & \phi(x) = 0 \\ 0 & |\phi(x)| < \varepsilon \\ \frac{1}{2\varepsilon} \left\{ 1 + \cos\left(\frac{\pi\phi(x)}{\varepsilon}\right) \right\}, & \text{otherwise.} \end{cases}
 \end{aligned} \tag{4}$$

In general, this algorithm could reliably extract the object contour if the user inputs appropriate markers. Namely, the

original interactive segmentation algorithm is more or less sensitive to the position and quantity of the user-inputs. Although many markers were used to cover the object features, in some regions it does not achieve satisfying results (see the third row of Figure 1). Moreover, it is tedious and time consuming in some cases.

**2.3. Saliency Detection Models.** During the last two decades, visual saliency detection and saliency map generation aiming to find out what attracts human's attention got broad interest in computer vision, especially for object detection or recognition from different scenes. A majority of computational models of attention follow the structure adapted from the Feature Integration Theory (FIT) [10] and the Guided Search model [11]. The saliency detection models fall into two general categories: local contrast based method and global contrast based methods.

Local contrast based methods investigate the rarity of image regions with respect to (small) local neighborhoods. Based on the highly influential biologically inspired early representation model introduced by Koch and Ullman [12], Itti et al. [13] define image saliency using central surrounded differences across multiscale image features. Ma and Zhang [14] propose an alternative local contrast analysis for generating saliency maps, which is then extended using a fuzzy growth

model. Harel et al. [15] normalize the feature maps of Itti et al., to highlight conspicuous parts and permit combination with other importance maps. Liu et al. [16] find multiscale contrast by linearly combining contrast in a Gaussian image pyramid. More recently, Goferman et al. [17] simultaneously model local low-level clues, global considerations, visual organization rules, and high-level features to highlight salient objects along with their contexts. Such methods using local contrast tend to produce higher saliency values near edges instead of uniformly highlighting salient objects.

Global contrast based methods evaluate saliency of an image region using its contrast with respect to the entire image. Zhai and Shah [18] define pixel-level saliency based on a pixel's contrast to all other pixels. However, for efficiency, they use only luminance information, thus ignoring distinctiveness clues in other channels. Achanta et al. [7] propose a frequency tuned method that directly defines pixel saliency using a pixel's color difference from the average image color. The elegant approach, however, only considers first order average color, which can be insufficient to analyze complex variations common in natural images. A recent excellent model proposed by Cheng et al. [19], which is named RC, calculated the saliency map by evaluating global contrast differences based on histogram.

We compared the abovementioned 5 state-of-the-art saliency detection methods. The comparison results are shown in Figure 2.

### 3. The Proposed Method: LRACSEP

For the issues pointed out in Section 2, in this paper, we focus our attention on the automatic acquisition of prior information. For one pixel in a saliency map, the saliency value is proportional to the intensity value. In other words, normally, for an image, pixels which have higher values in the corresponding saliency map are object pixels; conversely, they are background pixels. Inspired by this idea, we proposed our approach called localizing region-based active contours via salient edge points (LRACSEP). This strategy is intended mainly for the acquisition of prior information automatically instead of user-inputs.

Our purpose is to set the initial contour close to the object boundary. It is noted that the color boosting Harris detector yields the salient edge points. Consequently, we have to detect the salient object edge points firstly. For this purpose, we propose the core saliency map to find the salient object edge points. As is known to all, the initial contour of the level set is a closed curve. Therefore, we choose convex-hull polygon to embody the detected salient object points.

A general schematic framework of our proposed method (LRACSEP) is depicted in Figure 3. The major steps include (i) detecting the salient edge points; (ii) obtaining the core saliency map; (iii) finding the core edge points corresponding to the core saliency map; (iv) detecting the salient object edge points based on the core saliency map; (iv) using convex hull to generate the initial level set contour.

*3.1. Salient Edge Points Detection via the Color Boosting Harris Detector.* Traditional luminance-based saliency detection

methods incline to completely ignore the color information and thus are very sensitive to the background noises. van de Weijer et al. [20] analyze the statistical distribution of color derivative and propose a color saliency boosting function to enhance rare color edges or corners. Their goal is to incorporate color distinctiveness into salient point detection or, mathematically, to find the transformation for which vectors with equal information content have equal impact on the saliency function. The desired color saliency boosting function is obtained by

$$g(f_x) = \Lambda h(f_x), \quad (5)$$

where  $\Lambda$  is a  $3 \times 3$  diagonal matrix with  $\Lambda_{11} = \alpha$ ,  $\Lambda_{22} = \beta$ ,  $\Lambda_{33} = \gamma$ , and  $\Lambda_{11}^2 + \Lambda_{22}^2 + \Lambda_{33}^2 = 1$ ,  $h$  is one of the color transformations  $\tilde{S}$ ,  $\tilde{O}$ , or  $H$ , and  $f = (R, G, B)^T$  for a color image.

Meanwhile, the Harris corner detector [21] is a popular interest point detector due to its strong invariance to rotation, scale, illumination variation, and image noise. The Harris detector has been shown to outperform other detectors both on "shape" distinctiveness and repeatability.

The Harris corner detector is based on the local autocorrelation function of a signal, where the local autocorrelation function measures the local changes of the signal with patches shifted by a small amount in different directions. Thereby, the boosting color saliency theory can be applied to Harris detector. As can be seen in Figure 4, compared with the intensity-based feature detectors, the boosted color saliency points [20] are shown to be more stable and informative.

