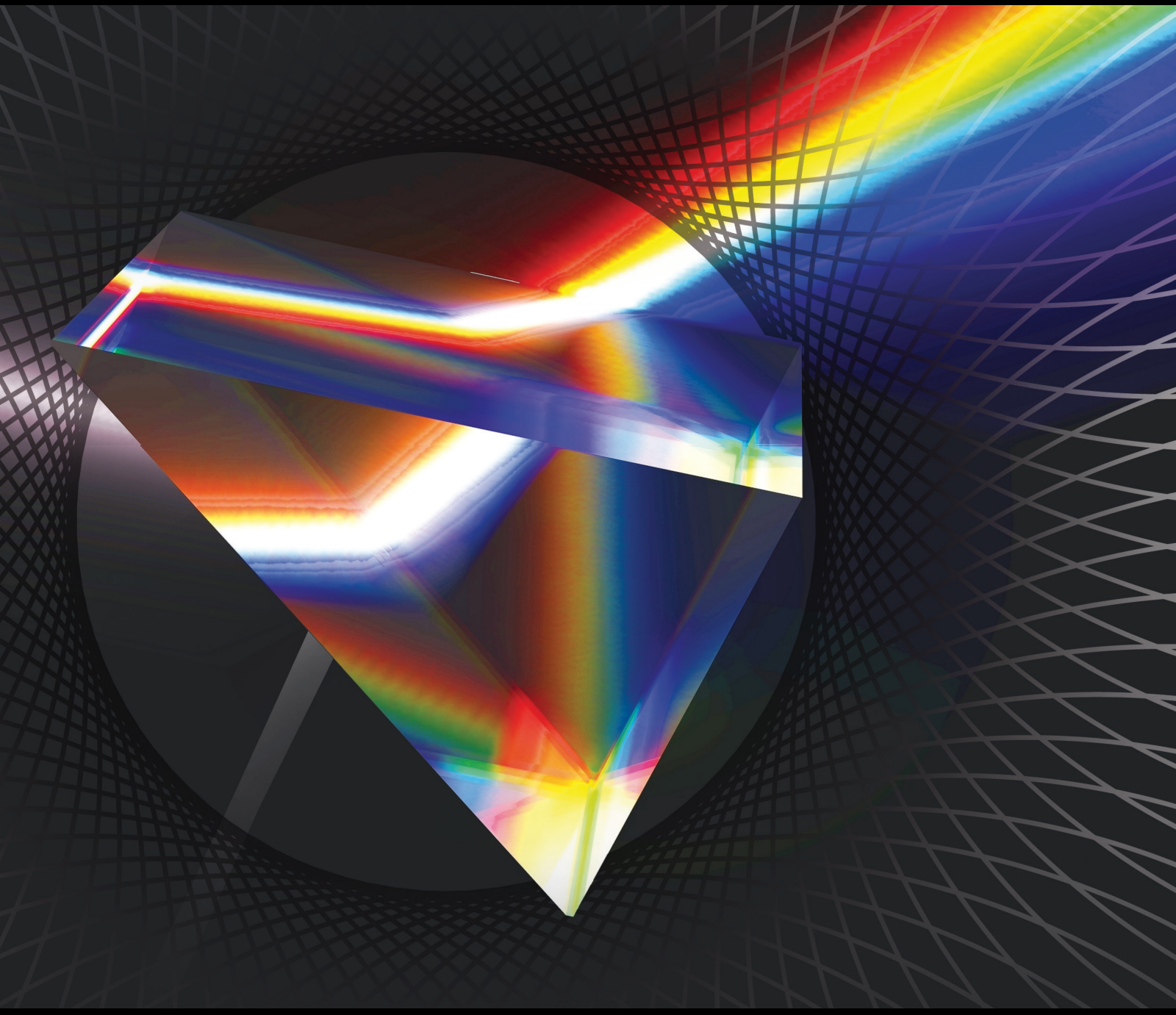
International Journal of Optics

Methods and Applications in Blur Detection and Classification

Lead Guest Editor: Muhammad Tariq Mahmood
Guest Editors: Abdul Majid and Aun Irtaza



Methods and Applications in Blur Detection and Classification

International Journal of Optics

Methods and Applications in Blur Detection and Classification

Lead Guest Editor: Muhammad Tariq Mahmood
Guest Editors: Abdul Majid and Aun Irtaza

Contents

Hindawi
International Journal of Optics
Volume 2021, Article ID 2785225, 13 pages
https://doi.org/10.1155/2021/2785225

Research Article

A Robust Approach for Blur and Sharp Regions' Detection Using Multisequential Deviated Patterns

Awais Khan,[1] **Ali Javed**,[1] **Aun Irtaza**,[1] **and Muhammad Tariq Mahmood**[2]

[1]*Department of Computer Science, University of Engineering and Technology Taxila, Taxila, Pakistan*
[2]*Future Convergence Engineering, School of Computer Science and Engineering, Korea University of Technology and Education, Cheonan, Republic of Korea*

Correspondence should be addressed to Muhammad Tariq Mahmood; tariq@koreatech.ac.kr

Received 23 April 2021; Revised 19 August 2021; Accepted 2 September 2021; Published 21 September 2021

Academic Editor: Sulaiman W. Harun

Blur detection (BD) is an important and challenging task in digital imaging and computer vision applications. Accurate segmentation of homogenous smooth and blur regions, low-contrast focal regions, missing patches, and background clutter, without having any prior information about the blur, are the fundamental challenges of BD. Previous work on BD has emphasized much effort on designing local sharpness metric maps from the images. However, the smooth/blurred regions having the same patterns as sharp regions make them problematic. This paper presents a robust novel method to extract the local metric map for blurred and nonblurred regions based on multisequential deviated patterns (MSDPs). Unlike the preceding, MSDP extracts the local sharpness metric map on the images at multiple scales using different adaptive thresholds to overcome the problems of smooth/blur regions and missing patches. By using the integral values of the image along with image masking and Otsu thresholding, highly accurate segmented regions of the images are acquired. We argue/hypothesize that the local sharpness map extraction by using direct integral information of the image is highly affected by the threshold selected for distinction between the regions, whereas MSDP feature extraction overcomes the limitations substantially by using automatic threshold computation over multiple scales of the images. Moreover, the proposed method extracts the relatively accurate sharp regions from the high-dense blur and noisy images. Experiments are conducted on two commonly used SHI and DUT datasets for blur and sharp region classifications. The results indicate the effectiveness of the proposed method in terms of sharp segmented regions. Experimental results of qualitative and quantitative comparisons of the proposed method with ten comparative methods demonstrate the superiority of our method. Moreover, the proposed method is also computationally efficient over state-of-the-art methods.

1. Introduction

With the exponential growth of digital image capturing devices, i.e., DSLR cameras, cellphone cameras, wearable cameras, etc., we have witnessed a massive collection of digital photos captured and uploaded on social media on a daily basis. A good quality photo must be sharp and not contain any degradation such as noise and blurred regions. As for many applications, we need to highlight the target object from the images. To make the object highlighted, many techniques are being used nowadays, e.g., using high-definition camera sensors, adding blurriness to the background objects, etc. However, the high-definition image sensors impose blurriness in the background of the images to make the foreground objects more prominent. Consequently, blurriness is being used as an editing effect that is added purposely in modern-day image capturing.

Image blurriness can be categorized into motion blur and defocus blur. The motion blur can occur due to two potential reasons: (a) when you try to capture the moving objects and (b) camera motion either intentionally or unintentionally, whereas the defocus blur usually occurs due to special effects used by the photographers to highlight the focus and out-of-focus regions in the image. It is a visual effect added by the photographer using highly sophisticated techniques to make the target object sharp and the rest of the image blur. An image contains useful information that can be used in various computer vision and image processing

applications, i.e., background tracing, text retrieval, image retrieval, person authentication, etc. However, blur affects the contrast and sharpness details of the image that made the retrieval of information challenging. Similarly, photos are used as the key evidence in a criminal investigation, where it can be very challenging to extract the immersed information about the target object(s) in the presence of high-density blurred regions. For this purpose, we need to classify the image into blur and nonblur regions initially. The information of the objects lying in the nonblur regions is more reliable in comparison with those in blurred regions, as the information can be distorted in blur regions like an increase of edge thickness. Thus, blur detection and sharpening are required for the accurate extraction of the information from the images.

Defocus blur detection (DBD) is the classification problem of intentionally added blur by the highly sophisticated modern-day digital cameras. This classification has been paid substantial attention due to their significant potential applications, e.g., object detection [1], image segmentation [2], object augmentation [3], etc. Defocus blur affects the information and the sharpness details of the image making it more challenging for objects/regions detection. The DBD without having any prior information about the blur densities, blur type, or sensor settings of the camera is a challenging task.

Existing blur image detection approaches are divided into two categories: single image detection and multi-image detection. For multi-image detection, the knowledge of the blur densities, type, sensor information of the cameras, and other additional information is required [4]. In contrast, a single image can be split into sharp and blur regions without having any prior information about the blur and the device used to capture that image [5, 6]. Moreover, the existing approaches presented for DBD can be categorized into frequency-based [7–13], depth-based [14–17], or local sharpness metric map-based for segmentation of blur and nonblur regions [6, 18–21]. In [22], Zhu et al. used the local coherent map generated by the evaluation of gradient fields of the local spectrum. However, the use of flat areas information and color edges is not enough for accurate DBD detection. In [23], Chakrabarti et al. proposed a Point Spread Function (PSF) using the local frequency analysis to obtain the segmented map for DBD. This method has the limitation of generating erroneously labeled regions of the image. Su et al. [24] presented a method called singular value decomposition (SVD) based on single thresholding on image features to detect the blurred and nonblurred regions. Similarly, Xiao et al. extended the single threshold SVD into multiscale SVD in [8]. The fusion-based method is used to overcome the smooth/blur region problems from the images. In [25], Golestaneh estimated the level of blurriness at each location using a method called high-frequency multiscale Fusion and Sort Transform (HiFST) based on gradient magnitudes.

Depth-based methods [9, 14–17] also proved to be effective in defocus blur detection using the information about the blur densities and blurry edges. In [9], Liu et al. presented different local features, i.e., association congruence, saturation, gradient histogram, and power bands to specify the type of blur from the images. In [26], a cross ensemble network is used along with a smaller defocus detector for diversity enhancement. However, this approach is computationally expensive and unable to differentiate the nonblur regions accurately in the presence of smooth regions. Furthermore, DBD measurements such as local variance, higher-order statistics, and variance of wavelength coefficient are also used in DBD with images containing a narrow depth of field (DOF) [27].

Most of the algorithms [6, 18, 19, 21] used the local sharpness metric approach for the detection of blur and nonblur regions. The local sharpness metric is like a filtering method such as energy function estimating the results based on the responses of blur energy from images. The low energy indicates the blur region, whereas the high energy represents the sharp region. In [12], Shi et al. introduced peculiar sharpness features, gradient histogram, and kurtosis span for DBD of local image regions. This method is unreliable and causes problems in the accurate detection of blur and sharp regions due to smooth homogenous regions. Zhu et al. analyze the blur from the images using PSF [22]. This method is unable to perform well in lightly blurred regions. In [21], Local Binary Patterns (LBP) are taken into consideration for defocus blur detection. The local metric map generated using the LBP segments the blur and nonblur regions but is unable to perform well in the presence of noisy images even in sharp regions. In our prior method [18], we proposed the Local Directional Mean Patterns (LDMP) to overcome the limitation of the sharp metric map in noisy situations. However, our prior method [18] is unable to detect simultaneous smooth blur and low-contrast regions.

Recently, deep learning methods have been heavily employed in various computer vision and image processing applications, i.e., saliency detection [1], semantic segmentation [2], automatic shadow detection [28] airplane detection using remote sensing images [29], ship detection from real-time images [30], vehicle detection [31], etc. The significance of deep learning algorithms is proven to be effective for defocus blur detection and segmentation, however, at the expense of increased computational cost. In [32], Kim introduced a deep learning method based on a convolution neural network (CNN) for the detection of sharp and blur regions of the image. The multiscale reconstruction loss function was used for the segmentation of blur regions. In [33], Park et al. encounter the DBD problem with patch level detection based on CNN. Unfortunately, the patch level DBD led to suppression in low-contrast regions. Tang et al. [34] proposed a Deep Neural Network (DNN) based technique Diffusion Network (DNet) that fused the refined features extracted by the networks to obtain the segmented blur and sharp regions. In [35], the author introduced global context-guided hierarchically residual feature refinement network "HRFRNet." The hierarchical features are used to enhance the final outcomes. Furthermore, a deep-guided fusion module is used for the refining process.

Accurate DBD has initiated extensive research interest from the last few years. However, it is still a significant yet

challenging computer vision problem. Although the aforementioned techniques can detect the defocus blur and nonblur regions, however, these approaches fail in certain cases, i.e., the presence of smooth blurry regions, missing sharp patterns, low contrast, etc. The DNN methods performed well for DBD, but all of these methods are computationally more complex and required high computational resources, i.e., GPUs, memory, etc. We aim to develop a robust method for DBD that can effectively extract the sharp targeted regions of the image in the presence of noise, smooth/blur, and low-contrast images. To address the aforementioned problems, we propose an efficient and robust Multisequential Deviated Patterns (MSDP) for accurate sharp region extraction from images at multiple scales. The extracted multiscale sharpness maps are further fused to get the refined map of the image. We used Otsu thresholding to segment the extracted sharp region into comparable binary representation. The proposed method is efficient due to using the local integral values of the images directly instead of using time-consuming matting approaches used by the preceding methods to segment the binary images. The major contributions of the proposed work are as follows:

(i) We propose efficient and robust multisequential deviated patterns for accurate blur detection from high-density blur and noisy images

(ii) For feature computation, we extract the sharpness metric using adaptive thresholding on multiple image scales to overcome the smooth blur and missing sharp region problem of manual thresholding

(iii) For image segmentation, we fused the multiscale sharpness maps extracted from MSDP along with the image masking

(iv) Rigorous experiments were performed against several state-of-the-art methods over the latest DUT and SHI datasets to prove the effectiveness of the system

The rest of the paper is organized as follows. Section 2 presents the details of the proposed method. Section 3 provides the discussion on results of different experiments conducted to evaluate the performance of our method. Finally, Section 4 concludes our work.

2. Methods

This paper presents a novel method based on the integral use of the image, which detects the blur and sharp regions from high-dense blurry and noisy images. The proposed method used the local window of different sizes based on the input image scale to extract the sharp regions in the image. Firstly, the image is divided into three different scales S_1, S_2 and S_3. Secondly, MSDP is used to extract a sharpness map from each scaled image using an adaptive threshold. Lastly, image masking is used over the extracted sharpness maps, and fusion is performed to produce a more accurate single sharpness map. Next, Otsu thresholding is used to retrieve the accurate binary images for comparison with state-of-the-art methods. The flow of the proposed method is shown in Figure 1.

2.1. Preprocessing and Image Scaling. The extraction of features using integral values of the image is challenging due to the influence of different factors, i.e., color, camera, object detail, etc., on the values. For accurate extraction of sharp regions, we used the RGB input image I_b consisting of sharp and blur regions. First, we convert the RGB color image into grayscale and apply the two-dimensional median filtering to reduce the noise.

$$\begin{aligned} I_b &= P_S + P_B, \\ I_g &= rgb2gray(I_b), \\ \mathrm{Im}_F &= medfilt2(I_g), \end{aligned} \tag{1}$$

where P_S and P_B denote the sharp and blur pixel values of the image, and I_g represents the grayscale image. Im_F represents the 2D median filtered image with reduced noise obtained after applying the 2D median filtering function ($medfilt2$). Next, we represent the image into three different scales (S_1, S_2 and S_3) after employing the $medfilt2$.

2.2. Multisequential Deviated Patterns (MSDPs). Selecting an appropriate threshold for integral extraction is a complicated task. For example, the use of a high threshold in local feature extraction leads to exclusion of low/lesser sharp regions, whereas the selection of low threshold value causes the inclusion of additional useless details in the features, i.e., background object and noise, etc. The relationship of high and low thresholds in integral feature extraction is shown in Figure 2. For MSDP maps, we computed the upper and lower patterns of the image. In integral extraction of features, the three-level thresholding is proven effective [18] as compared to the two-level thresholding [21]. Moreover, two-level thresholding is not effective in the presence of a high density of noise in the images. Therefore, we employ 3-level thresholding for the extraction of sharp integral upper and lower features of the images. We extract the upper and lower features as follows:

$$\begin{aligned} ULP &= \sum_{P=1}^{8} 3^p s(im_p - im_c), \\ \delta &= \sqrt{\frac{\sum (q_i - \mu)^2}{N}}, \end{aligned} \tag{2}$$

$$P(im_p, im_c, \delta) = \begin{cases} 1 \text{ if} & im_p \geq im_c + \delta, \\ -1 \text{ if} & im_p \leq im_c - \delta, \\ 0, \text{ if} & im_p > im_c - \delta \,\&\, im_p < im_c + \delta, \end{cases} \tag{3}$$

where 3^p and q_i represent the 3-level thresholds and integral values of the pixels, N and μ denote the total number of

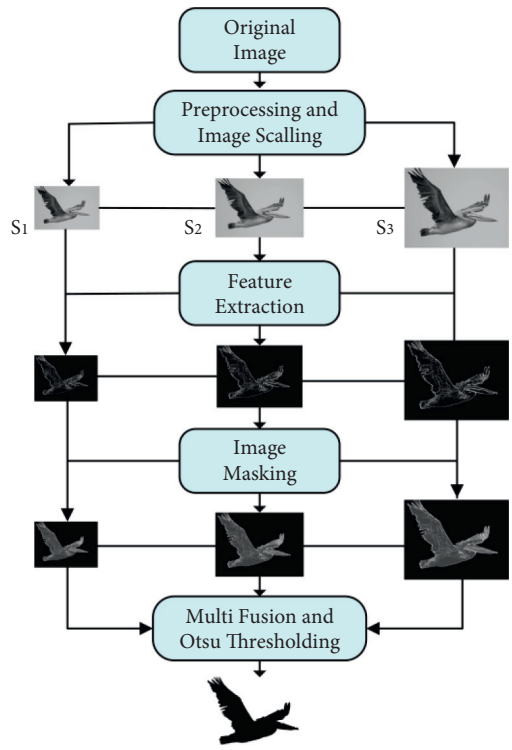


Figure 1: Flow diagram of the proposed method.

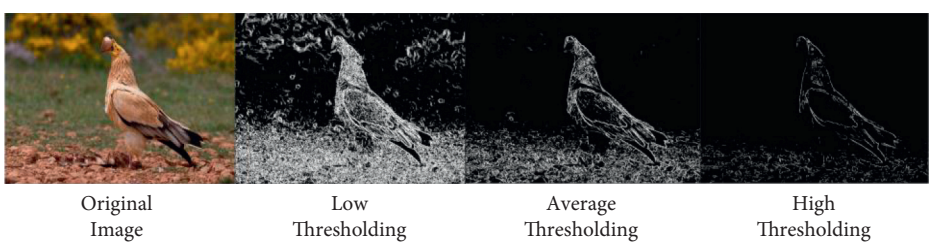


Figure 2: Impact of the threshold in local integral pattern extraction.

integral and mean values, and im_p and im_c are the neighboring and center pixels of the image windows, while δ represents the adaptive threshold, which is calculated by the sequential deviation of the existing window as shown in equation (2). Instead of adopting the threshold selection approach of the existing methods, we computed an adaptive threshold for each region of the image by rotating the extraction window over the image. The adaptive threshold is computed automatically by adding and subtracting the center pixel value im_c of the window along with the standard deviation δ as shown in equation (3). An adaptive threshold based on δ and im_c is responsible for the extraction of sharp pixels from the images; i.e., high threshold value leads to highly sharp regions, and low threshold value leads to the inclusion of noise and other unwanted content. In contrast, the preceding methods mostly used a hard-coded threshold value in their algorithms, which makes these methods unable to effectively extract the low-dense sharp regions locally [18, 21]. Consequently, we applied three-level thresholding with an adaptive threshold for the extraction of three values including 1, −1, and 0 from the image. For instance, as shown in Figure 3, if a 3×3 window is used over the image integral values having a central pixel value (im_c) of 23 with the deviation of the neighboring pixel (δ) of 10.2, then the range of extraction lies between 13.2 and 33.2, which is shown as $im_p > im_c - \delta \& im_p < im_c + \delta$ in equation (3). For three-level value extraction from the image, the neighboring pixel in the window lies between 13.2 and 33.2 and is converted into "0," whereas the integral value of 1 is assigned to values greater than the threshold $im_c + \delta$ (33.2 in Figure 3), and −1 is assigned to the values below the threshold $im_c - \delta$ (13.2 in Figure 3). After replacing the integral values of the images, we obtained $P(im_p, im_c, \delta)$, which contains three-level values comprising 1, 0, and −1. The overall extraction of three-level values is shown in Figure 3. For instance, the image with Qn window from the integral values of pixels where n represents the 3×3 window having 9 values is shown in equation (4).

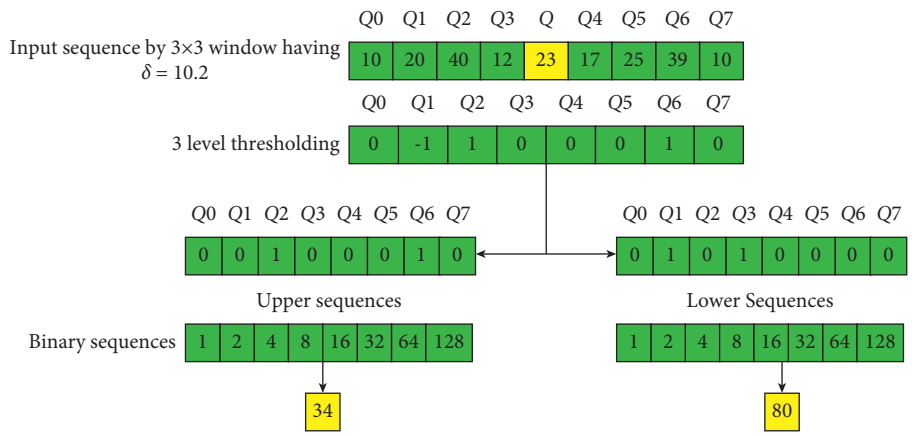


Figure 3: Extraction of upper and lower binary sequences.

$$\begin{aligned} \mathrm{Im}_F &= \sum_{i=1}^{n} Qn_{e\times e}, \\ Q_{3\times 3} &= \{Q0, Q1, Q2 \ldots Q7\}, \end{aligned} \tag{4}$$

where e denotes the dimension of the rotating window. In order to compute the sharp region, the upper and lower features of the image have to be computed separately. However, the proposed $P(im_p, im_c, \delta)$ extracts the combined features of the image. Therefore, to reduce the noise in the images, the three-level extracted patterns are further converted into two levels, i.e., upper and lower image patterns. For two levels of extraction, we need to replace the negative values from the obtained $P(im_p, im_c, \delta)$. For extracting the upper patterns of the image, we converted all -1 values from $P(im_p, im_c, \delta)$ into 0. And for lower patterns, 1 and -1 are replaced with 0 and 1 as shown in Figure 3. Ultimately, the resultant upper and lower patterns of the image are converted into binary bit streams using equation (5) that are further represented into their equivalent decimal values.

$$\begin{aligned} Upp_F &= \left\{0 \Leftrightarrow \text{if} \quad P\left(im_p, im_c, \delta\right) = -1\right\}, \\ Low_F &= \begin{cases} 1 \Leftrightarrow \text{if} \quad P\left(im_p, im_c, \delta\right) = -1, \\ 0 \Leftrightarrow \text{if} \quad P\left(im_c, im_c, \delta\right) = 1, \end{cases} \end{aligned} \tag{5}$$

where Upp_F denotes the upper features of the image, and Low_F represents the lower features. The $Z \times Z$ window is used to obtain the decimal values of upper and lower patterns using equation (6), where Z denotes the size of the window varying for each scale of the image. At last, we need to pick the sharp and blur patterns smartly by retrieving only the sharp pixels and neglecting the blur once. The proposed method uses the deviation of the pixels twice to observe the change in the patterns of blur and sharp regions. We computed the standard deviation of two-level patterns consisting of the upper and lower patterns of the image obtained from the last step. For this purpose, first, we have to convert the binary two-level patterns into their equivalent decimal numbers by using the window $Z \times Z$. All the neighboring pixels values in the window covert into their equivalent decimal number as follows:

$$\begin{aligned} \mathrm{Im}_{upp} &= \sum_{k=1}^{8} Upp_F\left(S_P\right) \times 2^{k-1}, \\ \mathrm{Im}_{low} &= \sum_{k=1}^{8} Low_F\left(S_P\right) \times 2^{k-1}, \end{aligned} \tag{6}$$

Im_{upp} and Im_{low} are the upper and lower patterns of equivalent decimal numbers from the extracted two-level binary patterns from equation (5). After that, we computed the deviation of the neighboring pixels again with decimal values of the upper and lower patterns computed from equation (6). The values higher than the deviation (δ) are considered as the sharp region values and retained, while the rest are neglected for being the blurry region. The extraction of Multilayered Sequential Pattern is shown in equations (7) and (8).

$$\begin{aligned} \delta_{upp} &= \sqrt{\sum (qi - \mu)^2 / N}, \\ \delta_{low} &= \sqrt{\sum (qi - \mu)^2 / N}, \end{aligned} \tag{7}$$

$$MSDP = \left\{\mathrm{Im}_{upp} > \delta_{upp} \oplus \mathrm{Im}_{low} > \delta_{low}\right\}, \tag{8}$$

where δ_{upp} and δ_{low} represent the deviation value of upper and lower patterns of the image, respectively. Finally, the sharpness map of sharp regions is extracted as $MSDP$ shown in equation (8). Moreover, we extracted three $MSDP$ sharpness maps for each scale of the image (S_1, S_2 and S_3) as shown in Figure 4.

2.3. Image Masking. The extracted features $MSDP$ contain only the sharp regions of the images at three scales S_1, S_2 and S_3. However, there are some regional variations among the images at all scales caused by the adaptive thresholding and different dimensions. The main reason for this variation is missing regions among the extracted maps. The empty integral values in the patterns usually occur due to the noise or background objects that ultimately lead to missing regions in the image. We applied a morphological operation on this image to fill the holes and gaps between the pixels. More

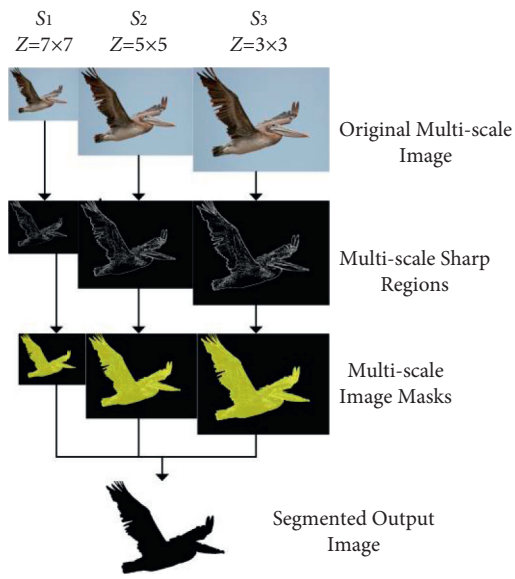


Figure 4: Multiscale image masking and Otsu thresholding.

specifically, we employed a binary filling operation to fill the gaps and holes using equation (9). Similarly, this filling process is applied at every scale (S_1, S_2 and S_3) of the image. In addition, we created an image mask containing sharp regions of the image before applying the Otsu thresholding for segmentation. For this purpose, we applied the Otsu global thresholding to select the sharp regions from the obtained image.

$$Trimap_{IM} = BF(MSDP). \tag{9}$$

2.4. Multifusion and Otsu Thresholding. In the last phase, we fused the scales (S_1, S_2 and S_3) into a single sharpness map and employed the Otsu thresholding [36] for binarization of the images. The masked image obtained from the previous phase is converted into a segmented binary image of the sharp and blur regions. The variance-based thresholding is used along with linear discriminant principles to segment the target object (foreground) from the heterogeneous and diverse background regions. The threshold for segmentation in Otsu thresholding is based on the variation of the integral values. The extracted patterns in $Trimap_{IM}$ are divided into two classes, i.e., sharp region and the background blur region. The global threshold is computed according to the variance of the classes. The regions with values higher than the threshold value are selected, whereas the regions below the threshold value are ignored. Finally, the highly segmented binary image with the sharp object as foreground and the black background is obtained as follows:

$$\begin{aligned} B_{img} &= c_0(\varepsilon)v_0^2(\varepsilon) + c_1(\varepsilon)v_1^2(\varepsilon), \\ c_0(\varepsilon) &= \sum_{j=1}^{rw-1} i(img), \\ c_1(\varepsilon) &= \sum_{j=\varepsilon}^{cl-1} i(img), \end{aligned} \tag{10}$$

where c_0 and c_1 are the integral values of sharp and blur region classes separated by a threshold ε, whereas v_0^2 and v_1^2 denote the variance of the classes. Overall, we extracted the MSDP from images at multiple scales (S_1, S_2 and S_3) using $Z \times Z$ windows of multiple sizes (i.e. 3×3, 5×5, 7×7). The reason behind the extraction of the same *MS DP* at different scales (S_1, S_2, S_3) is to overcome the missing region problem from the extracted sharp regions. The sharpness map is extracted from single *MSDP* image containing some missing areas inside the sharp regions, whereas the extraction of sharpness map over different scales overcomes this problem to some extent as discussed in Section 4. The intensity of the blur varies in each region of the image; i.e., some regions are highly affected by the blur, whereas regions far away from the sharp objects are less affected. Therefore, the extraction of the integral value is highly affected by the selected threshold value for classification of pixels as sharp and blur. We employed an adaptive threshold calculated from the deviated ratio of the patches along with the central pixel values of the regions. The combination of local central pixel values of the regions and the overall deviation between the pixels for thresholding make our method robust in the extraction of the sharp regions from the blurry images. The overall computation process at multiple scales of the image is shown in Figure 4.

3. Experimental Results

This section provides a discussion on the results of different experiments performed to measure the performance of the proposed method. We have provided a detailed comparison of qualitative and quantitative results along with the analysis of the computational complexity of our method. The details of the datasets and evaluation metrics are also presented in this section.

3.1. Datasets. The performance of our method is evaluated on two standard and commonly used datasets (Shi and DUT) for blur detection. Shi dataset [12] is the first public dataset collected and evaluated for blur detection [12]. This dataset [12] is used by almost all state-of-the-art descriptors to show the effectiveness of the methods. Shi dataset [12] contains 1000 blur images, where 704 images are partially blurred using defocus, and the rest of the images contain the motion blur. Additionally, the manually annotated ground truth images are also available along with the blur images. Most of the images in the Shi dataset are of 640×427 resolution.

DUT dataset [37] is the second commonly used publicly available dataset consisting of 500 images with defocus blur. Similarly, this dataset is also provided with manually annotated ground truth images. In comparison with the Shi dataset [12], the DUT dataset [37] is more challenging for blur detection and segmentation for various reasons; i.e., many of the images contain homogeneous smooth blur regions, have cluttered background, and are low in contrast images.

3.2. Evaluation Metrics. Three standard and commonly used metrics including precision, recall, and $F1$-score are used to evaluate the performance of the proposed method. We selected these metrics as adopted by the comparative methods for performance evaluation. We calculated the precision and recall as

$$pre = \frac{\mathrm{Im}_E \cap \mathrm{Im}_G}{\mathrm{Im}_E}, \quad rec = \frac{\mathrm{Im}_E \cap \mathrm{Im}_G}{\mathrm{Im}_G}, \tag{11}$$

where Im_E represents the pixels within the detected sharp areas, and Im_G corresponds to the pixels in the manually annotated ground truth image. Similarly, $F1$-score is computed as

$$F1 = 2 \times \frac{pre * rec}{pre + rec}, \tag{12}$$

where pre and rec denote the precision and recall of the proposed method.

3.3. Performance Evaluation of the Proposed Method. The objective of this experiment is to evaluate the effectiveness of the proposed method for blur detection on two diverse datasets. For this purpose, we computed the results of our method on the images of Shi [12] and DUT [37] datasets separately and reported the results in Table 1. In the case of Shi [12] dataset, the proposed method dominates in every comparison, i.e., quantitative analysis, qualitative analysis, and PR curve. However, in DUT [37] dataset, the precision of the proposed system slightly deteriorates. The DUT [37] dataset is more challenging than Shi dataset [12] due to exceeding homogenous and low cluttered regions. The homogenous regions are always difficult to locate in the local extraction of the regions. Although the precision of the proposed system is a little low, however, the other quantitative results (recall, $F1$-score, and computational cost) prove the effectiveness of the method.

3.4. Performance Comparison of the Proposed and Existing Methods. The objective of this experiment is to measure the robustness of the proposed method for blur detection over state-of-the-art methods. For this purpose, we have provided both the qualitative and quantitative analyses of the proposed and comparative approaches.

3.4.1. Qualitative Comparative Analysis. This experiment is designed to show the qualitative analysis of the proposed and comparative methods on Shi [12] and DUT [37] datasets. The visual quality of processed images is presented in Figures 5 and 6 to show the effectiveness of the proposed method over state-of-the-art methods. From the images depicted in Figures 5 and 6, we can observe that the proposed method can detect highly accurate results from Shi [12] and the DUT [37] datasets. Specifically, in Figure 6, we can see that images of DUT dataset [37] contain more homogenous smooth regions and low local cluttered regions that are effectively classified using our proposed method.

3.4.2. Quantitative Comparative Analysis. This experiment is designed to evaluate the performance of our method in terms of quantitative analysis. For this purpose, we used three standard metrics, i.e., precision, recall, and $F1$-score, to measure the performance of our system against the comparative methods. The precision-recall (PR) curve is calculated on the results of the proposed method from Shi [12] and DUT [37] datasets and presented in Figures 7 and 8, respectively. A separate PR curve comparison is shown for Shi [12] and DUT [37] datasets. The PR curves demonstrate that our method consistently outperforms all the comparative methods. On the other hand, our method effectively addresses the problems of homogenous smooth and low local cluttered regions in the images of a more challenging DUT dataset [37]. The PR curve of the proposed method on DUT [37] dataset continuously dominates throughout the period as shown in Figure 7.

Additionally, the $F1$-score is measured and compared for both the Shi [12] and DUT [37] datasets as shown in Figures 9 and 10. Although DNet [34] achieves almost comparable results as of the proposed method, however, our method is computationally very efficient over the DNet method [34] as shown in Table 2. This comparative analysis illustrates the superiority of the proposed method for defocus blur detection over the comparative approaches.

3.4.3. Computational Cost Analysis. Although DFD methods must be effective in terms of detecting blurred regions from the images, however, producing such accurate results in a minimum time is also crucial, especially in real-time applications. This experiment is designed to evaluate the efficiency of the proposed and comparative approaches for defocus blur detection. The results of this comparative analysis of time complexity for both the detection and segmentation are provided in Table 2. The proposed method not only dominates in terms of accurate blur detection over the state-of-the-art methods, but also executes exceptionally fast and computationally very efficient over comparative approaches. From the results, we can clearly observe that the proposed method has the 2nd lowest computational cost after LBP [21] as compared to all comparative methods. More precisely, [21] performs the best by achieving 5 seconds to segment an image into blur and sharp regions, whereas our method performed second best and achieved the time complexity of 7 sec. On the contrary, [22] performed the worst by taking the highest computational cost of 12 minutes. The main reason behind achieving such low time complexity of our method is the direct extraction of the local sharpness metric and removal of the time-consuming matting procedures used by several competitive methods. Moreover, image masking and Otsu thresholding used to produce the segmented maps are also very fast in the execution. In our method, the multiscale inference phase consumes the majority of the execution times.

Table 1: Precision, recall, and $F1$-score of the proposed method.

	Precision	Recall	$F1$-score
Shi dataset	0.91	0.90	0.92
DUT dataset	0.89	0.91	0.88

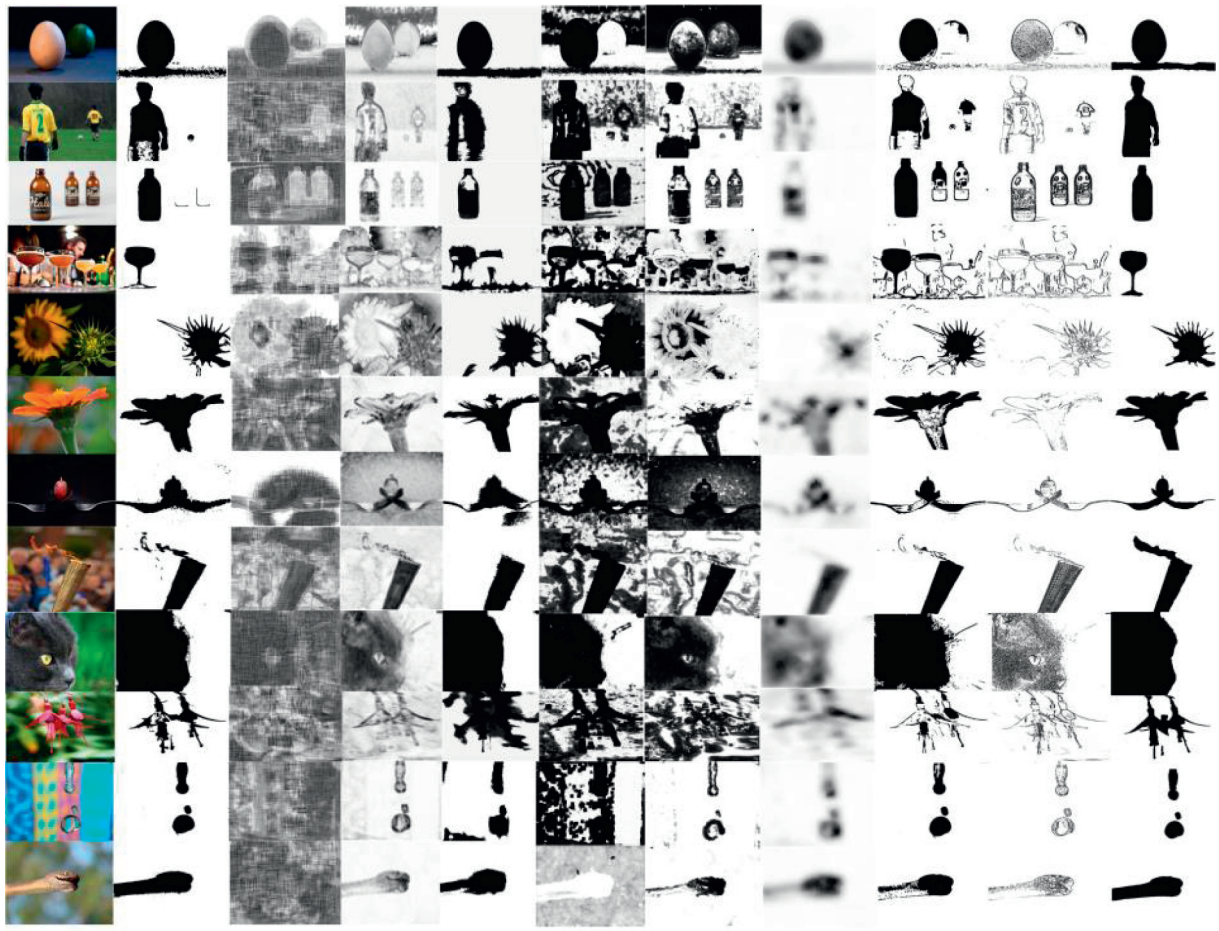


Figure 5: Visual comparison of the proposed method against comparative methods on the Shi dataset.

For this comparative analysis, we used the codes of comparative methods that are publicly available along with our own implementation. In the proposed method, the original image was scaled into 256×256, 128×128 and 64×64 dimensions for S_1, S_2 and S_3. Additionally, the window size $Z \times Z$ is selected as 3×3, 5×5 and 7×7 for image scale S_1, S_2 and S_3 respectively. The final values of window size Z for the specific scale Sn are selected after the detailed observations and experiments. Moreover, automatic global thresholding is used in image masking and Otsu thresholding for binarization of the images. We have implemented the proposed and comparative methods on Intel(R) Core (TM) m3-7Y30 CPU @ 1.00 GHz, 1.61 GHz with 8 GB memory system.

4. Discussion

The present study analyzes the findings about the selection of adaptive threshold and the impact of neighboring pixels in local extraction of the image regions. Experimental results demonstrate two facts. First, the global or hard-coded fix threshold value is not reliable for all types of images. Second, the neighboring pixels used to differentiate the integral values have a major impact on the extracted patterns. This is an important finding in the understanding of the local patterns and direct integral extraction from the images. Some sample results from LBP [21] and LDMP [18] are shown in Figure 11 to defend this fact. Figure 11 clearly demonstrates that the

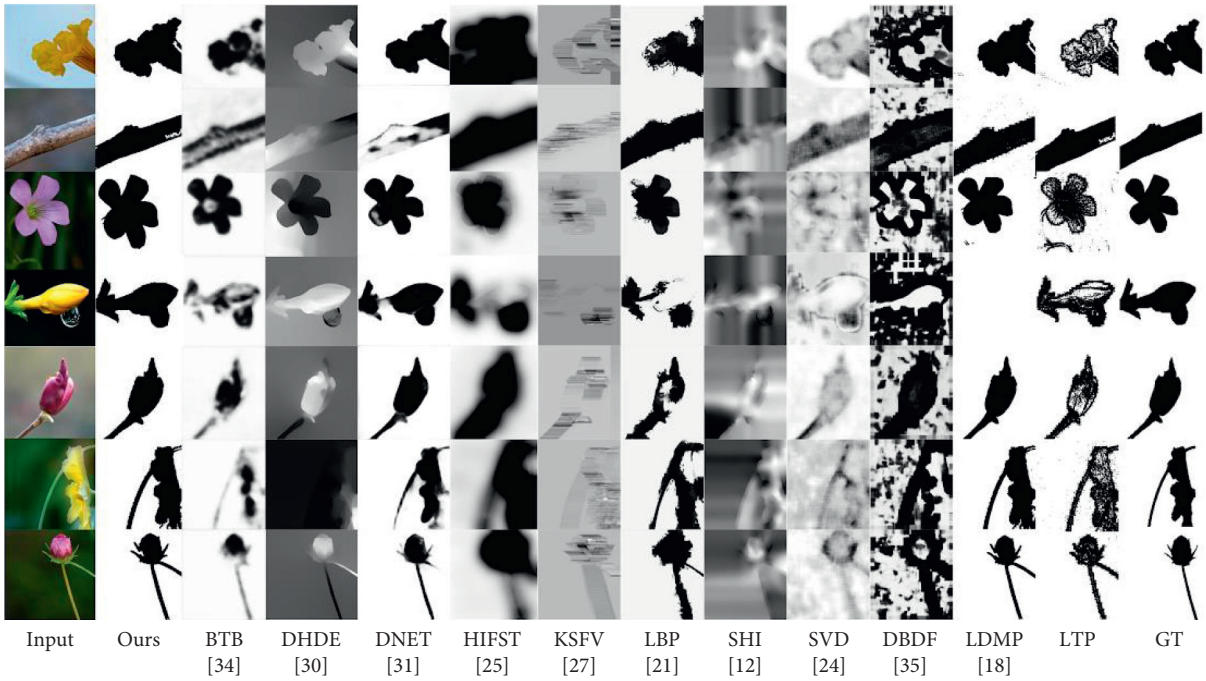


Figure 6: Visual comparison of the proposed method against comparative methods on the DUT dataset.