In this paper, we adopt the color boosting Harris points as salient points (Figure 4(d)) to catch the corners or marginal points of visual salient region in color image. The salient points provide us a coarse location of the salient areas. These points are denoted by  $SI(i)$ ,  $i = 1, 2, \dots, M$ . However, these points contain not only salient object points but also salient background points. The salient background points (from the tree in Figure 4) are noises for us to get the initial contour close to the object. Thus, the objective of our model is to distinguish object points from background points. It is exactly binary classification problem. Hereby, we will present a clustering method to find salient object points, which is based on the initial object seeds. Therefore, the objective is to select the most appropriate initial seeds. For this purpose, we present the core saliency map. The seeds of salient points are determined by the core saliency map.

*3.2. The Seeds Determined by the Core Saliency Map.* We choose the three prominent saliency models: RC, MZ, and FT. MZ is local prominent method while RC and FT are based on global contrast. The reason of choosing the two global contrast based models is that FT can output desirable results with very efficient computation while RC can well represent the regional contrast feature and is insensitive to local sudden changes.

As can be seen in Figure 5, the details highlighted by the three saliency maps ( $S^{RC}$ ,  $S^{MZ}$ , and  $S^{FT}$ ) are not the same. In spite of this, these saliency maps prefer to highlight the common parts of objects (referred to as core saliency map).

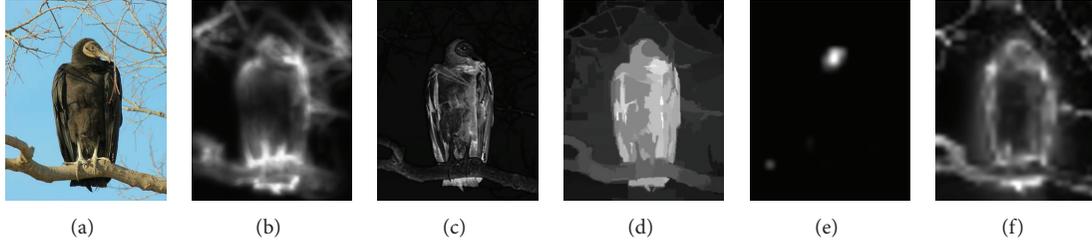


FIGURE 2: Saliency maps computed by different state-of-the-art methods. From the left to the right: original image, CA [17], FT [8], RC [19], IT [13], and GB [15] saliency maps.

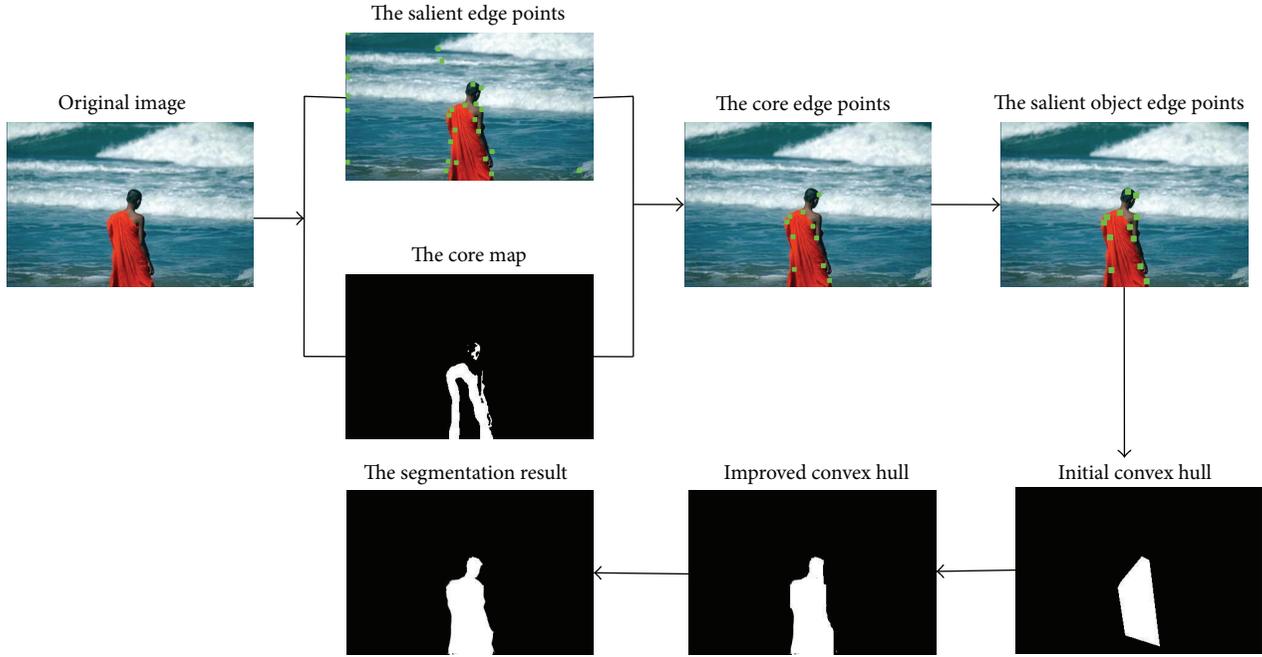


FIGURE 3: A general schematic framework of LRACSEP.