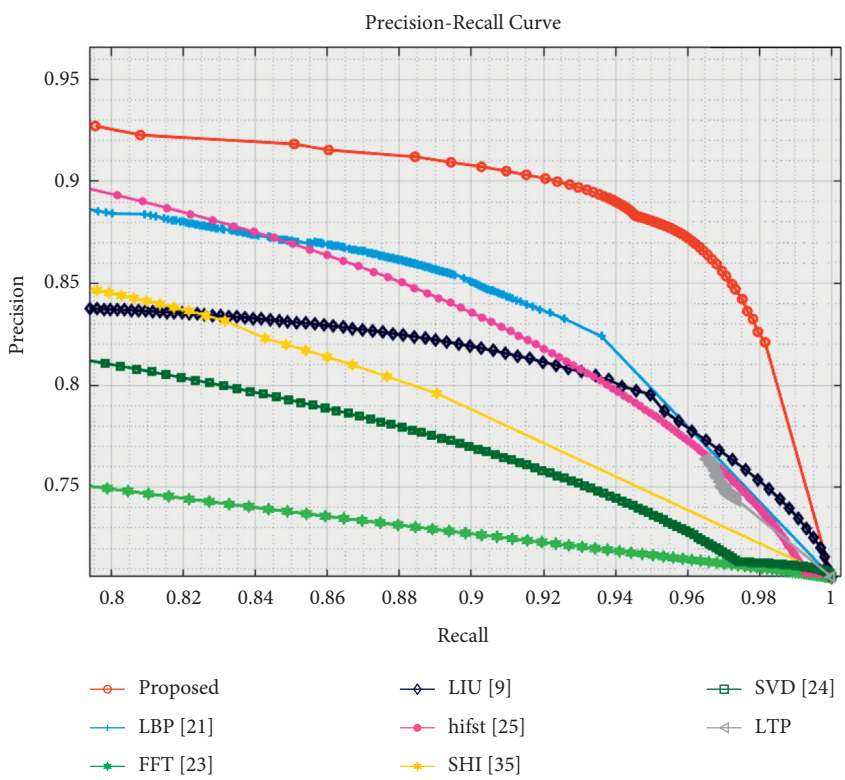


Figure 7: PR curve comparison on the Shi dataset.

comparative methods LBP [21] and LDMP [18] are unable to perform well for many images. Two methods [18, 21] used a similar approach of extraction from local integral values of the images but used either a fixed static threshold or from a specified range. This hard-coded threshold scheme results in performance degradation, as shown in

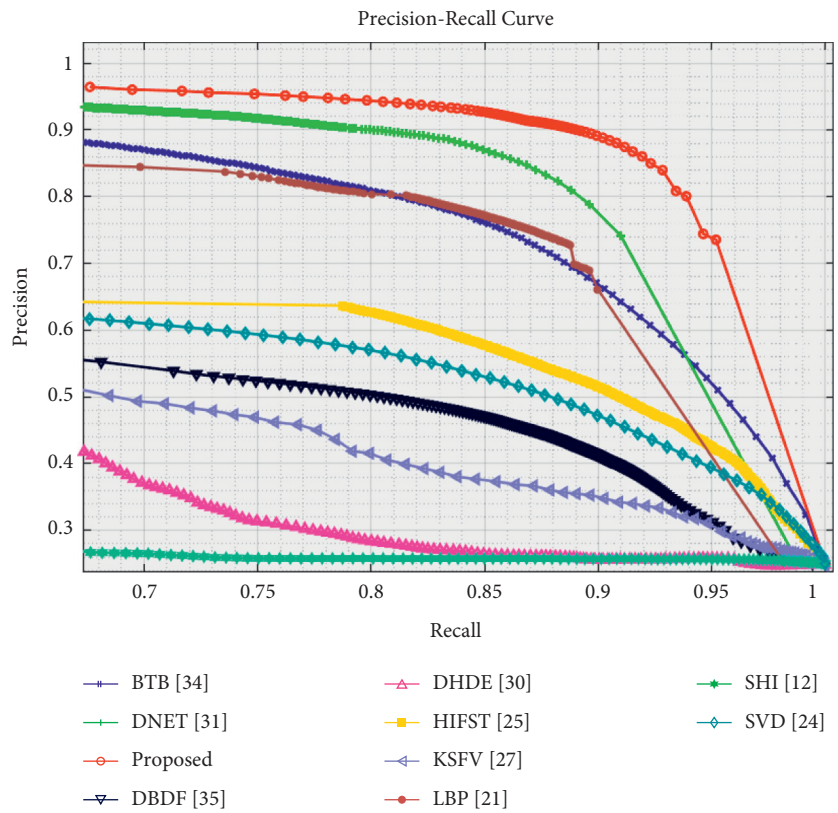


Figure 8: PR curve comparison on the DUT dataset.

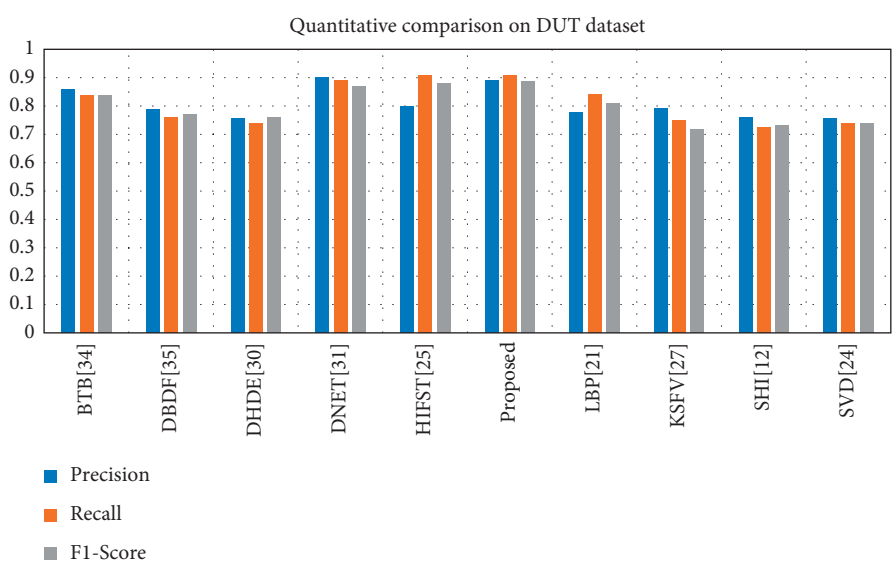


Figure 9: Quantitative comparison on the DUT dataset.

Figure 11 where [21] is unable to detect the regions in the image, whereas the proposed method used an adaptive threshold computed using the deviation between the neighboring pixels and performed very well. Our experiments further reveal that the results of [18, 21] indicate the gaps between the extracted sharp regions, whereas the proposed method produced the filled regions approximately. Extraction of the pixels using a small number of neighboring pixels can cause the overall selection of the region. We have used varying numbers of adjacent neighbors (3×3, 5×5, 7×7) to extract the regions based on the deviation between the neighbors. This provides significantly better results due to the large deviation of the region. Our results indicate that the pixel selection based on a small number of adjacent pixels affects the

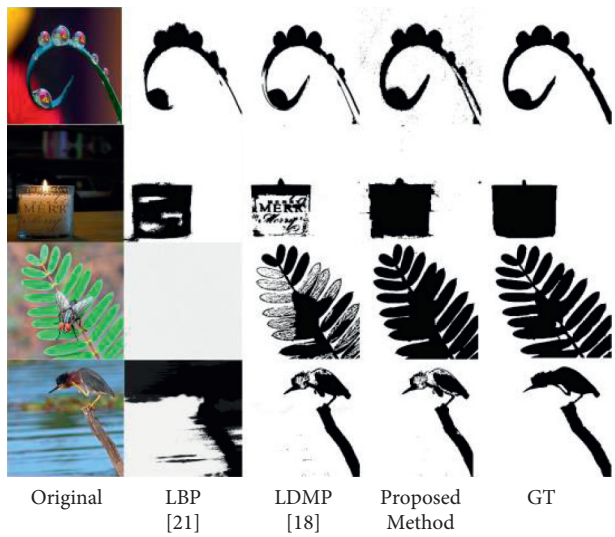


Figure 10: Comparison of local pattern-based approaches.

Table 2: Computational cost comparison of the proposed method against state-of-the-art methods.

DBD methods	Avg. computational cost (detection, segmentation)
DBDF [38]	7 m, 13 m
LBP [21]	5 s, 58 ms
Vu [13]	1 m 20 s, 35 s
SVD [24]	50 s, 55 s
Shi [12]	55 s
Zhou [28]	45 s
Zhu [22]	12 min
Proposed method	7 s

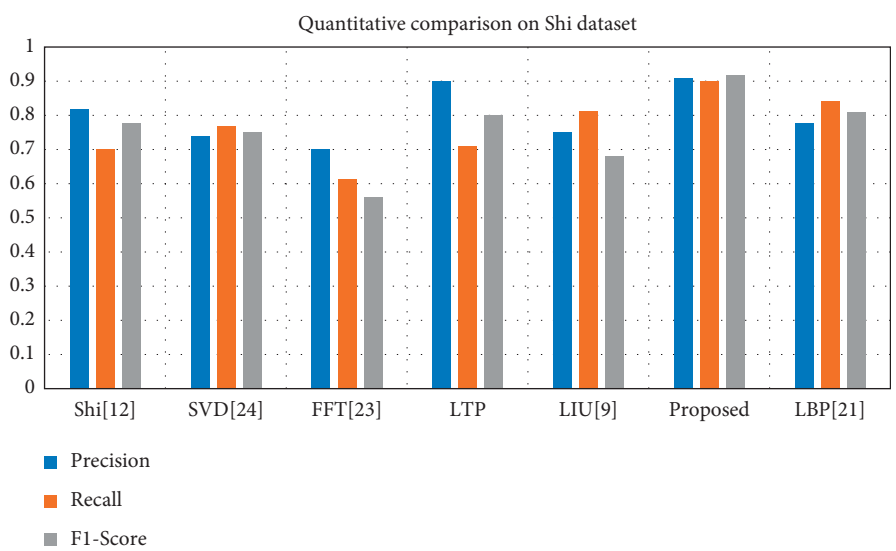


Figure 11: Quantitative analysis on the Shi dataset.

results, whereas using a large number of adjacent pixels to determine the pixel selection leads to good results. Existing defocus blur detection methods, based on using the integral values directly, fail to operate well for the motion blur. High variation in the integral values makes these methods unable to perform better on the motion blur. We aim to develop a unified

method in the future that can effectively detect both the motion and focus blur.

5. Conclusion

We have proposed an effective and efficient method for defocus blur detection problem without having the prior information about blur, camera configuration, pixels densities, etc. The local sharpness metric map is extracted directly from the images at different scales along with different patch sizes of the images. The local deviations between the neighboring pixels are used for the extraction of sharp regions. The automatic image masking and Otsu thresholding provide highly accurate and optimal segmented results over state-of-the-art methods. Our experimental results demonstrated that the proposed method performs far better than many hard-coded threshold-based algorithms. Additionally, the proposed method has a significant speed advantage over several comparative segmentation algorithms, i.e., alpha matting, KNN matting, global matting by GPU implementation, etc.

Data Availability

The datasets used and analyzed in this paper are publicly available.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by Education and Research Promotion Program of KOREATECH (2021).

References

[1] W. Wang, "Salient object detection in the deep learning era: an in-depth survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, In press, 2021.

[2] C. Li, W. Chen, and Y. Tan, "Point-sampling method based on 3D U-net architecture to reduce the influence of false positive and solve boundary blur problem in 3D CT image segmentation," *Applied Sciences*, vol. 10, no. 19, p. 6838, 2020.

[3] X. Chen, "Generative adversarial U-net for domain-free medical image augmentation," 2021, https://arxiv.org/abs/2101.04793.

[4] A. Garnica-Carrillo, "Multi-focus image fusion for multiple images using adaptable size windows and parallel programming," *Signal, Image and Video Processing*, vol. 14, no. 7, pp. 1293–1300, 2020.

[5] A. Shen, H. Dong, K. Wang, Y. Kong, J. Wu, and H. Shu, "Automatic extraction of blur regions on a single image based on semantic segmentation," *IEEE Access*, vol. 8, pp. 44867–44878, 2020.

[6] M. T. Mahmood, U. Ali, and Y. K. Choi, "Single image defocus blur segmentation using Local Ternary Pattern," *ICT Express*, vol. 6, no. 2, pp. 113–116, 2020.

[7] X. Chen, "Motion blur detection based on lowest directional high-frequency energy," in *Proceedings of the 2010 IEEE International Conference on Image Processing*, September 2010.

[8] H. Xiao, W. Lu, R. Li et al., "Defocus blur detection based on multiscale SVD fusion in gradient domain," *Journal of Visual Communication and Image Representation*, vol. 59, pp. 52–61, 2019.

[9] R. Liu, Z. Li, and J. Jia, "Image partial blur detection and classification," in *Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition*, June 2008.

[10] E. Kalalembang, K. Usman, and I. P. Gunawan, "DCT-based local motion blur detection," in *Proceedings of the International Conference on Instrumentation, Communication, Information Technology, and Biomedical Engineering*, November 2009.

[11] E. Mavridaki and V. Mezaris, "No-reference blur assessment in natural images using fourier transform and spatial pyramids," in *Proceedings of the 2014 IEEE International Conference on Image Processing (ICIP)*, October 2014.

[12] J. Shi, L. Xu, and J. Jia, "Just noticeable defocus blur detection and estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, June 2015.

[13] C. T. Vu, T. D. Phan, and D. M. Chandler, "${\bf S} _ {3} $: a spectral and spatial measure of local perceived sharpness in natural images," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 934–945, 2011.

[14] N. D. Narvekar and L. J. Karam, "A no-reference image blur metric based on the cumulative probability of blur detection (CPBD)," *IEEE Transactions on Image Processing*, vol. 20, no. 9, pp. 2678–2683, 2011.

[15] R. Huang, W. Feng, M. Fan, L. Wan, and J. Sun, "Multiscale blur detection by learning discriminative deep features," *Neurocomputing*, vol. 285, pp. 154–166, 2018.

[16] L. D'Andrès, J. Salvador, A. Kochale, and S. Susstrunk, "Non-parametric blur map regression for depth of field extension," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1660–1673, 2016.

[17] S. Gur and L. Wolf, "Single image depth estimation trained via depth from defocus cues," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, June 2019.

[18] A. Khan, "Defocus blur detection using novel local directional mean patterns (LDMP) and segmentation via KNN matting," *Frontiers of Computer Science*, 2020.

[19] A. Khan, "Segmentation of defocus blur using local triplicate Co-occurrence patterns (LTCoP)," in *Proceedings of the 2019 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*, December 2019.

[20] M. Ma, W. Lu, and W. Lyu, "Defocus blur detection via edge pixel DCT feature of local patches," *Signal Processing*, vol. 176, Article ID 107670, 2020.

[21] X. Yi and M. Eramian, "LBP-based segmentation of defocus blur," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1626–1638, 2016.

[22] X. Zhu, S. Cohen, S. Schiller, and P. Milanfar, "Estimating spatially varying defocus blur from a single image," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4879–4891, 2013.

[23] A. Chakrabarti, T. Zickler, and W. T. Freeman, "Analyzing spatially-varying blur," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010.

[24] B. Su, S. Lu, and C. L. Tan, "Blurred image region detection and classification," in *Proceedings of the 19th ACM international Conference on Multimedia*, Singapore, 2011.

[25] S. A. Golestaneh and L. J. Karam, "Spatially-varying blur detection based on multiscale fused and sorted transform coefficients of gradient magnitudes," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.

[26] W. Zhao, "Enhancing diversity of defocus blur detectors via cross-ensemble network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, June 2019.

[27] Y. Pang, H. Zhu, X. Li, and X. Li, "Classifying discriminative features for blur detection," *IEEE Transactions on Cybernetics*, vol. 46, no. 10, pp. 2220–2227, 2015.

[28] T. Zhou, H. Fu, C. Sun, and S. Wang, "Shadow detection and compensation from remote sensing images under complex urban conditions," *Remote Sensing*, vol. 13, no. 4, p. 699, 2021.

[29] Y. Zhong, Z. Zheng, A. Ma, X. Lu, and L. Zhang, "COLOR: cycling, offline learning, and online representation framework for airport and airplane detection using GF-2 satellite images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 12, pp. 8438–8449, 2020.

[30] R. Liu, W. Yuan, X. Chen, and Y. Lu, "An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system," *Ocean Engineering*, vol. 235, Article ID 109435, 2021.

[31] S. Arabi, A. Haghighat, and A. Sharma, "A deep-learning-based computer vision solution for construction vehicle detection," *Computer-Aided Civil and Infrastructure Engineering*, vol. 35, no. 7, pp. 753–767, 2020.

[32] B. Kim, "Defocus and motion blur detection with deep contextual features," in *Computer Graphics Forum*Wiley Online Library, Hoboken, NJ, USA, 2018.

[33] J. Park, Y.-W. Tai, D. Cho, and I. S. Kweon, "A unified approach of multi-scale deep and hand-crafted features for defocus estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.

[34] C. Tang, X. Zhu, X. Liu, L. Wang, and A. Zomaya, "Defusionnet: defocus blur detection via recurrently fusing and refining multi-scale deep features," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, June 2019.

[35] Y. Zhai, J. Wang, J. Deng, G. Yue, W. Zhang, and C. Tang, "Global context guided hierarchically residual feature refinement network for defocus blur detection," *Signal Processing*, vol. 183, Article ID 107996, 2021.

[36] X. Yang, X. Shen, J. Long, and H. Chen, "An improved median-based otsu image thresholding algorithm," *AASRI Procedia*, vol. 3, pp. 468–473, 2012.

[37] W. Zhao, "Defocus blur detection via multi-stream bottom-top-bottom fully convolutional network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, June 2018.

[38] J. Shi, L. Xu, and J. Jia, "Discriminative blur detection features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 2014.

Hindawi
International Journal of Optics
Volume 2021, Article ID 6641980, 12 pages
https://doi.org/10.1155/2021/6641980

Research Article

Optic Disc and Optic Cup Segmentation for Glaucoma Detection from Blur Retinal Images Using Improved Mask-RCNN

Tahira Nazir,[1] **Aun Irtaza**,[1] **and Valery Starovoitov**[2]

[1]*Department of Computer Science, University of Engineering and Technology, Taxila 47050, Pakistan*
[2]*United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Minsk 220012, Belarus*

Correspondence should be addressed to Tahira Nazir; tahira.nazir77@gmail.com

Received 1 January 2021; Accepted 14 July 2021; Published 22 July 2021

Academic Editor: Sulaiman W. Harun

Glaucoma is a fatal eye disease that harms the optic disc (OD) and optic cup (OC) and results into blindness in progressed phases. Because of slow progress, the disease exhibits a small number of symptoms in the initial stages, therefore causing the disease identification to be a complicated task. So, a fully automatic framework is mandatory, which can support the screening process and increase the chances of disease detection in the early stages. In this paper, we deal with the localization and segmentation of the OD and OC for glaucoma detection from blur retinal images. We have presented a novel method that is Densenet-77-based Mask-RCNN to overcome the challenges of the glaucoma detection. Initially, we have performed the data augmentation step together with adding blurriness in samples to increase the diversity of data. Then, we have generated the annotations from ground-truth (GT) images. After that, the Densenet-77 framework is employed at the feature extraction level of Mask-RCNN to compute the deep key points. Finally, the calculated features are used to localize and segment the OD and OC by the custom Mask-RCNN model. For performance evaluation, we have used the ORIGA dataset that is publicly available. Furthermore, we have performed cross-dataset validation on the HRF database to show the robustness of the presented framework. The presented framework has achieved an average precision, recall, *F*-measure, and IOU as 0.965, 0.963, 0.97, and 0.972, respectively. The proposed method achieved remarkable performance in terms of both efficiency and effectiveness as compared to the latest techniques under the presence of blurring, noise, and light variations.

1. Introduction

Glaucoma harms the optic nerve (ON) because of the imbalance of intraocular pressure within the eye. The affected nerve fibers result in deterioration of the retinal layer and give rise to the enlarged OD, that is, the part of the retina, and the OC is the main portion of the OD. Glaucoma is typically analysed by attaining the medical history of patients, determining intraocular pressure (IOP), conducting visual field loss tests, and manual assessment of OD employing ophthalmoscopy to investigate the shape and color of the ON [1]. The cup-to-disc ratio (CDR) is one of the key structural image cues reflected for glaucoma identification. The CDR compares the diameter of OC with the diameter of OD; less than 0.5 CDR considers the normal value [2]. So, timely detection of disease can avoid blindness [3]. Hence, clustering of the malicious area is not only advantageous for additional rigorous medical analysis by the ophthalmologist but also useful for designing the automated systems for disease categorization [4]. Initially, experts identify eye abnormalities through the manual examination of the glaucoma regions, by calculating the CDR, diameter, and boundaries variations [5]. However, due to the lack of available experts, timely identification of the eye abnormality is typically delayed [6], whereas early detection and treatment of the disease can save the victim from complete blindness. To tackle with mentioned challenges, the research community is targeting disease identification via Computer-Aided Diagnosis (CAD) based solutions.

In research, deep learning (DL) based approaches [3, 4, 7–20] have been utilized to identify glaucoma signs from the retinal images. In [21], an end-to-end RCNN

method for joint OD and OC segmentation was proposed. In joint-RCNN, OD and OC proposal networks were used to create bounding box (BB) proposals for OD and OC, respectively. The presented technique is computationally complex because it utilizes two distinct RCNNs to calculate the BBs of ROI regions. Therefore, a more reliable technique is required which can detect glaucoma affected region efficiently. In [22], a region-based pixel density calculation method was used for OD localization. Afterward, OD segmentation was performed through the Circular Hough Transform method. The procedure is efficient and robust to the segmentation of OD; however, its recognition performance is disturbed over the images having pathological distractions. In [3], the authors adapted DenseNet into a U-Net shaped framework for OD and OC segmentation. The method was comprised of three major phases, (i) preprocessing, (ii) FC-DenseNet model designing, and (iii) segmentation of OD and OC. In the first, the green channel was extracted from RGB images; after that, OD region within two OD diameters has been collected, which were utilized for the model training. In the second phase, the model has been built which was composed of three blocks, that is, dense and transition down and up. In the final phase, refinement was performed for the extraction of OD and OC through Softmax operation. The performance of the method [3] was evaluated over five different datasets and has achieved good results with a short testing time. However, the method [3] has some shortcomings: (i) calculation of OD centre being dependent on GT data, (ii) high training time, and (iii) training being done on small set. In [18], an eighteen-layer CNN architecture was proposed for glaucoma localization, which has two main components: (i) convolutional and max-pooling layer phase (ii) and fully connected layer phase. The method has evaluated 1426 images and achieved an accuracy of 98.13%. However, the method in [18] degrades performance on unseen samples and may not detect glaucoma at early stages.

In [15], Lu et al. presented a weekly and semisupervised segmentation method based on the Modified U-Net model for OD segmentation. Initially, the GrabCut technique was employed for the generation of the GTs. The U-Net model was improved by minimizing the original U-shape structure by adding a 2-dimensional convolutional layer at the end of the convolutional layer. This method needs a smaller amount of training, however, indicating less accuracy than other methods due to the lack of GTs. Elangovan et al. [23] have proposed the approach for glaucoma identification based on CNN which was consisted of 18 layers. The technique has different phases: preprocessing, key points computation, and classification. Initially, image resizing and data augmentation were performed; furthermore, rotation augmentation was applied to enhance the number of samples. Features were extracted through CNN which has four convolutional, two pooling, and a fully connected layer. For performance evaluation of the method, different datasets were used, namely, ORIGA, DRISHTI-GS1, RIM-ONE2, LAG, and ACRIMA. In [24], authors have presented the attention-based CNN (AG-CNN) technique for glaucoma recognition. In this paper [24], the authors have created a new database called large-scale attention-based glaucoma, which has a total of 11760 retinal images. All images were marked with negative or positive glaucoma. The AG-CNN method was comprised of two main stages; in the first phase, the attention prediction subnet was used to learn the ROI of glaucoma and then predicted the attention map. Secondly, the predicted map was utilized in the localized region, and then the feature map of this subnet was visualized to locate the pathological region. Lastly, the located region was merged with the anticipated attention to combining the input and subnet of glaucoma key points, for computing the binary labels of glaucoma. The method in [24] shows good performance and reduces the redundancy of fundus images; however, the method depends on the attention prediction subnet.

Existing techniques perform well over the standard datasets but not generalized well to real-world scenarios. The main reasons for performance degradation are the occurrence of blurring, noise, and light variations during the image capturing process, while the standard datasets are acquired in the control environment. In this work, our main motivation is to propose such techniques that can localize and segment the fundus samples under the presence of such factors. We have selected standard datasets like ORIGA and HRF databases which contain light variations and noise effects but lack the presence of blurriness. So, in this work, we have added blurriness in samples of mentioned datasets and proposed a novel technique, namely, Densenet-77 based [25] customized Mask-RCNN to detect and segment the OC and OD from fundus samples. The following are the main contributions of our work:

(1) The proposed method can precisely segment the OD and OC for glaucoma diagnosis from retinal images under the presence of blurring, noise, and light variations in input images.

(2) We have created the annotations which are essential for the training of the proposed model because available datasets do not have a BB and mask GTs.

(3) Accurate localization and segmentation of OD and OC due to effective region proposal network of Custom Mask-RCNN as it works in an end-to-end manner.

(4) Extensive results perform over challenging dataset ORIGA to show the robustness of the presented framework. Moreover, we have performed cross-dataset validation over the HRF database to demonstrate the generalization power of our technique to real-world scenarios.

2. Materials and Methods

The retinal images collected from different clinics can contain various artifacts like blurring, noise, out-of-focus images, or light variations, which must be removed to enhance the segmentation performance of the system. In our paper, we have employed the feature level set technique for correcting the bias field and applied the median filter to minimize the noisy effects from retinal images.

2.1. Preprocessing. The augmentation step is employed to increase the image samples in terms of data diversity. For this purpose, the input images are rotated at the angles of 0°, 90°, 180°, and 270° degrees, and Gaussian blur [26] is used over them to add blurriness.

Furthermore, we have generated the annotations for OC and OD regions. The GT mask along the retinal image is needed to detect glaucoma regions, that is, OD and OC for the training procedure. We used the VGG Image Annotator [27] tool to create a polygon mask for every image. Figure 1 presents a sample of images and related mask images. The annotations are saved in a JSON file that contains the set of polygon points for OD and OC regions. This file is utilized to generate a mask image related to each retinal image.

2.2. Localization and Segmentation of OD and OC Using Custom Mask-RCNN. Our objective is the automated detection and segmentation of OD and OC from fundus images with complicated backgrounds and under the presence of postprocessing operations without any human involvement. We aimed to identify glaucoma affected and nonaffected areas from a given sample by utilizing the Mask-RCNN [28] approach. The introduced approach (as shown in Figure 2) comprises the following steps: (1) key points computation, (2) region proposal network (RPN), (3) region of interest (ROI) classifier and bounding box regressor (BBR), and (4) OD and OC segmentation. The comprehensive explanation of all steps is described in the following.

2.3. Features Extraction. In our approach, we have used DenseNet-77 at the feature extraction level of the Mask-RCNN to compute the key points from a given sample. Utilizing DenseNet-77 for features computation exhibits an improvement in both the segmentation accuracy and computational complexity. The starting layers compute low-level key points from the images, that is, edge and corner information, and the deep layers calculate high-level key points, that is, structure and chrominance information. The extracted feature map is more enhanced through the FPN that calculates the key points with improved object representation at diverse scales for the RPN module.

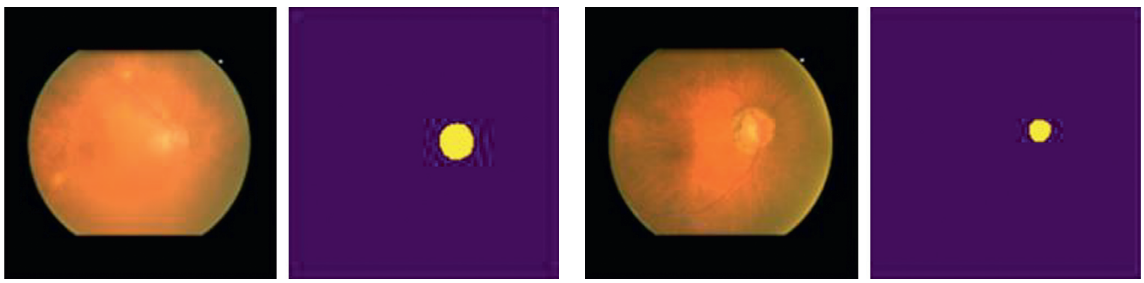
DenseNet [25] model is the advanced or improved form of Resnet, where the current layer belongs to all other layers. DenseNet contains the set of dense blocks, which remain consecutively linked with each other by using the extra convolutional and pooling layers among consecutive dense blocks. DenseNet can present the complex transformations which result in improving the issue of the absence of the target's position information for the top-level key points to some degree. DenseNet reduces the total parameters which makes them cost-effective. Furthermore, it supports the calculation of key points and encourages them to recycle, which makes them more suitable for region classification in retinal images. So, in this paper, we have employed the DenseNet-77 as a feature extractor for Mask-RCNN. The explanation of the DenseNet-77 model is shown in Figure 3. It also signifies the query sample size to be accommodated before computing key points from the allocated layer. The complete flow or description of the proposed method is presented in Algorithm 1.

The DenseNet-77 has two potential differences from traditional DenseNet: (i) it has a smaller number of parameters than the actual model and (ii) the layers within all dense block are adjusted to overcome with the computational complexity. Table 1 presents the detail of the training parameters for the Custom CenterNet.

2.4. Region Proposal Network. The calculated feature map from the previous step is passed as input to the RPN module to produce ROIs. Our work has used a 3×3 convolutional layer to scan the input sample by a sliding window to produce appropriate anchors that show the BB with varying scales and dispersed over the whole input sample. RPN module generates almost 20 k anchors of varying scales and dimensions which relate to each other to cover the entire image. A classifier is employed to decide whether an anchor holds the object or background (fg/bg). The BBR produces BBes according to the set intersection-over-union (IoU) value. Precisely, if the IoU value for an anchor is greater than 0.7 holding a GT box, then it is categorized positive; otherewise, it is marked as negative. The RPN module may generate overlapped areas; therefore, a nonmaximum suppression technique is used to keep the regions with the highest foreground score and discard the remaining insignificant parts. The final RoIs are passed to the succeeding step for performing classification.

2.5. ROI Classification and Bounding Box Regression. This module accepts two types of inputs which are the introduced RoI and feature map from previous steps. In contrast to the RPN module, this part is deeper and assigned a specific class to RoIs like glaucoma or nonglaucoma and improves the location of BB. The main objective of the BBR is to improve the location and dimension of the BB to correctly capture the glaucoma region. Typically, the margins of ROI do not overlap with the granularity of the feature map because of the reason that the computed feature map is shrunk k times from the actual image size. For resizing the feature maps, the ROIAlign layer is utilized to compute fixed-length key points vectors for random-sized candidate areas. For resizing, the ROIAlign layer employs the bilinear interpolation to evade misalignment problems that occurred in the ROI pooling layer which utilizes the quantization process.

2.6. Segmentation Mask. This module accepts positive marked ROIs by the ROI classifier as input and computes the segmentation mask with the dimension of 28×28 shown by floating values that hold more details as compared to binary masks. The GT masks are resized to 28×28 to compute the loss using the identified mask in the training step, which is later scaled up to match the actual size of the ROI BB to show the final mask.



(a) (b)

Figure 1: Sample original images and corresponding GT masks. (a) Optic disc. (b) Optic cup.

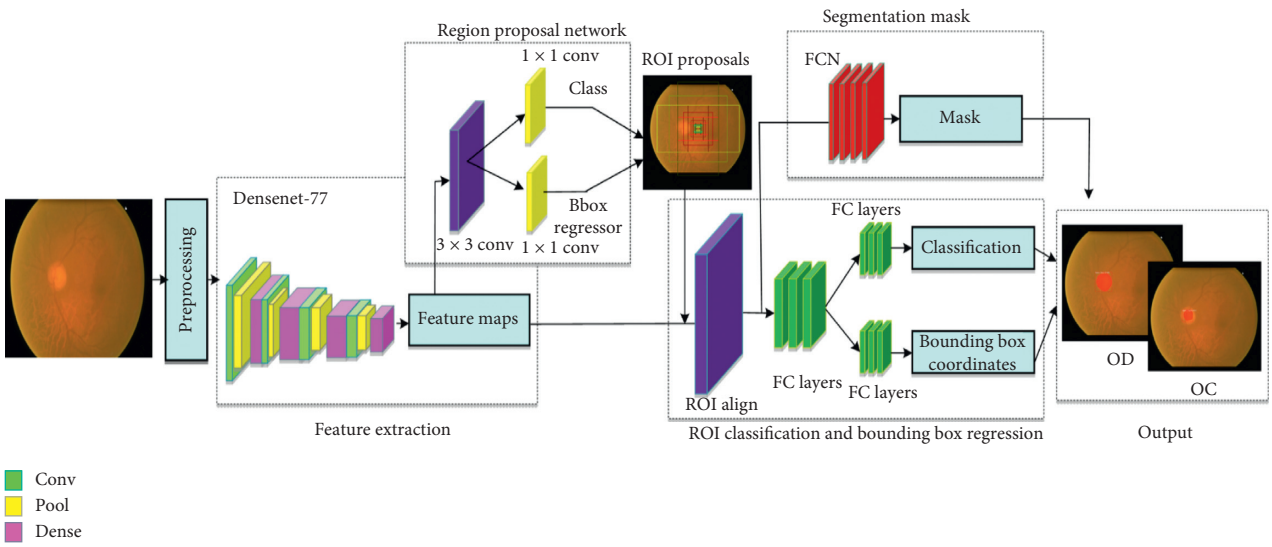


Figure 2: Architecture of the proposed approach.

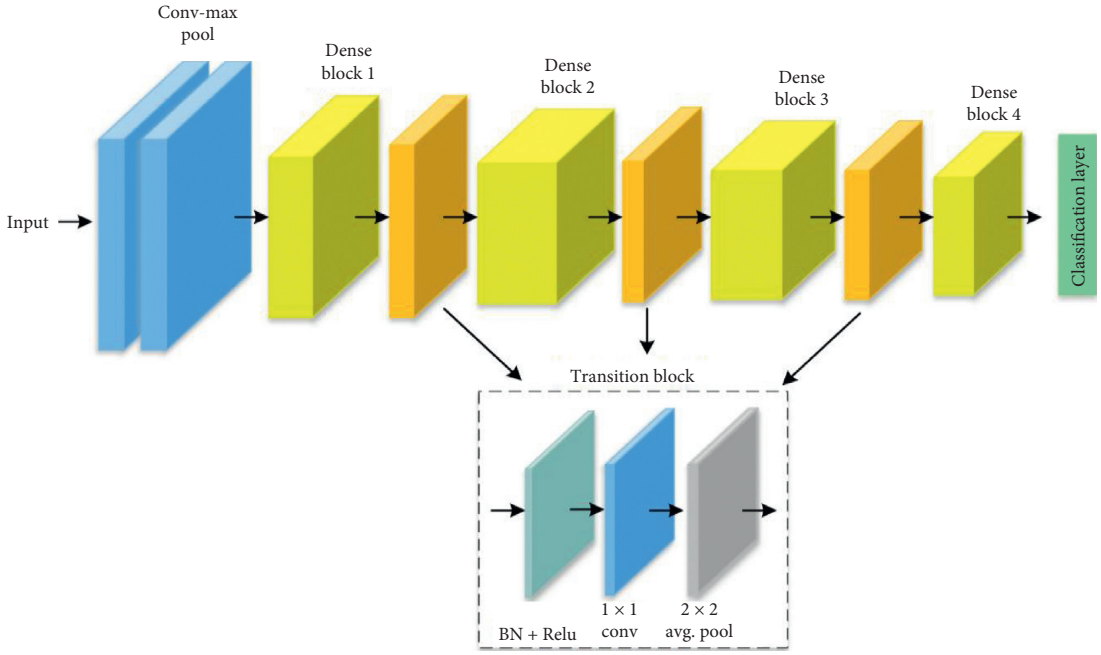


Figure 3: Structure of DenseNet-77.

```
START
INPUT: NS, annotation (orientation)
OUTPUT: Localized RoI, CMskDenseNet-77
  NS : Total image samples containing.
  annotation (orientation): Mask coordinates of the glaucoma regions in the retinal image
  Localized RoI : Region placement
  CMskDenseNet-77- : Custom Mask-RCNN network with DenseNe-77 key points
SampleResolution ← [x y]
// Computing Mask
μ← AnchorsComputation (NS, annotation)
// Customized MaskRCNN model
CMskDenseNet-77← DesignCustomDenseNet-77MaskRCNN (SampleResolution, μ)
[ Sr, St] ← database division into train and test section
 // Glaucoma Region recognition from Training part
For each sample f in ⟶Sr
  Compute DenseNet-77 keypoints ⟶ns
End For
Training CMskDenseNet-77over ns, and compute training time t_dense
∂_dense ← PreRegionLoc(ns)
Ap_dense ← Evaluate_AP (DenseNet-77, ∂_dense)
For each sample F in ⟶ St
   (a) compute features by employing trained model ¥⟶βI
  (b) [Mask, objectness_score, classLabel] ←Predict (βI)
  (c) Output sample along with Mask, class
  (d) ∂← [∂ Mask]
End For
Ap_¥← Evaluate framework ¥ using ∂
FINISH.
```

Algorithm 1: Steps for OD and OC segmentation with custom Mask-RCNN.

Table 1: Hypermeters details.

Framework parameters	Value
Epochs	30
Learning rate	0.001
Batch size	8
Confidence score threshold	0.2
Unmatched threshold	0.5

2.6.1. Multitask Loss. The presented framework uses a multitask loss L on all sampled ROIs given as follows:

$$L\ (\text{MaskRCNN}) = L_{\text{bclass}} + L_{\text{ref}} + L_{\text{smask}}. \tag{1}$$

Here L_{bclass}, L_{ref}, and L_{smask} demonstrate the box class labels estimation loss, BB refinement loss, and segmentation mask prediction loss, respectively. L_{bclass} presents the log loss of the two categories (glaucoma/nonglaucoma), given as follows:

$$L_{\text{bclass}}\left(P_t, l\right) = -\log\left[P_t\, l + \left(1 - P_t\right)\left(1 - l\right)\right]. \tag{2}$$

L_{bclass} is the log loss of the binary classification, where P_t presents the target prediction probability of whether the anchor t holds glaucoma and l shows the gt label. There are about 20 k anchors generated of distinct scales and sizes that correspond with each other to cover the image. If an anchor has intersection over union (IoU) higher than 0.5 with a ground-truth (GT) box, it is classified as a positive anchor; otherwise, it is negative. If several anchors overlap too much, we keep the one with the highest foreground score and discard the rest (referred to as nonmax suppression). Moreover, the value of l is 1 for true-marked anchors and 0 otherwise. The BB regression loss is given as follows:

$$L_{\text{lref}}\left(c_j, c_j^*\right) = \sum_{j\varepsilon\{x,y,w,h\}} \text{smooth}_{L1}\left(c_j - c_j^*\right), \tag{3}$$

where

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise.} \end{cases} \tag{4}$$

Here, vector c_j is presenting four dimensions of the estimated BB, and c_j^* is showing the dimensions of gt relating to the true-marked anchors. The smooth-$L1$ function is a robust $L1$ loss which is prone to outliers as compared to $L2$ loss. When regression targets are unbounded, training $L2$ loss leads to a gradient explosion and requires a carefully tuned learning rate. During the training of Mask-RCNN, the average cross-entropy loss is used which is calculated as follows:

$$L_{\text{mask}} = -\frac{1}{N^2} \sum_{1 \le x,\, y \le N} \left[p_{xy} \log V_{xy}^k + \left(1 - p_{xy}\right) \log\left(1 - V_{xy}^k\right)\right], \tag{5}$$

where p_{xy} is the pixel value at the location (x, y) in a gt mask of size $N \times N$ and for the same pixel, V_{xy}^{k} is presenting its estimated value in the mask obtained for class k ($k = 1$ for glaucoma region and 0 for nonglaucoma region) [28].

3. Results and Discussion

We have implemented the model using Keras and TensorFlow libraries with DenseNet-77 and FPN for feature extraction. We initialized the model using pretrained weights obtained from the COCO dataset and employed transfer learning to fine-tune the model on retinal datasets for OD and OC segmentation. For experimentation, we used a 70–30 ratio that is randomly divided into training (70%) and test (30%) sets.

3.1. Dataset. The evaluation experiments of the system were performed on the ORIGA "Online Retinal Fundus Image Database for Glaucoma Analysis" dataset [29]. The details of dataset are presented in Table 2. The dataset have a total of 650 images in which 168 are glaucomatous samples and the remaining 482 are nonglaucomatous samples and gathered from the "Eye Research Institute, Singapore." In each image, OD and OC regions are marked by experts using a vertical and nonrotated ellipse. The sample images are shown in Figure 4.

3.2. Evaluation Parameters. The proposed method is assessed by employing the intersection over union (IOU) as described in Figure 5. A shows the GT rectangle, and B denotes the estimated rectangle with ROI regions.

The first decision for the region is identified when the value of IOU is greater than 0.5; otherwise, it is not recognized. The average precision (AP) is mostly employed in evaluating the precision of object detectors, that is, R–CNN, SSD, and YOLO. The geometrical explanation of precision is shown in Figure 6. In our framework of the detection of glaucoma regions, AP depends on the idea of IOU [30].

3.3. Results. This section presented the details of results achieved after performing the experiments over diverse samples with light, color, region sizes variations, and the presence of blurring. For OD, to show the detection accuracy of the presented framework, the visual results are reported in Figure 7. It can be observed from the results that the proposed method can accurately localize the OD regions from the healthy areas despite discontinuous or blurry boundaries and artifacts in fundus images. Moreover, the Mask-RCNN method can precisely segment the OD regions by overcoming the challenges of location, shape, and size.

Furthermore, the visual results for OC segmented regions are shown in Figure 8. From the reported results, it can be visualized that our method can accurately localize and segment the OC regions under the different conditions due to a representative set of features extraction by DenseNet-77 and segmentation power of Mask-RCNN. However, its localization and segmentation power may slightly decrease for samples with intense color variations which results in color-matching with healthy regions.

The proposed method can accurately recognize the OD and OC with an average accuracy of 0.965 on the ORIGA dataset. Moreover, the proposed technique can precisely segment the OD and OC by overcoming the challenges of blurriness and variations in location, size, and shape.

To further understand the performance of our method, we have used the evaluation parameters i.e., accuracy, precision, recall, F-measure, and IOU. Table 3 demonstrates the results or proposed approach. We can observe that the presented framework has achieved an average precision, recall, F-measure, and IOU as 0.965, 0.963, 0.97, and 0.972, respectively. Moreover, the confusion matrix of the proposed approach is presented in Figure 9.

3.4. DenseNet-77 Framework Evaluation. We performed an analysis to evaluate the robustness of the DenseNet-77 framework for eye disease detection by comparing it with other DL approaches. To accomplish this, the accuracy of the introduced Mask-RCNN with DenseNet-77 is compared with other base models, that is, Inception-v4 [31], VGG-16 [32], ResNet-101 [33], ResNet-152 [33], and DenseNet-121 [34].

Table 4 shows the comparative analysis of the presented method with other frameworks in both the aspect of model parameters and detection accuracy. The results of this comparative analysis indicate that the custom Mask-RCNN with DenseNet-77 works better than the Inception-v4, VGG-16, ResNet-50, ResNet-101, ResNet-152, and DenseNet-121. Moreover, from Table 4, it can be seen that VGG-16 has the highest model parameters, whereas ResNet-152 is the most expensive approach in terms of execution time. On the contrary, the presented framework with the DenseNet-77 model is economically most efficient and took only 1067 seconds for execution. The main reason for the efficient performance of DenseNet-77 is having a shallow architecture that employs efficient reuse of framework parameters without using redundant key point maps. Such structure of DenseNet-77 results in the extensively minimum number of framework parameters, whereas the comparative techniques suffer from high economical cost and unable to show efficient classification performance for the samples with noise, blurring, scale, and angle variations. Therefore, the presented technique better tackles the issues of comparative models by introducing a robust network for feature extraction and shows complicated transformations perfectly, leading to enhanced detection accuracy in postprocessing attacks as well. From the conducted analysis, it can be summarized that our customize Mask-RCNN with DenseNet-77 framework exhibits better performance than the other deep learning models in both terms of accuracy and efficacy.

3.5. Evaluation of the Custom Mask-RCNN Model. In this section, we have compared the performance of the introduced methodology with other region-based segmentation methods, that is, RCNN and Faster-RCNN over the ORIGA database, and results are reported in Figure 10. The RCNN is computationally complex as it randomly generates region

Table 2: Dataset details.

Attribute	Value
Total images	650
Glaucoma images	168
Normal images	482
Resolution	3072×2048
Ground truths	OD and OC regions

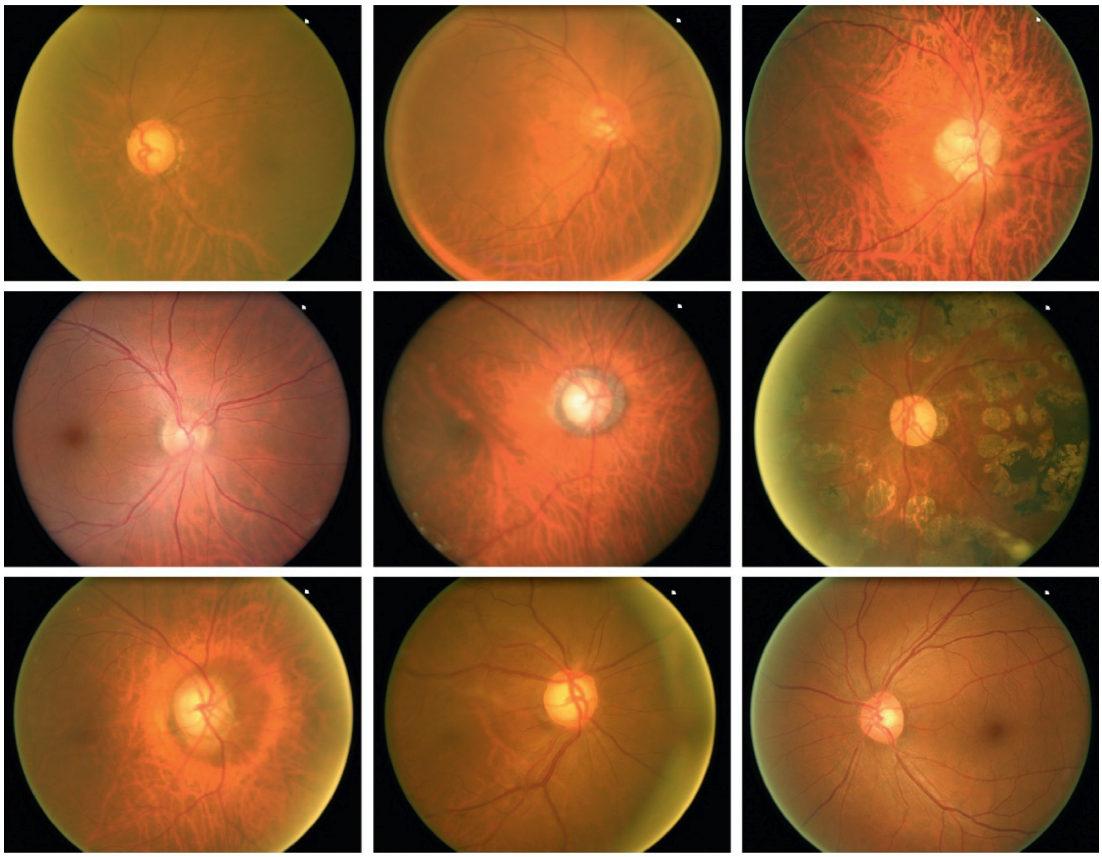


Figure 4: Sample images of the ORIGA dataset.

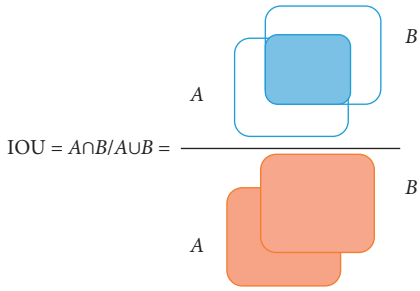


Figure 5: IOU Venn diagram.

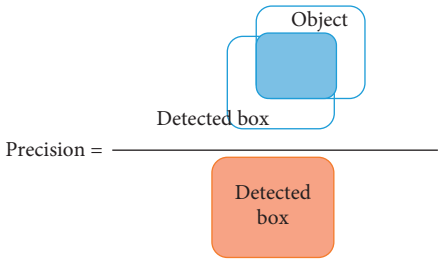
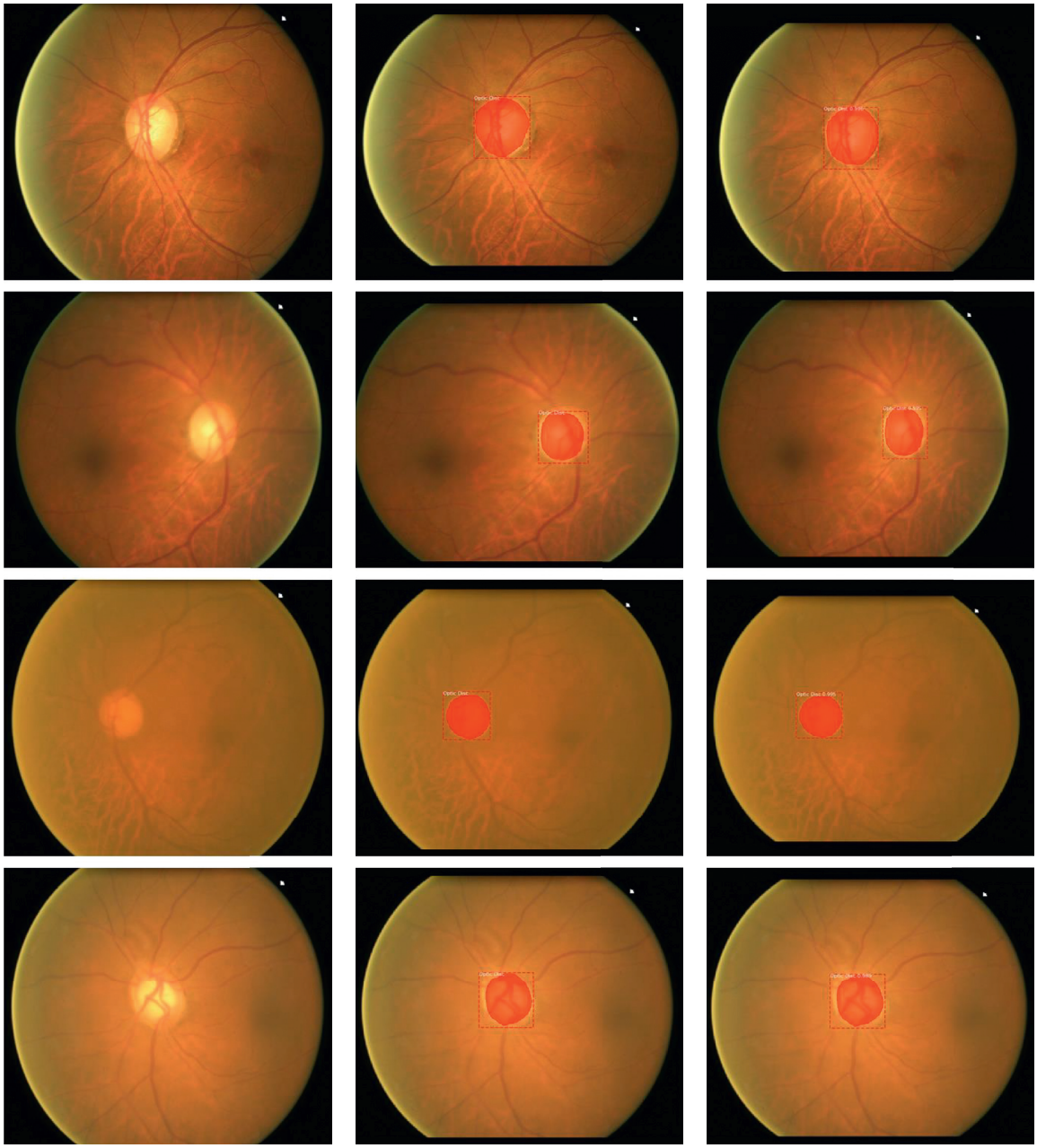


Figure 6: Geometrical representation of precision.

proposals (2000 per image) and uses a selective search algorithm for classification. The Faster-RCNN automatically extracts the region proposals using the RPN and shares the convolutional layer among class and BB network to reduce the computational cost. The traditional Mask-RCNN offers an added advantage over Faster-RCNN by providing an automated segmentation mask as well but is unable to capture the robust set of features under the postprocessing attacks. Therefore, the presented DenseNet-77 based Mask-RCNN performs well in comparison to traditional Mask-RCNN as DenseNet can capture the complex transformations with more accuracy which results in better automated segmentation and localization of



(a) (b) (c)

Figure 7: Visual results of OD segmentation. (a) Input images. (b) Annotated images. (c) Output images.

glaucoma regions. Moreover, our model is easier to train and adds a very small overhead over Mask-RCNN.

3.6. Comparative Analysis. Here, we have compared the performance of our model with the existing approaches over the ORIGA dataset. The proposed technique uses deep features that are more discriminating and reliable and provide a more effective representation of glaucoma regions over other methods. For performance evaluation, we evaluate our approach against the work of Bajwa et al. [1], Jiang et al. [21], Xu et al. [35], and Fu et al. [8]. These techniques are capable of detecting glaucoma from retinal images. However, they require intense training and exhibit lower accuracy for training samples with the class imbalance problem. The comparison results are presented in Table 5. Our framework has acquired the highest average precision, recall, and AUC, that is, 0.965, 0.963, and 0.96, respectively, that signifies the reliability of the proposed method in comparison with other methods. Unlike these methods, our model performs segmentation on the localized ROIs, which limits the space of segmentation and uses the ROIAlign layer which ultimately improves the accuracy of the final segmentation result.

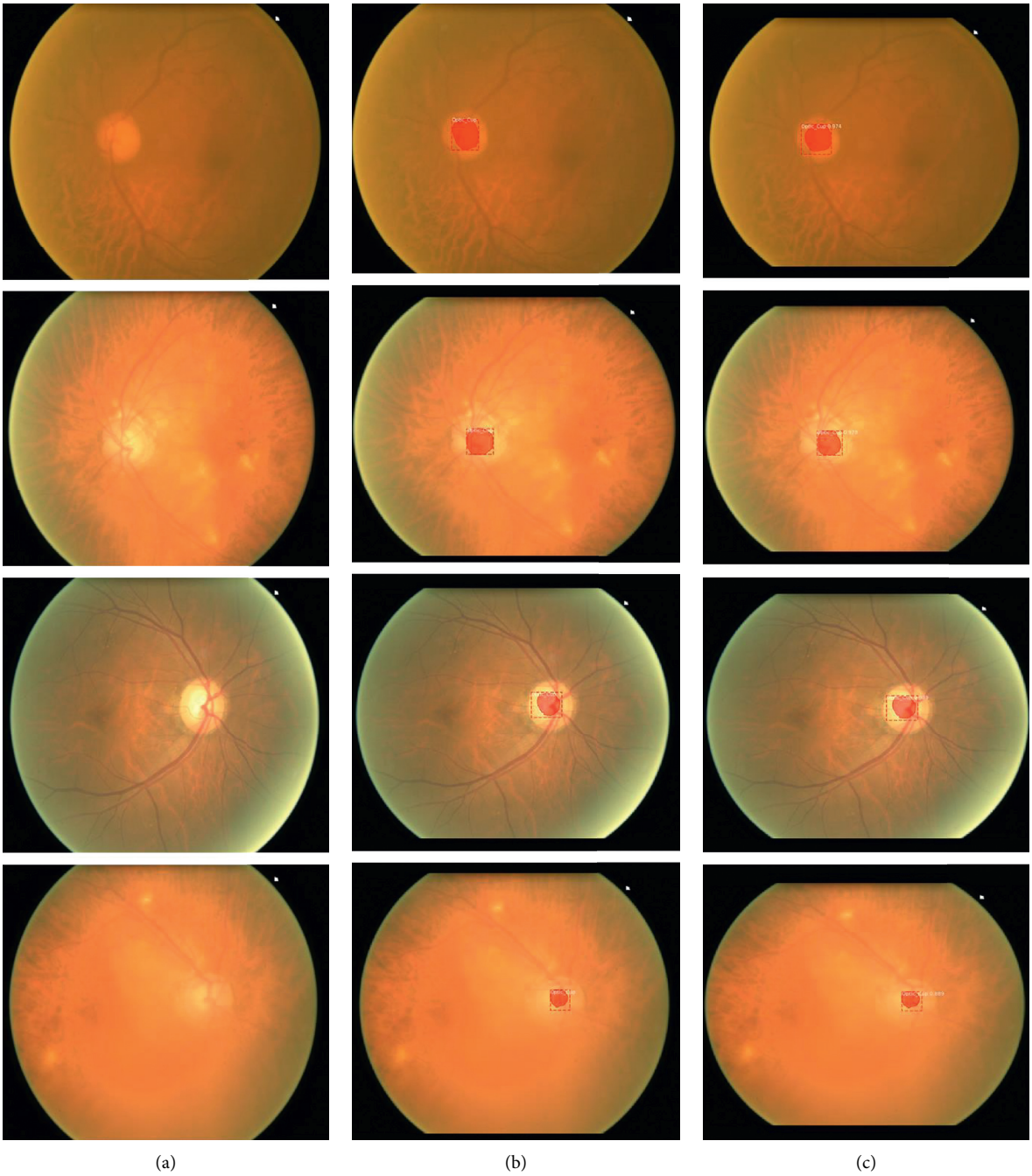


Figure 8: Visualization results of OC segmentation. (a) Input images. (b) Annotated images. (c) Output images.

Table 3: Proposed method results.

Method	Accuracy	Precision	Recall	F-measure	IOU
OD	0.979	0.959	0.969	0.953	0.981
OC	0.951	0.971	0.957	0.987	0.963
Average	0.965	0.965	0.963	0.970	0.972

3.7. Cross-Dataset Validation. To further evaluate the performance of the proposed method, we trained our method on the ORIGA dataset, and testing is performed on the HRF dataset [36]. The dataset contains 45 retinal images in which 15 images are healthy, 15 images are affected with diabetic retinopathy, and 15 images are affected by glaucoma.

We have plotted the box plot for evaluation of the cross dataset in Figure 11; the accuracy of the test and train is

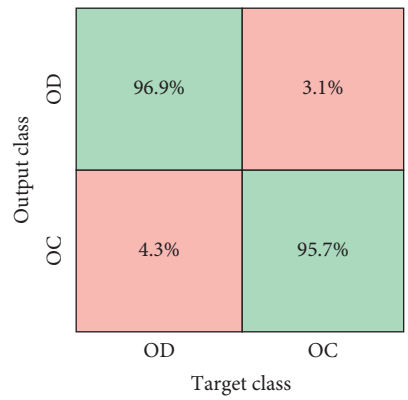


Figure 9: Confusion matrix.

Table 4: Comparative analysis of the proposed approach with base models.

Parameters	Inception-V4	VGG-16	ResNet-101	ResNet-152	DenseNet-121	DenseNet-77
Total parameters (million)	41.2	119.6	42.5	58.5	7.1	6.2
Training loss	0.0102	0.5069	$4.1611e^{-04}$	$2.4844e^{-04}$	$5.6427e^{-04}$	$6.442e^{-04}$
Test loss	0.0686	0.6055	0.02082	0.0246	0.0159	0.0085
Training accuracy	99.74%	83.86%	99.99%	100%	100%	100%
Test accuracy	98.08%	81.83%	99.66%	99.59%	99.75%	99.983%
Processing time (s)	4042	1051	2766	4366	2165	1067

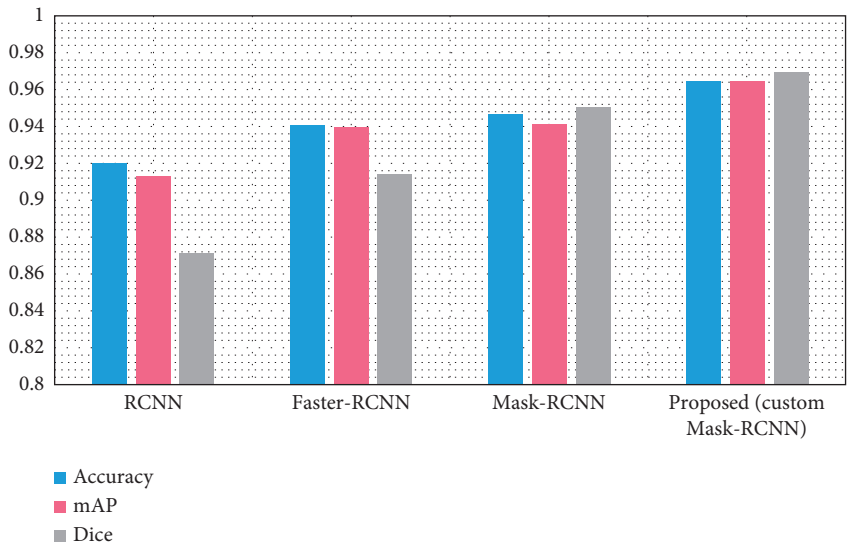


Figure 10: Comparison with other RCNN methods.

Table 5: Comparison with other techniques.

Method	Recall	Precision	AUC
Bajwa et al. [1]	0.71	—	0.860
Jiang et al. [21]	—	0.937	0.854
Xu et al. [35]	0.58	—	0.830
Fu et al. [8]	0.84	0.92	0.910
Proposed	0.963	0.965	0.970

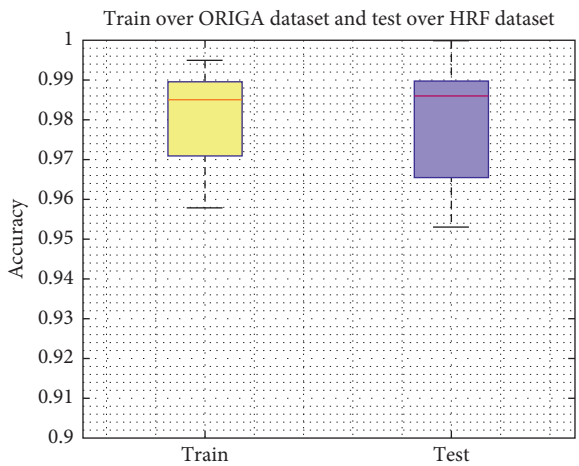


Figure 11: Cross-dataset validation results.

spreading across the number line into quartiles, median, whisker, and outliers. According to the figure, we achieved an average accuracy of 98% for training and 97.7% for testing which exhibits that our proposed work outperforms the unknown samples as well. Therefore, it can be concluded that the introduced framework is robust to OD and OC localization and segmentation.

4. Conclusions

In this paper, we presented a deep learning technique to customize Mask-RCNN for precise and automated segmentation of OD and OC from the retinal images. We introduce the DenseNet-77 model at the feature computation level of Mask-RCNN to compute the more diverse key points which assist in accurately localizing the OD and OC regions under the various sample conditions. We have tested our framework over a challenging database, namely, ORIGA, and performed cross-dataset validation on the HRF database to show its robustness. The results exhibit that improved Mask-RCNN can compute deep features with effective representation of glaucoma regions over existing systems and serves as a new automated tool for diagnostic purposes. Moreover, both the qualitative and quantitative results show that Custom Mask-RCNN works better than the base framework. Although our approach has presented better OD and OC detection accuracy, however, it can be further enhanced by the inclusion of other latest DL-based techniques like EfficientNet. Furthermore, we plan to extend our work to other medical abnormalities.

Data Availability

Data sharing is not applicable to this article as authors have used publicly available datasets, whose details are included in the Experimental Results section of this article.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

[1] M. N. Bajwa, M. I. Malik, S. A. Siddiqui et al., "Two-stage framework for optic disc localization and glaucoma classification in retinal fundus images using deep learning," *BMC Medical Informatics and Decision Making*, vol. 19, no. 1, p. 136, 2019.

[2] N. B. Prakash and D. Selvathi, "An efficient detection system for screening glaucoma InRetinal images," *Biomedical and Pharmacology Journal*, vol. 10, no. 1, pp. 459–465, 2017.

[3] B. Al-Bander, B. Williams, W. Al-Nuaimy, M. Al-Taee, H. Pratt, and Y. Zheng, "Dense fully convolutional segmentation of the optic disc and cup in colour fundus for glaucoma diagnosis," *Symmetry*, vol. 10, no. 4, p. 87, 2018.

[4] X. Chen, Y. Xu, D. W. K. Wong, T. Y. Wong, and J. Liu, "Glaucoma detection based on deep convolutional neural network," in *Proceedings of the 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 715–718, Milan, Italy, August 2015.

[5] J. Martins, J. S. Cardoso, and F. Soares, "Offline computer-aided diagnosis for Glaucoma detection using fundus images targeted at mobile devices," *Computer Methods and Programs in Biomedicine*, vol. 192, Article ID 105341, 2020.

[6] V. S. Mary, E. B. Rajsingh, and G. R. Naik, "Retinal Fundus Image Analysis for Diagnosis of Glaucoma: A Comprehensive Survey," *IEEE Access*, vol. 4, 2016.

[7] T. Nazir, A. Irtaza, A. Javed, H. Malik, D. Hussain, and R. A. Naqvi, "Retinal image analysis for diabetes-based eye disease detection using deep learning," *Applied Sciences*, vol. 10, no. 18, p. 6185, 2020.

[8] H. Fu, J. Cheng, Y. Xu et al., "Disc-aware ensemble network for glaucoma screening from fundus image," *IEEE Transactions on Medical Imaging*, vol. 37, no. 11, pp. 2493–2501, 2018.

[9] J. I. Orlando, E. Prokofyeva, M. del Fresno, and M. B. Blaschko, "Convolutional neural network transfer for automated glaucoma identification," in *Proceedings of the 12th International Symposium on Medical Information Processing and Analysis*, vol. 10160, p. 101600U, Tandil, Argentina, December 2017.

[10] B. Al-Bander, W. Al-Nuaimy, M. A. Al-Taee, and Y. Zheng, "Automated glaucoma diagnosis using deep learning approach," in *Proceedings of the 2017 14th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 207–210, Marrakech, Morocco, March 2017.

[11] A. Li, J. Cheng, D. W. K. Wong, and J. Liu, "Integrating holistic and local deep features for glaucoma classification," in *Proceedings of the 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1328–1331, Orlando, FL, USA, August 2016.

[12] Q. Abbas, "Glaucoma-deep: detection of glaucoma eye disease on retinal fundus images using deep learning," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, pp. 41–45, 2017.

[13] R. Hemelings, B Elen, J. Barbosa-Breda et al., "Accurate prediction of glaucoma from colour fundus images with a convolutional neural network that relies on active and transfer learning," *Acta Ophthalmologica*, vol. 98, no. 1, pp. e94–e100, 2020.

[14] M. D. Abràmoff, Y. Lou, A. Erginay et al., "Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning," *Investigative Opthalmology & Visual Science*, vol. 57, no. 13, pp. 5200–5206, 2016.

[15] Z. Lu and D. Chen, "Weakly supervised and semi-supervised semantic segmentation for optic disc of fundus image," *Symmetry*, vol. 12, no. 1, p. 145, 2020.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[17] X. Chen, Y. Xu, S. Yan, D. W. K. Wong, T. Y. Wong, and J. Liu, "Automatic Feature Learning for Glaucoma Detection Based on Deep Learning," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 669–677, Munich, Germany, October 2015.

[18] U. Raghavendra, H. Fujita, S. V. Bhandary, A. Gudigar, J. H. Tan, and U. R. Acharya, "Deep convolution neural network for accurate diagnosis of glaucoma using digital fundus images," *Information Sciences*, vol. 441, pp. 41–49, 2018.

[19] H. Fu, J. Cheng, Y. Xu, D. W. K. Wong, J. Liu, and X. Cao, "Joint optic disc and cup segmentation based on multi-label deep network and polar transformation," *IEEE Transactions on Medical Imaging*, vol. 37, no. 7, pp. 1597–1605, 2018.

[20] A. Sevastopolsky, "Optic disc and cup segmentation methods for glaucoma detection with modification of U-Net convolutional neural network," *Pattern Recognition and Image Analysis*, vol. 27, no. 3, pp. 618–624, 2017.

[21] Y. Jiang, L. Duan, Z. Gu et al., "JointRCNN: a region-based convolutional neural network for optic disc and cup segmentation," *IEEE Transactions on Biomedical Engineering*, vol. 67, Article ID 2913211, 2019.

[22] R. G. Ramani and J. J. Shanthamalar, "Improved image processing techniques for optic disc segmentation in retinal fundus images," *Biomedical Signal Processing and Control*, vol. 58, Article ID 101832, 2020.

[23] P. Elangovan and M. K. Nath, "Glaucoma assessment from color fundus images using convolutional neural network," *International Journal of Imaging Systems and Technology*, 2020.

[24] L. Li, "A large-scale database and a CNN model for attention-based glaucoma detection," *IEEE Transactions on Medical Imaging*, vol. 39, no. 2, pp. 413–424, 2019.

[25] Y. Wang, H. Li, P. Jia, G. Zhang, T. Wang, and X. Hao, "Multi-scale DenseNets-based aircraft detection from remote sensing images," *Sensors*, vol. 19, no. 23, p. 5270, 2019.

[26] J. Flusser, S. Farokhi, C. Höschl, T. Suk, B. Zitová, and M. Pedone, "Recognition of images degraded by Gaussian blur," *IEEE Transactions on Image Processing*, vol. 25, no. 2, pp. 790–806, 2015.

[27] A. Zisserman, "Vgg image annotator (VIA)," [Online]. Available: https://www.robots.ox.ac.uk/~vgg/software/via. [Accessed: 03-May-2020], 2020.

[28] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969, Venice, Italy, October 2017.

[29] Z. Zhang, "Origa-light: an online retinal fundus image database for glaucoma analysis and research,," in *Proceedings of the 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 3065–3068, Buenos Aires, Argentina, September 2010.

[30] T. Nazir, A. Irtaza, J. Rashid, M. Nawaz, and T. Mehmood, "Diabetic retinopathy lesions detection using faster-RCNN from retinal images," in *Proceedings of the 2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH)*, pp. 38–42, Riyadh, Saudi Arabia, March 2020.

[31] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, San Francisco, CA, USA, February 2017.

[32] W. Yu, K. Yang, Y. Bai, T. Xiao, H. Yao, and Y. Rui, "Visualizing and comparing AlexNet and VGG using deconvolutional layers," in *Proceedings of the 33 rd International Conference on Machine Learning*, New York, NY, USA, June 2016.

[33] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," 2016, https://arxiv.org/abs/1605.07678.

[34] I. Allaouzi and M. Ben Ahmed, "A novel approach for multi-label chest X-ray classification of common thorax diseases," *IEEE Access*, vol. 7, pp. 64279–64288, 2019.

[35] Y. Xu, S. Lin, D. W. K. Wong, J. Liu, and D. Xu, "Efficient reconstruction-based optic cup localization for glaucoma screening," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 445–452, Nagoya, Japan, September 2013.

[36] A. Budai, R. Bock, A. Maier, J. Hornegger, and G. Michelson, "Robust vessel segmentation in fundus images," *International Journal of Biomedical Imaging*, vol. 2013, Article ID 154860, 11 pages, 2013.

Hindawi
International Journal of Optics
Volume 2021, Article ID 4151482, 19 pages
https://doi.org/10.1155/2021/4151482

Research Article

Noise Resilient Local Gradient Orientation for Content-Based Image Retrieval

Samina Bilquees,[1] Hassan Dawood,[1] Hussain Dawood,[2] Nadeem Majeed,[3] Ali Javed,[4] and Muhammad Tariq Mahmood[5]

[1]*Department of Software Engineering, University of Engineering and Technology, Taxila 47080, Pakistan*
[2]*Department of Computer and Network Engineering, College of Computer Science and Engineering, University of Jeddah, Jeddah 21589, Saudi Arabia*
[3]*Punjab University College of Information Technology (PUCIT), University of the Punjab, Lahore, Pakistan*
[4]*Department of Computer Science, University of Engineering and Technology, Taxila 47050, Pakistan*
[5]*Future Convergence Engineering, School of Computer Science and Engineering, Korea University of Technology and Education, Cheonan, Republic of Korea*

Correspondence should be addressed to Muhammad Tariq Mahmood; tariq@koreatech.ac.kr

Received 29 April 2021; Revised 25 June 2021; Accepted 5 July 2021; Published 14 July 2021

Academic Editor: E. Bernabeu

In a world of multimedia information, where users seek accurate results against search query and demand relevant multimedia content retrieval, developing an accurate content-based image retrieval (CBIR) system is difficult due to the presence of noise in the image. The performance of the CBIR system is impaired by this noise. To estimate the distance between the query and database images, CBIR systems use image feature representation. The noise or artifacts present within the visual data might confuse the CBIR when retrieving relevant results. Therefore, we propose Noise Resilient Local Gradient Orientation (NRLGO) feature representation that overcomes the noise factor within the visual information and strengthens the CBIR to retrieve accurate and relevant results. The proposed NRLGO consists of three steps: estimation and removal of noise to protect the local visual structure; extraction of color, texture, and local contrast features; and, at the end, generation of microstructure for visual representation. The Manhattan distance between the query image and the database image is used to measure their similarity. The proposed technique was tested using the Corel dataset, which contains 10000 images from 100 different categories. The outcomes of the experiment signify that the proposed NRLGO has higher retrieval performance in comparison with state-of-the-art techniques.

1. Introduction

The advent of multimedia tools has made it easier to access a wider range of images with a variety of information. CBIR (content-based image retrieval) uses low-level image properties to search a huge database for images that suit the user's demands. Color, edges, and orientation are considered low-level image attributes [1]. CBIR is widely used in different applications like medical imaging [2], E-commerce [3], and digital libraries [4].

Text-based image retrieval (TBIR) and sketch-based image retrieval are two image retrieval methodologies. TBIR search images on the basis of labels and keywords. It is mostly used in Google Images. Therefore, it become difficult to describe whole image content in words and it may show irrelevant content. Manual annotation becomes difficult when there is large database [5]. A sketch-based image retrieval system is used to find images on the basis of sketch content drawn by user [6]. Soft Histogram of Edge Local Orientation (S-HELO) uses local orientation computation for improving the performance of sketch-based image retrieval [7]. The proposed histogram of line relationship (HLR) solves the appearance problem between sketches and images. It removes the noisy edges by choosing suitable edge shape that best corresponds to object boundaries [8]. Manual annotations of text-based and user-based sketch

techniques cause incompatible image retrieval. Xie et al. proposed analogy relevance feedback method based on the user intentions that is helpful in maximizing retrieval results [9]. Case Based Long Term Learning (CB-LTL) method of relevance feedback [10] incorporates user intentions and increases retrieval performance. Mandal et al. proposed signature-based bag of visual words (S-BoVW) that employs jumble of words with image's texture and color to retrieve similar images [11]. BoVW based methods improve the retrieval performance but their computation is expensive.

In this paper, we propose Noise Resilient Local Gradient Orientation (NRLGO) that overcomes the noise factor within the visual information and strengthens the CBIR to retrieve accurate and relevant results. In our proposed NRLGO, the color, texture, and local contrast information were used for feature representation of visual data. While semantic attributes are estimated through calculating the correlation between visual features and regular structure data. NRLGO has noise resilience characteristics and saves the local structures of visual information from noise by finding an uncertain state. The major contribution of NRLGO is summarized as follows:

(1) The NRLGO holds a noise resilient attribute by finding the uncertain bit within the visual information which is significant to increase the CBIR system's performance.

(2) NRLGO computes pixel value and difference and then multiplies them by the same constant value. In this way, the bright changes will be canceled and the discriminative feature representation is improved.

(3) Multiresolution gradient orientations of NRLGO improve the gradient magnitude by removing noise factor and detecting the edges.

(4) NRLGO estimates the gradient orientation through dividing the θ value into T dominant orientations. Thus, gradient orientation strengthens the visual feature representation of image.

$$\arctan 2\left(u_s^{11}, u_s^{10}\right) = \begin{cases} \theta, & u_s^{11} > 0 \text{ and } u_s^{10} > 0, \\ \pi - \theta, & u_s^{11} > 0 \text{ and } u_s^{10} < 0, \\ \theta - \pi, & u_s^{11} < 0 \text{ and } u_s^{10} < 0, \\ -\theta, & u_s^{11} < 0 \text{ and } u_s^{10} > 0. \end{cases} \quad (1)$$

The remainder of the paper is structured as follows: The related work is given in Section 2. The suggested NRLGO is presented in Section 3 which discusses the attributes of NRLGO against noise, illumination variation, and rotation variation. Section 4 summarizes the experimental findings and our contribution against state-of-the-art techniques. In Section 5, discussion is given, and Section 6 concludes our contribution.

2. Related Work

The CBIR system plays a vital role in retrieving images that are related from large dataset that are semantically and contextually similar to search image. The CBIR systems extract visual feature representation and estimate the similarity distance for scoring the images that are comparable from the visual database. In the CBIR systems, visual feature descriptors were used in two ways: local and global descriptors. The local descriptors extract image's local interest points while global descriptors take the complete image for feature extraction [12]. Different CBIR methods are used to achieve efficient retrieval performance. The scale invariant feature transform (SIFT) [13] was introduced as a local descriptor, and it is invariant to scale and rotation. To deal with the difficulty of image retrieval using a high-dimensional feature vector, the PCA-SIFT [14] was used. The Hessian matrix and distance ratio were used to solve the computational issue of speeded up robust features (SURF) [15]. The local binary pattern (LBP) based descriptor was proposed by [16]; however, LBP is applicable for gray-scale images only. Moreover, LBP does not perform accurate prediction in presence of noise. This limitation of LBP was overcome by multichannel decoded LBP [17]. The multichannel decoded LBP [17] performs computation on the color and gray-scale images. To further improve the performance of LBP, the fusion of color histogram with LBP was applied to achieve better retrieval performance for color images [18]. To minimize noise in regular patterns, the local ternary pattern (LTP) [19] was developed. The LTP extracts the spatial features in three directions while LBP extract textual features in two directions. The extended local ternary pattern (ELTP) [20] is robust and overcomes the noise within the images. The scale invariant local binary pattern (SILTP) [21] was used to extract local features for gray-scale images and estimate the pixel difference for complex background scenes. Edge detection Sobel filter calculates orientation of edges but has low signal to noise ratio [22]. Edges obtained from Sobel filter have large thickness which gives mismatched images. Canny filter [23] solved the issue of thickness of edges. It has good signal to noise ratio.

Color and image edges are sensitive to the human visual system. In HSV color space, CDH is utilized to improve human visual perception. Entropy is used for feature selection, and correlation is developed between the features after feature selection [24]. The use of visual feature discrimination to measure the similarity between the image passed by the user and images in the database is proposed in an approach based on weight learner [25]. Spatial pyramid matching (SPM) is proposed for spatial distribution of images that increased the retrieval accuracy, but when image alignment is not done properly, it results in rotation and translation variance. Geometric relation is established on the basis of center of image among a group of similar words [26]. Color feature is fused with texture feature to achieve highest retrieval performance as a single attribute is not more resilient for obtaining efficient retrieval performance. As a result, for color extraction, color moment is used, and Gabor descriptor is used to extract the texture feature. After that, features are represented using the Color and Edge Directivity Descriptor (CEDD) [27]. The images are first transformed to RGB color space. Texture information is obtained with the help of evaluated local binary pattern and by predefined prepattern unit. This texture information is then

fused with color channels to increase the retrieval performance [28]. Feature extraction [29] is described by fusing top-hat transforms and local binary patterns for other image processing cases to analyze the color images. Top-hat transform is used to extract the shape, and color local binary pattern is used for texture classification.

In [30], extraction of color feature is done by color difference histogram, and Gabor descriptor is used for the texture classification. Joint property of color with the texture increases the retrieval accuracy. Rotation invariant and gray-scale invariant property is achieved by using the texture feature with color feature. Another technique that uses the color and texture features is developed for image retrieval. HSV and Lab color space are used to extract the color attribute, and texture classification is done by using Haralick features in RGB and gray-scale images [31]. Plant disease [32] is identified by using color, and texture classification is done by using gray-level cooccurrence matrix (GLCM).

CBIR systems are used to represent the visual information in global context based on different deep learning techniques. Bilinear convolutional neural network was proposed for efficient retrieval performance that consists of CNN architecture with bilinear root pooling [33]. Similarly, another deep learning method for CBIR which extracts global features through two parallel CNN models was presented. In [34] SPoC global descriptor was developed through aggregation of deep learning local features for CBIR. The cross-dimensional CNN weights were aggregated to represent the global information for CBIR [35]. The triplet network [36] was used for optimizing the ReLU max aggregation convolutions (R-MAC). The R-MAC involves the pooling of regions of image to cover the point of interest. Moreover, the deep CNN model [37] was used to compress the image descriptors and activate the layers of CNN. The multilayered CNN was used for features extraction, and features were encoded with VLAD encoding scheme [38]. The CNN based CBIR systems require large amount of data for model learning and robust machines for computation of loss and hyperparameters, while local features and distance based CBIR approaches require no clustering and are computationally less expensive than deep learning methods.

The CBIR systems have been studied for decades. Still, occlusion, cluttered background, viewpoint variation, and noise make the retrieval process a challenging task. The literature reveals that all the existing methods for CBIR systems were established using clear visual information, and little effort was made to remove the noise within the visual information. However, presence of noise in visual information is responsible for degraded performance of local or global descriptors (deep learning methods). This is due to the fact that the visual information might be occluded with noise, or maliciously designed data was introduced within the visual information to deceive the CBIR models. Therefore, noise resilient local descriptor is required to overcome the noise or artifacts within the images for CBIR.

3. Noise Resilient Local Gradient Orientation

The CBIR systems compute the discriminative low-level attribute of image including color, edges, and patterns. The distribution of pixels within the visual information is uniformed and represents the discriminative attributes of visual information like color, texture, and patterns. The uniform distribution of pixels is sensitive to small uncertain changes and these small uncertain changes disturb the uniform pixel distribution and cause nonuniform patterns within the image. This uncertain noise confuses the CBIR models to retrieve relevant information. Therefore, we proposed Noise Resilient Local Gradient Orientation (NRLGO) to overcome the noise patterns and sustain the uniform pattern of the pixels. The proposed NRLGO consists of three steps: estimation of noise and protection of the local visual structure; then low-level feature extraction (color, texture, and contrast); and generation of microstructures, as illustrated in Figure 1. The nonuniform noise signal was estimated, and resilient model was established through error correction method. Then, discriminative color and texture features were estimated through quantizing HSV color space and local gradient orientation (LGO). Moreover, NRLBP exacts contrast information through quantizing V values of HSV. The Manhattan distance between the search image and the database image is used to determine their similarity. The proposed NRLGO method is completely described in Figure 2.

3.1. Noise Resilient Local Binary Pattern. The proposed NRLBP establishes a noise resilient attribute through computing the small pixel difference and estimating the uncertain bit Y and uncertain state. The uncertain bit Y and uncertain state were estimated on the basis of corrected noise-free bits within the image. The intensity difference x_p is calculated between center pixel and its corresponding neighboring pixel through the three-state fuzzy logic's B_{p-1}^N representation, and s is threshold which is chosen as 2 for NRLBP [39].

$$B_{p-1}^N = \begin{cases} 1, & \text{if } x_p \geq s, \\ U, & \text{if } \left|x_p\right| < s, \\ 0, & \text{if } x_p \leq -s. \end{cases} \tag{2}$$

The noise factor is considered as uncertain bit and occurs either from 0 to 1 or from 1 to 0. The uncertain bit $Y = (y_1, y_2, \ldots, y_n)$ represents the feature vector where $y_i, y_i \in \{0, 1\}$ is formed by variable$n$. The uncertain state $U(Y)$ can be mathematically represented by the following equation:

$$B_{p-1}^N B_{p-2}^N, \ldots\ldots\ldots\ldots, B_1^N B_0^N = U(Y). \tag{3}$$

The regular visual patterns involve edges, edge-ends, corners, etc. The regular patterns exist more frequently than irregular patterns within the images. As regular expression for normal visual patterns exists, it is possible to estimate the regular expression of uncertain state U from the regular visual patterns present within the image.

NRLBP corrects the irregular patterns to regular patterns by removing noise on the basis of error correction mechanism. For example, the image shown in Figure 1 holds an

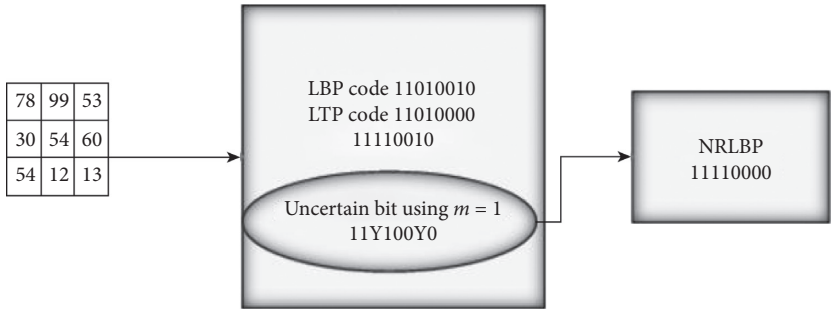


Figure 1: NRLBP encoding scheme in addition to comparison of LBP and LTP. NRLBP uniform code is formed by finding the uncertain bit Y at $m = 1$, and s is chosen as 2 for LTB and NRLBP.