For any pixel  $p$ , the core saliency map ( $S_{\text{coresaliency}}(p)$ ) is computed as

$$S_{\text{coresaliency}}(p) = S^{\text{RC}}(p) \times S^{\text{MZ}}(p) \times S^{\text{FT}}(p). \quad (6)$$

For convenient show, we propose the core map ( $S_{\text{core}}(p)$ ), which is the binarization of  $S_{\text{coresaliency}}(p)$ :

$$S_{\text{core}}(p) = B(S_{\text{coresaliency}}(p)), \quad (7)$$

where  $B(\cdot)$  is the binarization operator. To binarize  $S_{\text{coresaliency}}(p)$ , we introduce adaptive threshold  $T$  which is determined as

$$T = \frac{2}{h \times w} \sum_p S_{\text{coresaliency}}(p), \quad (8)$$

where  $h$  and  $w$  are the height and width of the image, respectively. The corresponding core map is exactly Figure 5(d).

Pixels which are included in the core map are highly likely to be parts of the object. Consequently, the points

which are included not only in the salient edge points (white dots in Figure 4(d)) but also in the core map are labeled as foreground seeds. These seeds are indicated by blue dots in Figure 6.

### 3.3. The Salient Object Edge Points Detection and Using Convex Hull

As known to all, each superpixel is a perceptually consistent unit; that is, all pixels in a superpixel are most likely uniform in color and texture. For this reason, provided that one of the color boosting Harris points is in the same superpixel with the foreground seed, this point should be treated as the salient object edge points. According to this strategy, the search of the salient object edge points is shown in Figure 7. We can observe that the points in the left-hand part of Figure 7(c) are omitted in Figure 7(d).

Then the convex hull (Figure 8(a)) is used to embody these salient object edge points. The contour of the convex hull (green line in Figure 8(b)) is chosen as the initial contour

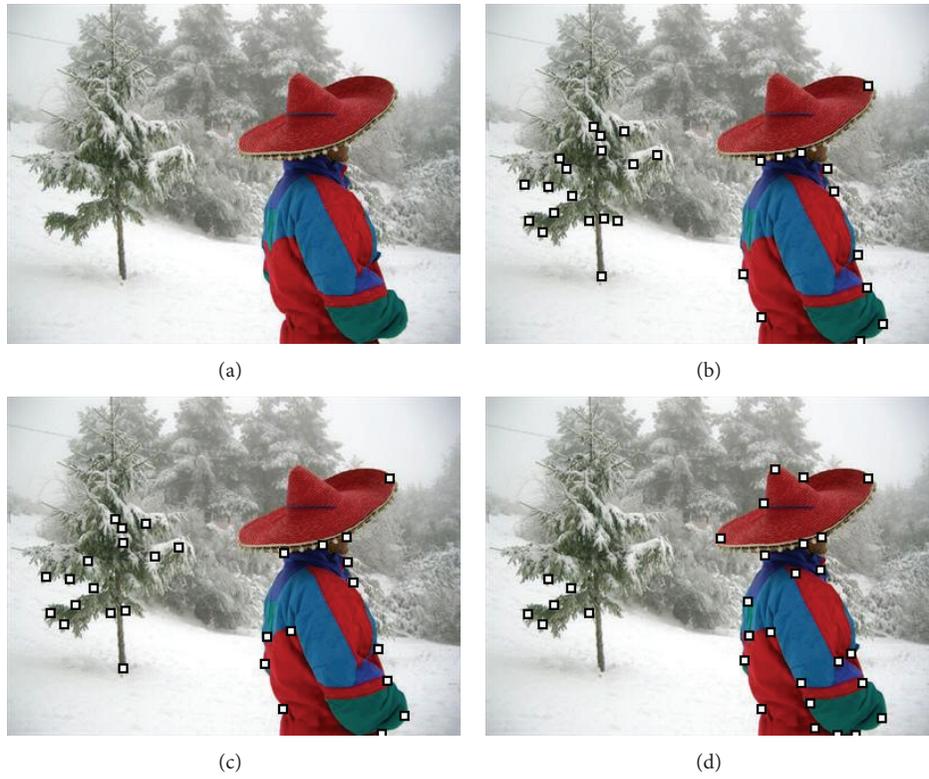


FIGURE 4: The salient edge points. (a) Original image; (b) luminance-based Harris points; (c) color based Harris points; (d) the color boosting Harris points.