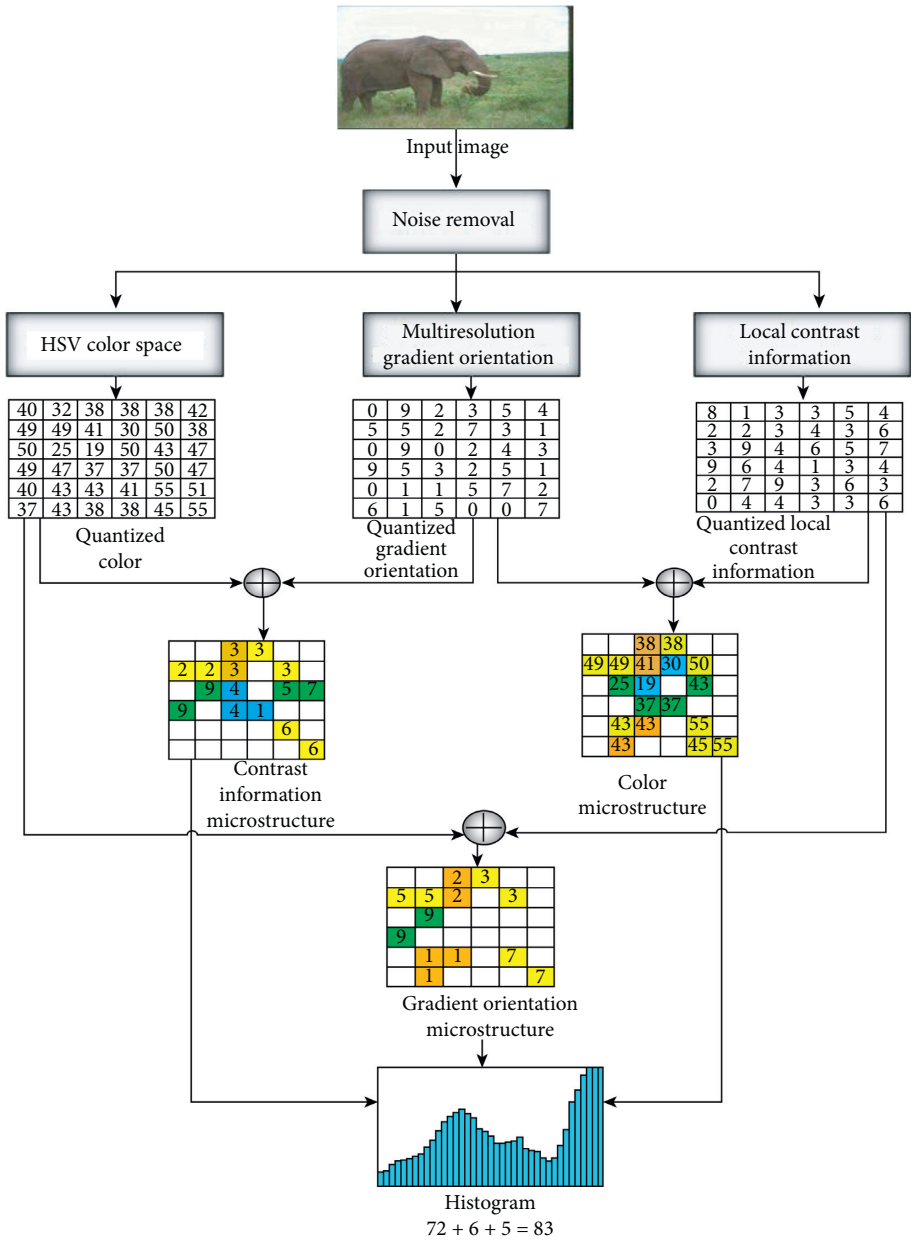


Figure 2: Visual representation of proposed NRLGO descriptor for CBIR system. The proposed NRLGO consists of three steps: estimation and removal of noise to protect the local visual structure; then the color and texture feature extraction; and, at the end, microstructure generation.

uncertain code 11**Y**100**Y**0. In order to remove the noise, there is a need to predict the uncertain bit and then modify the uncertain bit to form uniform patterns, for example, 11110000 and11111000. Suppose that φ_r represents the set of all regular LBP patterns. For instance, the image consists of 58 regular patterns. On the basis of uncertain state $U(Y)$, a list of NRLBP codes is provided as follows:

$$R_{\text{NRLBP}} = \left\{U(Y) | y \in \{0,1\}^t\right\}, \quad U(Y) \in \varphi_r. \tag{4}$$

The uniform code of 11110000 is obtained after error correction, and noise was removed. After that, the histogram of NRLBP for a local region of image was formed, and the number of elements in R_{NRLBP} is represented by n. The bin of histogram was representing each element of R_{NRLBP} which was added by $1/n$ when $t > 0$. When $t = 0$, the irregular bin was added by 1. The same process was repeated for each pixel in the region. Furthermore, NRLBP has reduced histogram bins from 118 to 59 bins. The feature vector of the local image region can be generated by summing the feature vectors collected from each pixel in the image region.

3.2. Feature Extraction. Feature extraction includes the extraction of color, texture, and local contrast information. Converting RGB to HSV color space extracts the color characteristic. HSV (high-saturation value) color space is quantized into 72 colors with, 8, 3, and 3 bins. Texture classification is done by using multiresolution gradient orientation. It improves the gradient magnitude by removing noise factor and detecting the edges. The V value of the HSV color space is used to acquire local contrast information.

3.2.1. Color Feature. The chrominance signals of visual information hold significant discriminative information to represent the images in CBIR models. To reflect the chrominance attribute of visual information, different color spaces are used like YCbCr, HSV, HIS, Lab. The RGB color model is made up of three components: red, green, and blue. In proposed method, HSV color space was selected to segregate the hue, saturation, and intensity of image chrominance.

In addition, the HSV color model represents the chrominance attribute of image, and it has cylindrical representation with hue (H), saturation (S), and value (V). The hue component of the HSV color space describes the wavelength of colors ranging from 0 to 360, beginning with red at 0, yellow at 60, green at 120, cyan at 180, blue at 240, and magenta at 300. The saturation component describes the chrominance saturation level between 0 and 1, where 0 denotes gray and 1 denotes primary color. The value V component describes the intensity values between 0 and 1, where 1 reflects black and 1 reflects white. The difference between HSV and RGB is that the HSV separates the intensity values from the color element. The RGB system is based on three color elements: red, green, and blue.

The HSV color space is color invariant because the V component of HSV represents the image's brightness and is independent of color [24]. With 8, 3, and 3 bins, the HSV color space is quantized into 72 colors. The quantization is essential in persevering images from light and intensity and also reduces the time complexity.

$$\begin{aligned} q_h &= \left\{h \times \left(\frac{b_h}{\max_h}\right)\right\}, \\ q_s &= \left\{s \times \left(\frac{b_s}{\max_s}\right)\right\}, \\ q_v &= \left\{v \times \left(\frac{b_v}{\max_v}\right)\right\}, \end{aligned} \tag{5}$$

where b_h, b_s, and b_v represent the quantized bins of hue, saturation, and value component of HSV [12].

$$cm = q_h \times (b_s \times b_v) + q_s \times b_v + q_v. \tag{6}$$

The quantized values for hue (h), saturation (s), and value (v) of HSV in color map cm are q_h, q_s, and q_v. n_c is the number of quantized colors. s and t are spatial coordinates in quantized color map q_{cm}.

$$q_{cm}(j) = \{(s,t) | (s,t) \in_{cm} = j, \quad 0 \le j \le n_c - 1\}. \tag{7}$$

3.2.2. Local Gradient Orientation (LGO). For gradient computation, the orientation component of Weber local descriptor (WLD) was applied to HSV color model. This is due to the fact that by using the gray-scale images majority of chrominance information is lost. Therefore, the local gradient orientations were extracted, which holds rotation invariant attribute from chrominance part and is beneficial to obtain discriminative features. The gradient orientation [40] was calculated from the angle between the reference axis and the vector in horizontal and vertical location x.

$$\theta(y_c) = \alpha_s^1 = \arctan\left(\frac{u_t^{11}}{u_t^{10}}\right), \tag{8}$$

$f_{10} = \begin{bmatrix} -1 \\ +1 \end{bmatrix}$ and $f_{11} = \begin{bmatrix} -1 \\ +1 \end{bmatrix}$. When filters f_{10} and f_{11} are applied to an input image, then outputs u_t^{11} and u_t^{10} are obtained. $u_t^{10} = y_5 - y_1$ and $u_s^{11} = y_7 - y_3$.

θ is quantized into T gradient orientations. Firstly, mapping is done before quantization $f: \theta \longrightarrow \theta'$: $\theta' = \arctan 2(u_s^{11}, u_s^{10}) + \pi$ and

$$\arctan 2\left(u_s^{11}, u_s^{10}\right) = \begin{cases} \theta, & u_s^{11} > 0 \text{ and } u_s^{10} > 0, \\ \pi - \theta, & u_s^{11} > 0 \text{ and } u_s^{10} < 0, \\ \theta - \pi, & u_s^{11} < 0 \text{ and } u_s^{10} < 0, \\ -\theta, & u_s^{11} < 0 \text{ and } u_s^{10} > 0, \end{cases} \tag{9}$$

$\theta \in [-\pi/2, \pi/2]$, and $\theta' \in [0, 2\pi]$. Mapping takes value of θ under consideration.

Quantization function is given as follows.

$\varphi_s = f_r(\theta') = (2q/Q)\pi$, and $q = \mathrm{mod}(\theta'/(2\pi/T) + t(1/2))$. If $T = 8$ then $\varphi_s = (s\pi)/4$, $(s = 0, 1, T-1)$.

The multiscale gradient orientation is useful for extracting gradient orientation to describe different granular aspects of visual representation. Multiscale gradient orientation is able to improve the discriminative ability of a single resolution. It is obtained by using square neighbors of P pixels, considering the length of $(2R+1)$. As illustrated in Figure 3, P denotes the set of neighbors, and R denotes the spatial resolution.

The multiscale gradient orientation was obtained through combining the histograms of different operators at varying P and$R$. By varying the P and R, whole image pixel values are considered. A multiscale analysis of WLD orientation can be done using the data generated by several operators of varying scales (P, R). Despite the fact that the operator is based on a squared symmetric neighbor set of P members on a square with side length $(2R+1)$, it can also be used for a circular situation. In general, it can improve the discrimination of a single resolution (P, R).

The WLD orientation is used for texture estimation, and it also eliminates the effect of noise. The WLD orientation reduces the noise and illumination variation by computing the differences between the pixel and its neighbors. WLD orientation is tolerant of the presence of noise in an image. The influence of noise is reduced by using a WLD orientation, which is analogous to smoothing in image processing. Furthermore, the sum of its p-neighbor differences is divided by the current pixel's intensity, reducing the influence of noise in an image.

WLD orientation has also been established to mitigate the impact of changing brightness. It calculates the differences between the center pixel and its neighbors. As a result, a brightness modification that adds a constant to each image pixel has no effect on the disparities in values. WLD orientation, on the other hand, is responsible for dividing the differences. Thus, when each pixel value is multiplied by a constant, the differences are also multiplied by the same constant, canceling out the contrast change. As a result, the description is not affected by changes in brightness.

Finally, quantized orientation map incorporating multiscale granularity is represented as follows:

$$q_{os} = \frac{1}{2}\left(o_{mg}\right) \times \frac{N_o}{180},$$
$$q_{os}(f) = \{(s,t) | s,t \in q_{os} = j\}, \quad 0 \le N_o = 1, \tag{10}$$

where s and t are spatial coordinates of orientation structure and j is the resultant value obtained across quantized value $N_o = 6$.

3.2.3. Local Contrast Information. The pixel intensity is used to describe the local contrast information [41], as the visual information depends on intensity, and specific contrast range improves the visibility of visual information and increases the performance of CBIR systems. The V element of HSV color model was used to extract the intensity feature. This is due to the fact that V component of HSV color model called luma separates the chrominance element of image. The local contrast information map for input image $f(s,t)$ is calculated as [42]

$$\mathrm{lcm}(s,t) = \sum_{s=1}^{U}\sum_{t=1}^{v} \max(p(s,t), \max(H(s,t), b(s,t))). \tag{11}$$

From the mathematical expression represented in (11), the constants s and t are the coordinates of input image $f(s,t)$. The dimensions of input image are represented by m and n. Then, the local contrast information was quantized at 10 levels to retain the salient information. The final intensity map was obtained and mathematically represented by the following expression:

$$q_{\mathrm{lcm}}(s,t) = IM(s,t) * n_i,$$
$$q_{\mathrm{lcm}}(s,t)_l = \{(s,t) | (s,t) \in q_{lcm}(s,t)\} = l, \quad 0 \le l \le n_i - 1. \tag{12}$$

3.3. Interlinked Microstructure Identification. Natural images contain large color, edge, and shape attribute, which are considered as low-level image attributes. More relevant images are retrieved from large datasets on the basis of color, edge, and shape attribute of image. Human eye shows sensitivity toward color and orientation features. Orientation reflects an informative content in image. Strong orientation gives a uniform pattern. As most natural standard images do not contain strong orientation, there is no uniform pattern. Natural images contain uniform and non-uniform patterns contributing spatial information, forming the microstructures. Orientation is not enough to discuss the image's spatial characteristics, so color, edge, orientation, and local contrast information are fused to describe the richer image contents. In this paper, microstructures are proposed on the basis of color, orientation, and local contrast information attributes. Color feature and microstructure which is obtained by associating texture and local contrast information are joined to obtain micro-color structure. Orientation attributes and microstructure which is derived by linking color and intensity features are joined to obtain micro-orientation structure. Micro-intensity map is derived by intensity features and micro-map which is derived by linking orientation and intensity features. As microstructures are combination of color, orientation, and intensity features, they will increase the retrieval performance. Quantized orientation and quantized intensity ranges are used to derive micro-color map. The whole orientation $O_{\mathrm{orientation}}$ is divided into 3×3 small grids [43]. Orientation is scale and rotation invariant, so it is used to derive color map. As orientation is quantized at level 6, it can vary from 0 to 5. For horizontal and vertical location, pixel of 3lengths is defined for each grid. Suppose that $G\wedge_{ii} = G_{ii} + G_{ii+1} + G_{ii+2} + \cdots G_P$ are the grids which involve $O_{\mathrm{orientation}}$ where $ii = 1, 2, \ldots\ldots P$ where P represents length of grid. Similarly, suppose that $G\wedge_{jj} = G_{jj} + G_{jj+1} + G_{jj+2} + \cdots G_Q$ are the grids which involve $O_{\mathrm{intensity}}$ where $jj = 1, 2, \ldots\ldots Q$

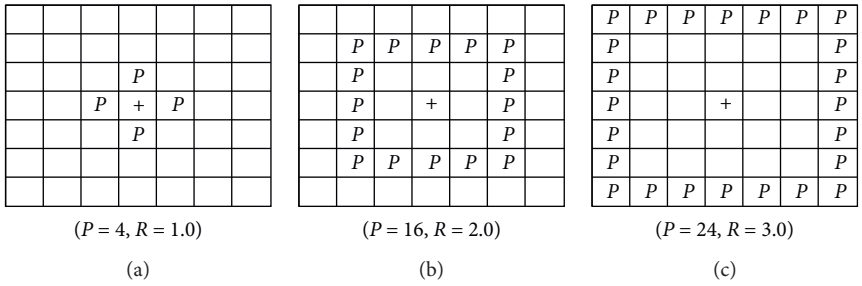


Figure 3: Squared symmetric neighborhood for different (P, R). P represents neighbors and R represents the spatial resolution. Multi-resolution analysis is done by choosing P as 4, 16, and 24 and R as 1.0, 2.0, and 3.0.

where Q represents length of grid. Suppose that G_{ii} is the grid of length 3×3 moved on quantized orientation and is the grid of length 3×3 moved on quantized local contrast information. Relationship is established between the pixels in the center and the pixels in the surrounding areas as a result of resemblance measures, which are helpful in finding the regular patterns. Suppose that cp_o, cp_c are the central pixels and k_i and k_j are neighboring pixels of quantized orientation and intensity microstructures. While moving grid of 3×3, neighbors of quantized orientation and intensity have the same value as central pixels of orientation and intensity; then, uniform or regular patterns and microstructure basic block are obtained. If no values of neighboring pixels of k_i and k_j are the same as those of the central pixels cp_o, cp_c, then irregular patterns are formed and microstructure basic block is not obtained. Five steps are followed to achieve a finally single image's microstructure.

(1) Starting from $(0,0)$, a grid of size 3×3 is moved on both quantized orientation and intensity from the left to the right and from the top to the bottom. The micro-map is established at $(0,0)$ and is labeled as $M^1(s,t)$ on the basis of regular patterns in both quantized orientation and intensity where $0 \le s \le m-1$, $0 \le t \le n-1$.

(2) At $(0,1)$, a grid of size 3×3 is moved on both quantized orientation and intensity from the left to the right and from the top to the bottom. The micro-map is established at $(0,1)$ and is labeled as $M^2(s,t)$ on the basis of regular patterns in both quantized orientation and intensity where $0 \le s \le m-1$, $1 \le t \le n-1$.

(3) At $(1,0)$, a grid of size 3×3 is moved on both quantized orientation and intensity from the left to the right and from the top to the bottom. The micro-map is established at $(1,0)$ and is labeled as $M^3(s,t)$ on the basis of regular patterns in both quantized orientation and intensity where $1 \le s \le m-1$, $0 \le t \le n-1$.

(4) At $(1,1)$, a grid of size 3×3 is moved on both quantized orientation and intensity from the left to the right and from the top to the bottom. The micro-map is established at $(1,1)$ and is labeled as $M^4(s,t)$ on the basis of regular patterns in both quantized orientation and intensity where $1 \le s \le m-1$, $1 \le t \le n-1$.

(5) Final micro-map of image is obtained and demonstrated as (x) by combining all four maps.

4. Feature Quantization

Extraction of discriminative set of attributes for retrieving images is a difficult task. After obtaining discriminative set of features, the second task is the image's attribute representation, that is, how to represent the extracted features. In this work, microstructures are suggested for feature representation. The main steps involved in forming correlated microstructures are shown in Figure 4.

4.1. Color Feature. Microstructure's value for the input image $f(s,t)$ is demonstrated as $f^1(s,t) = a$, where $a \in 0, 1, 2, \ldots\ldots, m-1$ and m reflects the dimensionality of micro-color structure which is obtained on the basis of the relationship between microstructure of quantized orientation and intensity. The equation used to derive the microstructure features is as follows:

$$G(a) = \frac{L_1\{f(Pa_0) = a_0, f(Pa_i) = a_i | Pa_i - Pa_0| = 1\}}{8\,\overline{L_1}\{f(Pa_0) = a_0\}}, \quad a_0 = a_i, i = \{1, 2, 3, \ldots 8\}. \tag{13}$$

Every block of input image is 3×3, supposing that Pa_o reflects the central pixel with position $Pa_o = (s_o, t_o)$ and Pa_i reflects the neighboring pixels of central pixel with position $Pa_i = (s_i, t_i)$. The value for the central pixel is $Pa_o = e$ neighboring pixels and denotes the number of cooccurring values for a_o and a_i. $\overline{L}_1$ is used to represent the number of occurrences of a_o. The micro-color structure is derived by establishing the relationship between microstructure image

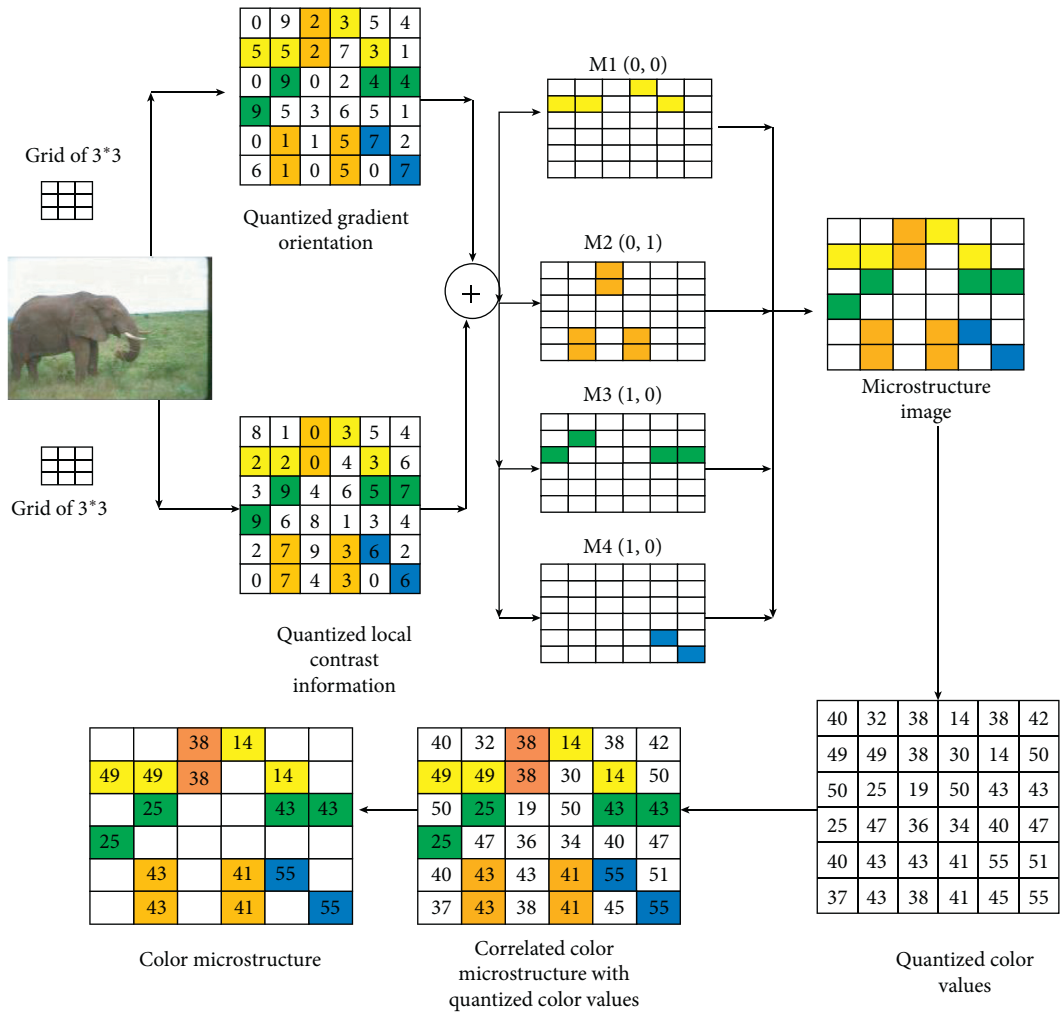


Figure 4: Correlated microstructure. It is formed by correlating the quantized color, orientation, and local contrast information quantized values.

$f^1(s,t)$ and quantized color structure q_{cs}. Final micro-color structure is obtained from quantized color map's pixel values, which are in regular region present in $f^1(s,t)$. The features of micro-color structure have 72 dimensions.

4.2. Orientation Feature. Microstructure's value for the input image $f(s,t)$ is demonstrated as $f^2(s,t) = b$, where $b \in 0, 1, 2, \ldots, m-1$, and m reflects the dimensionality of micro-color structure which is obtained on the basis of the relationship between microstructure of quantized orientation and intensity. The equation used to derive the microstructure features is as follows:

$$G(b) = \frac{L_2\{f(Pb_0) = a_0, f(Pb_i) = a_i | Pb_i - Pb_0| = 1\}}{8\overline{L_2}\{f(Pb_0) = b_0\}}, \quad b_0 = b_i, i = \{1, 2, 3, \ldots, 8\}. \tag{14}$$

Every block of input image is 3×3, supposing that Pb_o reflects the central pixel with position $P_o = (s_o, t_o)$ and Pb_i reflects the neighboring pixels of central pixel with position $P_i = (s_i, t_i)$. The value for the central pixel is $Pb_o = e$ neighboring pixels and denotes the number of cooccurring values for b_o and b_i. $\overline{L}_2$ is used to represent the number of occurrences of bo. The micro-color structure is derived by establishing the relationship between microstructure image $f^2(s,t)$ and quantized orientation structure q_{os}. Final micro-color structure is obtained from quantized color map's pixel

values, which are in regular region present in $f^2(s,t)$. The features of micro-orientation structure have 6 dimensions.

4.3. Local Contrast Information. Microstructure's value for the input image $f(s,t)$ is demonstrated as $f^3(s,t) = c$, where $c \in 0, 1, 2, \ldots, m-1$, and m reflects the dimensionality of micro-color structure which is obtained on the basis of the relationship between microstructure of quantized orientation and intensity. The equation used to derive the microstructure features is as follows:

$$G(c) = \frac{L_3\{f(Pc_0) = c_0, f(Pc_i) = c_i | Pc_i - Pc_0| = 1\}}{8\overline{L_3}\{f(Pc_0) = c_0\}}, \quad c_0 = c_i, i = \{1, 2, 3, \ldots, 8\}. \tag{15}$$

Every block of input image is 3×3, supposing that $\mathrm{Pc_o}$ reflects the central pixel with position $P_o = (s_o, t_o)$ and Pc_i reflects the neighboring pixels of central pixel with position $P_i = (s_i, t_i)$. The value for the central pixel is $Pc_o = e$ neighboring pixels and denotes the number of cooccurring values for c_o and c_i. $\overline{L}_3$ is used to represent the number of occurrences of c_o. The micro-color structure is derived by establishing the relationship between microstructure image $f^3(s,t)$ and quantized local contrast structure q_{lcs}. Final micro-color structure is obtained from quantized color map's pixel values, which are in regular region present in $f^3(s,t)$. The features of micro local contrast information structure have 10 dimensions.

5. Experimental Results and Findings

In this section, experiment is performed on Corel dataset to evaluate the retrieval performance of NRLGO method. The performance of NRLGO is given as follows.

5.1. Dataset. Different databases are used for evaluating the retrieval accuracy of images. In our proposed work, Corel database is used for measuring efficiency of NRLGO. Corel database consists of Corel 1k, Corel 5k, and Corel 10k. Corel 1k contains 1000 images of different contents such as Africans, mountains, elephants, buildings, beaches, and horses. There are 10 categories and 100 images per category with dimension of 256×384. Corel 5k is derived from Corel 10k and contains 5000 images of different contents such as waves, trees, lions, ducks, and food. There are 50 categories and 100 images per category. Corel 10k is a large dataset and contains 10000 images of different contents such as 250 cars, flowers, trains, furniture, butterflies, and tractors. There are 100 categories and 100 images per category with dimension of 192×128.

5.2. Evaluation Parameters. Precision and recall are the most important parameters for evaluating retrieval performance. Precision of input image is described as ratio of number of images retrieved that are similar to total number of images retrieved. Recall of input image is defined as ratio of similar images retrieved to total number of similar images.

$$P_I = \frac{N_s}{M}, \qquad R_I = \frac{N_s}{M_s}, \tag{16}$$

where M reflects the total number of images that have been retrieved and M_s reflects the total number of similar images that have been retrieved. Average precision and recall are also calculated to measure the image retrieval performance being 100; N_c describes the total number of images of Corel dataset in each category; and values 10, 50, and 100 describe the number of categories of Corel dataset. Average retrieval precision and recall are calculated by (18) and (20).

$$\mathrm{RP}_I = \frac{1}{M_c}\sum_{j=1}^{M_c} P_I, \tag{17}$$

$$\mathrm{ARP} = \frac{1}{Q_c}\sum_{k=1}^{Q_c} \mathrm{RP}_I, \tag{18}$$

$$\mathrm{RR}_I = \frac{1}{M_c}\sum_{j=1}^{M_c} R_I, \tag{19}$$

$$\mathrm{ARR} = \frac{1}{Q_c}\sum_{k=1}^{Q_c} \mathrm{RR}_I. \tag{20}$$

F-score has characteristics of both precision and recall, and it combines precision and recall into single similarity measure as shown in the following:

$$F_s = \frac{2 \times P_I \times R_I}{P_I + R_I}. \tag{21}$$

5.3. Distance Metric. Manhattan distance (L1) is used to measure the similarity between the search image and the images in dataset by using the following equation:

$$D(u,v) = \sum_{i=1}^{N_i} |u_I - v_i|, \tag{22}$$

$u_i = u_1, u_2, \ldots\ldots, u_N$ is the collection of attributes extracted from database's images and $v_i = v_1, v_2, \ldots\ldots, v_N$ is the collection of attributes extracted from search image, where W_i and $V_j = [1, 2, 3, \ldots, N_i]$, where $N_i = 83\,(72 + 6 + 5)$ is the dimension of feature vector. Manhattan distance is good to measure similarity for a large image database by reducing the computational costs.

5.4. Performance of NRLGO. Color, texture, and local contrast information are used to retrieve relevant images from large dataset. In proposed NRLGO, color, edges, and local contrast information are extracted from the color image. The effectiveness of NRLGO is obtained by using 192, 128, 108, and 72 color quantization levels; 6, 12, 18, 24, 30, and 36 orientation quantization levels; and 10, 15, and 20 local contrast information quantization levels.

Tables 1–3 demonstrate the efficient performance on Corel dataset with 5000 images by changing color and texture levels by keeping local contrast information values fixed at 5, 10, and 15. In Table 1 highest precision of 65.82% is achieved when $N_c = 72$ and $N_o = 6$, and in Table 2 highest precision of 65.32% is achieved when $N_c = 128$ and $N_o = 6$. Moreover, in Table 3 highest precision of 65.58% is achieved when $N_c = 128$ and $N_o = 24$. So, in the proposed NRLGO, we obtained the descriptor of dimensions $(72 + 6 + 5 = 83)$ by setting values for color $N_c = 72$, texture $N_t = 6$ and local contrast information $N_{ci} = 5$. It is observed that, in Tables 1–3, ARP decreases when $N_c = 128$; N_{ci} is set to 5, 10, and 15; and N_o values are selected from 6 to 36. This decrement is basically because of sensitivity to visual system toward continuously changing texture orientations. In some cases, nonuniform pattern of average retrieval precision is noticed; for example, in Table 3, when $N_c = 72$ and $N_{ci} = 15$, average retrieval precision firstly increases for N_o from 6 to 18, then decreases at $N_o = 24$, and again increases for N_o from 30 to 36. This irregular pattern occurred because of noise due to quantization that increases variability within the class and decreases the performance of NRLGO.

Table 4 shows the ARP obtained on Corel 5k by varying N_o from 6 to 36 and N_c from 16 to 32, 64, and 128 in RGB color space. Highest precision is achieved when color quantization is 128 and texture level is 6 at intensity level 5. It is shown from the table that HSV color space gave higher retrieval performance. As RGB color channels ignore the color characteristics, irregular pattern is obtained. Thus, by using HSV color space, average precision and recall have been increased. HSV color space is used in NRLGO. Not only is it effective toward user intentions, it also takes standard color characteristics into considerations. Hence, HSV color space increases average retrieval performances. Table 5 demonstrates retrieval accuracy obtained on Corel 1k, Corel 5k, and Corel 10k by using different similarity measures. In the proposed method, NRLGO, bin-by- bin similarity measures are applied. Similarity is measured between the search image and database image by using Manhattan distance. L1 increases the retrieval performance: performance of 83.5% on Corel dataset with 1000 images, 65.82% on Corel dataset with 5000 images, and 53.07% on Corel dataset with 10000 images. L1 does not perform any square or square root calculation, so it has high performance for large datasets, while L2 has poor performance at higher distance values and has high computation. It is clear from Table 5 that average retrieval accuracy is smaller on Corel dataset with 5000 images and Corel dataset with 10000 images by using Euclidean distance.

Euclidean distance includes the square operation, so it does not perform well on larger distance values and is also computationally expensive. Furthermore, on Corel 1k, retrieval accuracy is less at square chord. Color, texture, and local contrast information attributes are selected for proposed method NRLGO. However, NRLGO performance is observed at different color, texture, and local contrast information's combinations. ARP and ARR at 7 different combinations are shown in Table 6 for performance comparisons.

It is evident that when the combination of color, texture, and local contrast information is used, better average retrieval precision (ARP) and average retrieval recall (ARR) are achieved. Besides, poor performance is achieved when using only color, texture, or local contrast information. Local contrast information is an essential feature as it distinguishes images on the basis of difference in illumination; however, it is less emphasized for efficient image retrieval methods. Standard images contain more color and intensity features. Therefore, contrast information is useful to distinguish between the images with the same contrast and the images with different contrast information. Proposed method NRLGO gave better retrieval performance in all combinations of color, texture, and local contrast information as compared to other descriptors on Corel dataset with 1000, 5000, and 10000 images. The retrieval accuracy of NRLGO descriptor is observed on Corel 1k in comparison with MTH [44], MSD [43], CMSD [2], CDH [45], SED [46], and ENN [47] descriptors. Table 7 shows the efficient retrieval accuracy of NRLGO in comparison to the state-of-the-art techniques at different categories of Corel dataset with 1000 images. Texture is penetrating feature in beach and mountain images. It is observed that beach category has variation in textural classifications. Therefore, human perception is more sensitive toward textural variations.

Table 7 shows that dinosaur's category has 100% retrieval performance. Category-wise result evaluation shows that average retrieval precision is low at beach category, i.e., 50.75%, in comparison with MTH, MSD, SED, ENN, CDH, LeNET-F6, and CMSD. In beach category, most of mountain category images are retrieved, so this results in decrement of ARP in both mountain and beach category. In Table 8, 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 images are retrieved; a total of 100 images are retrieved per category; and R is the relevant images retrieved. Highest precision is achieved at top 10 images retrieved, and lowest precision is achieved at 100 images retrieved. To achieve multiresolution property that increases the retrieval accuracy, we performed experiment by varying P and R as shown in Table 9. Retrieval accuracy is increased when we select the value of P equal to 24 and R equal to 3.0. Similarly, we used different threshold values and achieved highest precision and recall when we

Table 1: Average retrieval precision and average retrieval recall of NRLGO at varying values of gradient orientation and color quantization levels by keeping local contrast information fixed at 5 on Corel 5k in HSV.

	Texture classification (Corel 5k, contrast information = 5)											
Color quantization	Precision (%)						Recall (%)					
	6	12	18	24	30	36	6	12	18	24	30	36
192	62.1	63.11	63.10	62.60	62.68	62.79	7.46	7.57	7.57	7.51	7.52	7.53
128	63.49	64.12	63.80	64.23	63.16	63.54	7.61	7.69	7.65	7.70	7.58	7.62
108	62.73	62.51	64.05	62.66	62.70	62.40	7.52	7.50	7.68	7.52	7.52	7.48
72	65.8	63.3	63.6	63.4	63.6	64.5	7.89	7.59	7.64	7.61	7.64	7.74

Table 2: Average retrieval precision and average retrieval recall of NRLGO at varying values of gradient orientation and color quantization levels by keeping local contrast information fixed at 10 on Corel 5k in HSV.

	Texture classification (Corel 5k, contrast information = 10)											
Color quantization	Precision (%)						Recall (%)					
	6	12	18	24	30	36	6	12	18	24	30	36
192	64.05	64.26	63.94	64.01	63.47	63.35	7.68	7.71	7.67	7.68	7.61	7.60
128	65.32	64.38	65.04	65.05	64.62	64.56	7.89	7.72	7.80	7.80	7.75	7.74
108	64.71	64.15	63.68	65.07	63.68	63.63	7.76	7.69	7.64	7.80	7.64	7.63
72	64.87	65.30	64.67	65.40	64.35	65.37	7.78	7.83	7.76	7.85	7.72	7.84

Table 3: Average retrieval precision and average retrieval recall of NRLGO at varying values of gradient orientation and color quantization levels by keeping local contrast information fixed at 15 on Corel 5k in HSV.

	Texture classification (Corel 5k, contrast information = 15)											
Color quantization	Precision (%)						Recall (%)					
	6	12	18	24	30	36	6	12	18	24	30	36
192	64.59	63.76	63.96	64.21	64.06	63.85	7.75	7.65	7.67	7.70	7.68	7.66
128	65.47	65.51	64.94	65.58	64.56	64.75	7.85	7.86	7.79	7.87	7.74	7.77
108	64.80	64.30	64.94	63.90	64.69	65.09	7.77	7.71	7.79	7.66	7.76	7.81
72	65.30	65.32	65.46	64.75	65.45	65.16	7.83	7.83	7.85	7.77	7.85	7.82

Table 4: Average retrieval precision and average retrieval recall of NRLGO at varying values of gradient orientation and color quantization levels by keeping local contrast information fixed at 5 on Corel 5k in RGB.

	Texture classification (Corel 5k, contrast information = 5)											
Color quantization level	Precision (%)						Recall (%)					
	6	12	18	24	30	36	6	12	18	24	30	36
128	60.1	60.5	59.9	59.5	59.8	59.7	7.07	7.10	7.11	7.06	7.13	7.09
64	58.5	57.4	57.8	55.2	56.9	58.5	6.80	6.69	6.83	6.81	6.81	6.85
32	53.5	50.2	55.0	55.6	54.7	49.4	6.20	6.23	6.31	6.30	6.27	6.25
16	44.4	48.7	46.6	47.4	47.6	48.5	5.30	5.39	5.47	5.50	5.55	5.57

Table 5: Average retrieval precision and average retrieval recall of NRLGO by varying combinations of gradient orientation and color quantization levels by keeping contrast information fixed at 5 on Corel 5k in RGB.

Dataset	Performance	Distance or similarity metrics					
		Chi-square	Square chord	MTH	Euclidean distance	Shepherd	Manhattan
Corel 1k	Precision (%)	73.5	76.70	83.45	97.91	83.20	83.50
	Recall (%)	8.64	9.20	10.01	11.75	9.98	10.02
Corel 5k	Precision (%)	55.42	52.89	59.64	19.91	64.00	65.8
	Recall (%)	6.83	6.34	7.15	2.39	7.68	7.89
Corel 10k	Precision (%)	47.91	40.96	49.93	10.00	52.72	53.07
	Recall (%)	5.53	4.91	5.99	12.00	6.32	6.36

Table 6: Performance of NRLGO at Corel dataset with 1000, 5000, and 1000 images at different combinations of color, gradient orientation, and local contrast information vector.

Dataset	Performance	Color	Texture	Contrast	Texture + contrast	Color + texture	Color + contrast	Proposed
Corel 1k	Precision (%)	76.8	54.6	52.8	59.4	79.5	78.25	83.5
	Recall (%)	9.02	6.38	6.10	6.90	9.22	9.30	9.96
Corel 5k	Precision (%)	60.2	27.5	31.5	39.5	61.6	61.8	65.8
	Recall (%)	7.15	3.06	3.89	4.50	7.30	7.25	7.89
Corel 10k	Precision (%)	48.5	20.5	25.9	28.6	50.7	49.3	53.07
	Recall (%)	5.51	2.30	3.05	3.45	5.80	5.95	6.36

Table 7: Category-wise performance comparison of NRLGO at Corel 1k with state-of-the-art descriptors.

Category	Performance	MTH [44]	MSD [43]	SED [46]	ENN [47]	CDH [45]	CMSD [12]	LeNet-F6 [48]	Proposed
African	Precision	69.17	83.33	82.50	85	77.50	86.66	80	88.25
	Recall	8.30	10.00	9.90	9.00	9.30	10.40	8.00	10.30
Beach	Precision	61.67	43.33	28.33	75	56.67	42.08	60	50.75
	Recall	7.40	5.20	3.40	7.00	6.80	5.05	6.5	5.15
Building	Precision	45.83	63.33	47.50	70	47.50	81.66	75	85.00
	Recall	5.50	7.60	5.70	6.00	5.70	9.80	8.00	9.95
Bus	Precision	68.33	76.67	73.33	75	71.67	81.66	80	83.70
	Recall	8.20	9.20	8.80	7.00	8.60	9.80	9.00	9.95
Dinosaur	Precision	100.00	100.00	90.00	100	100	100	100	100
	Recall	12.00	12.00	10.80	12	12.00	12.00	12.00	12.00
Elephant	Precision	70.83	65.00	55.00	75	62.50	72.08	85	100
	Recall	8.50	7.80	6.60	7.00	7.50	8.65	10.00	12.00
Flower	Precision	75.00	86.67	72.50	80	60.83	84.16	75	86.20
	Recall	9.00	10.40	8.70	8.00	7.30	10.10	8.00	10.20
Horse	Precision	100.00	97.50	62.50	85	91.67	94.16	90	95.20
	Recall	12.00	11.70	7.50	9.00	11.00	11.30	11.00	11.40
Mountain	Precision	39.17	29.17	40.00	65	44.17	50.00	60	60.05
	Recall	4.70	3.50	4.80	5.00	5.30	6.00	6.5	6.10
Food	Precision	52.50	76.67	64.17	55	45.00	92.91	75	94.9
	Recall	6.30	9.20	7.70	3.00	5.40	11.15	8.00	11.38
Average	Precision	68.25	72.16	61.58	76.5	65.75	78.54	78	83.5
	Recall	8.19	8.66	7.39	7.3	7.89	9.42	8.7	9.96

Table 8: Top 10 to 100 images are retrieved using NRLGO method for precision parameter.

P class	10	20	30	40	50	60	70	80	90	100
African	0.65	0.90	0.73	0.75	0.70	0.66	0.64	0.51	0.61	0.60
Beach	0.40	0.75	0.66	0.60	0.60	0.54	0.52	0.50	0.47	0.46
Building	1	0.95	0.86	0.70	0.72	0.71	0.71	0.58	0.67	0.58
Bus	1	1	1	1	0.98	0.91	0.91	0.86	0.82	0.81
Dinosaur	1	1	1	1	1	1	1	1	1	1
Elephant	1	0.50	0.76	0.60	0.60	0.51	0.48	0.45	0.43	0.33
Flower	1	0.95	0.93	0.87	0.78	0.71	0.64	0.67	0.55	0.52
Horse	1	0.60	1	1	0.88	0.96	0.93	0.98	0.84	0.82
Mountain	0.60	0.70	0.60	0.57	0.44	0.51	0.48	0.57	0.46	0.44
Food	0.70	0.60	0.60	0.57	0.62	0.60	0.58	0.57	0.56	0.53
Average (R\P)	0.835	0.795	0.741	0.76	0.732	0.71	0.68	0.669	0.64	0.609

selected the threshold value equal to 2. NRLGO extracts noise-robust features that remain invariant to variation in scale, illumination, and orientation. Experiment is performed on Corel dataset by adding Gaussian noise for texture recognition. Gaussian Noise of 5, 10, and 15 (%) is added to dinosaur color image

In Table 9, by using different values of P and R, the multiresolution property is analyzed. Highest accuracy is observed when we choose the value of P equal to 24 and R equal to 3.

In Table 10, different values of s are used. Highest precision and recall are observed when we set threshold value equal to $s = 2$.

NRLGO firstly converts the noisy color images to gray-scale images and then extracts the edges perfectly even in the presence of noise. It extracts the edges under different noise

Table 9: Retrieval accuracy observed at 9 different angles at Corel 1k dataset.

(P, R)	(8, 1)	(16, 2)	(24, 3)	Average
Accuracy	82.4	83.5	84.6	83.5

Table 10: Retrieval accuracy observed at different threshold values using Corel 1k dataset. Highest retrieval accuracy is achieved at $s = 2$.

Noise resistant	Threshold values (s)					
	1	2	3	4	6	10
Precision (%)	83.4	83.5	82.8	82.3	82.7	83.2
Recall (%)	9.91	9.96	9.94	9.88	9.75	9.94

values as shown in Figure 5. Accordingly, the accuracy of NRLGO is good in comparison to state-of-the-art descriptors on Corel dataset with 1000 images.

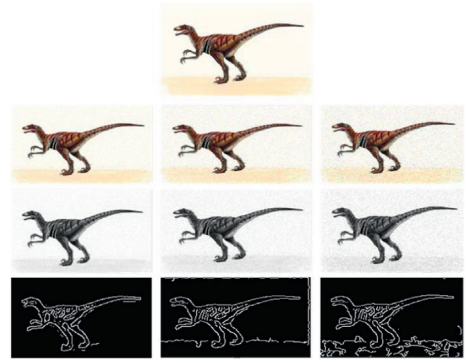


Figure 5: Extraction of edges in presence of Gaussian Noise. Firstly, Gaussian noise of 5, 10, and 15 (%) is inserted. Secondly, conversion of RGB to gray scale is done; then, NRLGO is applied to detect the edges.

6. Discussion

An image retrieval approach, based on Noise Resilient Local Gradient Orientation, that reduces the noise is proposed. Experiment is performed using Corel dataset, i.e., Corel dataset with 1000, 5000, and 10000 images, and the NRLGO achieves higher retrieval accuracy in comparison to state-of-the-art image techniques based on feature extraction techniques.

6.1. Comparison with Other Techniques. Natural images consist of regular and irregular textural patterns. Different textural descriptors like GLCM [25], Gabor features [28], SQ-Spatiogram [49], CMSD [2], and GMM-mSpatiogram [49] are proposed to check the performance on these regular and irregular patterns. It is observed that these textural descriptors perform well on regular patterns; their performance is not good at irregular patterns.

Table 11 demonstrates the retrieval accuracy of proposed NRLGO descriptor with uniform codes as compared to state-of-the-art methods, namely, GLCM [50], SQ-Spatiogram [49], GMM-mSpatiogram [49], SED [25], CMSD [2], and CPV-THF [42]. Experimental result shows that increase in retrieval accuracy of 27.8%, 24.13%, 14%, 9.5%, 2.66%, 1.9%, and 5.5% is achieved as compared to GLCM, SQ-Spatiogram [49], GMM-mSpatiogram [49], SED, CMSD, and CPV-THF on Corel 5k. On Corel 10k, for GLCM, SED, CMSD, CPV-THF, and LeNET-F6 [48], retrieval accuracy increase of 21.17%, 15.43%, 5.82%, 6.67%, 2.82%, and 79% is observed. Table 12 shows retrieval performance of proposed descriptor with respect to texture, color, and shape in comparison with state-of-the-art descriptor at Corel dataset with 5000 images and Corel dataset with 10000 images. Retrieval accuracy of NRLGO descriptor as compared to other descriptors is 29.6%, 26.4%, 29.98%, 17.36%, 27.32%, 5.52%, 2.82%, and 12.06% greater on Gabor [51], EHD [8], color moment [18], LBP [52], STH [53], and MTSD [54] at Corel dataset with 5000 images. At Corel dataset with 10000 images, the accuracy is 23.97%, 20.77%, 26.69%, 18.06%, 5.04%, and 1.12% on Gabor, EHD, color moment, LBP, STH, and MTSD.

Increase in retrieval precision and recall shows that the proposed method outperforms the shape, orientation, and color techniques. Figures 6–8 demonstrate performance of NRLGO with state-of-the-art techniques in form of curve using precision and recall parameters. x-axis is labeled as recall (%) and y-axis as precision (%). Precision-recall curves are drawn opposite to each other. For example, if average retrieval precision of descriptor is high, then obtained curve is drawn away from the point of origin. If ARP is low, then precision-recall will be short. We have also measured performance of NRLGO on the basis of F-score (%) with state-of-the-art methods like GA [55], GMM-mSpatiogram [49], ODBTC [56], Tetrolet Transform [57] and BiCBIR [58] as shown in Table 13 with the increase of F-score of 0.087%, 0.062%, 0.053%, 0.03%, and 0.015%.

When there is uncertainty in the performance of a descriptor, curve with different turning point is obtained. When the performance of two descriptors is the same in image category, then the curves overlapped at some point. NRLGO outperforms on Corel 1000, Corel 5000, and Corel 10000 the other state-of-the-art descriptors as shown in Figure 6–8. These figures indicate that the ARP of proposed NRLGO is high, so precision-recall curve lies away from the point of origin. Proposed method NRLGO has a curve with few turning points and outperforms other descriptors for image categories of each dataset. MTH, SED, and MTH are textural descriptors used for efficient image retrieval including the correlation between texture and color, so local contrast information is missing. As correlation achieved by using intensity feature, the poor performance of MTH, SED, and STH is achieved. GLCM, EHD, and Gabor features are textural descriptors; they gave poor performance on natural images because using texture descriptor demonstrates limited texture classification of an image. Gabor filter is also used in textural classification due to the high relation between its cooccurrence and texture. EOAC does not consider textural characteristics of an image. MSD established correlation between color and orientation only and does not

Table 11: Performance of NRLGO is compared with uniform pattern-based state-of-the-art methods on Corel dataset with 5000 images and Corel dataset with 10000 images.

Dataset	Performance	CPV-THF [42]	GLCM [50]<	SQs [49]	GMMs [49]	SED [46]	CMSD [12]	Proposed
Corel 5k	Precision (%)	63.90	38.0	41.67	51.80	56.3	63.14	65.8
	Recall (%)	7.66	1.08	5.13	6.22	21.3	7.57	7.89
Corel 10k	Precision (%)	52.28	31.9	37.64	47.25	46.4	50.25	53.07
	Recall (%)	6.27	.63	4.51	5.67	18.3	6.03	6.36

Table 12: Performance comparison of NRLGO with color, texture, and shape based state-of-the-art techniques on Corel 5k and Corel 10k.

Database	Parameters	Gabor [51]	EHD [8]	CM [18]	LBP [52]	STH [53]	MTSD [54]	Proposed
Corel 5k	Precision (%)	36.2	39.4	35.82	48.44	60.28	62.98	65.8
	Recall (%)	4.35	4.74	4.29	28.87	7.23	9.45	7.89
Corel 10k	Precision (%)	29.1	32.3	26.38	35.01	48.03	51.95	53.07
	Recall (%)	3.50	3.88	3.16	18.59	5.76	7.79	6.36

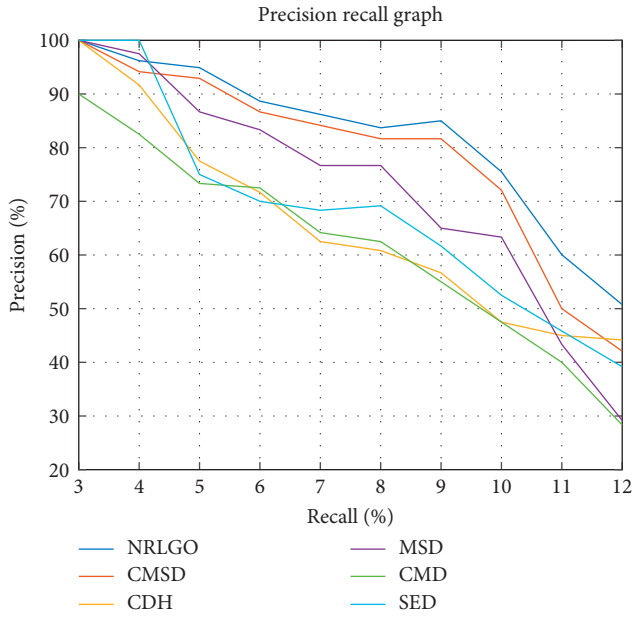


Figure 6: Performance of NRLGO is compared with state-of-the-art descriptors on Corel 1k using precision and recall parameter.

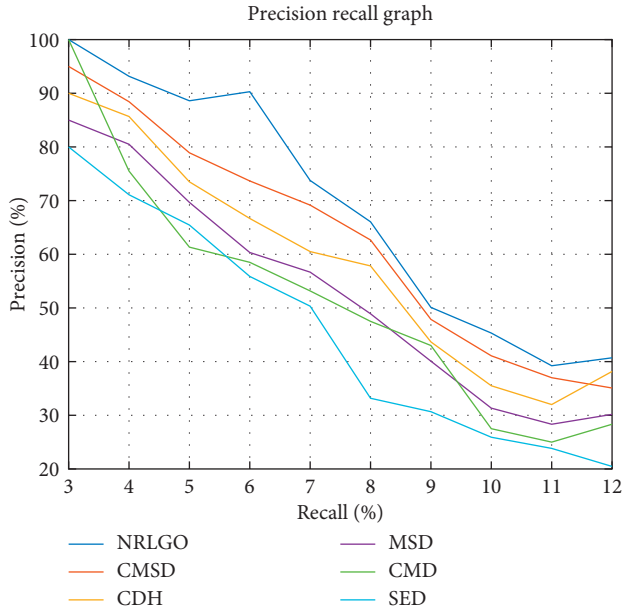


Figure 7: Performance of NRLGO is compared with state-of-the-art descriptors on Corel 5k using precision and recall parameters.

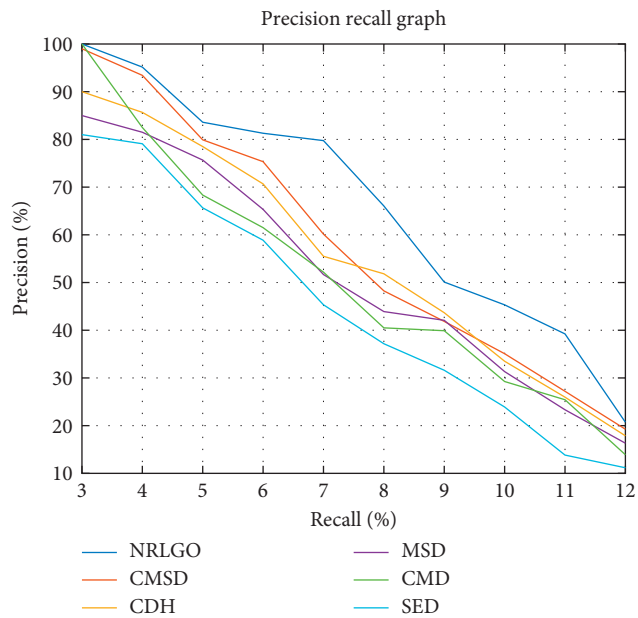


Figure 8: Performance of NRLGO is compared with state-of-the-art descriptors on Corel 10k using precision and recall parameters.

Table 13: F-score is compared with state-of-the-art descriptors at Corel 1k.

Category	GA [55]	ODBTC [56]	GMM + spatiogram [49]	Tetrolet transform [57]	BiCBIR [58]	Proposed
African	0.187	0.282	0.242	0.317	0.317	0.321
Beach	0.179	0.155	0.217	0.200	0.250	0.265
Building	0.203	0.227	0.235	0.183	0.283	0.311
Bus	0.298	0.295	0.297	0.333	0.333	0.352
Dinosaur	0.328	0.331	0.333	0.333	0.333	0.30
Elephant	0.193	0.244	0.235	0.300	0.300	0.342
Flower	0.300	0.321	0.316	0.333	0.333	0.321
Horse	0.260	0.313	0.306	0.333	0.333	0.342
Mountain	0.171	0.158	0.241	0.250	0.267	0.286
Food	0.231	0.269	0.263	0.333	0.317	0.380
Average	0.235	0.260	0.269	0.292	0.307	0.322

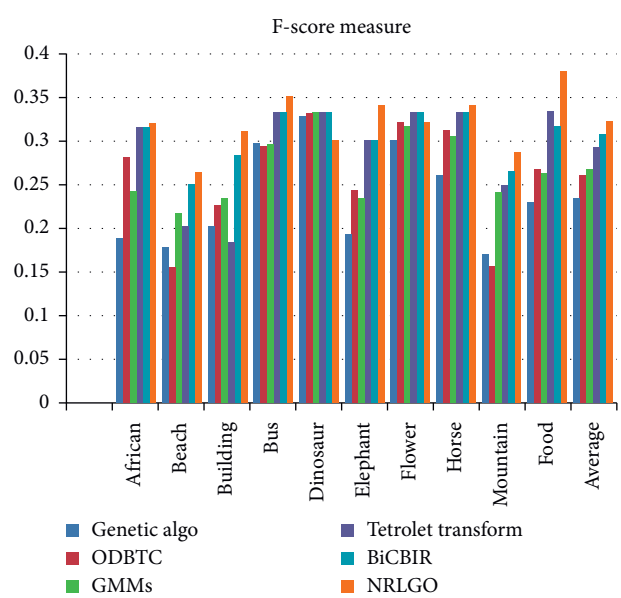


Figure 9: F-score is calculated for each category in Corel 1k and compared with the state-of-the-art techniques.

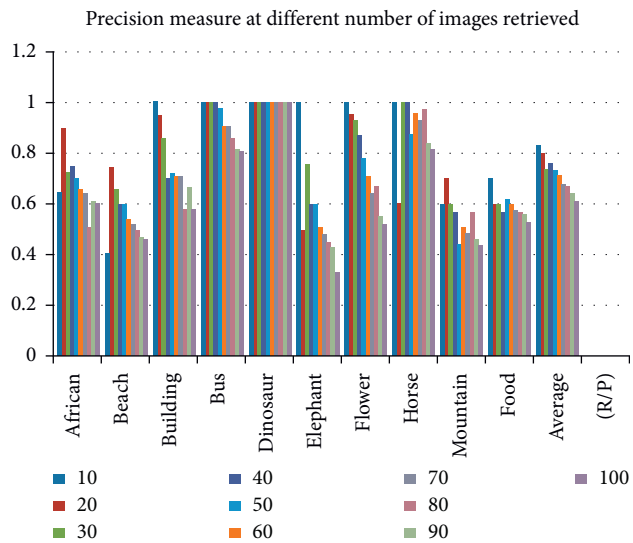


Figure 10: Precision is calculated for the number of images retrieved at Corel 1k.

Figure 11: Retrieval of top 12 images as a result of search image selected from African category.

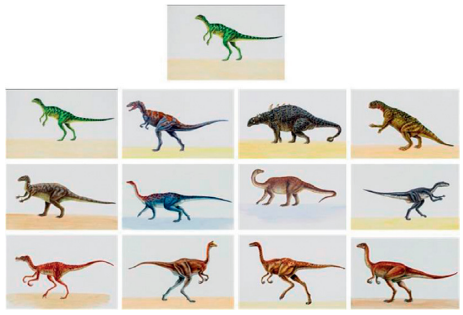


Figure 12: Retrieval of top 12 images as a result of search image selected from dinosaur category.

Figure 13: Retrieval of top 12 images as a result of search image selected from bus category.

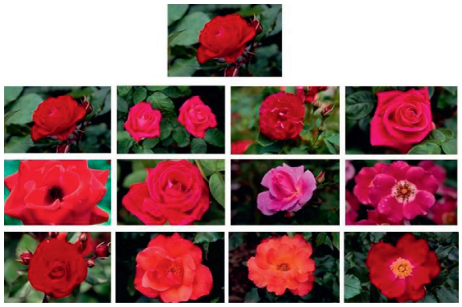


Figure 14: Retrieval of top 12 images as a result of search image selected from flower category.

take local contrast information into account. The color moment (CM) approach considers only the spatial data of pixels that are near boundary of an image. LBP takes only textural characteristics of an image. A GUI application is established; it takes query image from African, dinosaur, bus, and flower category by the user and then retrieves 12 images from each category. Performance of NRLGO is observed on the basis of image's features like color, texture, and shape (uniform-nonuniform).

In Figure 9, F-score at Corel 1k is calculated and then its average is obtained. It shows that highest F-score of 0.322 is achieved by using NRLGO method in comparison with state-of-the-art descriptors. Figure 10 shows that highest precision of 0.835 is achieved at retrieval of top 10 images and lowest precision of 0.60 is achieved at retrieval of 100 images. It shows that retrieval performance decreased when the retrieval number of images increased from 10 to 100.

Retrieval of top 12 images done against the search images selected from the African, dinosaur, bus, and flower category is shown in Figures 11–14.

In the proposed NRLGO method, NRLBP solves the issue of small pixel difference. It is used to save the local structures from noise by finding an uncertain state. It finds the value of uncertain state on the basis of corrected bits of LBP code. Uniform codes describe the image local structure and nonuniform codes describe the noise patterns. Thus, for finding uncertain state, error correction method that recovers the nonuniform patterns is used. Color feature is obtained in HSV color space by quantizing input image into 72 quantization levels. Texture classification is done by using local gradient orientation. Multiscale gradient orientation is used to extract multiple granularity features. It will reduce noise and illumination variations. V component of HSV is used to extract the intensity attribute. Similarity measure and efficient indexing are done by using Manhattan distance. Color, texture, and local contrast information micro-maps are used to demonstrate extracted features. They describe the important content that is within the uniform region and hide the irrelevant content.

7. Conclusion

A feature descriptor named Noise Resilient Local Gradient Orientation is proposed in this paper for improving the retrieval performance. NRLGO relies on noise removal, color, texture classifications, and local contrast information. Small pixel difference causes noise in color image which leads to changing code abruptly resulting in poor feature extraction. NRLBP protects the local structure from noise by finding the uncertain state first. HSV color space is quantized into 72 levels to obtain the color attribute. Texture classification is done by using LGO approach at 6 quantization levels, which quantize orientation further into supreme orientation. V component of HSV is quantized into 10 levels to obtain the local contrast information. Incorporation of multiresolution gradient orientation gave better texture detection, which increases the relationship between color, texture, and local contrast information. To achieve the resemblance between search image and image in the database, Manhattan distance is used. Correlation is established between color, texture, and local contrast information. For image characteristic representation, correlation is developed between the color, orientation, and local contrast data. Microstructures are developed on the basis of correlation information to develop more fine details in subject field. Multiresolution orientation increased the retrieval performance of proposed NRLGO. NRLGO extracts noise-robust features that remain invariant to variation in scale, illumination, and orientation. Experimental results show that NRLGO has improved retrieval performance of texture, local contrast information, and color features in comparison with state-of-the-art descriptors on Corel with 1000 images, 5000 images, and 10000 images.

Data Availability

The dataset Corel 5k is used during this study which is publicly available at corel5k.20091111.tar.bz2 and https://sites.google.com/site/dctresearch/Home/content-based-image-retrieval.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding this work.

Acknowledgments

This work was supported by Education and Research Promotion Program of KOREATECH (2021).

References

[1] R. Ashraf, "Content based image retrieval by using color descriptor and discrete wavelet transform," 2018.

[2] A. Shinde, A. Rahulkar, and C. Patil, "Content based medical image retrieval based on new efficient local neighborhood wavelet feature descriptor," *Biomedical Engineering Letters*, vol. 9, no. 3, pp. 387–394, 2019.

[3] H. Road, "E-commerce," 2016.

[4] G. Wan and Z. Liu, "Content-based information retrieval and digital libraries," *Information Technology and Libraries*, vol. 27, no. 1, pp. 41–47, 2008.

[5] J. Ren and Y. Shen, "A novel image retrieval based on representative colors," *Image Processing and Computer Vision*, vol. 16, pp. 102–107, 2003.

[6] M. Indu and K. V. Kavitha, "Survey on sketch based image retrieval methods," in *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, pp. 1–4, Nagercoil, India, March 2016.

[7] S. A. Orand, "Sketch based image retrieval using a soft computation of the histogram of edge local orientations (s-helo)," *Computer Vision Research Group*, vol. 32, pp. 2998–3002, 2014.

[8] S. Wang, J. Zhang, T. X. Han, Z. Miao, and Z. Miao, "Sketch-based image retrieval through hypothesis-driven object boundary selection with HLR descriptor," *IEEE Transactions on Multimedia*, vol. 17, no. 7, pp. 1045–1057, 2015.

[9] H. Xie, Y. Ji, and Y. Lu, "An analogy-relevance feedback cbir method using multiple features," 2016.

[10] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "Long term learning in image retrieval systems using case based reasoning," *Engineering Applications of Artificial Intelligence*, vol. 35, pp. 26–37, 2014.

[11] R. Mandal, P. P. Roy, U. Pal, and M. Blumenstein, "Bag-of-visual-words for signature-based multi-script document retrieval," *Neural Computing and Applications*, vol. 31, no. 10, pp. 6223–6247, 2019.

[12] H. Dawood, M. H. Alkinani, A. Raza, H. Dawood, R. Mehboob, and S. Shabbir, "Correlated microstructure descriptor for image retrieval," *IEEE Access*, vol. 7, no. c, pp. 55206–55228, 2019.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 1–28, 2004.

[14] Y. Ke and R. Sukthankar, "PCA-SIFT : a more distinctive representation for local image descriptors," 2004.

[15] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[16] O. A. Vatamanu, M. Frandeş, D. Lungeanu, and G. I. Mihalaş, "Content based image retrieval using local binary pattern operator and data mining techniques," *Studies in Health Technology and Informatics*, vol. 210, pp. 75–79, 2015.

[17] S. R. Dubey, S. K. Singh, R. K. Singh, and S. Member, "Multichannel decoded local binary patterns for content-based image retrieval," *IEEE Transactions on Image Processing*, vol. 25, no. 9, pp. 4018–4032, 2016.

[18] S. R. Singh and S. Kohli, "Enhanced CBIR using color moments, HSV histogram, color auto correlogram, and gabor texture," *International Journal of Computer Systems*, vol. 2, no. 5, pp. 161–165, 2015.

[19] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," 2007.

[20] W.-H. Liao, "Region description using extended local ternary patterns," in *Proceedings of the2010 20th International Conference on Pattern Recognition*, pp. 1007–1010, Istanbul, Turkey, August 2010.

[21] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikäinen, and S. Z. Li, "Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1301–1306, San Francisco, CA, USA, June 2010.

[22] S. D. Thepade and Y. D. Shinde, "Improvisation of content based image retrieval using color edge detection with various gradient filters and slope magnitude method," in *Proceedings of the 2015 International Conference on Computing Communication Control and Automation*, pp. 625–628, Mumbai, India, February 2015.

[23] P. Bao, L. Lei Zhang, and X. Xiaolin Wu, "Canny edge detection enhancement by scale multiplication," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 9, pp. 1485–1490, 2005.

[24] H. Qazanfari, H. Hassanpour, and K. Qazanfari, "Content-based image retrieval using HSV color space features," *International Journal of Computer and Information Technology*, vol. 13, no. 10, pp. 537–545, 2019.

[25] A. Al-mohamade and O. Bchir, "Multiple query content-based image retrieval using relevance feature weight learning," 2020.

[26] B. Zafar, R. Ashraf, N. Ali et al., "A novel discriminating and relative global spatial image representation with applications in CBIR," *Applied Sciences*, vol. 8, no. 11, pp. 2242–2323, 2018.

[27] R. Ashraf, M. Ahmed, U. Ahmad et al., "MDCBIR-MF: multimedia data for content-based image retrieval by using multiple features," *Multimedia Tools and Applications*, vol. 79, no. 13-14, pp. 8553–8579, 2020.

[28] N. Hor and S. Fekri-Ershad, "Image retrieval approach based on local texture information derived from predefined patterns and spatial domain information," 2019, http://arxiv.org/abs/1912.12978.

[29] F. Tajeripour, M. Saberi, and S. Fekri-Ershad, "Developing a novel approach for content based image retrieval using modified local binary patterns and morphological transform," *The International Arab Journal of Information Technology*, vol. 12, no. 6, pp. 574–581, 2015.

[30] N. Tadi Bani and S. Fekri-Ershad, "Content-based image retrieval based on combination of texture and colour information extracted in spatial and frequency domains," *The Electronic Library*, vol. 37, no. 4, pp. 650–666, 2019.

[31] M. Singha, "Content based image retrieval using color and texture," *Signal & Image Processing: An International Journal*, vol. 3, no. 1, pp. 39–57, 2012.

[32] N. Ahmed, H. M. S. Asif, and G. Saleem, "Leaf image-based plant disease identification using color and texture features," 2021, http://arxiv.org/abs/2102.04515.

[33] A. Alzu'bi, A. Amira, and N. Ramzan, "Content-based image retrieval with compact deep convolutional features," *Neurocomputing*, vol. 249, pp. 95–105, 2017.

[34] A. B. Yandex and V. Lempitsky, "Aggregating local deep features for image retrieval," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1269–1277, Santiago, CH, USA, August 2015.

[35] Y. Kalantidis, C. Mellina, and S. Osindero, "Cross-dimensional weighting for aggregated deep convolutional features," *Lecture Notes in Computer Science*, vol. 9913, pp. 685–701, 2016.

[36] B. Leibe, J. Matas, N. Sebe, and M. Welling, "Preface," *Lecture Notes in Computer Science(Including Subser. Notes Bioinformatics)*, vol. 9906, 2016.

[37] M. Tzelepi and A. Tefas, "Deep convolutional learning for content based image retrieval," *Neurocomputing*, vol. 275, pp. 2467–2478, 2018.

[38] J. Y.-H. Ng, F. Yang, and L. S. Davis, "Exploiting local features from deep networks for image retrieval," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 53–61, Boston, MA, USA, June 2015.

[39] J. Ren, X. Jiang, J. Yuan, S. Member, and J. Yuan, "Noise-resistant local binary pattern with an embedded error-correction mechanism," *IEEE Transactions on Image Processing*, vol. 22, no. 10, pp. 4049–4060, 2013.

[40] J. Shiguang Shan, W. L. D. Chu He, Z. Guoying, M. Pietikäinen, C. Xilin, and G. Wen, "WLD: a robust local image descriptor," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1705–1720, 2010.

[41] J. Qiu, H. Xu, and Z. Ye, "Color constancy by reweighting image feature maps," *IEEE Transactions on Image Processing*, vol. 29, pp. 5711–5721, 2020.

[42] A. Raza, H. Dawood, H. Dawood, S. Shabbir, R. Mehboob, and A. Banjar, "Correlated primary visual texton histogram features for content base image retrieval," *IEEE Access*, vol. 6, pp. 46595–46616, 2018.

[43] G.-H. Liu, Z.-Y. Li, L. Zhang, and Y. Xu, "Image retrieval based on micro-structure descriptor," *Pattern Recognition*, vol. 44, no. 9, pp. 2123–2133, 2011.

[44] G.-H. Liu, L. Zhang, Y.-K. Hou, Z.-Y. Li, and J.-Y. Yang, "Image retrieval based on multi-texton histogram," *Pattern Recognition*, vol. 43, no. 7, pp. 2380–2389, 2010.

[45] G.-H. Liu and J.-Y. Yang, "Content-based image retrieval using color difference histogram," *Pattern Recognition*, vol. 46, no. 1, pp. 188–198, 2013.

[46] X. Wang and Z. Wang, "A novel method for image retrieval based on structure elements' descriptor," *Journal of Visual Communication and Image Representation*, vol. 24, no. 1, pp. 63–74, 2013.

[47] R. Ashraf, K. Bashir, A. Irtaza, and M. Mahmood, "Content based image retrieval using embedded neural networks with bandletized regions," *Entropy*, vol. 17, no. 6, pp. 3552–3580, 2015.

[48] H. Liu, B. Li, X. Lv, and Y. Huang, "Image retrieval using fused deep convolutional features," *Procedia Computer Science*, vol. 107, pp. 749–754, 2017.

[49] S. Zeng, R. Huang, H. Wang, and Z. Kang, "Image retrieval using spatiograms of colors quantized by Gaussian Mixture Models," *Neurocomputing*, vol. 171, pp. 673–684, 2016.

[50] E. M. Gebejes, "Master, and a samples, "texture characterization based on grey-level Co-occurrence matrix," *Informatics and Management Science*, vol. 65, pp. 375–378, 2013.

[51] Z.-C. Huang, P. P. K. Chan, W. W. Y. Ng, and D. S. Yeung, "Content-based image retrieval using color moment and Gabor texture feature," in *Proceedings of the 2010 International Conference on Machine Learning and Cybernetics*, pp. 719–724, Qingdao, China, July 2010.

[52] P. Srivastava and A. Khare, "Integration of wavelet transform, Local Binary Patterns and moments for content-based image retrieval," *Journal of Visual Communication and Image Representation*, vol. 42, pp. 78–103, 2017.

[53] A. Raza, T. Nawaz, H. Dawood, and H. Dawood, "Square texton histogram features for image retrieval," *Multimedia Tools and Applications*, vol. 78, no. 3, pp. 2719–2746, 2019.

[54] M. Zhao, H. Zhang, and J. Sun, "A novel image retrieval method based on multi-trend structure descriptor," *Journal of Visual Communication and Image Representation*, vol. 38, pp. 73–81, 2016.

[55] M. E. Elalami, "A novel image retrieval model based on the most relevant features," *Knowledge-Based Systems*, vol. 24, no. 1, pp. 23–32, 2011.

[56] J. M. Jing-Ming Guo and H. Prasetyo, "Content-based image retrieval using features extracted from halftoning-based block truncation coding," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 1010–1024, 2015.

[57] J. Pradhan, S. Kumar, A. K. Pal, and H. Banka, "A hierarchical CBIR framework using adaptive tetrolet transform and novel histograms from color and shape features," *Digital Signal Processing*, vol. 82, pp. 258–281, 2018.

[58] S. Singh and S. Batra, "An efficient bi-layer content based image retrieval system," *Multimedia Tools and Applications*, vol. 79, no. 25-26, pp. 17731–17759, 2020.

Hindawi
International Journal of Optics
Volume 2021, Article ID 9946809, 16 pages
https://doi.org/10.1155/2021/9946809

Research Article

Infrared Image Deblurring Based on Generative Adversarial Networks

Yuqing Zhao, Guangyuan Fu, Hongqiao Wang, Shaolei Zhang, and Min Yue

Xi'an Research Institute of High-Tech, Shaanxi 710025, China

Correspondence should be addressed to Yuqing Zhao; zoe_rabbi@126.com

Received 7 March 2021; Accepted 26 April 2021; Published 4 May 2021

Academic Editor: Muhammad Tariq Mahmood

Blind deblurring of a single infrared image is a challenging computer vision problem. Because the blur is not only caused by the motion of different objects but also by the relative motion and jitter of cameras, there is a change of scene depth. In this work, a method based on the GAN and channel prior discrimination is proposed for infrared image deblurring. Different from the previous work, we combine the traditional blind deblurring method and the blind deblurring method based on the learning method, and uniform and nonuniform blurred images are considered, respectively. By training the proposed model on different datasets, it is proved that the proposed method achieves competitive performance in terms of deblurring quality (objective and subjective).

1. Introduction

The main reason of motion blur is that there is rapid relative motion between the camera and the captured object during the exposure time. The blurring of images will reduce the perceptual quality of human beings. It also has a negative impact on advanced visual tasks such as object detection and semantic understanding. Image deblurring is a common and important problem in the field of image processing and computer vision. However, due to the complexity of motion blur processing, most existing methods may not produce satisfactory results when the blur kernel is complex, and the details of the required clear image are abundant. In addition, because the infrared (IR) imaging system is more complex than the natural imaging system, the degradation degree of infrared images is relatively high, such as Gaussian blur, motion blur, and noise pollution. Therefore, infrared image deblurring plays an important role in the IR imaging system. Some researchers are dedicated to hardware-based research for infrared image deblurring. In literature [1], the fluttered shutter is used to solve the problem of infrared image deblurring. Literature [2] uses an ordinary inertial measurement unit (IMU) to estimate the trajectory of the camera movement during the exposure time. Oswald-Tranta et al. [3] used the parameterized Wiener filter method to blur the infrared images obtained from the infrared detector of the microbolometer. Oswald-Tranta also committed to obtaining accurate temperature measurements by deblurring infrared images [4]. Wang et al. [5] used the iterative Wiener filter to estimate the PSF filter of motion blur in infrared images. The deblurring method based on infrared imaging hardware equipment is more expensive. Therefore, the algorithm-based deblurring of infrared images is more widely used. Luo et al. [6] developed a new infrared blurred image restoration model based on the principle of nonuniform exposure. In order to eliminate the motion blur of the image and restore the image, Jing et al. [7] proposed an infrared target motion deblurring method based on the Haar wavelet transform. Liua et al. [8] proposed a method of using Lp-quasi-linear norm and the overlapping sparse total variation method to blur infrared images.

Inspired by the great progress of traditional blind deblurring methods and learning-based blind deblurring methods recently, we propose a method based on GAN and channel prior discrimination. Specifically, the innovation of this article is summarized as follows:

(i) A channel-based inverse prior discrimination is proposed. And this method is built into a new framework of the GAN. It improves the blind deblurring performance of infrared images.

(ii) Different blur types are caused by the motion of the camera or object. In view of this situation, two different methods were used to synthesize two kinds of blurred datasets.

(iii) In the experimental stage, we conducted extensive experiments which were carried out on two different datasets. The method proposed in this article is compared with the other four advanced methods qualitatively and quantitatively.

2. Related Work

2.1. Image Deblurring. The solutions to deblurring problems are mainly divided into two types: blind deblurring and nonblind deblurring. The early related work is mainly nonblind deblurring, that is, the ambiguity function is assumed to be known. Most of these algorithms rely on the Lucy–Richardson algorithm and Wiener or Tikhonov filter which are sensitive to noise to perform deconvolution operation and obtain IS estimation. However, in reality, ambiguity functions are often uncertain. It is unrealistic to find the ambiguity function for each pixel. Therefore, a lot of recent works are focused on blind deblurring. The first modern bold attempt was Fergus et al.'s [9] variational Bayesian method to eliminate uniform camera shake. In the past decade, many methods [10–20] have solved the blur caused by camera shake by considering the uniform blur on the image. This kind of algorithm first estimates camera motion according to the induced blur kernel and then reverses the effect by performing deconvolution operation. Unfortunately, these algorithms are usually unable to eliminate nonuniform motion blur.

In fact, due to the camera rotation, radial camera motion, depth of field change, or rapid movement of objects, images taken in the field may experience more complex heterogeneous blur. Therefore, most existing nonuniform blind deblurring methods [21–26] are based on specific motion models. For example, Gupta et al. [27] proposed to model camera motion as a motion density function. The blurring kernel of spatial variables can be derived directly from it. By specifying a prior of sparsity and compactness in density, an optimization problem is formulated, and the density function and deblurred image can be solved iteratively. A new projection motion path model is proposed in [28, 29]. Another method to eliminate spatial variation ambiguity is to estimate through block-by-block blurring kernel [30–32]. Segmented blurring estimation [24, 33] also considers the spatial variation blur caused by the object movement.

In recent years, some methods based on the convolutional neural network (CNN) have appeared [23, 34–42]. Schuler et al. [39] made the first heuristic attempt, focusing on unified blind deblurring, including modules for feature extraction, blurring kernel estimation, and clear image estimation. Sun et al. [40] used the CNN to estimate the blurring kernel. Chakrabarti [43] put forward another advanced method. This method learnt to predict the plural Fourier coefficients of the deconvolution filter of the input patches of blurred images and then used the traditional optimization strategy to estimate the global blurring kernel from the restored patches. And Gong et al. [34] used the fully convolved network movement flow to estimate. All these methods use the CNN to estimate unknown ambiguity functions. Recently, Noroozi et al. [23] and Nah et al. [44] adopted the kernel-free end-to-end method, using the multiscale CNN to directly remove images. Tao et al.'s latest work [42] expands the multiscale CNN from [37] to scale the recursive CNN to realize image deblurring, and the effect is impressive. Ramakrishnan et al. [38] used a combination of pix2pix framework [45] and densely connected convolution network [46] to perform blind kernel-free image deblurring. These methods can deal with different sources of blur. Since Ramakrishnan et al., the success of GAN in image restoration has also affected the deblurring of single image. Ramakrishnan et al. [38] firstly solved the problem of image deblurring by referring to the idea of image translation [45]. Recently, Kupyn et al. [36] introduced DeblurGAN; it is developed by Wasserstein GAN [47] with gradient penalty and perceived loss.

2.2. GAN. Generative adversarial network, commonly known as GAN, was proposed by Goodfellow [48] and inspired by the zero-sum game in game theory. This game has achieved many exciting results in image restoration [49]. After style conversion [45, 50, 51], it can even be used in other fields. The system includes a generator G and a discriminator D; they constitute a minmax game for two. The generator tries to capture the potential actual data distribution and outputs new data samples, while the discriminator tries to distinguish whether the input data come from the real data distribution. The minmax game with the value function $V(G, D)$ is represented by the following formula [1]. Both generator and discriminator can be constructed based on the CNN and trained based on the above ideas.

$$\min_G \max_D V(G, D) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_x(z)}[\log(1 - D(G(z)))], \tag{1}$$

where $p_{\text{data}}(x)$ is the real data distribution, $p_{(x)}(z)$ is the model distribution, and the input z is a sample from a simple noise distribution.

GAN is known for its ability to preserve textural details in images, create solutions that are close to the real image, and be perceptually persuasive. Literature [51] was further developed; it is based on the conditional GAN [52] and trains a cyclic consistency goal. This target generates a more realistic image in the task of image migration. Inspired by this idea, Isola [45] put forward the earliest idea of image deblurring based on the GAN. Recently, great progress has been made in the related fields of image super-resolution [53] and image restoration [54] by applying the GAN.

2.3. Dark Channel Prior Algorithm. He et al. [55] proposed a defogging algorithm (DCP) based on the dark channel prior. DCP is based on the assumption that most nonsky patches of outdoor fog-free images contain some pixels. These pixels have very low intensity in at least one color channel. For any image I, its dark channel $I_{\text{dark}}(x)$ is given by the following formula:

$$I_{\text{dark}}(x) = \min_{x\in\Omega(x)}\left(\min_{c\in(r,g,b)} I^c(x)\right), \tag{2}$$

in which $\Omega(x)$ represents a local color block centered on x and I^c is the c-th color channel of i. The optical channel proposed in a similar article [56] is based on the assumption that the most blurred image block contains some pixels with very bright intensity in at least one color channel. For any image I, its optical channel $I_{\text{bright}}(x)$ is as follows:

$$I_{\text{bright}}(x) = \max_{x\in\Omega(x)}\left(\max_{c\in(r,g,b)} I^c(x)\right). \tag{3}$$

Many methods use dark channels and bright channels to complete image defogging [55, 56], and they are also used to estimate the blurring kernel in conventional blind image deblurring [15, 57]. In [15], Pan et al. proposed to use the regularization term based on L_0 additionally on the dark channel image to improve the gradient-based L_0-minimization blind deblurring method [11]. In [57], Yan et al. further combined and used L_0-based regularization in both dark and bright channel images.

3. Method

In this work, the purpose of the infrared image deblurring model is to restore a clear image when only the blurred infrared image is given. In this paper, the architecture, proposed in [51], is used to build two sets of GAN models. The generators are G_{B2S}: $I_B \longrightarrow I_S$ and G_{S2B}: $I_S \longrightarrow I_B$. G_{B2S} restores clear images from blurred images, while G_{S2B} generates blurred images from clear images. The discriminators are D_B and D_S. D_B tries to distinguish whether the input is a blurred image, while D_S tries to distinguish whether the input is sharp. The architecture of the proposed method is shown in Figure 1. The input in the method is the blurred image and clear image. The clear image is sent to the generator G_{S2B} to generate the corresponding blurred image. The generated blurred image is sent to the generator G_{B2S} to generate a deblurred image. The generated deblurred image and the real clear image are sent to the discriminator D_S together to identify true and fake. The real blurred image is input into the generator G_{B2S} to generate a deblurred image. The generated deblurred image is sent to the generator G_{S2B} to synthesize the blurred image. The synthesized blurred image and the real blurred image are sent to the discriminator D_B to determine the authenticity. Through continuous iteration, the generator can generate more realistic deblurred images. The algorithm flow is summarized as Algorithm 1.

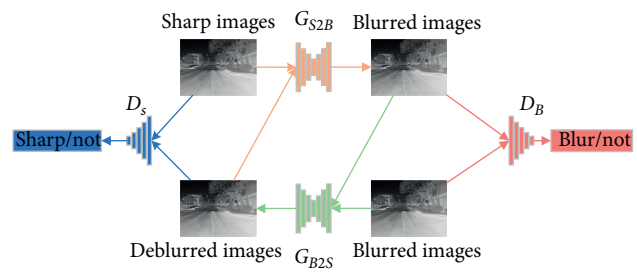


Figure 1: The overall structure of the proposed method.

3.1. Model Architecture. The method proposed by us includes two pairs of GAN. The model architecture of one pair is shown in Figure 2; it includes two deep convolutional neural network (DCNN) modules. The generator is similar to that proposed by Johnson et al. [50], including two step convolution blocks with a step size of 0.5, nine residual blocks, and two transposed convolution blocks. An instantiation standardization layer (IN) is added after the convolution layer of each convolution module except the ResBlocks. The network structure of the discriminator is the same as that of [45]. It includes five convolution modules; except the last module, each convolution layer is followed by an IN layer and a LeakyReLU layer.

As we all know, both BN and IN layers use a batch of mean and variance to normalize features during training and use the estimated mean and variance of the whole training dataset during testing. One of the potential motivations for applying BN or IN is to accelerate the training of deep neural networks (DNNs). However, recent work [58] on single-image super-resolution points out that the BN layer will bring artifacts in training and testing stages. Especially, these artifacts are more likely to occur with the deepening of the network and training under the framework of the GAN. When turned to blind deblurring, the above empirical discussion shows that the IN layer will bring similar artifacts, that is, irregular block color shift. Therefore, no IN or BN layer is introduced in the residual block, as shown in Figure 3. The network configuration of the generator and discriminator is shown in Tables 1 and 2.

3.2. Loss Function

3.2.1. Adversarial Loss. Adversarial loss includes generator adversarial loss and discriminator adversarial loss, where generator adversarial loss is defined as follows:

$$\mathscr{L}_{G_{\text{adv}}} = \sum_{n=1}^{N} -\log\left[D_B\left(\hat{I}_B\right)\right] + \sum_{n=1}^{N} -\log\left[D_S\left(\hat{I}_S\right)\right]. \tag{4}$$

Among them, the first item is the adversarial loss between the reconstructed blurred image $\hat{I}_B$ and the discriminator D_B. The second term is the adversarial loss between the reconstructed sharp image $\hat{I}_S$ and the discriminator D_S. The least square loss is better than the mean square loss in the image style conversion task. Therefore, the discriminator uses the least square loss as adversarial loss:

Input: clear image I_S; blurred image I_B
Output: deblurred image $\widehat{I}_S$ and synthesized blurred image $\widehat{I}_B$ discriminator judgment result;
(1) **for** epoch = 1, . . ., 200 **do**
(2) Sample real clear image I_S and real blurred image I_B from the training dataset
(3) I_S is sent to the generator G_{S2B} to generate a blurred image $\widetilde{I}_B$
(4) I_B is sent to the generator G_{B2S} to generate a sharp image $\widetilde{I}_S$
(5) $\widetilde{I}_B$ is sent to the generator G_{B2S} to reconstruct the sharp image $\widehat{I}_S$
(6) $\widetilde{I}_S$ is sent to the generator G_{S2B} to reconstruct the blurred image $\widehat{I}_B$
(7) Update the discriminator D_S and D_B
(8) Update the generator G_{S2B} and G_{B2S}
(9) **end for**

Algorithm 1: Deblurring method proposed in this work.

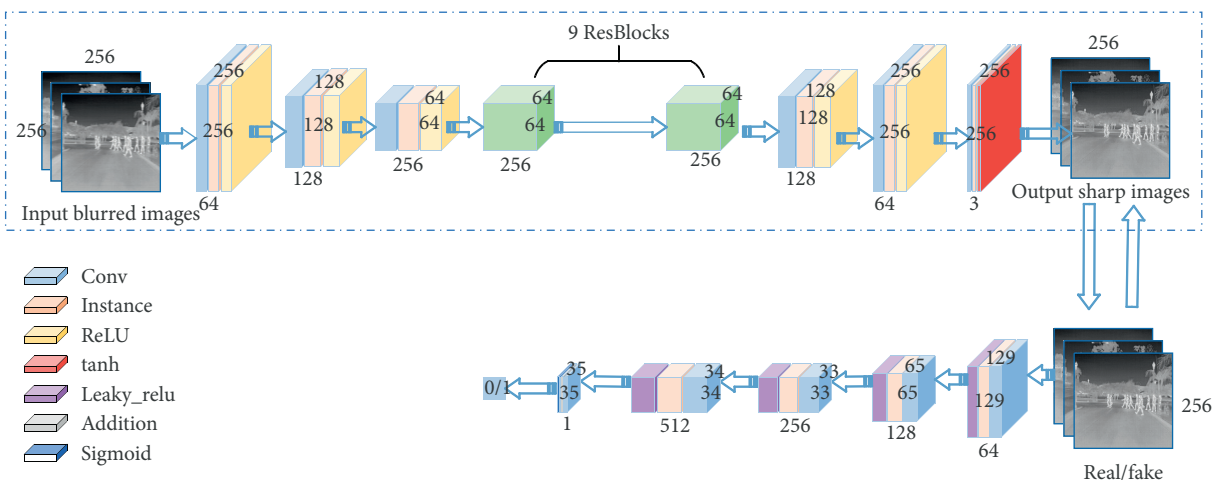


Figure 2: The model architecture of the method is proposed. Top: generator model. Bottom: discriminator model.

$$\mathscr{L}_{D_{\text{adv}}} = \frac{1}{2}\left\{\mathbb{E}_{I_B \sim p_{\text{data}}(I_B)}\left[D_B(I_B) - 1\right]^2 + \mathbb{E}_{\widetilde{I}_S \sim p_z(\widetilde{I}_S)}\left[D_B\left(G_{S2B}(\widetilde{I}_S)\right)\right]^2\right\} + \frac{1}{2}\left\{\mathbb{E}_{I_S \sim p_{\text{data}}(I_S)}\left[D_S(I_S) - 1\right]^2 + \mathbb{E}_{\widetilde{I}_B \sim p_z(\widetilde{I}_B)}\left[D_S\left(G_{B2S}(\widetilde{I}_B)\right)\right]^2\right\}. \tag{5}$$

Among them, the first term is the loss function of the discriminator D_B error identification, and the second term is the loss function of the discriminator D_S error identification.