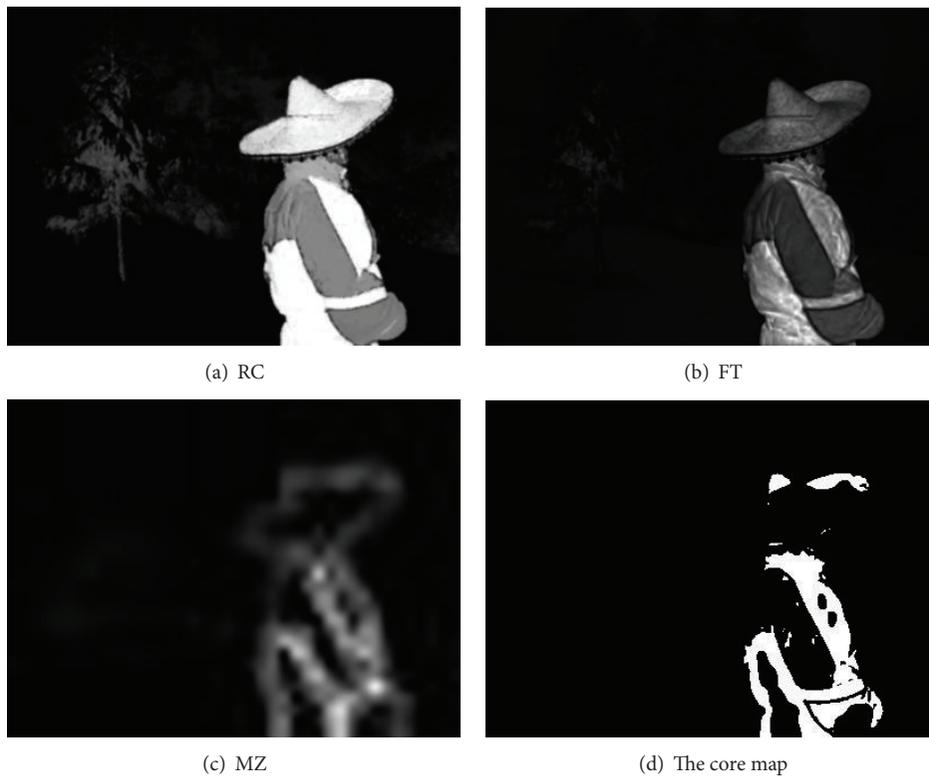


FIGURE 5: Three state-of-the-art saliency maps and the core map.



FIGURE 6: The foreground seeds determined by the core map.

of LRAC model. This initial contour is not sufficiently close to the boundary of this object.

**3.4. Improved Convex Hull by an Edge-Preserving Filter.** Given input image  $I$  and initial convex hull  $M_i$ , we want to get a refined convex hull. We note that solving this problem is similar to the image matting method. Therefore, our goal can be achieved by minimizing

$$E(M_R) = (M_R - M_i)^T \Lambda (M_R - M_i) + M_R L M_R, \quad (9)$$

where  $M_R$  denotes the output that we want to get,  $M_i$  is the initial convex hull,  $\Lambda$  is a diagonal matrix encoded with the weights of the constraints, and  $L$  is the matting Laplacian matrix [21]. The  $(i, j)$ th element of  $L$  is given by

$$\sum_{k:(i,j) \in \omega_k} \left( \delta_{ij} - \frac{1}{|\omega|} \left( 1 + (I_i - \mu_k)^T \left( \sum_k + \varepsilon U_k \right)^{-1} \cdot (I_j - \mu_k) \right) \right), \quad (10)$$

where  $\delta_{ij}$  is Kronecker delta,  $i$  and  $j$  are pixel indexes of input image  $I$ ,  $\mu_k$  is a  $3 \times 1$  mean vector of the colors in a squared window  $\omega_k$  with dimensions  $r \times r$ , centered at pixel  $k$ ,  $\sum_k$  is a  $3 \times 3$  covariance matrix,  $U_k$  is a  $3 \times 3$  identity matrix,  $|\omega|$  denotes the number of pixels in the window  $\omega_k$ , and  $\varepsilon$  is a smoothness parameter.

As can be seen from Figure 9(a), the initial contour (the contour of the convex hull) is close to the object boundary. It gives rise to the fact that LRAC model provides good segmentation performance and the times of iterative steps are reduced in contour evolution.

We use more images to better show the performance of our improved convex hull. They are shown in Figure 10. It is obvious that the obtained convex hull is more close to the real object than the initial convex hull.

## 4. Experiments

In order to verify the proposed method, we have evaluated the results of our approach on the publicly available database

provided by Achanta et al. [7]. This database includes 5000 images, originally containing labeled rectangles from nine users drawing a bounding box around what they consider the most salient object. There is a large variation among images, including natural scenes, animals, indoor, and outdoor. To the best of our knowledge, the database is the largest of its kind and has ground truth in the form of accurate human-marked labels for salient regions. For consistency in these experiments, we chose  $\lambda = 0.15$  in all trials to weight the influence of contour smoothness.

**4.1. Comparison and Evaluation.** Firstly, to measure the segmentation performance of LRACSEP algorithm comprehensively, we compare LRACSEP algorithm with the Grabcut [22] algorithm using more saliency maps, that is, the abovementioned 9 state-of-the-art saliency maps. Grabcut is very useful for image segmentation and one can get satisfactory results when giving a very informative input. It enabled users to roughly annotate (e.g., using a rectangle) a region of interest and then use Grabcut to extract a precise image mask. To automatically initialize Grabcut, we use a segmentation obtained by binarizing the saliency map using a fixed threshold. We set the threshold to 0.3 empirically. Once initialized, we iteratively run Grabcut 4 times to improve the segmentation result. Figure 11 shows the comparison results.

Here, we use the precision, recall, and  $F$ -measure to evaluate the performance of our proposed model. Given a ground-truth segmentation map  $G = [g_1, g_2, \dots, g_n]$  and the detected segmentation map  $S = [s_1, s_2, \dots, s_n]$  for an image, we have

$$\begin{aligned} \text{precision} &= \frac{\sum_x g_x s_x}{\sum_x s_x}, \\ \text{recall} &= \frac{\sum_x g_x s_x}{\sum_x g_x}. \end{aligned} \quad (11)$$

$F$ -measure, a harmonic mean of precision and recall, is a measure that combines precision and recall. It is calculated as follows:

$$F_\beta = \frac{(1 + \beta) \text{precision} \times \text{recall}}{\beta \times \text{precision} + \text{recall}}, \quad (12)$$

where  $\beta$  is a positive parameter to decide the importance of precision over recall in computing the  $F$ -measure.