3.2.2. Loss of Circular Perception Consistency. For the general GAN, it is necessary to compare the reconstructed image and the original image in the training stage with a certain metric as content loss. The common choice of content loss is pixel-space loss, and the simplest is L1 or L2 loss. Because this kind of loss often produces excessively smooth pixel-space output, this leads to blurring artifacts on the generated image. This brings negative factors to the deblurring task, so the circular perception consistency loss suggested in [58] is adopted. The purpose of circular perception consistency loss is to preserve the original image structure by looking at the combination of high-level and low-level features extracted from the second and fifth pooling layers of the VGG-16 system [59]. Under the constraints of generator G_{B2S}: $I_B \longrightarrow I_S$ and generator G_{S2B}: $I_S \longrightarrow I_B$, the following formula of circular perception consistency loss is given:

$$\mathscr{L}_{\text{cycle\_perceptual}} = \mathscr{L}_{\text{cycle\_perceptual1}} + \mathscr{L}_{\text{cycle\_perceptual2}}, \tag{6}$$

$$\mathscr{L}_{\text{cycle\_perceptual1}} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} \left(\phi_{i,j}(I_s)_{x,y} - \phi_{i,j}\left(G_{B2S}(I_B)\right)_{x,y}\right)^2, \tag{7}$$

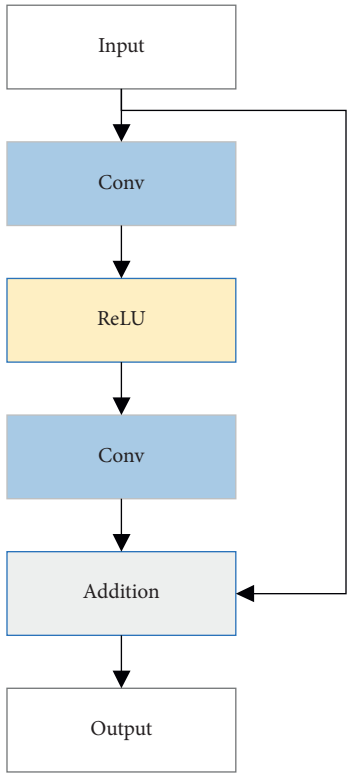


Figure 3: The ResBlocks with no IN and BN layers are not introduced.

Table 1: The layer structure and parameters of the generator.

Layer (type)	Output shape	Parameters
ReflectionPad2d-1	[−1, 3, 262, 262]	0
Conv2d-2	[−1, 64, 256, 256]	1,792
InstanceNorm2d-3	[−1, 64, 256, 256]	0
ReLU-4	[−1, 64, 256, 256]	0
Conv2d-5	[−1, 128, 128, 128]	73,856
InstanceNorm2d-6	[−1, 128, 128, 128]	0
ReLU-7	[−1, 128, 128, 128]	0
Conv2d-8	[−1, 256, 64, 64]	295,168
InstanceNorm2d-9	[−1, 256, 64, 64]	0
ReLU-10	[−1, 256, 64, 64]	0
ReflectionPad2d-11	[−1, 256, 66, 66]	0
Conv2d-12	[−1, 256, 64, 64]	590,080
ReLU-13	[−1, 256, 64, 64]	0
ReflectionPad2d-14	[−1, 256, 66, 66]	0
Conv2d-15	[−1, 256, 64, 64]	590,080
ResidualBlock-16	[−1, 256, 64, 64]	0
ConvTranspose2d-65	[−1, 128, 128, 128]	295,040
InstanceNorm2d-66	[−1, 128, 128, 128]	0
ReLU-67	[−1, 128, 128, 128]	0
ConvTranspose2d-68	[−1, 64, 256, 256]	73,792
InstanceNorm2d-69	[−1, 64, 256, 256]	0
ReLU-70	[−1, 64, 256, 256]	0
ReflectionPad2d-71	[−1, 64, 262, 262]	0
Conv2d-72	[−1, 3, 256, 256]	1,731
Tanh-73	[−1, 3, 256, 256]	0

Table 2: The layer structure and parameters of the discriminator.

Layer (type)	Output shape	Parameters
Conv2d-1	[−1, 64, 128, 128]	1,792
LeakyReLU-2	[−1, 64, 128, 128]	0
Conv2d-3	[−1, 128, 64, 64]	73,856
InstanceNorm2d-4	[−1, 128, 64, 64]	0
LeakyReLU-5	[−1, 128, 64, 64]	0
Conv2d-6	[−1, 256, 32, 32]	295,168
InstanceNorm2d-7	[−1, 256, 32, 32]	0
LeakyReLU-8	[−1, 256, 32, 32]	0
Conv2d-9	[−1, 512, 31, 31]	1,180,160
InstanceNorm2d-10	[−1, 512, 31, 31]	0
LeakyReLU-11	[−1, 512, 31, 31]	0
Conv2d-12	[−1, 1, 30, 30]	4,609

$$\mathscr{L}_{\text{cycle\_perceptual2}} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} \left(\phi_{i,j}(I_B)_{x,y} - \phi_{i,j}(G_{S2B}(I_S))_{x,y}\right)^2. \tag{8}$$

Among them, $\mathscr{L}_{\text{cycle\_perceptual1}}$ is the cycle perception consistency loss of the generator G_{B2S}; $\mathscr{L}_{\text{cycle\_perceptual2}}$ is the cycle perception consistency loss of the generator G_{S2B}. The goal is to make the reconstructed image and the input image as close as possible. $\phi_{i,j}$ is the feature map obtained by the VGG-16 network from the i-th largest pooling layer after the j-th convolutional layer. $W_{i,j}$ and $H_{i,j}$ are the corresponding dimensional feature maps.

3.2.3. Prior Loss Based on the Dark Channel and Bright Channel. Using the bright channel and dark channel presented in formulas (2) and (3), the following two different energies are defined:

$$\text{Energy}_{\text{dark}}(I) = \left(\frac{\sum_x I_{\text{dark}}^2(x)}{M * N}\right)^{1/2}, \tag{9}$$

$$\text{Energy}_{\text{bright}}(I) = \left(\frac{\sum_x I_{\text{bright}}^2(x)}{M * N}\right)^{1/2}, \tag{10}$$

in which M and N are channel sizes. $I_{\text{dark}}(x)$ is defined by formula (2). $I_{\text{bright}}(x)$ is defined by formula (3). It is verified by He et al. and Xu et al. [55, 56] that clear images have lower dark energy and higher bright energy. In order to test the resolvability of (9) and (10) between the infrared clear image I_S and the corresponding blurred image I_B, the images of the FLIR_ADAS_1_3 dataset are calculated. The results of 8862 clear and blurring image pairs show that $\text{Energy}_{\text{dark}}(I_S) < \text{Energy}_{\text{dark}}(I_B)$ and $\text{Energy}_{\text{bright}}(I_S) > \text{Energy}_{\text{bright}}(I_B)$. In order to visualize the calculation results, 200 images were randomly selected, and the sum curves were provided, as shown in Figure 4.

Based on this conclusion, it is considered that clear images and blurred images can be distinguished by dark energy and bright energy defined in (9) and (10). In order to improve the GAN from the perspective of domain knowledge, the prior judgment of the traditional blind image deblurring method is taken as the training loss function:

$$\mathscr{L}_{DC\ P}(G_{B2S}(I_B)) = \text{Energy}_{\text{dark}}(G_{B2S}(I_B)), \tag{11}$$

$$\mathscr{L}_{BCP}(G_{S2B}(I_S)) = \text{Energy}_{\text{bright}}(G_{S2B}(I_S)). \tag{12}$$

Combining formulas (4)–(12), the final losses adopted in this article are as follows:

$$\mathscr{L}_G = \lambda_1 \mathscr{L}_{G_{\text{adv}}} + \lambda_2 \mathscr{L}_{\text{cycle\_perceptual}} + \lambda_3 (\mathscr{L}_{DC\ P} + \mathscr{L}_{BCP}), \tag{13}$$

$$\mathscr{L}_D = \mathscr{L}_{D_{\text{adv}}}. \tag{14}$$

In formula (13), λ_1, λ_2, and λ_3 are the weights of the loss function. According to the experimental results, they are $\lambda_1 = 0$, $\lambda_2 = 10$, and $\lambda_3 = 10^3$, respectively.

4. Experiment

All models are implemented by the PyTorch deep learning framework. FLIR_ADAS_1_3 dataset and LTIR dataset are used to train on a desktop with 2.20 GHz × 40 Intel Xeon (r) Silver4114 CPU, GeForce GTX 1080Ti, and 64GiB memory. In this section, the experimental results are introduced and compared with the results of mainstream methods. In addition, qualitative results are provided on real images.

4.1. Synthetic Blurring Dataset. There are two types of blurred images: the overall image is blurred due to the movement of the imaging device, and the partial image is blurred due to the movement of the imaging object. In order to verify that our deblurring method is effective for both types of blur, we simulate the two types of image blur through two different schemes.

For the overall image blur caused by the motion of the imaging device, we choose to use a linear blur kernel to create a synthetic blur image. Sun et al. [40] created a composite blurred image by convolving a clear natural image with one of 73 possible linear motion kernels. Xu et al. [60] also used the linear motion kernel to create synthetic blurred images. Chakrabarti [61] created a blurring kernel by

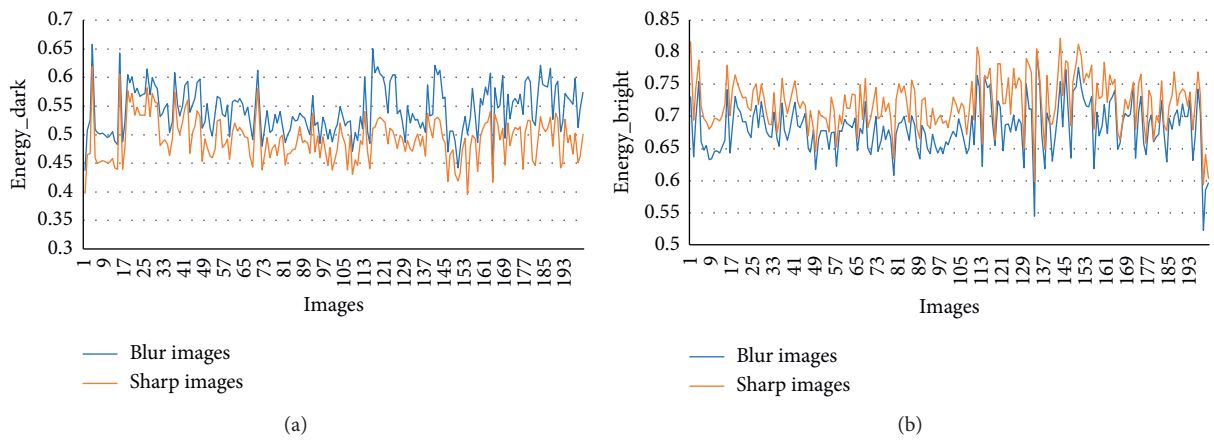


Figure 4: Graph of Energy_dark and Energy_bright on the test image randomly selected from the FLIR dataset: (a) dark energy and (b) bright energy.

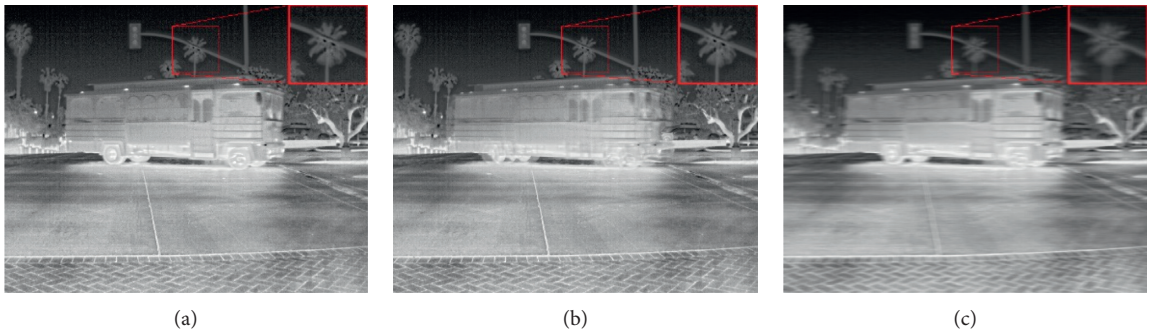
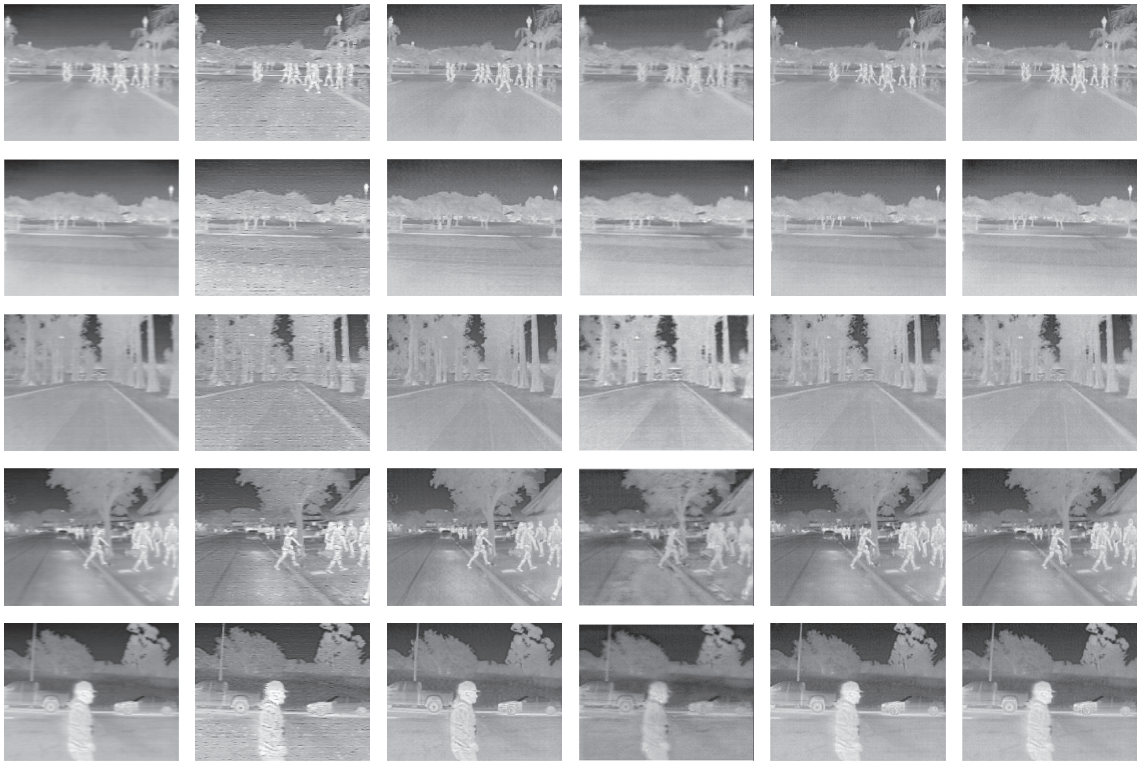


Figure 5: Comparison of synthetic blurred images by different methods. (a) Real clear image. (b) Blurred image synthesized by the average frame method. (c) Blurred image synthesized by the blur kernel method.

sampling six random points and fitting splines to them. Levin et al. provided eight blurring kernels [62] that have been used for multiple datasets. However, the maximum blurring kernel size of these eight blurring kernels is 41×41, which is relatively small in practice. Therefore, we follow the algorithm in [63] to generate four uniform blur kernels from 51×51 to 101×101 by sampling random 6D camera trajectories. Then, a convolution model with 1% Gaussian noise is used to synthesize a blurred image.

For the local image blur caused by the motion of the imaging object, we choose to use the average frame of the video sequence to simulate. This is a typical method of simulating blurred image pairs [23, 37]. This method can create realistic blurred images but only limits the image space to scenes with video sequences; this makes the dataset limited. Figure 5 shows a comparison of two different blur types. The blurred image generated by averaging frames shows the blur caused by moving objects and static background. The car in Figure 5(b) is blurred, but the surrounding trees are clear. The blur kernel method simulates the motion blur of the whole image caused by the motion of the camera. In Figure 5(c), the car and the surrounding trees are blurred. In order to verify the universality of our algorithm, we use the blur kernel to synthesize blurred images for the LTIR dataset and use two synthetic methods of average frame and the blur kernel for the FLIR dataset to simulate motion blur. The blurred dataset synthesized by the blur kernel method is used as the FLIR-A dataset; the blurred dataset synthesized by the average frame method is used as the FLIR-B dataset.

4.2. FLIR_ADAS_1_3 Dataset Results. FLIR_ADAS_1_3 datasets provide annotated thermal imaging datasets and corresponding unannotated RGB images for training and verifying neural networks. Data are acquired by using the RGB camera and thermal imaging camera installed on the vehicle. The dataset contains a total of 14,452 infrared images, of which 10,228 are from multiple short videos, and 4224 are from a video with a length of 144 s. All videos come from streets and highways. The sampling rate of most pictures is two frames per second. The frame rate of the video is 30 frames per second. When there are few targets in a few environments, the sampling rate is 1 frame per second. In the experiment, 8862 8-bit infrared images are divided into 7090 image training sets and 1772 image test sets. Figure 6



(a) (b) (c) (d) (e) (f)

Figure 6: Deblurring results of the FLIR-A dataset. The columns from left to right are the original blurred image, results of Schuler et al. [39], Kupyn et al. [36], Zhu et al. [51], Engin et al. [64], and the results of the method proposed in this article. (a) Blurred image. (b) DeepDeblur. (c) DeblurGAN. (d) CycleGAN. (e) Cycle-Dehaze. (f) Ours.

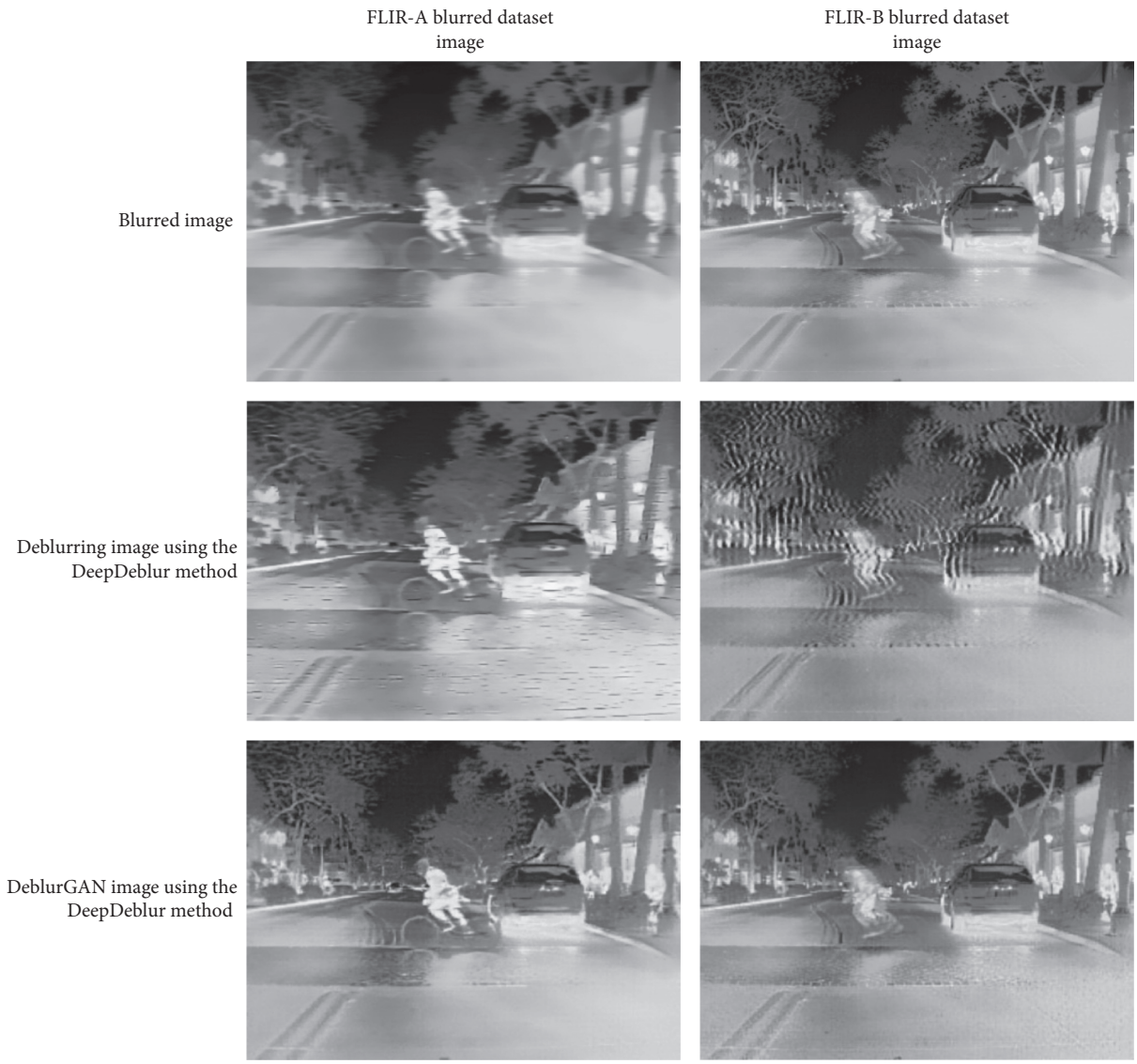
Table 3: Comparison of quantitative deblurring performance on FLIR-A datasets.

	DeepDeblur	DeblurGAN	CycleGAN	Cycle-Dehaze	Ours
SSIM	0.8916	0.9899	0.9190	0.9788	0.9985
PSNR (dB)	17.48	26.91	20.45	21.03	28.79
Time (s)	40.03	1.05	4.59	7.01	0.14

shows the test images on the FLIR-A blurred dataset, and the quantitative results are shown in Table 3.

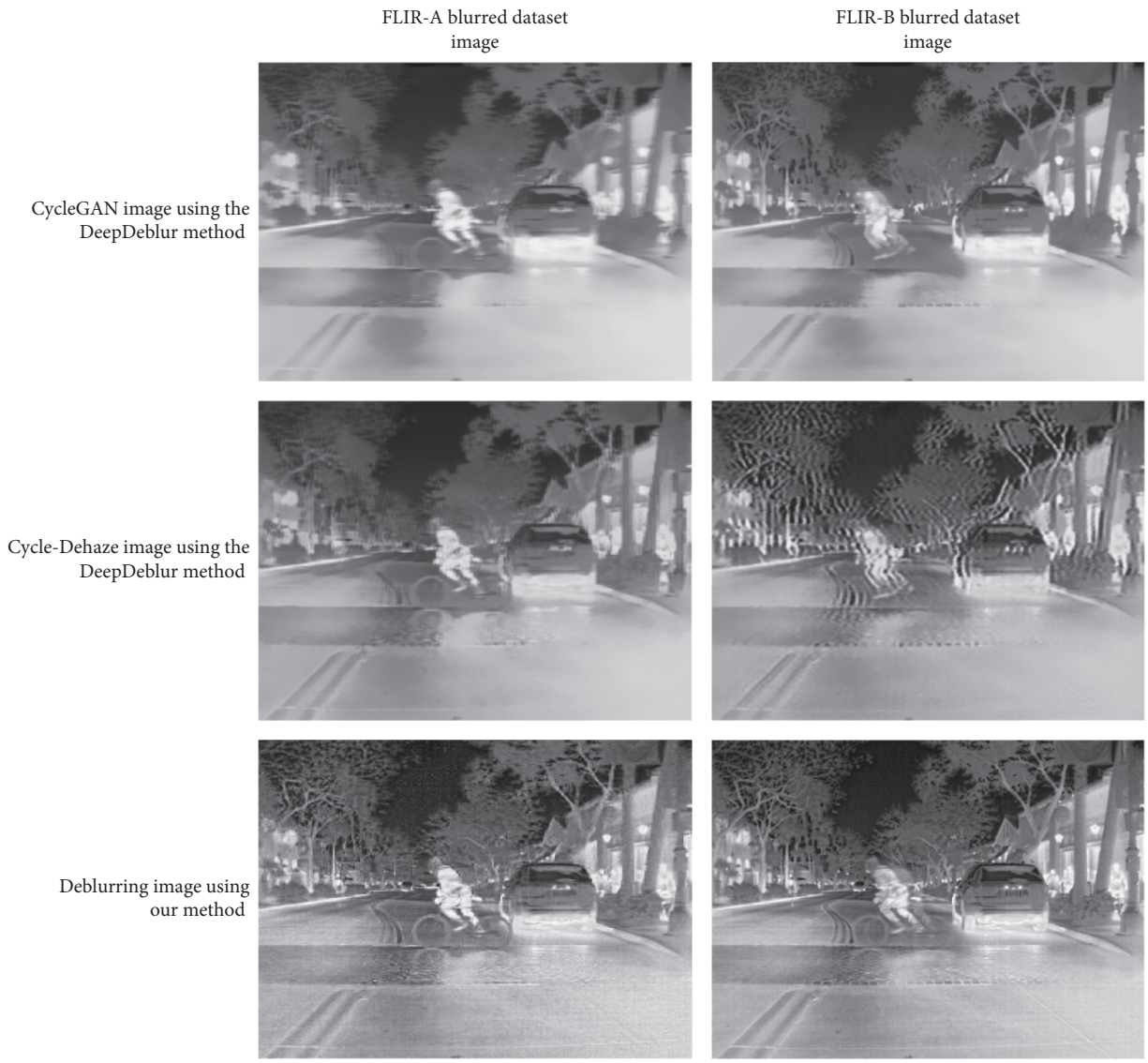
In order to further compare the deblurring effects of various methods on different types of blurred images, we compare the deblurring results of FLIR-A and FLIR-B blurred datasets. Figure 7 shows the deblurred images of different methods on the two types of blurred datasets, and the evaluation indicators are shown in Table 4. It can be seen from the subjective and objective results that our method has better deblurring performance than several other methods. This result is particularly obvious on the FLIR-B blurred dataset. For partially blurred images caused by the motion of the imaging object, the deblurring effect of other methods is significantly reduced, the original clear background becomes more blurred, and the blurred area does not achieve the ideal deblurring effect. However, our method can restore the blurred area clearly while keeping the background clear. This has a lot to do with the idea of channel prior discrimination adopted in our method. The channel prior discrimination algorithm is based on local color patches. This makes our method have better deblurring performance in the local blurred image.

4.3. LTIR_v1_0 Dataset Results. LTIR dataset is a thermal infrared dataset used to evaluate the tracking of a single object (STSO) in a short time. Currently, only one version is available. Version 1.0 consists of 20 infrared thermal sequences with an average length of 563 frames. This dataset is a subchallenge of the 2015 Visual Object Recognition (VOT) Challenge. In the experiment, 11,262 8-bit images are divided into a training set of 9010 images and a test set of 2252 images. Figure 8 shows the test image on the LTIR dataset. The quantitative results are shown in Table 5.



(a)

Figure 7: Continued.



(b)

Figure 7: Comparison of deblurring results between FLIR-A and FLIR-B blurred datasets.

Table 4: Comparison of the deblurring performance of various methods on different types of blurred images.

		DeepDeblur	DeblurGAN	CycleGAN	Cycle-Dehaze	Ours
FLIR-A	SSIM	0.8916	0.9899	0.9190	0.9788	0.9985
	PSNR	17.48	26.91	20.45	21.03	28.79
FLIR-B	SSIM	0.7458	0.8161	0.7997	0.8364	0.9589
	PSNR	16.76	18.47	17.20	19.51	21.22

4.4. Ablation Research and Analysis. We conduct ablation research on the effect of the loss function component in the deblurring method proposed in this paper. The results are summarized in Table 6. We can see that our proposed dark channel and bright channel a priori determination components are steadily improving PSNR and SSIM. In

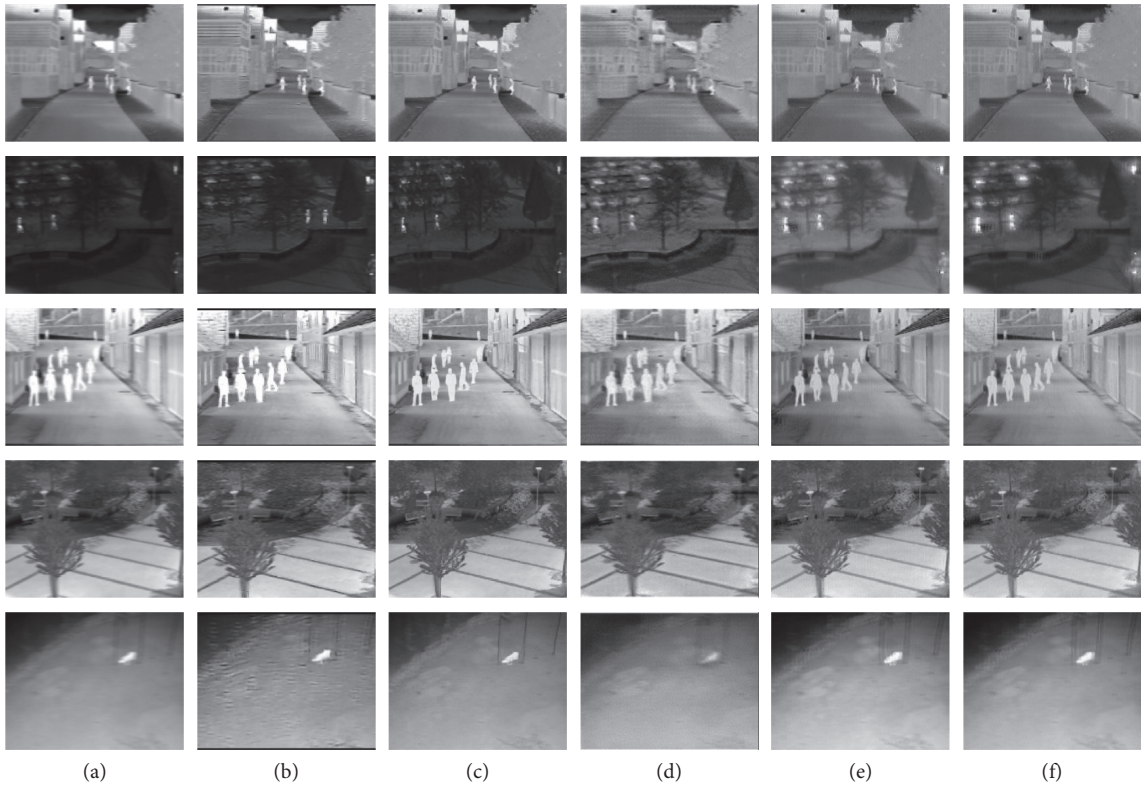


Figure 8: Deblurring results of the LTIR dataset. The columns from left to right are the original blurred image, the results of Schuler et al. [39], Kupyn et al. [36], Zhu et al. [51], Engin et al. [64], and the results of the proposed method in this article. (a) Blurred image. (b) DeepDeblur. (c) DeblurGAN. (d) CycleGAN. (e) Cycle-Dehaze. (f) Ours.

Table 5: Comparison of quantitative deblurring performance on LTIR datasets.

	DeepDeblur	DeblurGAN	CycleGAN	Cycle-Dehaze	Ours
SSIM	0.7535	0.8576	0.6977	0.7110	0.9697
PSNR (dB)	15.85	22.48	17.51	10.55	25.85
Time (s)	37.62	0.82	3.56	5.74	0.06

Table 6: Ablation study of the channel prior loss function.

	SSIM		PSNR (dB)	
	FLIR dataset	LTIR dataset	FLIR dataset	LTIR dataset
Remove the dark channel prior loss function	0.9805	0.7463	21.66	13.95
Remove the bright channel prior loss function	0.9823	0.8818	22.47	22.73
Replace perceptual loss with L1 loss	0.9344	0.9191	19.43	20.20
Replace perceptual loss with L2 loss	0.9421	0.9256	19.25	20.64
Ours	0.9985	0.9697	28.79	25.85

particular, the dark channel a priori determination module contributes the most. When we replace the perceptual loss function with L1 and L2 loss functions, the average SSIM and PSNR both decrease. It can be seen from Figure 9 that the deblurred image generated after replacing the perceptual loss function with the L1 and L2 loss function is too smooth. In summary, in the deblurring task, the perceptual loss function is more suitable than the L1 and L2 loss functions.

4.5. Use Advanced Vision Tasks to Compare Deblurring Results. Basic vision tasks, including image deblurring, serve for advanced vision tasks. In order to further verify the effectiveness of our method, we match the deblurred images generated by several methods with real clear images. Scale-Invariant Feature Transformation (SIFT) is a representation of Gaussian image gradient statistics in the field of feature points and is a commonly used image local feature

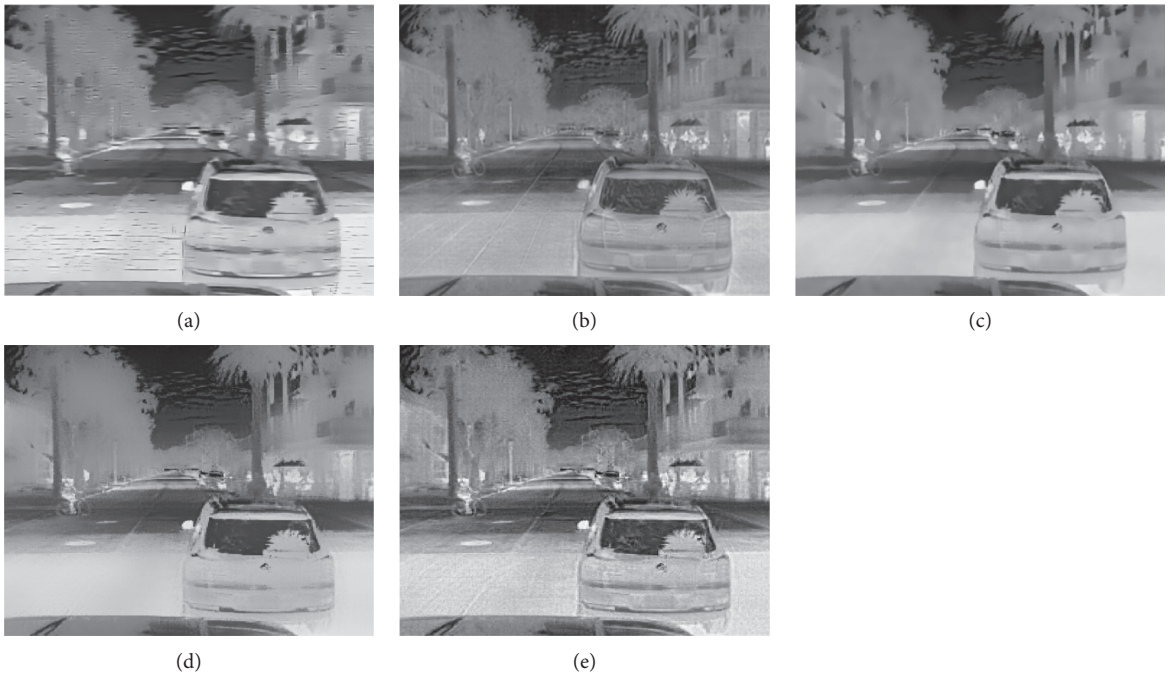


Figure 9: Effect picture of the ablation experiment. (a) Remove the dark channel prior loss function. (b) Remove the bright channel prior loss function. (c) Replace perceptual loss with L1 loss. (d) Replace perceptual loss with L2 loss. (e) Ours.

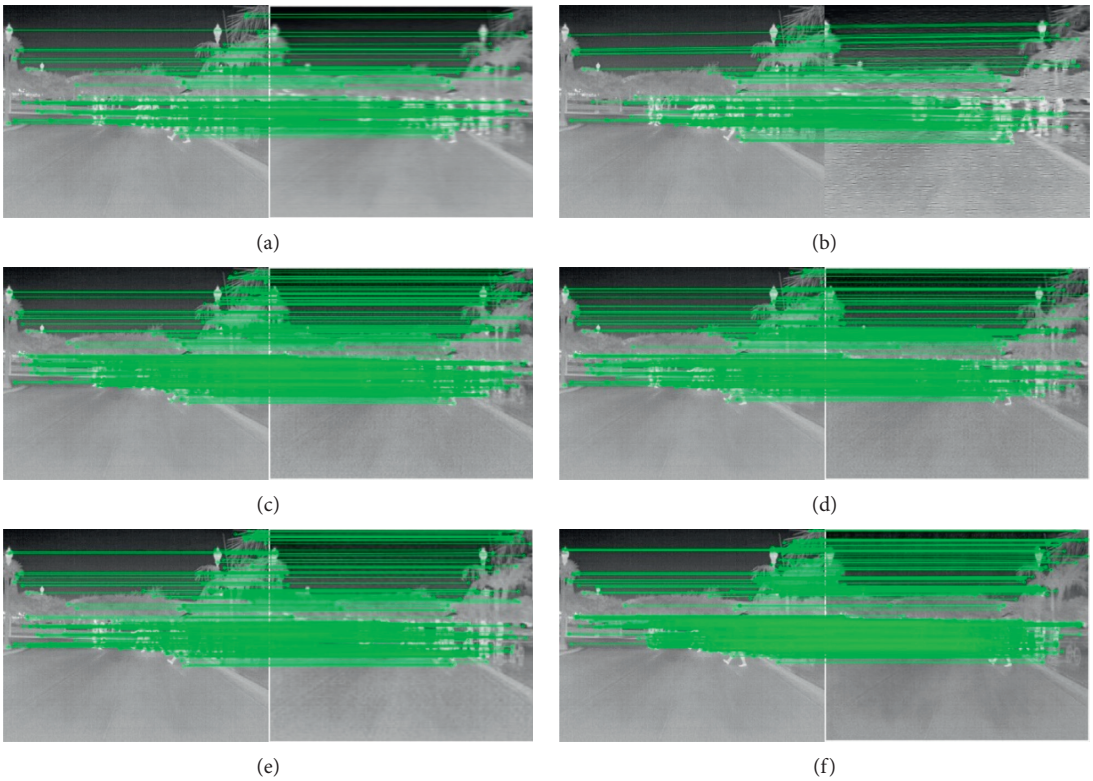
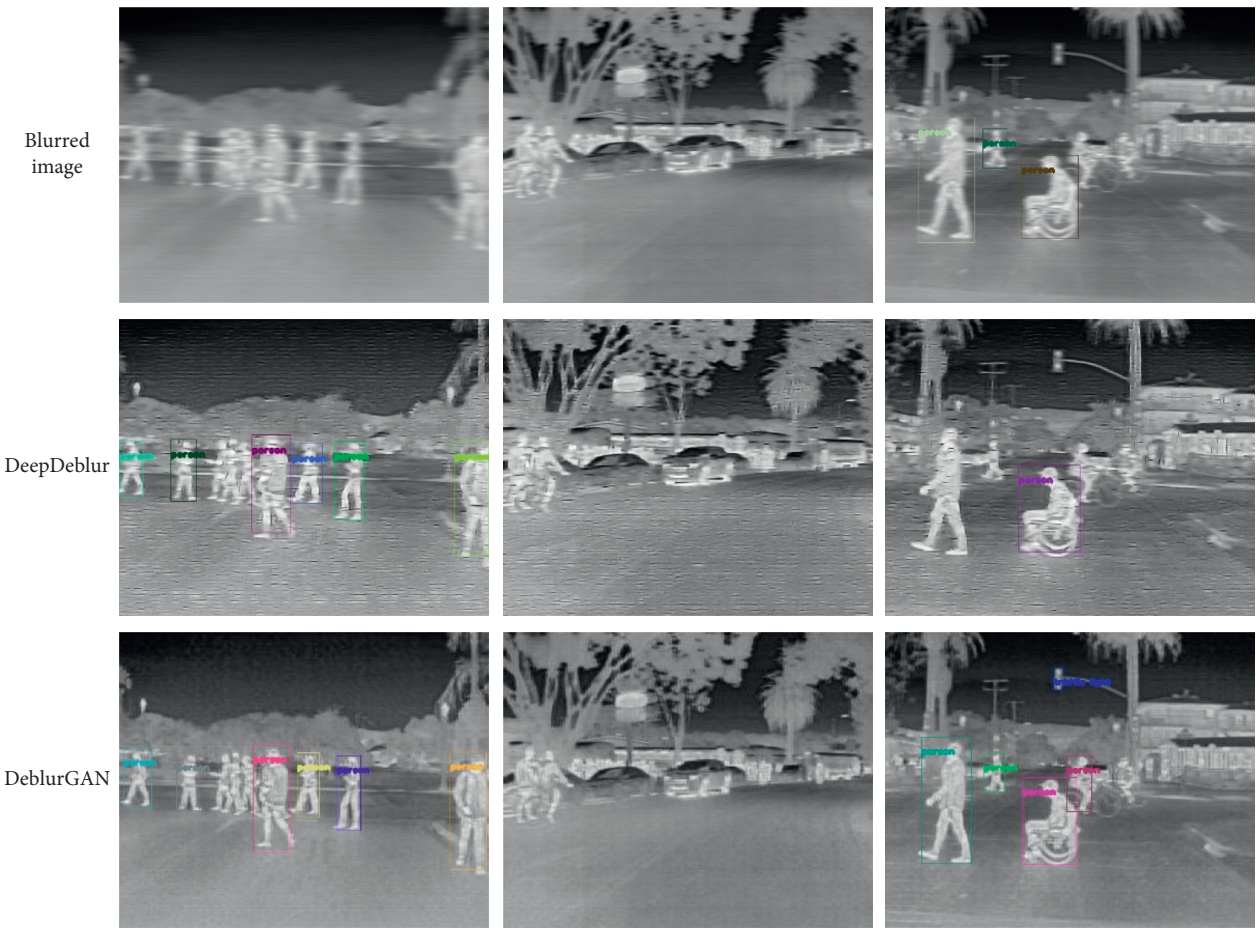
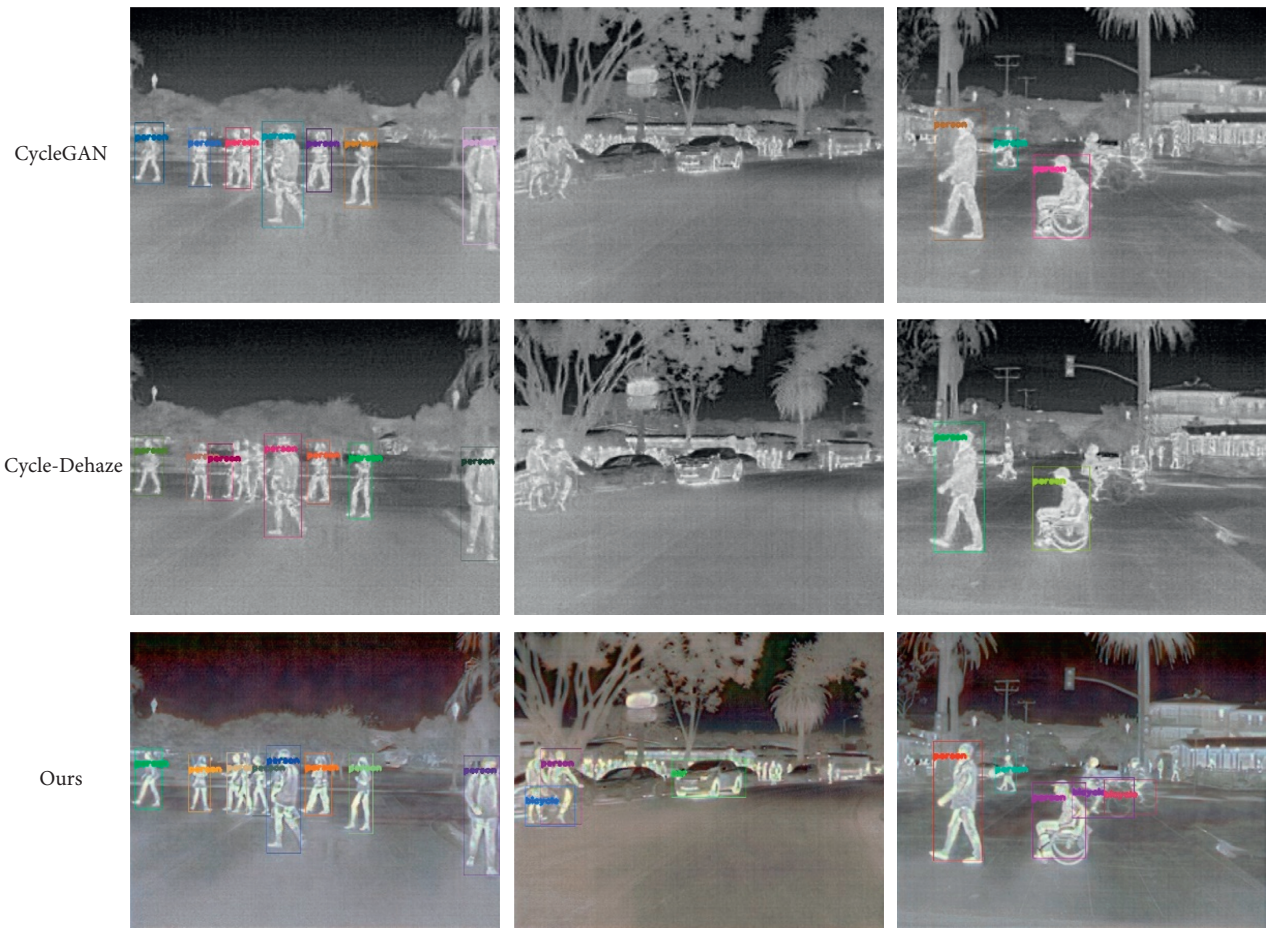


Figure 10: Deblurring image matching results (the images on the left are real and clear images). (a) Blurred image matching result. (b) DeepDeblur method deblurred image matching result. (c) DeblurGAN method deblurred image matching result. (d) CycleGAN method deblurred image matching result. (e) Cycle-Dehaze method deblurred image matching result. (f) Our method deblurred image matching result.