We use  $\beta = 0.3$  [19] in our work for fair comparison. The segmentation performance is compared in Figure 11. It is shown in the figure that the proposed method significantly outperforms the abovementioned 9 models with respect to precision, recall, and  $F$ -measure.

As seen in Figure 11, the Grabcut using RC saliency map is better than other saliency maps based Grabcut. For the convenience of visual inspection of the segmentation performance, the LRACSEP method is compared with the Grabcut on RC saliency map on a group of images (see Figure 12). As shown in Figure 12, the Grabcut on RC saliency map yields high false-positive (i.e., the background areas misclassified to object areas) and false-negative (i.e., the object areas misclassified to background areas) rates. In

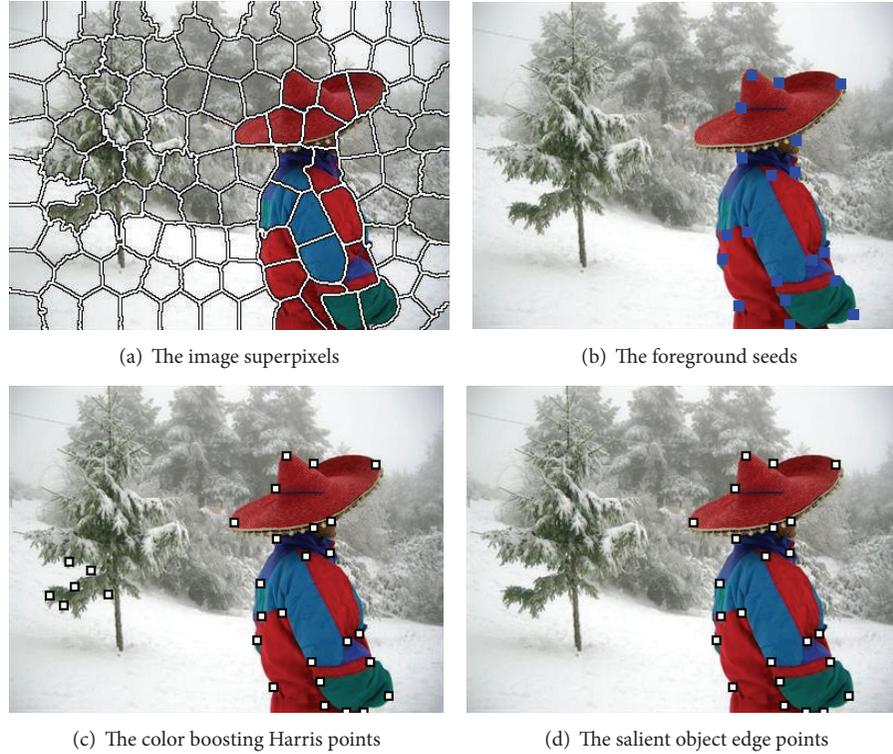


FIGURE 7: The salient object edge points detection.



FIGURE 8: The initial contour of our model. (a) The convex hull which contains the salient points in Figure 7(d) and (b) the contour of the obtained convex hull as the initial contour.

contrast with that, the proposed algorithm robustly works even with complicated cluttered background. Such favorable segmentation results can be achieved since we use localizing region-based active contour model which can achieve subpixel accuracy of object boundaries. Additionally, for the Grabcut on RC saliency map, the performance of saliency map affects the final segmentation result.

We secondly measure the segmentation performance of the proposed algorithm, as compared with existing competitive automatic salient object segmentation methods, such as Fu's method [6] and Achanta's segmentation method [7]. Figure 13 shows the segmentation performance of the three methods. It is shown in the figure that the proposed method

significantly outperforms the state-of-the-art algorithms with respect to precision, recall, and  $F$ -measure.

**4.2. The Comparison of Iteration Times.** To verify the effectiveness of our method, we compare LR with the abovementioned two state-of-the-art algorithms: Fu's method [6] and Achanta's segmentation method [7]. The average numbers of iterations are depicted in Figure 14. It can be observed that our method is more efficient. The reason for the advantage of our method is that our method makes use of the salient edge points, while the other two methods are based on the saliency maps. The computation of saliency map is consuming.

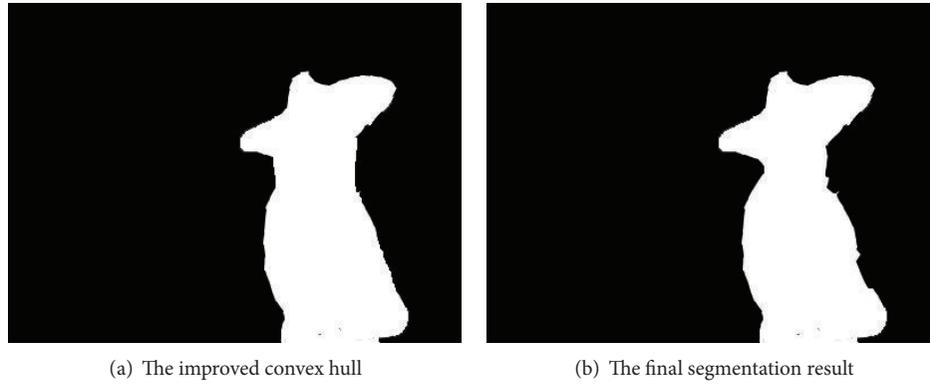


FIGURE 9: The segmentation results of our model.