(a)

Figure 11: Continued.



(b)

Figure 11: Deblurring image target detection results.

extraction algorithm. In the matching result, the number of matching points can be used as a criterion for matching quality, and the corresponding matching points can also determine the similarity of the local features of the two images. Figure 10 shows the result of matching the deblurred image with the real clear image through the SIFT algorithm. It can be seen from the quantity that the deblurred image produced by our proposed method obtains more correct matching pairs than other methods.

In this experiment, we use the classic YOLO [65] method for deblurring image target detection (Figure 11). As can be seen, the proposed method to generate a blurred image has better detection result, and more targets can be detected.

5. Conclusion

Blind deblurring of a single infrared image is still a challenging computer vision problem. In this work, a method based on the GAN and channel prior discrimination is proposed for the problem of infrared image deblurring. Different from the previous deblurring work, we combine traditional blind deblurring and blind deblurring methods based on learning methods. Considering the different types of blur caused by the motion of the imaging device and the imaging object, extensive experiments were carried out on different public datasets. Experimental results show that the proposed method is more competitive than other popular image deblurring methods in terms of deblurring quality (subjective and objective) and efficiency.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. Agrawal, *Motion Deblurring: Motion Deblurring Using Fluttered Shutter*, Cambridge University Press, Cambridge, UK, 2014.

[2] N. Joshi, S. B. Kang, C. Lawrence Zitnick, and R. Szeliski, "Image deblurring using inertial measurement sensors," *ACM Transactions on Graphics*, vol. 29, no. 4CD, pp. 30–39, 2010.

[3] B. Oswald-Tranta, M. Sorger, and P. O'Leary, "Motion deblurring of infrared images from a microbolometer

camera," *Infrared Physics & Technology*, vol. 53, no. 4, pp. 274–279, 2010.

[4] B. Oswald-Tranta, "Temperature reconstruction of infrared images with motion deblurring," *Journal of Sensors and Sensor Systems*, vol. 7, no. 1, pp. 13–20, 2018.

[5] N. Wang, W. Jing, Y. Zhang, and X. Sun, "Restoration of the infrared image blurred by motion," in *Proceedings of the 2016 SPIE Society of Photo-optical Instrumentation Engineers*, Jinhua, China, October 2016.

[6] Y. Luo, T. Xu, N. Wang, and F. Liu, "Restoration of non-uniform exposure motion blurred image," in *Proceedings of the 2014 International Symposium on Optoelectronic Technology & Application*, Beijing, China, November 2014.

[7] L. I. Jing, M. Wang, J. Sha, and B. Xujmet, "Research on wavelet transform based motion deblurring method of infrared target," 2016.

[8] X. Liua, Y. Chena, Z. Penga, and J. Wu, "Total variation with overlapping group sparsity and lp quasinorm for infrared image deblurring under salt-and-pepper noise," *Journal of Electronic Imaging*, vol. 28, no. 4, Article ID 043031, 2018.

[9] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 787–794, 2006.

[10] D. Perrone and P. Favaro, "Total variation blind deconvolution: the devil is in the details," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 2014.

[11] L. Xu, S. Zheng, and J. Jia, "Unnatural L0 sparse representation for natural image deblurring," in *Proceedings of the 2013 IEEE Conference on Computer Vision & Pattern Recognition*, Portland, OR, USA, June 2013.

[12] W. S. Lai, J. J. Ding, Y. Y. Lin, and Y. Y. Chuang, "Blur kernel estimation using normalized color-line priors," in *Proceedings of the 2015 IEEE Computer Vision & Pattern Recognition*, Boston, MA, USA, June 2015.

[13] W. S. Lai, J. B. Huang, Z. Hu, N. Ahuja, and M. H. Yang, "A comparative study for single image blind deblurring," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, June 2016.

[14] T. Michaeli and M. Irani, "Blind deblurring using internal patch recurrence," in *Proceedings of the 2014 European Conference on Computer Vision*, Zurich, Switzerland, September 2014.

[15] J. Pan, D. Sun, H. Pfister, and M. H. Yang, "Blind image deblurring using dark channel prior," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2016.

[16] J. Pan, H. Zhe, Z. Su, and M. H. Yang, "Deblurring text images via L0-regularized intensity and gradient prior," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 2014.

[17] D. Perrone and P. Favaro, "A logarithmic image prior for blind deconvolution," *International Journal of Computer Vision*, vol. 117, no. 2, pp. 159–172, 2016.

[18] D. Perrone and P. Favaro, "A clearer picture of total variation blind deconvolution," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 6, pp. 1041–1055, 2015.

[19] W.-Z. Shao, H.-S. Deng, Q. Ge, H.-B. Li, and Z.-H. Wei, "Regularized motion blur-kernel estimation with adaptive sparse image prior learning," *Pattern Recognition*, vol. 51, no. C, pp. 402–424, 2016.

[20] W. Zuo, D. Ren, D. Zhang, S. Gu, and L. Zhang, "Learning iteration-wise generalized shrinkage-thresholding operators for blind deconvolution," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1751–1761, 2016.

[21] Z. Hu, L. Xu, and M. H. Yang, "Joint depth estimation and camera shake removal from single blurry image," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 2014.

[22] T. H. Kim and K. M. Lee, "Segmentation-free dynamic scene deblurring," in *Proceedings of the 2014 Computer Vision & Pattern Recognition*, Columbus, OH, USA, June 2014.

[23] M. Noroozi, P. Chandramouli, and P. Favaro, "Motion deblurring in the wild," in *Proceedings of the 2017 German Conference on Pattern Recognition*, Basel, Switzerland, September 2017.

[24] J. Pan, H. Zhe, Z. Su, H. Y. Lee, and M. H. Yang, "Soft-segmentation guided object motion deblurring," in *Proceedings of the 2016 Computer Vision & Pattern Recognition*, Las Vegas, NV, USA, June 2016.

[25] O. Whyte, "Non-uniform deblurring for shaken images: derivation of parameter update equations for blind de-blurring," 2010.

[26] S. Zheng, X. Li, and J. Jia, "Forward motion deblurring," in *Proceedings of the 2013 IEEE International Conference on Computer Vision*, Sydney, Australia, December 2013.

[27] A. Gupta, N. Joshi, C. L. Zitnick, M. F. Cohen, and B. Curless, "Single image deblurring using motion density functions," in *Proceedings of the 2010 European Conference on Computer Vision*, Heraklion, Crete, Greece, September 2010.

[28] Y. W. Tai, P. Tan, and M. S. Brown, "Richardson-lucy deblurring for scenes under a projective motion path," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1603–1618, 2011.

[29] H. Zhang, D. Wipf, and Y. Zhang, "Multi-image blind deblurring using a coupled adaptive sparse prior," in *Proceedings of the 2013 Computer Vision & Pattern Recognition*, Portland, OR, USA, June 2013.

[30] M. Hirsch, C. J. Schuler, S. Harmeling, and B. Schlkopf, "Fast removal of non-uniform camera shake," in *Proceedings of the 2011 International Conference on Computer Vision*, Barcelona, Spain, November 2011.

[31] M. Hirsch, S. Sra, B. Scho¨Lkopf, and G. Spemannstrae, "Efficient filter flow for space-variant multiframe blind deconvolution," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA, June 2010.

[32] H. Ji and K. Wang, "A two-stage approach to blind spatially-varying motion deblurring," in *Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition*, Providence, RI, USA, June 2012.

[33] A. Levin, "Blind motion deblurring using image statistics," in *Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems*, Vancouver, British Columbia, Canada, December 2006.

[34] D. Gong, J. Yang, L. Liu et al., "From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3806–3815, Honolulu, HI, USA, July 2017.

[35] M. Hradiš, "Convolutional neural networks for direct text deblurring," in *Proceedings of the 2015 British Machine Vision Conference*, Swansea, UK, September 2015.

[36] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "DeblurGAN: blind motion deblurring using conditional adversarial networks," in *Proceedings of the 2018*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, June 2018.

[37] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.

[38] S. Ramakrishnan, S. Pachori, A. Gangopadhyay, and S. Raman, "Deep generative filter for motion deblurring," in *Proceedings of the 2017 IEEE International Conference on Computer Vision Workshop (ICCVW)*, Venice, Italy, October 2017.

[39] C. J. Schuler, M. Hirsch, S. Harmeling, B. Scholkopf, and M. Intelligence, "Learning to deblur," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 7, pp. 1439–1451, 2016.

[40] J. Sun, W. Cao, Z. Xu, and J. Ponce, "Learning a convolutional neural network for non-uniform motion blur removal," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, June 2015.

[41] P. Svoboda, M. Hradis, L. Marsik, and P. Zemcik, "CNN for license plate motion deblurring," in *Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP)*, Phoenix, AZ, USA, September 2016.

[42] X. Tao, H. Gao, Y. Wang, X. Shen, J. Wang, and J. Jia, "Scale-recurrent network for deep image deblurring," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, June 2018.

[43] A. Chakrabarti, "A neural approach to blind motion deblurring," in *Proceedings of the European Conference on Computer Vision*, Amsterdam, The Netherlands, October 2016.

[44] S. Nah, T. Hyun Kim, and K. Mu Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.

[45] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.

[46] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.

[47] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," 2017, http://arxiv.org/abs/1701.07875.

[48] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," 2014, http://arxiv.org/abs/1406.2661.

[49] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.

[50] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-resolution," in *Proceedings of the 2016 European Conference on Computer Vision*, Amsterdam, The Netherlands, October 2016.

[51] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, October 2017.

[52] B. Dai, S. Fidler, R. Urtasun, and D. Lin, "Towards diverse and natural image descriptions via a conditional GAN," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, October 2017.

[53] C. Ledig, L. Theis, F. Huszar et al., "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.

[54] R. Yeh, C. Chen, T. Y. Lim, M. Hasegawa-Johnson, and M. Do, "Semantic image inpainting with perceptual and contextual losses," 2016, http://arxiv.org/abs/1607.07539.

[55] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, 2010.

[56] Y. Xu, X. Guo, H. Wang, F. Zhao, and L. Peng, "Single image haze removal using light and dark channel prior," in *Proceedings of the 2016 IEEE/CIC International Conference on Communications in China (ICCC)*, Chengdu, China, July 2016.

[57] Y. Yan, W. Ren, Y. Guo, R. Wang, and X. Cao, "Image deblurring via extreme channels prior," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.

[58] X. Wang, K. Yu, and S. Wu, "ESRGAN: enhanced super-resolution generative adversarial networks," 2018, http://arxiv.org/abs/1809.00219.

[59] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, http://arxiv.org/abs/14091556.

[60] L. Xu, J. S. J. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pp. 1790–1798, Montreal, Canada, December 2014.

[61] A. Chakrabarti, *A Neural Approach to Blind Motion Deblurring*, Springer International Publishing, Cham, Switzerland, 2016.

[62] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, June 2009.

[63] U. Schmidt, C. Rother, S. Nowozin, J. Jancsary, and S. Roth, "Discriminative non-blind deblurring," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 604–611, Portland, OR, USA, June 2013.

[64] D. Engin, A. Genc, and H. K. Ekenel, "Cycle-dehaze: enhanced CycleGAN for single image dehazing," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2018.

[65] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2016.

Hindawi
International Journal of Optics
Volume 2021, Article ID 5591728, 15 pages
https://doi.org/10.1155/2021/5591728

Research Article

An Appearance Invariant Gait Recognition Technique Using Dynamic Gait Features

Hajra Masood and Humera Farooq

Department of Computer Science, Bahria University Karachi Campus, Karachi, Pakistan

Correspondence should be addressed to Hajra Masood; hajra.cs@gmail.com

Received 10 February 2021; Revised 15 March 2021; Accepted 10 April 2021; Published 3 May 2021

Academic Editor: Muhammad Tariq Mahmood

Gait recognition-based person identification is an emerging trend in visual surveillance due to its uniqueness and adaptability to low-resolution video. Existing gait feature extraction techniques such as gait silhouette and Gait Energy Image rely on the human body's shape. The shape of the human body varies according to the subject's clothing and carrying conditions. The clothing choice changes every day and results in higher intraclass variance and lower interclass variance. Thus, gait verification and gait recognition are required for person identification. Moreover, clothing choices are highly influenced by the subject's cultural background, and publicly available gait datasets lack the representation of South Asian Native clothing for gait recognition. We propose a Dynamic Gait Features extraction technique that preserves the spatiotemporal gait pattern with motion estimation. The Dynamic Gait Features under different Use Cases of clothing and carrying conditions are adaptable for gait verification and recognition. The Cross-Correlation score of Dynamic Gait Features resolves the problem of Gait verification. The standard deviation of Cross-Correlation Score lies in the range of 0.12 to 0.2 and reflects a strong correlation in Dynamic Gait Features of the same class. We achieved 98.5% accuracy on Support Vector Machine based gait recognition. Additionally, we develop a multiappearance-based gait dataset that captures the effects of South Asian Native Clothing (SACV-Gait dataset). We evaluated our work on CASIA-B, OUISIR-B, TUM-IITKGP, and SACV-Gait datasets and achieved an accuracy of 98%, 100%, 97.1%, and 98.8%, respectively.

1. Introduction

Gait recognition for person identification is gaining importance because it is distinct enough for biometric identification and difficult to hide or morph. While face, eyes, and fingerprints biometrics are morphed with face masks, oversized glasses, and gloves. Gait recognition for visual surveillance includes biometric identification [1, 2], gender recognition [3–5], ethnicity classification [6], age group estimation [7–9], and suspect identification in forensics [10, 11].

Gait biometric-based person identification is challenging due to variance in the viewing angle, the direction of walk, speed of the walk, clothing, and carrying items. Among all these challenges, the subject's appearance is critical because it varies daily and alters his/her body's shape. The shape of the subject's body is the primary visual cue for gait feature extraction. Additionally, loose clothing reduces the gait dynamics' visibility, such as self-occlusion due to long coats and gowns, reducing the lower limb's visibility, while carrying items like handbags and satchel adds swinging motion as dynamic noise.

The challenge of gait recognition robust to appearance variance introduces two significant issues: higher intraclass variance and lower interclass variance. Higher intraclass variance refers to the phenomenon in which the same subject looks differently in different clothing combinations. Lower intraclass variance refers to the phenomenon in which different subjects look similar in similar clothing combinations. Higher intraclass variance requires gait verification before gait recognition. This paper proposed Dynamic Gait Features (DGF) extraction that preserves

spatiotemporal gait dynamics with subpixel motion estimation. The contribution of the presented work is outlined as follows:

(1) A novel gait feature extraction approach, Dynamic Gait Features (DGF), is presented. The Dynamic Gait Features preserve spatiotemporal gait dynamics with the help of the subpixel motion estimation technique. The effectiveness of the Dynamic Gait Features for Gait verification is statistically proved with Cross-Correlation Score (CCS).

(2) The Cross-Correlation Score is utilized as a feature vector for Support Vector Machine classifier-based gait recognition. The accuracy of our work is comparable with the existing state-of-the-art techniques.

(3) A new dataset named SACV-Gait is developed to capture the appearance variance induced by South Asian clothing. Dynamic Gait Features are evaluated on the CASAI-B, OUISIR-B, TUM-IITKGP, and SACV-Gait datasets.

The rest of the paper is organized as follows: Section 2 summarizes the existing Gait feature extraction techniques. Section 3 explains the material and methods adapted in our work. Sections 4 and 5 comprise the results and discussion of our work. Section 6 briefly explains the crux of our research work.

2. Existing Work

Gait recognition techniques rely on shape-dependent feature extraction such as Gait Energy Image (GEI) and Gait Silhouette. The shape of the human body varies due to clothing and carrying items and results in gait recognition techniques' performance degradation.

Research studies that utilize GEI for gait recognition include different GEI variants such as Multiscale Gaussian Blur Gait Energy Image (MGEI) and Skeleton Gait Energy Image. Choudhury and Tjahjadi [12] adapted multiscale Gaussian Gait Energy Image (MGEI) for clothing invariant gait recognition. In [13], Bashir et al. adapted Canonical Correlation strength of GEI for gait feature learning across different views. Wu et al. [14] employed Deep Neural Networks with GEI for gait recognition with different walking conditions. In [15], Xu et al. adapted capsule network and GEI for gait recognition robust to multiwalking conditions and multiclothes condition. In [16], Yu et al. employed Deep Neural Networks with stacked multilayer autoencoders to synthesize gait features robust to view clothing and carrying conditions. In [17], Zhang et al. adapted a Long Short Term Memory based autoencoder network for pose based gait feature learning. Yao et al. [18] utilized Skeleton Gait Energy Image (SGEI) and convolutional neural networks for gait recognition with varying clothing conditions.

Research studies that utilize gait silhouette for gait recognition include region-based feature learning, 3D gait modeling, and optical flow field-based gait feature extraction. Chai et al. [19] utilized region-based variance of gait silhouette and Nearest Neighbor classifier. Kastaniotis et al. [20] extracted histogram gait features and kernel Hilbert based feature space for sparse representation for gait recognition. In [21], El-Alfy et al. transformed contours into curvature and developed normal distance maps. Tang et al. [22] utilized contours for the 3D gait feature and adapted multilinear subspace classifiers for gait recognition. In [22, 23], 3D gait modeling with sparse reconstruction is adapted for gait recognition robust to view and clothing variance. In [24], Yu et al. adapted optical flow field and histograms for gait recognition robust to appearance variance. Mahfouf et al. [25] computed optical flow gait features for neural network-based gait recognition. In [26], Wang et al. utilized gait silhouette as a set of three images and adapted a multichannel neural network. Liao et al. [27] proposed pose based temporal spatial network for gait recognition robust to appearance variance.

The adaptation of GEI helped to preserve spatial features, but the temporal variance is not addressed. Similarly, Gait Silhouette's adaptation restrains feature extraction to the contour level, while flat regions are not considered for gait feature extraction. We proposed the Dynamic Gait Features that preserved the gait pattern's spatiotemporal nature and captured motion estimation between gait images.

3. Materials and Methods

3.1. Preparation of Gait Datasets. We have evaluated the proposed framework on CASIA-B [28], OUISIR-B [29], TUM-IITKGP [30], and SACV-Gait dataset. The CASIA-B dataset is considered as a benchmark for the evaluation of gait recognition techniques. The CASIA-B dataset has three Use Cases of appearance variance named normal, bag, and long coat. We have considered these Use Cases as a point of reference for appearance variance and selected similar Use Cases from OUISIR-B, TUM-IITKGP, and SACV-Gait datasets. These three Use Cases define the impact of clothing on the subject's body shape. The first Use Case represents fitted clothing, such as a trouser shirt. The second Use Case represents clothing and carrying items that bring a slight change in the subject's body shape, such as jackets, bags, and loose pants. The third Use Case represents loose clothing that brings a significant change in the shape of the subject's body, such as long coats, gowns, abbaya, and kurta. Table 1 summarizes the three Use Cases for each dataset.

OUISIR-B gait dataset has captured the appearance variance in 32 combinations categorized into Use Cases 1, 2, and 3. Table 2 describes the codes of clothing combinations considered from the OUISIR-B dataset. Figure 1 depicts (top to bottom) use case scenarios of CASIA-B, OUISIR-B, TUM-IITKGP, and SACV datasets to evaluate the proposed work.

3.2. SACV-Gait Dataset. The SACV-Gait dataset captures South Asian ethnic clothing and accessories such as long shirts, abbaye, scarves, dupatta, and hats. The SACV-Gait dataset captured clothing and carrying items in four Use Cases named fitted clothing, fitted clothing with a bag, loose clothing, and loose clothing with a bag.

Table 1: Description of Use Cases 1, 2, and 3 in CASIA-B [28], TUM-IITKGP [30], OUISIR-B [29], and SACV.

Dataset	Use Case 1	Use Case 2	Use Case 3
CASIA-B [28]	Normal	Bag	Long coat
OUISIR-B [29]	Normal	Loose	Long coat
TUM-IITKGP [30]	Normal	Bag	Gown
SACV-Gait (proposed work)	Fitted	Fitted with bag	Knee down

3.2.1. Ethical Data Collection and Usage. The ethical review committee of Bahria University has approved the data collection for research purposes under application number ERC/ES/002. It assures that the procedures adapted for data collection are not harmful to the participant and SACV-Gait data is collected for research purposes solely.

3.2.2. Equipment. We have used a surveillance camera model Grasshopper S2-GE-20S4M-C manufactured by Point Grey's (FLIR Vision) for data collection. The video data has a screen resolution of 1600×1200 pixels with 8-bit depth and a frame rate of 30 fps.

3.2.3. Data Collection Environment. The indoor data collection setup established in Bahria University, Karachi, has a scene depth (distance between the camera and subject) of 6 meters, and participants have walked on a 12-meter long path. We installed a vision camera at the height of 2 meters, and it captured side view gait data. The viewing angle between the subject and camera changed as 45°, 90°, and 135° at the start, middle, and end of the path. Figure 2 illustrates the camera setup for gait data collection.

3.2.4. Subject Statistics. A total of 145 students participated in the research. 121 out of 145 subjects have been selected after preprocessing. We have captured gait in four Use Cases. The SACV-Gait dataset has male and female participants in 4 : 1 ratio. The age group of participants ranges between 18 and 25 years.

3.2.5. Use Cases. The SACV-Gait dataset has 121 subjects under four different Use Cases such as fitted clothing, fitted clothing with a bag, loose clothing, and loose clothing with a bag.

3.2.6. Gradual View Variance. SACV-Gait data have captured the effects of gradual view variance. According to the scene depth, length of the path, and location of the vision camera, the viewing angle at the start, middle, and end of the walking course has been observed at 45°, 90°, and 135°.

3.2.7. Other Covariates. The participants walked along a straight path in two directions (from right to left and from left to right). The surveillance camera's adaptation for data collection provided slightly tilted images similar to real life surveillance videos. Figure 3 captures the male subject of SACV-Gait dataset in different Use Cases. Figure 4 captures the female subject of the SACV-Gait dataset in different Use Cases. Figure 5 represents the subject's walk from left to right and from right to left direction.

3.3. Dynamic Gait Feature-Based Gait Verification and Recognition. The presented research has adapted Dynamic Gait Feature extraction, Cross-Correlation Score analysis for gait verification, and Support Vector Machine based gait recognition. Figure 6 represents the complete framework of gait verification and recognition. The steps of Gait verification and recognition algorithm is provided in Algorithm 1.

3.3.1. Preprocessing. The preprocessing of gait data is performed for foreground extraction and gait cycle detection. The image differencing technique [30] is adapted for foreground extraction. Gait cycle is defined as the time interval between successive instances of initial foot-to-floor contact by the same foot [31]. For gait cycle detection, we have considered two consecutive local minima of the bounding box as the start and endpoints of the gait cycle. Figure 7 represents a complete gait cycle after preprocessing.

3.3.2. Gait Feature Extraction. The process of Dynamic Gait Feature extraction has been performed with subpixel motion estimation [32] on gait images. Following are the steps taken for motion estimation for Dynamic Gait Features:

(1) Initial parameters are set for subpixel motion estimation, such as block space, search space, and gait images.

(2) Sum of Absolute Difference computation has been adapted for motion estimation.

(3) Motion estimation refinement has been performed with Tyler Series partial derivation.

(4) The estimated motion is referred to as Dynamic Gait Features. Gait signature contains features extracted from the complete gait cycle.

(5) The Dynamic Gait Features of each subject in different Use Cases were computed for Gait Signature development.

(6) The Gait Signature was developed by concatenating Dynamic Gait Features

We have implemented Dynamic Gait Feature extraction with subpixel motion estimation on consecutive gait images "GI." Let Gait cycles be captured "n" images denoted as GI_1—GI_n. Equations (1) and (2) summarize Tyler series-based derivative computation for motion estimation. Tyler series has simplified the complex task of multivariate derivation into partial derivatives linear functions. Equation (3) defines Dynamic Gait Feature extraction under different Use Cases where k is the number of Use Cases.

For CASIA-B, OUISIR-B, and TUM-IITKGP, $k = 3$.

For the SACV-Gait dataset, $k = 4$

Table 2: Clothing types in OUISIR-B gait dataset categorized in Use Cases 1, 2, and 3 [28].

Use Case	Clothing combination: upper-lower-accessory
Normal	0: CP-CW, 2: RP-HS, 3: RP-HS-Ht, 4: RP-HS-Cs, 9: RP-F, N: SP-HS, P: SP-Pk, X: RP-FS-Ht, Y: RP-FS-Cs, Z: SP-FS
Loose	A: RP-Pk, B: RP-DJ, C: RP-DJ-Mf, D: CP-HS, F: CP-FS, G: CP-Pk, H: CP-DJ, I: BP-HS, K: BP-FS, L: BP-Pk, M: BP-DJ, R: RC-RC, S: Sk-HS, T: Sk-FS, U: Sk-Pk, V: Sk-DJ
Long coat	5: RP-LC, 6: RP-LC-Mf, 7: RP-LC-Ht, 8: RP-LC-Cs, E: CP-LC, J: BP-LC

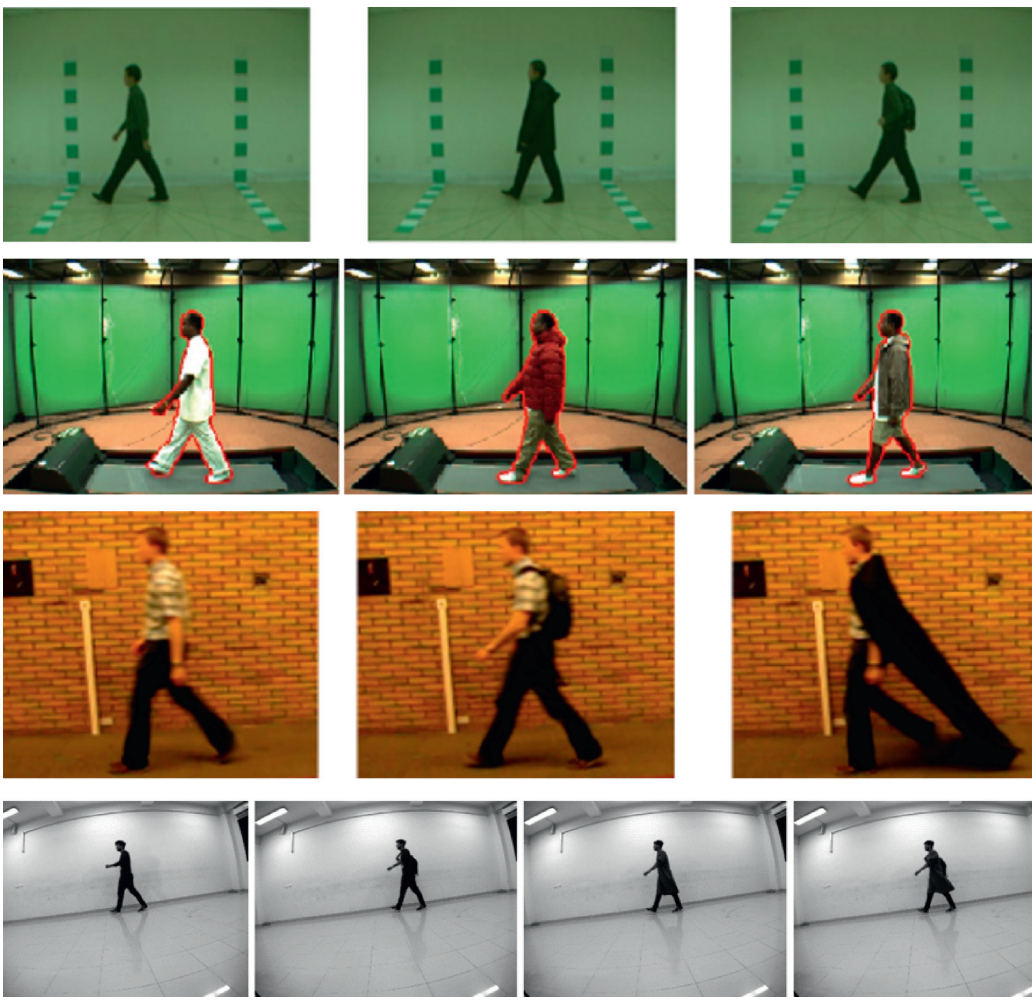


Figure 1: (Top to bottom) use cases of CASIA-B. OUISIR-B, TUM-IITKGP, and SACV dataset.

$$[\mathrm{d}fx, \mathrm{d}fy] = \mathrm{gradient}(f), \tag{1}$$

$$F(x + \mathrm{d}x, y + \mathrm{d}y) = f(x, y) + \mathrm{d}x\frac{\mathrm{d}f}{\mathrm{d}x} + \mathrm{d}y\frac{\mathrm{d}f}{\mathrm{d}y}, \tag{2}$$

$$\mathrm{DGF} = \sum_{\text{use case}=1}^{k} \text{Motion Estimation}\,(\mathrm{IG}_1, \mathrm{IG}_2), (\mathrm{IG}_2, \mathrm{IG}_3), \ldots, (\mathrm{IG}_{n-1}, \mathrm{IG}_n), \tag{3}$$

In equations (1) and (2), derivation between consecutive gait images (GI) has been considered as Dynamic Gait Features. For derivative computation, the Gait Image is considered as $f(x, y)$. The partial derivative of $f(x)$ and $f(y)$ represents the motion estimated between consecutive gait images as mentioned in equation (2). The estimated motion is referred to as Dynamic Gait Features. Gait signature contains features extracted from the complete gait cycle. The Gait Signature was developed by concatenating Dynamic Gait Features between consecutive GI such as $(\mathrm{GI}_1, \mathrm{GI}_2), \ldots,$

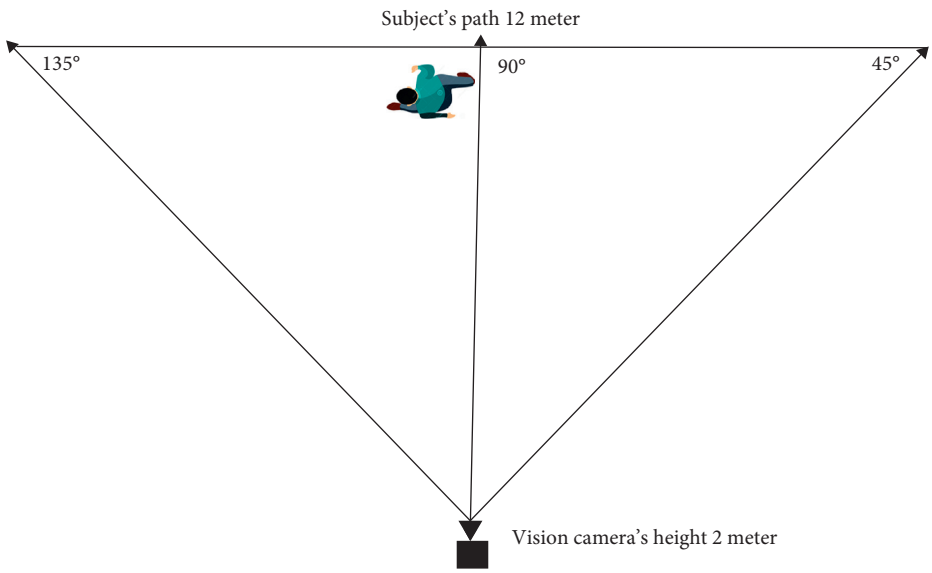


Figure 2: Indoor data collection setup for SACV-Gait dataset.

Figure 3: The male subject of the SACV-Gait dataset in different appearance.

Figure 4: The female subject of the SACV-Gait dataset in different appearances.

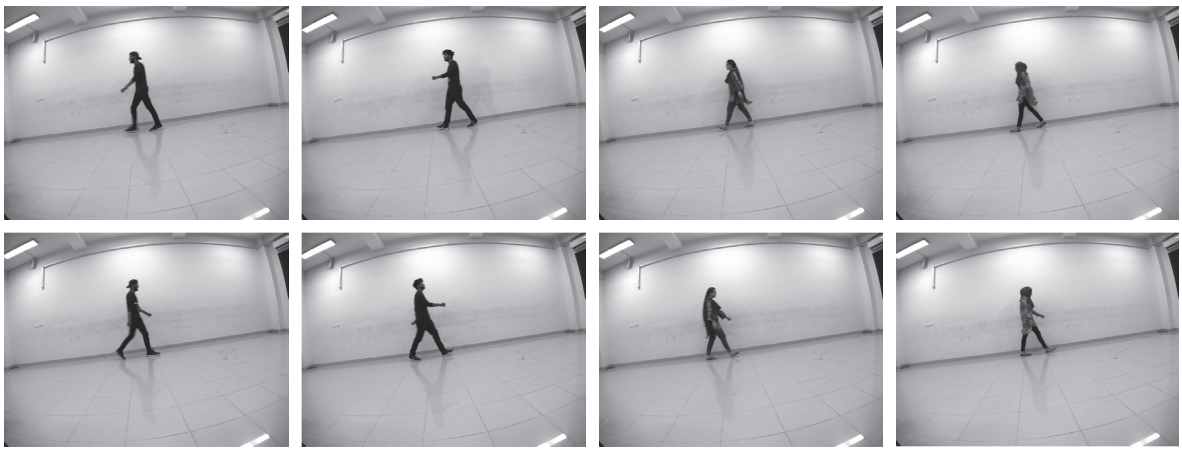


Figure 5: The subjects of SACV-Gait dataset are walking in L-R and R-L directions.

Input: Video Frames of Gait Data
Output: Gait Recognition
Step 1: Preprocessing
Foreground Extraction: $\mathrm{GI}(x, y) = \mathrm{BG}(x, y) - I$
Gait Cycle Detection: Gait Cycle $= \sum_{i=1}^{n} \mathrm{GI}_i$
Step 2: Gait Feature Extraction
Motion Estimation: $F(x + \mathrm{d}x, y + \mathrm{d}y) = f(x, y) + \mathrm{d}x(\mathrm{d}f/\mathrm{d}x) + \mathrm{d}y(\mathrm{d}f/\mathrm{d}y)$
Gait Signature Extraction: Gait cycle $= \mathrm{GI}_1, \mathrm{GI}_2, \mathrm{GI}_3, \mathrm{GI}_4, \ldots, \mathrm{GI}_n$
$\mathrm{DGF} = \sum_{uc=1}^{k}$ Motion Estimation $(\mathrm{GI}_1, \mathrm{GI}_2), (\mathrm{GI}_2, \mathrm{GI}_3), \ldots, (\mathrm{GI}_{n-1}, \mathrm{GI}_n)$
Step 3: Gait Verification
Cross − Correlation Strength Analysis:
$\text{Cross} - \text{Correlation}_{\text{use case 1,use case 2}} = (\text{Covariance}_{\text{use case 1,use case 2}} / \sqrt{\text{Mean Variance}_{\text{use case 1,use case 2.}}})$
Step 4: Gait recognition:
Support Vector Machine based multiclass classification
Return: Gait Recognition based person identification

Algorithm 1: DGF-based gait verification and recognition.

(GI_{n-1} and GI_n), where n is the total number of images in the gait cycle. Figure 8 shows the Dynamic Gait Features extracted from consecutive frames of the gait cycle.

3.3.3. Cross-Correlation Strength Analysis for Gait Verification. The Cross-Correlation Scores between Dynamic Gait Features of different Use Cases are computed to analyze intraclass feature consistency. The standard deviation and relative standard deviation of Cross-Correlation Score helped determine that Dynamic Gait Features of the same subject in different Use Cases are correlated or inconsistent. The lower standard deviation of the Cross-Correlation Score ($\mathrm{SD} < 0.3$) has indicated that Dynamic Gait Features under different appearances belong to the same subject. Relative standard deviation values (30%–70%) have indicated the spread of Dynamic Gait Features in the feature space. The Cross-Correlation Score helped infer that intraclass Dynamic Gait Features are correlated and consistent enough for gait verification. Equations (4)–(6) summarize the computation of covariance, mean-variance, and Cross-Correlation Score.

$$\text{covariance}_{\text{use case 1,use case 2}} = \frac{1}{N} \sum_{t=1}^{N} \left(X_{\text{use case 1}} - \overline{X}_{\text{use case 1}}\right)_t \left(X_{\text{use case 2}} - \overline{X}_{\text{use case 2}}\right)_t, \tag{4}$$

$$\text{mean variance}_{\text{use case 1,use case 2}} = \sqrt{\sum_{t=1}^{N} \left(X_{\text{use case 1}} - \overline{X}_{\text{use case 1}}\right)_t \left(X_{\text{use case 2}} - \overline{X}_{\text{use case 2}}\right)_t}, \tag{5}$$

$$\text{Cross} - \text{Correlation}_{\text{use case 1,use case 2}} = \text{CCS} = \frac{\text{covariance}_{\text{use case 1,use case 2}}}{\sqrt{\text{mean variance}_{\text{use case 1,use case 2}}}}, \tag{6}$$

$$\text{Standard deviation} = \text{SD} = \frac{\sum_{i=1}^{n} \sqrt{\left(\text{CCS}_i - \overline{\text{CCS}}\right)^2}}{n-1}, \tag{7}$$

$$\text{Relative standard deviation} = \frac{\text{SD}}{\overline{\text{CCS}}} \times 100. \tag{8}$$

(1) Cross-Correlation Score (CCS). The Cross-Correlation Score provides the similarity between two time series data. The Cross-Correlation Score provides statistical evidence to analyze that Dynamic Gait Features of the same subject under different Use Cases are strongly correlated or not. The range of values for the Cross-Correlation Score varies between −1 and 1.

(2) Standard Deviation (SD). The standard deviation score reflects the variance of data points from their mean. The lower standard deviation of Cross-Correlation Score

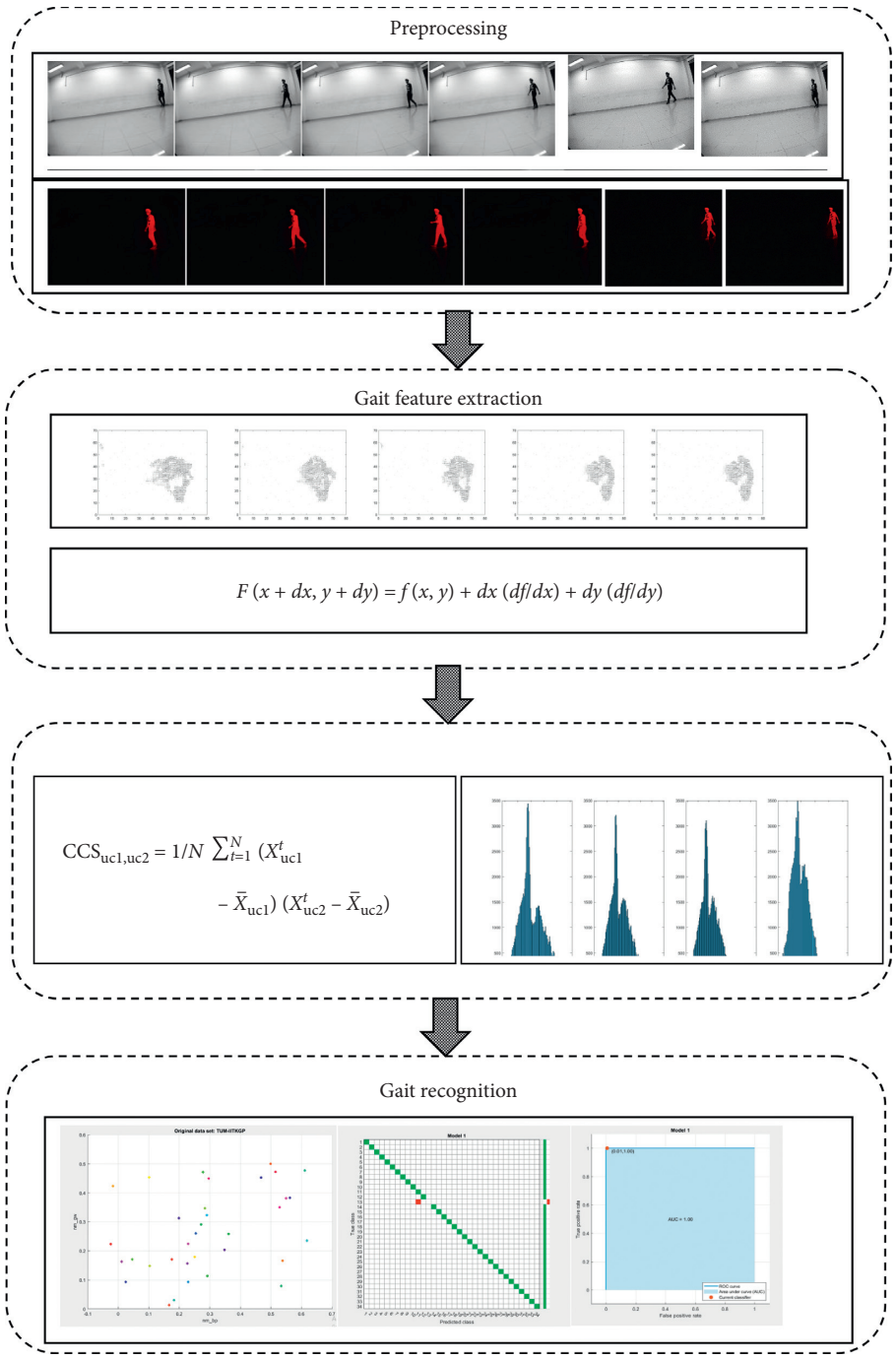


Figure 6: The workflow of DGF- and SVM-based gait verification and recognition.

statistically proves that Dynamic Gait Features are consistent and highly correlated despite significant appearance variance. The maximum standard deviation of the Cross-Correlation Score observed was 0.3. The computation of SD of Cross-Correlation Score is explained in equation (7).