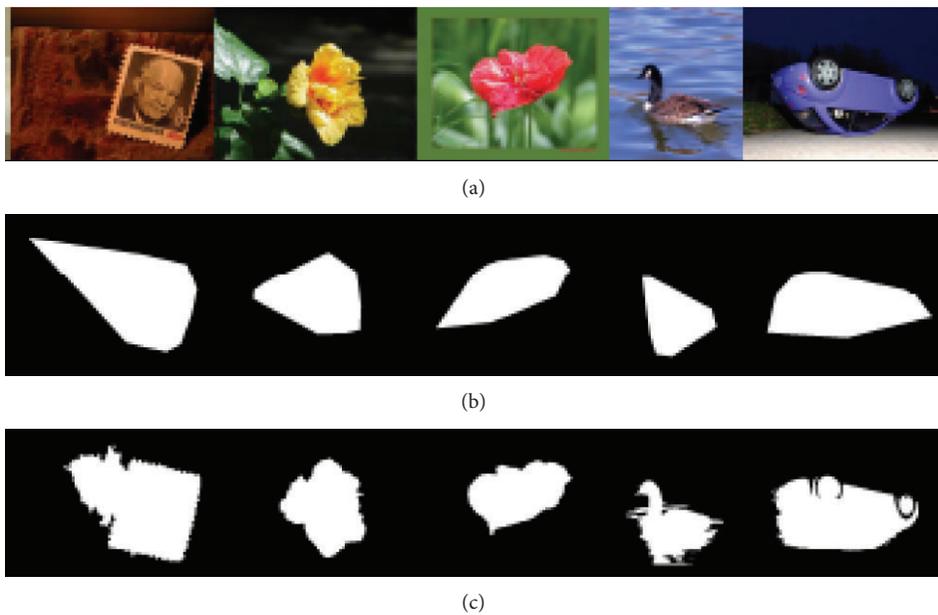


FIGURE 10: Visualization of the improved convex hulls. The images from top to bottom are the original input images, the initial convex hulls  $M_i$ , and the improved convex hulls  $M_R$ .

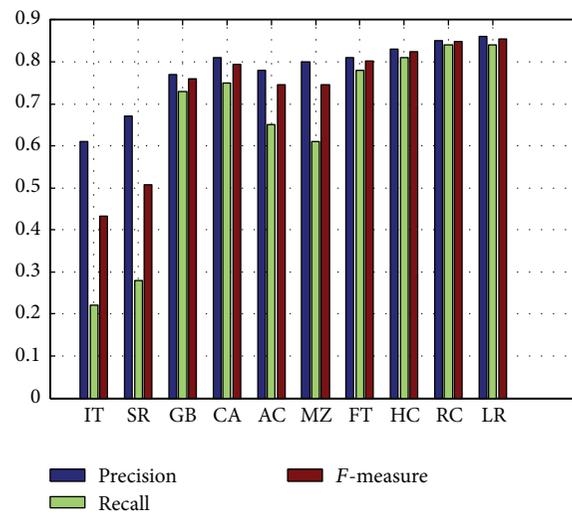


FIGURE 11: Precision-recall bars for the proposed algorithm and the Grabcut using different saliency maps. Our method, LRACSEP (LR), shows high precision, recall, and  $F$ -measure values over the 1000-image database.

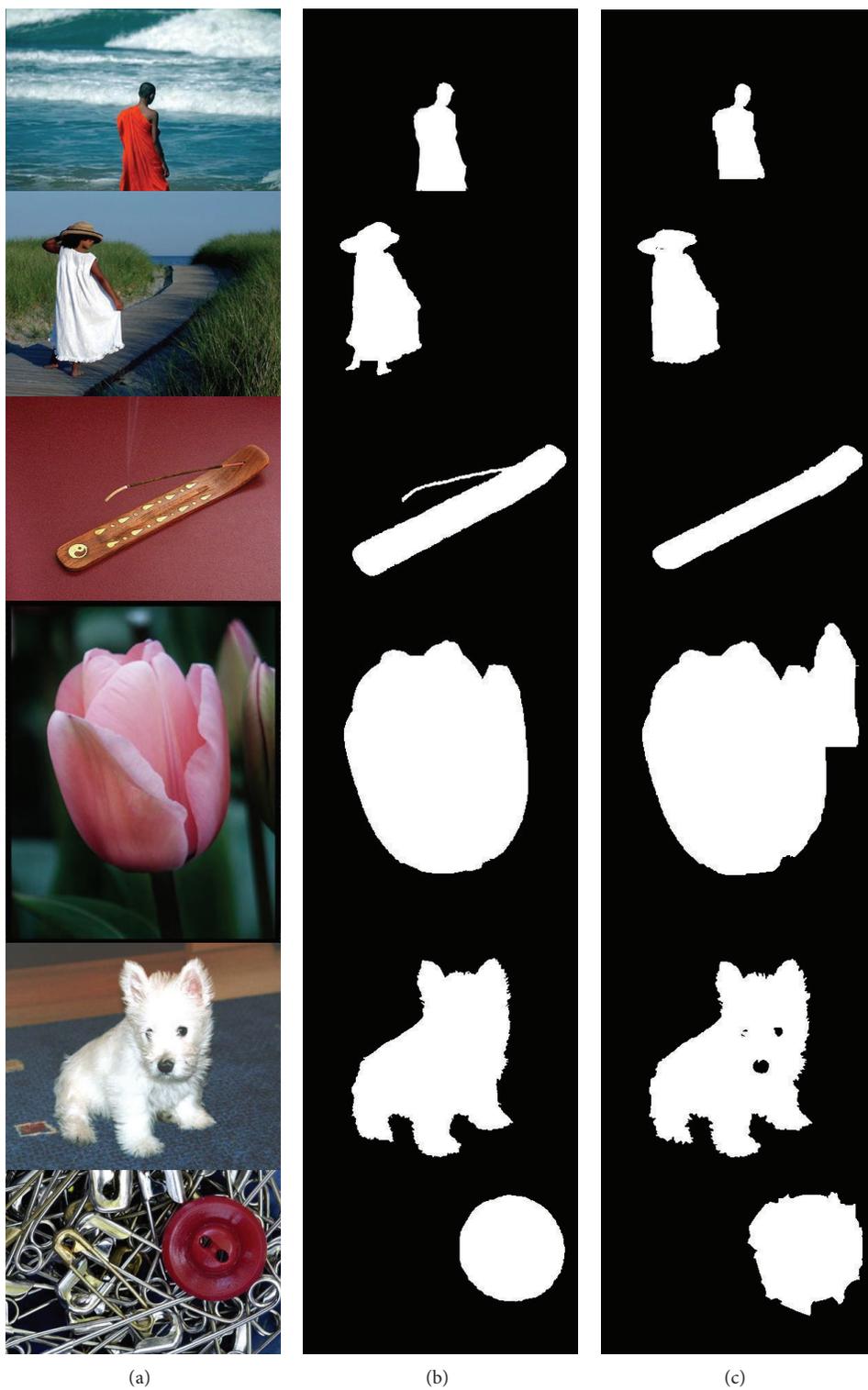


FIGURE 12: The segmentation results by LACSEP and the Grabcut on RC saliency map. (a) Original image, (b) LACSEP, and (c) the Grabcut on RC saliency map.

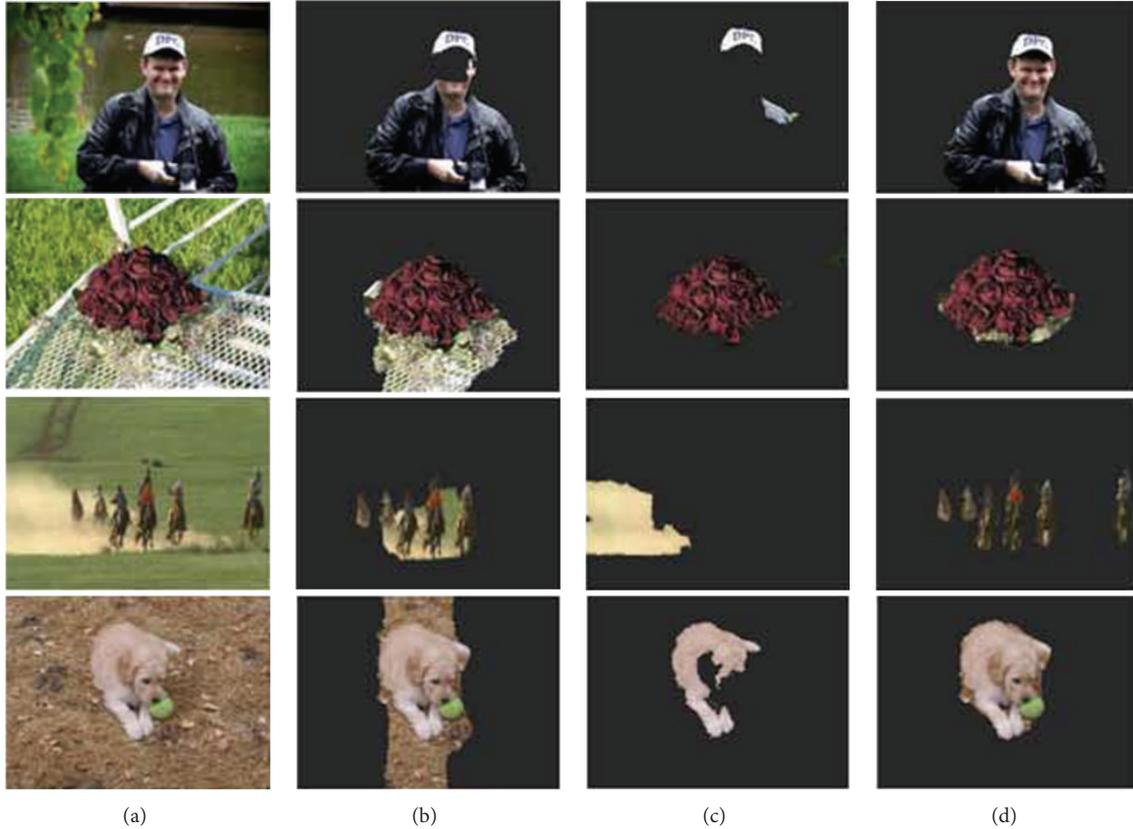


FIGURE 13: Results of object segmentation. The leftmost is the original image. The segmentation results from the second left to right are obtained from [6, 7] and the proposed algorithm.