(3) Relative Standard Deviation (RSD). The relative standard deviation represents the difference between standard deviation and the mean of data points. Lower relative standard deviation indicates tightly clustered data points, and higher standard deviation values indicate the spread of data points. The relative standard deviation score lies in the range from 30% to 70%. Higher relative standard deviation values indicate that Cross-Correlation Score Gait features require nonlinear classifiers for Gait recognition. Equation (8) explains the computation of RSD. The higher relative standard deviation values indicate the spread of features and suitability of nonlinear hyperplanes for classification.

Table 3: Standard deviation and relative standard deviation score of CCS for gait verification.

Dataset	Pair (Use Case 1, Use Case 2)	Mean	Variance	SD	RSD = SD/mean %
CASIA-B	Pair 1 (normal, bag)	0.4	0.015	0.12	0.12/0.4 = 0.3 = 30%
	Pair 2 (normal, long coat)	0.4	0.014	0.2	0.2/0.4 = 0.5 = 50%
	Pair 3 (bag, long coat)	0.6	0.04	0.2	0.2/0.6 = 0.33 = 33%
OUISIR-B	Pair 1 (normal, loose)	0.2	0.02	0.141	0.141/0.2 = 0.7 = 70%
	Pair 2 (normal, long coat)	0.19	0.02	0.144	0.144/0.19 = 0.7 = 70%
	Pair 3 (loose, long coat)	0.16	0.02	0.14	0.14/0.16 = 0.8 = 80%
TUM-IITKGP	Pair 1 (normal, back pack)	0.29	0.03	0.18	0.18/0.29 = 0.6 = 60%
	Pair 2 (normal, gown)	0.27	0.02	0.14	0.14/0.27 = 0.5 50%
	Pair 3 (back pack, gown)	0.28	0.02	0.15	0.15/0.28 = 0.5 = 50%
SACV	Pair 1 (fitted, fitted with bag)	0.42	0.05	0.23	0.23/0.42 = 0.54 = 54%
	Pair 2 (fitted, knee down)	0.44	0.048	0.22	0.22/0.44 = 0.5 = 50%
	Pair 3 (knee down, knee down with bag)	0.52	0.056	0.23	0.23/0.52 = 0.44 = 44%
	Pair 4 (fitted with bag, knee down with bag)	0.47	0.038	0.19	0.19/0.47 = 0.4 = 40%

Figure 7: Gait data after foreground extraction and gait cycle detection.

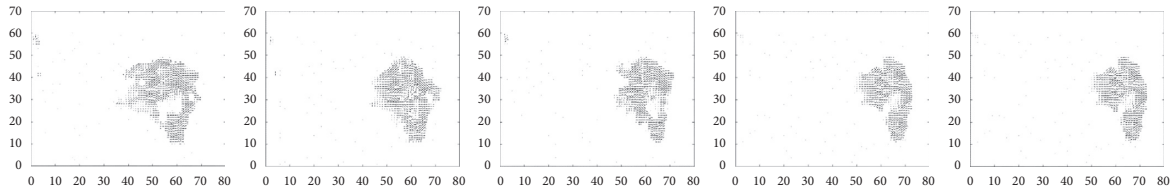


Figure 8: DGF extraction through motion estimation between consecutive gait images.

3.3.4. Classification with Support Vector Machine for Gait Recognition. We adapt Cross-Correlation Score with quadratic Support Vector Machine classifier for Gait recognition. Support Vector Machine classifier works by defining relative decision boundaries between two classes. The quadratic Support Vector Machine kernels utilize hyperplanes similar to the hyperparameters of Deep Neural Networks for classification. Deep learning-based approaches [17, 27, 33, 34] for Gait recognition also adapt hyperparameters-based Gait recognition.

Dynamic Gait Feature extraction for the SACV-Gait dataset with computational details are as follows.

We consider the gait cycle of different lengths for each dataset. Such as in the SACV-Gait dataset, the gait cycle length varies between 7 and 11 images. While in CASIA-B and OUISIR-B datasets, it lies in the range of 21 to 36 images.

Cycle length = $n = 7$ (GI_1–GI_7)

Image size of SACV = $1200 \times 1451 \times 3$

Feature vector Dynamic Gait Features between 2 consecutive gait images = 60×72 double

Feature vector Dynamic Gait Features of complete cycle = 300×72 double

Cross-Correlation Score of Dynamic Gait Features vectors in 4 different Use Cases = 1×4 double.

Features are computed and stored in the double format as it captures the change in magnitude in detail.

4. Results

We evaluated Dynamic Gait Feature-based gait verification and recognition on CASIA-B, OUISIR-B, TUM-IITKGP, and the SACV-Gait dataset for evaluation.

4.1. Experimental Results on CASIA-B. CASIA-B Gait dataset [28] consists of 124 subjects with three Use Cases named normal, long coat, and bag captured from a 90° viewing angle. The Cross-Correlation Scores of pair 1 (normal, bag), pair 2 (normal, long coat), and pair 3(bag, long coat) were further analyzed with standard deviation and relative standard deviation. The standard deviation score of pair 1, pair 2, and pair 3 was 0.12, 0.2, and 0.2. The relative standard deviation of pair 1, pair 2, and pair 3 was 30%, 50%, and 33%. We adapted Cross-Correlation Score with a Support Vector Machine for gait recognition and achieved 100%. Figure 9 represents the standard deviation plotted with a normal distribution curve for pair 1, pair 2, and pair 3. Figure 10 summarizes the scatter plot, confusion matrix, and receiver operating characteristic curve of gait recognition.

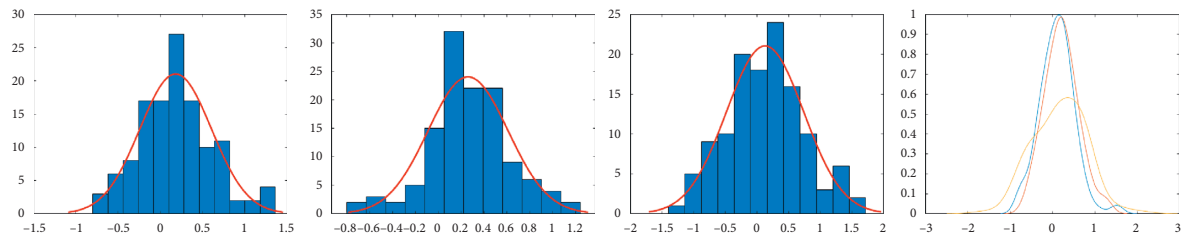


Figure 9: Standard deviation and normal distribution curve of pair 1, pair 2, and pair 3.

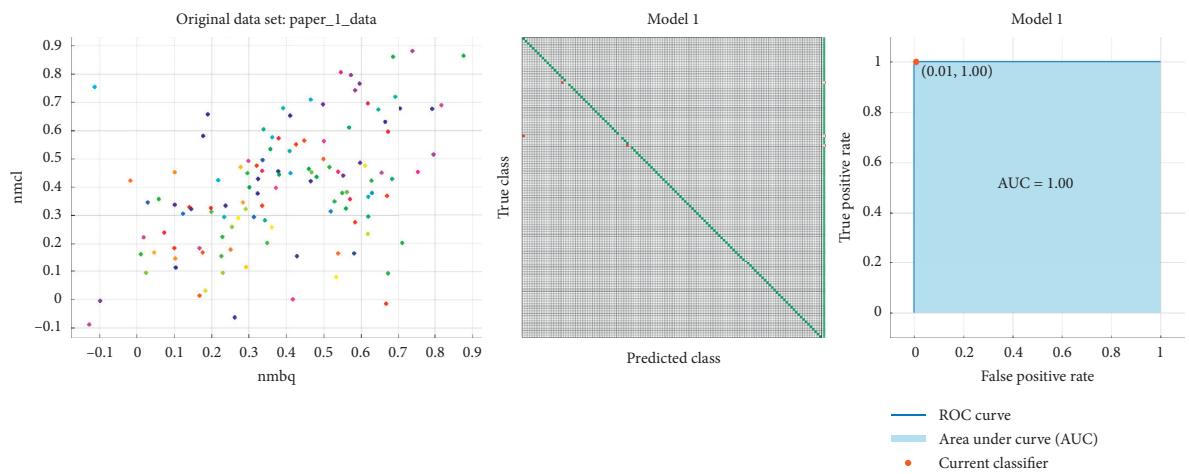


Figure 10: Gait recognition with Cross-Correlation Score, confusion matrix, and ROC curve of CASIA-B.

4.2. Experimental Results on OUISIR-B. OUISIR-B [29] dataset contains 65 subjects in 32 different types of clothing combinations. We classify the clothing combinations into three Use Cases as normal, loose, and long coat. Table 2 mentions the dress codes under each use case. The Cross-Correlation Scores computation of pair 1 (normal, loose), pair 2 (normal, long coat), and pair 3 (loose, long coat) were further analyzed with standard deviation and relative standard deviation. The standard deviation score of pair 1, pair 2, and pair 3 was 0.141, 0.144, and 0.140. The relative standard deviation of pair 1, pair 2, and pair 3 was 70%, 70%, and 80%. We adapted the Cross-Correlation Score with the Support Vector Machine for gait recognition and achieved 100% accuracy. Figure 11 represents the normal distribution curve of pair 1, pair 2, and pair 3. Figure 12 summarizes the scatter plot, confusion matrix, and receiver operating characteristic curve of gait recognition.

4.3. Experimental Results on TUM-IITKGP. The TUM-IITKGP dataset [30] contains 35 subjects with three relevant Use Cases: normal, bag, and gown. We further analyzed Cross-Correlation Scores of pair 1 (normal, bag), pair 2 (normal, gown), and pair 3 (bag, gown) with standard deviation and relative standard deviation. The standard deviation of pair 1, pair 2, and pair 3 was 0.141, 0.144, and 0.140. The relative standard deviation of pair 1, pair 2, and pair 3 was 60%, 50%, and 50%. Figure 13 represents the normal distribution curve of pair 1, pair 2, and pair 3. We adapted the Cross-Correlation Score with the Support Vector Machine and achieved 97.1% accuracy. Figure 14 summarizes the scatter plot, confusion matrix, and receiver operating characteristic curve of gait recognition.

4.4. Experimental Results on SACV-Gait Dataset. The SACV-Gait dataset contains 121 subjects under four Use Cases of fitted, fitted with the bag, knee down, and knee down with the bag. We further analyzed Cross-Correlation Scores of pair 1 (fitted, fitted with bag), pair 2 (fitted, knee down), pair 3 (knee down, knee down with bag), and pair 4 (fitted with the bag, knee down with bag) with standard deviation and relative standard deviation. The standard deviation score of pair 1, pair 2, pair 3, and pair 4 was 0.23, 0.22, 0.23, and 0.19. The relative standard deviation of pair 1, pair 2, and pair 3 was 54%, 50%, 44%, and 40%. We adapted Cross-Correlation Score with the Support Vector Machine classifier and achieved 98.8% accuracy. Figure 15 represents the normal distribution curve of pair 1, pair 2, pair 3, and pair 4. Figure 16 summarizes the scatter plot, confusion matrix, and receiver operating characteristic curve of gait recognition.

5. Discussion

5.1. Gait Verification. This research work adapted Dynamic Gait Features for gait verification and recognition. For Gait verification, we computed the Cross-Correlation Score between different Use Cases. The standard deviation of Cross-Correlation Score helped to understand intraclass feature dispersion. The Cross-Correlation Score with a low standard deviation score showed the consistency of Dynamic Gait

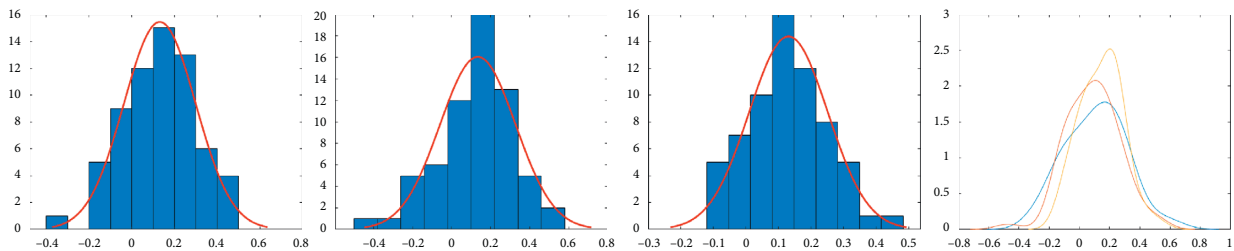


Figure 11: Standard deviation and normal distribution curve of pair 1, pair 2, and pair 3.

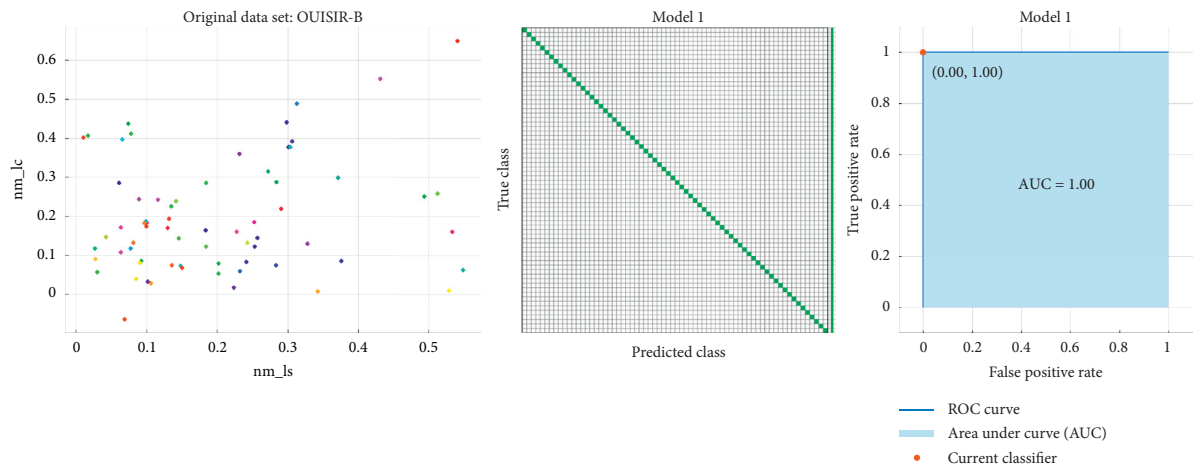


Figure 12: Gait recognition with Cross-Correlation Score, confusion matrix, and ROC curve of OUISIR-B.

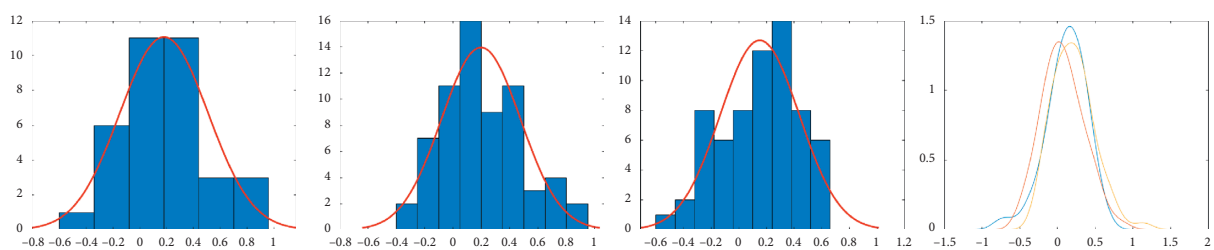


Figure 13: Standard deviation and normal distribution curve of pair 1, pair 2, and pair 3.

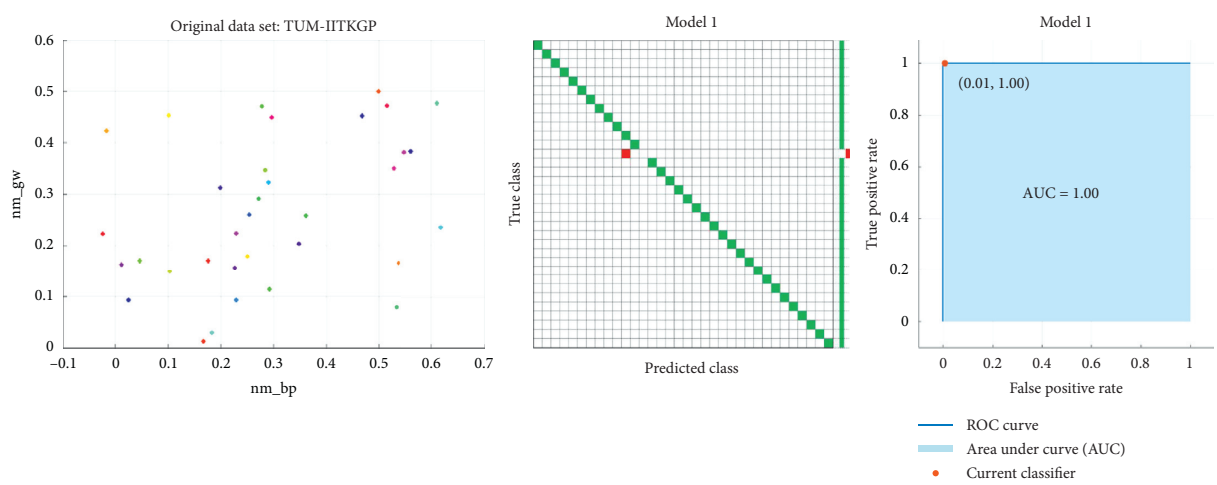


Figure 14: Gait recognition with Cross-Correlation Score, confusion matrix, and ROC curve of TUM-IITKGP.

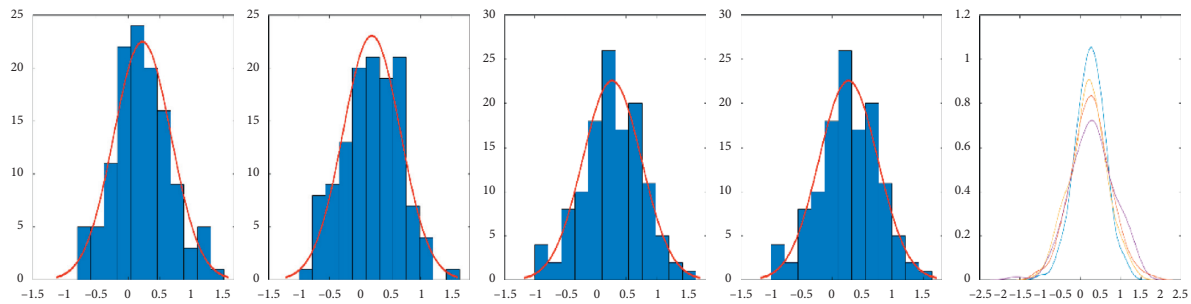


Figure 15: Standard deviation and normal distribution curve of pair 1, pair 2, pair 3, and pair 4.

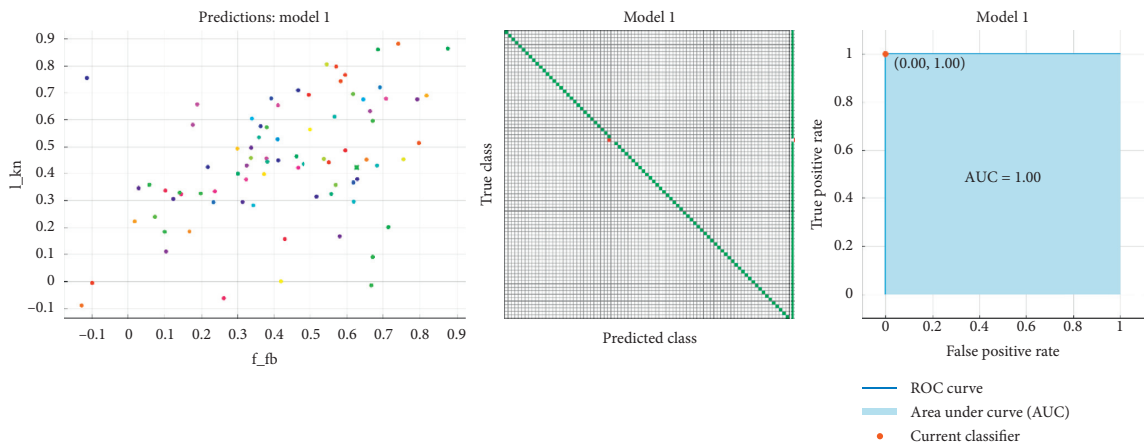


Figure 16: Gait recognition with Cross-Correlation Score, confusion matrix, and ROC curve of SACV-Gait.

Features despite significant appearance variance. Overall, the standard deviation of Cross-Correlation Score was in the range of 0.12 to 0.23. That indicates the correlation and consistency between Dynamic Gait Features of different Use Cases.

Gait verification on pair 1 (normal, bag), pair 2 (normal, long coat), and pair 3 (bag, long coat) of the CASIA-B dataset resulted in standard deviation scores of 0.12, 0.2, and 0.2. The overall variance in standard deviation was 0.08. The relative standard deviation score of pair 1, pair 2, and pair 3 was observed as 30%, 50%, and 33%. The lower standard deviation showed a higher correlation between DGF features within the same class. Relative standard deviation's higher values indicate significant variance between Use Cases and their impact on feature dispersion.

Gait verification on pair 1 (normal, loose), pair 2 (normal, long coat), and pair 3 (loose, long coat) of the OUISIR-B gait dataset resulted in standard deviation scores of 0.141, 0.144, and 0.14. Overall variance in standard deviation was 0.003. The relative standard deviation values of pair 1, pair 2, and pair 3 were observed as 70%, 70%, and 80%. The standard deviation score was the lowest, and RSD was the highest among all datasets. The lower standard deviation score validated the adaptability of the Cross-Correlation Score for gait verification. In contrast, higher relative standard deviation values indicated the heterogeneous and spatially diverse nature of the clothing combinations considered in the OUISIR-B dataset.

Gait verification on pair 1 (normal, bag), pair 2 (normal, gown), and pair 3 (bag, gown) of the TUM-IITKGP dataset resulted in a standard deviation score of 0.18, 0.14, and 0.15. The overall variance in standard deviation was 0.04. The relative standard deviation of pair 1, pair 2, and pair 3 was observed as 60%, 50%, and 50%.

Gait verification with pair 1 (fitted, fitted with bag), pair 2 (fitted, knee down), pair 3 (knee down, knee down with bag), and pair 4 (fitted with the bag, knee down with bag) of SACV-Gait dataset resulted in standard deviation score of 0.23, 0.22, 0.23, and 0.19. The overall variance in standard deviation was 0.04. The relative standard deviation of pair 1, pair 2, pair 3, and pair 4 was 54%, 50%, 44%, and 40%. Table 3 summarizes the standard deviation and relative standard deviation score for the Cross-Correlation Score of Dynamic Gait Features.

5.1.1. Standard Deviation (SD). The standard deviation score of all datasets lies in the range from 0.12 to 0.23. Collectively, the standard deviation score was less than 0.3. The lower standard deviation indicates that Dynamic Gait Features are highly correlated despite significant appearance variance. Standard deviation reflects the intraclass consistency of Dynamic Gait Features and assures that gait features extracted from different appearances belong to the same subject (Gait verification). Figure 17 illustrates the standard deviation of Cross-Correlation Score computed for pair 1, pair 2, and pair 3.

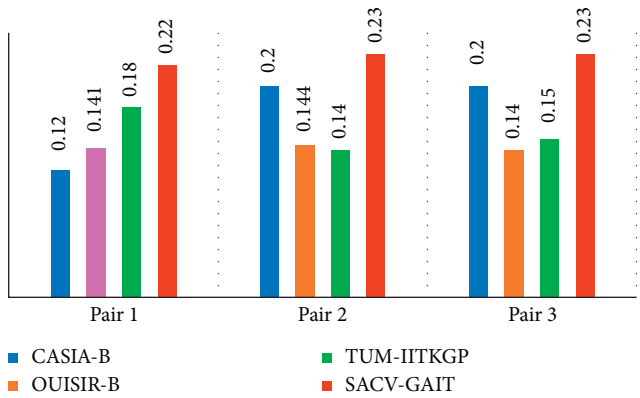


Figure 17: SD of CCS observed in pair 1, pair 2, and pair 3.

5.1.2. The Variance in Standard Deviation (SD). Collectively, the variance in standard deviation ranged from 0.003 to 0.1. This pattern indicated that Cross-Correlation Score is an effective way to develop statistically consistent gait features while capturing a wide range of clothing combinations.

5.1.3. The Relative Standard Deviation (RSD). The relative standard deviation score for Cross-Correlation Score lies in the range from 30% to 80%. Those higher values of relative standard deviation score reflected the Cross-Correlation Score variance due to significantly different Use Cases. The standard deviation score reflected the intraclass consistency of Dynamic Gait Features and assured that gait features extracted from different appearances belong to the same subject. Table 3 summarizes the standard deviation and relative standard deviation score of Cross-Correlation Score. Figure 18 represents relative standard deviation of Cross-Correlation Score computed for pair 1, pair 2, and pair 3.

5.2. Gait Recognition. For gait recognition, we compute the Cross-Correlation Score between Dynamic Gait Features under different Use Cases. This Cross-Correlation Score was utilized as the feature vector for Support Vector Machine based gait recognition. We achieved 98%, 97.1%, 100%, and 98.8% accuracy on CASIA-B, OUISIR-B, TUM-IITKGP, and SACV-Gait datasets. The proposed gait features are evaluated on CASIA-B, OUISIR-B, and TUM-IITKGP and achieve 98%, 97.1%, and 100% accuracy. Although these datasets are available in binary format and motion estimation at the global level was detected from the contour area, the presented gait features' consistency and discriminability are significant.

The Dynamic Gait Features computed from contours are similar to normal distance mapping [35] as both techniques encode gait dynamics at the contour level. In normal distance maps, normal vectors depend on the curvature between two successive contour points [36]. The Dynamic Gait Features adaptation with motion estimation at the global level helped to encode gait dynamics at different levels and provided stable gait features regardless of high scene depth. Motion estimation with optical flow-based approaches [22, 24] performed well and provided efficient pixel flow tracking. The optical flow-based approach also interprets dynamic noise as motion due to the brightness constancy constraint. Additionally, the flat regions of silhouette images did not contribute to motion estimation due to spatial smoothness constraints [37].

5.3. Comparison with Existing Work. The proposed work performed gait recognition on CASIA-B, OUISIR-B, and TUM-IITKGP with 98%, 97.1%, and 100% accuracy. The Cross-Correlation Score of Dynamic Gait Feature enabled us to resolve higher intraclass variance and perform gait verification. We utilized the Cross-Correlation Score as a feature vector for Support Vector Machine based Gait recognition. Table 4 summarizes the accuracy of our and existing gait recognition techniques.

The accuracy of Dynamic Gait Features-based Gait recognition is comparable with existing feature extraction techniques such as GEI [12, 17, 38] and Gait silhouette [22, 23, 26]. State of the artwork reported in recent years [12, 17, 23, 25], and [38] are evaluated on CASIA-B dataset and achieved accuracy of 89%, 92.6%, 96%, 99%, and 90.43%, respectively. We achieved 98% accuracy on the CASIA-B dataset. Figure 19 summarizes the accuracy of our work and its comparison with existing work.

CASIA-B is a benchmark for the evaluation of gait recognition techniques. The research works reported in [22, 27, 39, 40] and presented work is evaluated for the normal, bag, and long coat-based Use Cases of CASIA-B separately. The accuracy of [22] on CASIA-B's Use Cases normal, bag, and the long coat was 99%, 96%, and 95%. Similarly, the accuracy for Use Cases of TUM-IITKGP normal, bag, and gown cases was 99%, 80%, and 65%. Over the accuracy of [22] declines for second and third Use Case of CASIA-B and TUM-IITKGP. Accuracy of [27] is reported as 96% for normal, 79% for the bag, and 61% for a long coat. Similarly, the accuracy of [39] for normal, bag, and the long coat was 97.58%, 70.16%, and 56.45%. The accuracy of [40] is 98%, 90%, and 64% for the normal, bag, and long coat. Figure 20 graphically shows this consistent accuracy of our work and its comparison with [22, 27, 39, 40].

The research contributions of [18, 27] and presented work also reflects the adaptability of gait dynamics-based

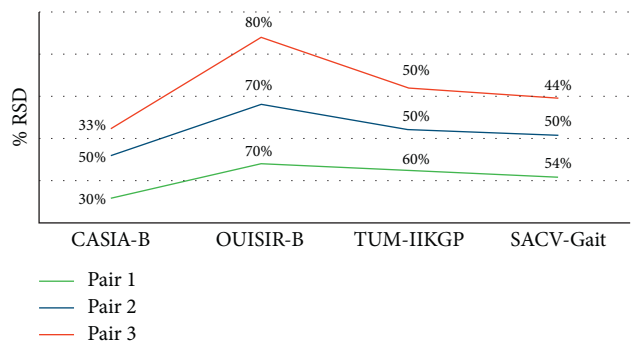


Figure 18: Relative standard deviation computed for pair 1, pair 2, and pair 3.

Table 4: Gait recognition accuracy achieved by our work and existing work.

Research work	CASIA-B	TUM-IITKGP	OUISIR-B
TTGS + MCCNN [26]	99%	—	—
3D Gait model + partial Similarity [22]	99%	99%	—
	96%	80%	—
	95%	65%	
	Average = 96.6%		
3D Gait + Sparse reconstruction [23]	96%	—	—
GEI + PCA + WRSL [12]	89%	—	—
GEI + DRL + CNN [17]	92.6%	—	—
GEI + MSCNN [38]	90.43%	—	—
Effective joints + LSTM + CNN [27]	96%	—	—
	79%		
	61%		
	Average = 79.6%		
Pose + LSTM + CNN [39]	97.58%	—	—
	70.16%		
	56.45%		
	Average = 74.7%		
Optical flow, PCA, LDA [40]	98%		
	90%		
	64%		
	Average = 84%		
Our work (DGF, CCS, SVM)	98%	97.1%	100%

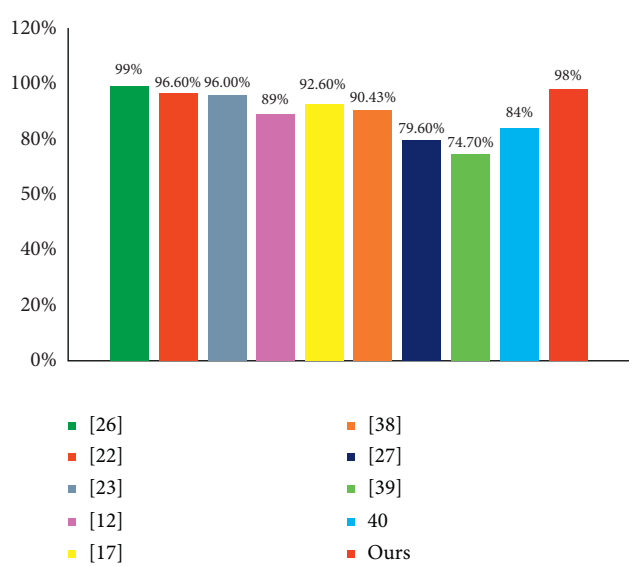


Figure 19: The performance evaluation of existing and presented work on CASIA-B.

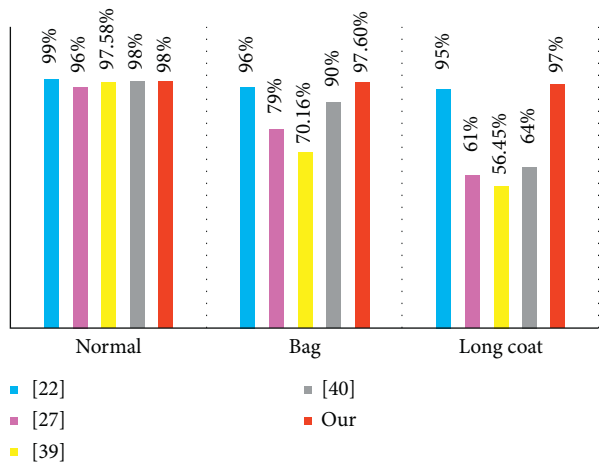


Figure 20: Comparison of our work with [22, 27, 39, 40].

feature extraction techniques for appearance invariant gait recognition. The Dynamic Gait Features are extracted through motion estimation at the global level with varying search space sizes. Motion estimation at the global level helps to encode gait dynamics at different levels and provides stable gait features regardless of high scene depth. Motion estimation with optical flow-based approaches [22, 24] depends on pixel flow tracking and lacks robustness to dynamic noise as optical flow-based techniques interpret dynamic noise as motion due to the brightness constancy constraint. Additionally, the flat regions of silhouette images did not contribute to motion estimation due to spatial smoothness constraints [37].

6. Conclusion and Future Work

The standard deviation score and percentage accuracy for gait verification and recognition reflect Cross-Correlation Score's effectiveness for multiclass classification problems with higher intraclass and lower interclass variance. The DGF builds consistency within the same class despite significant appearance variance. Adaptation of subpixel motion estimation preserves the spatiotemporal gait features. Additionally, the summation of DGF extracted under different appearances is a better approach than handcrafted feature extraction. In our work, the Cross-Correlation Score of Dynamic Gait Features reduces the feature dimensionality and computational complexity. Our future work includes the adaptation of DGF with neural network-based feature learning across different views and appearances.

Data Availability

The dataset will be available for future studies related to gait recognition.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] I. Rida, N. Almaadeed, and S. Almaadeed, "Robust gait recognition: a comprehensive survey," *IET Biometrics*, vol. 8, no. 1, pp. 14–28, 2018.

[2] I. Bouchrika, "A survey of using biometrics for smart visual surveillance: gait recognition," *Surveillance in Action*, pp. 3–23, 2018.

[3] T. Liu, X. Ye, and B. Sun, "Combining convolutional neural network and support vector machine for gait-based gender recognition," in *Proceedings of the 2018 Chinese Automation Congress (CAC)*, IEEE, Xi'an, China, 2018.

[4] K. Kitchat, N. Khamsemanan, and C. Nattee, "Gender classification from gait silhouette using observation angle-based GEIs," in *Proceedings of the 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, IEEE, Bangkok, Thailand, 2019.

[5] E. R. H. P. Isaac, S. Elias, S. Rajagopalan, and K. S. Easwarakumar, "Multiview gait-based gender classification through pose-based voting," *Pattern Recognition Letters*, vol. 126, pp. 41–50, 2019.

[6] D. Zhang, Y. Wang, and B. Bhanu, "Ethnicity classification based on gait using multi-view fusion," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, IEEE, San Francisco, CA, USA, 2010.

[7] C. Xu, Y. Makihara, G. Ogi, X. Li, Y. Yagi, and J. Lu, "The OU-ISIR gait database comprising the large population dataset with age and performance evaluation of age estimation," *IPSJ Transactions on Computer Vision and Applications*, vol. 9, no. 1, pp. 1–14, 2017.

[8] X. Li, Y. Makihara, C. Xu, Y. Yagi, and M. Ren, "Gait-based human age estimation using age group-dependent manifold learning and regression," *Multimedia Tools and Applications*, vol. 77, no. 21, pp. 28333–28354, 2018.

[9] A. Sakata, N. Takemura, and Y. Yagi, "Gait-based age estimation using multi-stage convolutional neural network," *IPSJ Transactions on Computer Vision and Applications*, vol. 11, no. 1, pp. 1–10, 2019.

[10] I. Bouchrika, J. N. Carter, and M. S. Nixon, "Towards automated visual surveillance using gait for identity recognition and tracking across multiple non-intersecting cameras," *Multimedia Tools and Applications*, vol. 75, no. 2, pp. 1201–1221, 2016.

[11] N. M. van Mastrigt, K. Celie, A. L. Ruifrok, A. C. C. Ruifrok, and Z. Geradts, "Critical review of the use and scientific basis of forensic gait analysis," *Forensic Sciences Research*, vol. 3, no. 3, pp. 183–193, 2018.

[12] S. D. Choudhury and T. Tjahjadi, "Robust view-invariant multiscale gait recognition," *Pattern Recognition*, vol. 48, no. 3, pp. 798–811, 2015.

[13] K. Bashir, T. Xiang, and S. Gong, "Cross view gait recognition using correlation strength," in *Proceedings of the BMVC*, Wales, UK, 2010.

[14] Z. Wu, Y. Huang, L. Wang, X. Wang, and T. Tan, "A comprehensive study on cross-view gait based human identification with deep CNNS," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 209–226, 2016.

[15] Z. Xu, W. Lu, Q. Zhang, Y. Yeung, and X. Chen, "Gait recognition based on capsule network," *Journal of Visual Communication and Image Representation*, vol. 59, pp. 159–167, 2019.

[16] S. Yu, H. Chen, Q. Wang, L. Shen, and Y. Huang, "Invariant feature extraction for gait recognition using only one uniform model," *Neurocomputing*, vol. 239, pp. 81–93, 2017.

[17] Z. Zhang, L. Tran, X. Yin et al., "Gait recognition via disentangled representation learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 2019.

[18] T. Huynh-The, C.-H. Hua, N. A. Tu, and D.-S. Kim, "Learning 3D spatiotemporal gait feature by convolutional network for person identification," *Neurocomputing*, vol. 397, pp. 192–202, 2020.

[19] Y. Chai, Q. Wang, J. Jia, and R. Zhao, "A novel human gait recognition method by segmenting and extracting the region variance feature," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06)*, IEEE, Hong Kong, China, 2006.

[20] D. Kastaniotis, I. Theodorakopoulos, and S. Fotopoulos, "Pose-based gait recognition with local gradient descriptors and hierarchically aggregated residuals," *Journal of Electronic Imaging*, vol. 25, no. 6, Article ID 063019, 2016.

[21] H. El-Alfy, I. Mitsugami, and Y. Yagi, "Gait recognition based on normal distance maps," *IEEE Transactions on Cybernetics*, vol. 48, no. 5, pp. 1526–1539, 2017.

[22] J. Tang, J. Luo, T. Tjahjadi, and F. Guo, "Robust arbitrary-view gait recognition based on 3D partial similarity matching," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 7–22, 2016.

[23] J. Luo, J. Tang, T. Tjahjadi, and X. Xiao, "Robust arbitrary view gait recognition based on parametric 3D human body reconstruction and virtual posture synthesis," *Pattern Recognition*, vol. 60, pp. 361–377, 2016.

[24] C. C. Yu, C. H. Cheng, and K. C. Fan, "A gait classification system using optical flow features," *Journal of Information Science and Engineering*, vol. 30, no. 1, pp. 179–193, 2014.

[25] Z. Mahfouf, H. F. Merouani, I. Bouchrika, and N. Harrati, "Investigating the use of motion-based features from optical flow for gait recognition," *Neurocomputing*, vol. 283, pp. 140–149, 2018.

[26] X. Wang, J. Zhang, and W. Q. Yan, "Gait recognition using multichannel convolution neural networks," *Neural Computing and Applications*, vol. 32, no. 18, pp. 14275–14285, 2019.

[27] R. Liao, C. Cao, E. B. Garcia, S. Yu, and Y. Huang, "Pose-based temporal-spatial network (PTSN) for gait recognition with carrying and clothing variations," in *Proceedings of the Chinese Conference on Biometric Recognition*, Springer, Shenzhen, China, 2017.

[28] S. Yu, D. Tan, and T. Tan, "Modelling the effect of view angle variation on appearance-based gait recognition," in *Proceedings of the Asian Conference on Computer Vision*, Springer, Hyderabad, India, 2006.

[29] Y. Makihara, H. Mannami, A. Tsuji et al., "The OU-ISIR gait database comprising the treadmill dataset," *IPSJ Transactions on Computer Vision and Applications*, vol. 4, pp. 53–62, 2012.

[30] M. Hofmann, S. Sural, and G. Rigoll, "Gait recognition in the presence of occlusion: a new dataset and baseline algorithms," in *Proceedings of the WSCG'2011*, Plzen, Czech Republic, 2011.

[31] S. Gong, C. Liu, Y. Ji, B. Zhong, Y. Li, and H. Dong, *Advanced Image and Video Processing Using MATLAB*, Springer, Berlin, Germany, 2018.

[32] S. H. Chan, D. T. Vo, and T. Q. Nguyen, "Subpixel motion estimation without interpolation," in *Proceedings of the ICASSP*, Dallas, TX, USA, 2010.

[33] J. P. Singh, S. Jain, S. Arora, and U. P. Singh, "Vision-based gait recognition: a survey," *IEEE Access*, vol. 6, pp. 70497–70527, 2018.

[34] R. Martín-Félez and T. Xiang, "Gait recognition by ranking," in *Proceedings of the European Conference on Computer Vision*, Springer, Florence, Italy, 2012.

[35] W. Liu, C. Zhang, H. Ma, and S. Li, "Learning efficient spatial-temporal gait features with deep learning for human identification," *Neuroinformatics*, vol. 16, no. 3-4, pp. 457–471, 2018.

[36] A. Sud, N. Govindaraju, R. Gayle, E. Andersen, and D. Manocha, "Surface distance maps," in *Proceedings of the Graphics Interface 2007*, Montréal, Canada, 2007.

[37] D. Sun, S. Roth, J. Lewis, and M. J. Black, "Learning optical flow," in *Proceedings of the European Conference on Computer Vision*, Springer, Marseille, France, 2008.

[38] L. Yao, W. Kusakunniran, Q. Wu, J. Zhang, Z. Tang, and W. Yang, "Robust gait recognition using hybrid descriptors based on skeleton gait energy image," *Pattern Recognition Letters*, 2019.

[39] R. Liao, S. Yu, W. An, and Y. Huang, "A model-based gait recognition method with body pose and human prior knowledge," *Pattern Recognition*, vol. 98, Article ID 107069, 2020.

[40] Z. Luo, T. Yang, and Y. Liu, "Gait optical flow image decomposition for human recognition," in *Proceedings of the 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*, IEEE, Chongqing, China, 2016.