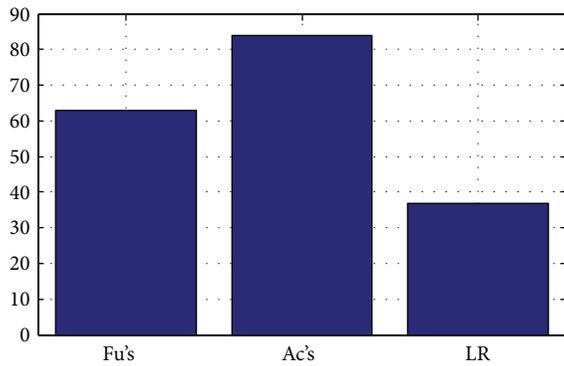


FIGURE 14: The number of iterations comparison between the two different initialization methods.

## 5. Conclusions and Future Work

In this paper, we propose a novel automatic approach to extract interesting objects from natural images. This approach uses the salient edge points as the prior knowledge. It makes the original semisupervised segmentation method LRAC become unsupervised. Our main contributions are threefold: the first is that the core saliency map is proposed to determine the foreground seeds; the second is that salient object edge points are detected by the foreground seeds; the

last is that the proposed framework can apply any active contour model to segment the salient object automatically. From the experimental results, our method is better than several state-of-the-art saliency-based segmentation methods on the public database. In contrast with existing interactive segmentation approaches that require considerable user interaction, the proposed method does not require it; that is, the segmentation task is fulfilled in a fully automatic manner.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work is sponsored by the National Natural Science Foundation of China (NSFC) no. 61402192, JiangSu Qing Lan Project, Six Talent Peaks project in Jiangsu Province, Natural Science Foundation of the Jiangsu Higher Education Institutions of China (Grant no. 14KJB520006), Jiangsu 333 Project, the Science & Technology Fund of Huai'an under the Grants nos. HAG2013059, HANZ2014006, and HAG2014028, the open fund of Jiangsu Provincial Key Laboratory for Advanced Manufacturing Technology (HGAMTL-1401), and

the open fund of Jiangsu Provincial Key Laboratory for Interventional Medical Devices (JRI405).

## References

- [1] A. Kovacs and T. Sziranyi, "Harris function based active contour external force for image segmentation," *Pattern Recognition Letters*, vol. 33, no. 9, pp. 1180–1187, 2012.
- [2] R. Ksantini, B. Boufama, and S. Memar, "A new efficient active contour model without local initializations for salient object detection," *EURASIP Journal on Image and Video Processing*, vol. 2013, article 40, 2013.
- [3] E. N. Mortensen and W. A. Barrett, "Intelligent scissors for image composition," in *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*, pp. 191–198, ACM, 1995.
- [4] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proceedings of the 8th International Conference on Computer Vision*, pp. 105–112, July 2001.
- [5] S. Lankton and A. Tannenbaum, "Localizing region-based active contours," *IEEE Transactions on Image Processing*, vol. 17, no. 11, pp. 2029–2039, 2008.
- [6] Y. Fu, J. Cheng, Z. Li, and H. Lu, "Saliency cuts: an automatic approach to object segmentation," in *Proceedings of the IEEE International Conference on Pattern Recognition*, pp. 1–4, Tampa, Fla, USA, December 2008.
- [7] R. Achanta, S. Hemamiz, F. Estraday, and S. S¸usstrunky, "Frequency-tuned salient region detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pp. 1597–1604, Miami, Fla, USA, June 2009.
- [8] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proceedings of the IEEE 8th International Conference on Computer Vision (ICCV '01)*, vol. 1, pp. 105–112, IEEE, Vancouver, Canada, July 2001.
- [9] W. Wen, C. He, M. Li, and Y. Zhan, "Adaptively active contours based on variable exponent  $L^{p(|v|)}$  norm for image segmentation," *Mathematical Problems in Engineering*, vol. 2012, Article ID 490879, 20 pages, 2012.
- [10] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive Psychology*, vol. 12, no. 1, pp. 97–136, 1980.
- [11] J. M. Wolfe and T. S. Horowitz, "What attributes guide the deployment of visual attention and how do they do it?" *Nature Reviews Neuroscience*, vol. 5, no. 6, pp. 495–501, 2004.
- [12] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," *Human Neurobiology*, vol. 4, no. 4, pp. 219–227, 1985.
- [13] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.
- [14] Y.-F. Ma and H.-J. Zhang, "Contrast-based image attention analysis by using fuzzy growing," in *Proceedings of the 11th ACM International Conference on Multimedia (MM '03)*, pp. 374–381, November 2003.
- [15] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS '06)*, pp. 545–552, 2006.
- [16] T. Liu, Z. Yuan, J. Sun et al., "Learning to detect a salient object," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, pp. 353–367, 2011.
- [17] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 2376–2383, San Francisco, Calif, USA, June 2010.
- [18] Y. Zhai and M. Shah, "Visual attention detection in video sequences using spatiotemporal cues," in *Proceedings of the 14th Annual ACM International Conference on Multimedia (MULTIMEDIA '06)*, pp. 815–824, ACM, 2006.
- [19] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 409–416, June 2011.
- [20] J. van de Weijer, T. Gevers, and A. D. Bagdanov, "Boosting color saliency in image feature detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 150–156, 2006.
- [21] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference*, vol. 15, pp. 147–151, Manchester, UK, September 1988.
- [22] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut -interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.