

## Research Article

# Facial Expression Recognition Using Kernel Entropy Component Analysis Network and DAGSVM

Xiangmin Chen,<sup>1,2</sup> Li Ke ,<sup>1</sup> Qiang Du,<sup>1</sup> Jinghui Li,<sup>1</sup> and Xiaodi Ding<sup>1</sup>

<sup>1</sup>College of Electrical Engineering, Shenyang University of Technology, Shenyang 110870, China

<sup>2</sup>Department of Information and Control Engineering, Shenyang Institute of Science and Technology, Shenyang 110101, China

Correspondence should be addressed to Li Ke; keli@sut.edu.cn

Received 24 October 2020; Revised 21 November 2020; Accepted 30 November 2020; Published 5 January 2021

Academic Editor: Wei Wang

Copyright © 2021 Xiangmin Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Facial expression recognition (FER) plays a significant part in artificial intelligence and computer vision. However, most of facial expression recognition methods have not obtained satisfactory results based on low-level features. The existed methods used in facial expression recognition encountered the major issues of linear inseparability, large computational burden, and data redundancy. To obtain satisfactory results, we propose an innovative deep learning (DL) model using the kernel entropy component analysis network (KECANet) and directed acyclic graph support vector machine (DAGSVM). We use the KECANet in the feature extraction stage. In the stage of output, binary hashing and blockwise histograms are adopted. We sent the final output features to the DAGSVM classifier for expression recognition. We test the performance of our proposed method on three databases of CK+, JAFFE, and CMU Multi-PIE. According to the experiment results, the proposed method can learn high-level features and provide more recognition information in the stage of training, obtaining a higher recognition rate.

## 1. Introduction

Facial expression recognition (FER) has great application potential in emotional computing, intelligent robot, intelligent monitoring, and clinical medicine [1, 2]. A FER system mainly consists of image acquisition, image preprocessing, image feature extraction, and classification [3], among which the image feature extraction directly exerts an impact on the training of classifiers and the performance of the whole recognition system. The effective feature extraction method can be used to improve recognition accuracy. In essence, expression feature extraction refers to the conversion of a high-dimensional facial expression image vector into a low-dimensional vector with a lot of distinguished information. Many algorithms used to extract image features are commonly applied to facial expression recognition [4–10].

Among the numerous feature extraction algorithms, principal component analysis (PCA) [11] is essentially a classic K-L transform and widely applied to extract features from statistics in pattern recognition. Its principle is to find

an optimal orthogonal transformation to minimize the mean squared error for transformed data. In general, PCA can represent the main information of the initial data by using a few principal components. However, PCA has weak robustness for some images with large illumination changes and complicated facial expression changes. In addition, PCA loses plenty of useful information because it cannot be used for nonlinear problem. As for the nonlinear problem case, the initial data can be mapped onto the higher-dimensional space through some nonlinear mapping. The linear dependence between the data in the high-dimensional space is utilized to fit the nonlinear dependence in the original data to the greatest extent, and then PCA can be applicable in this high-dimensional space. This is the kernel PCA or KPCA for short [12, 13]. But, it cannot learn high-level features and achieve good performance.

Recently, deep learning (DL) [14] has achieved a great success in terms of computer vision and pattern recognition. The DL has been gradually applied to feature extraction of FER. The convolutional Neural Network (CNN)

[15–18] is a very basic DL model and represents a sort of the artificial neural network. The structure of the CNN is similar to the connectivity model of neurons in the human brain, which makes the network model less complicated and minimizes weight amount. Such merit is more evident when the multidimensional image is inputted into the network. This model makes the image as an input into the network directly and reduces the complexity of feature extraction. However, the CNN has a complex hierarchical structure, slow convergence, and excessive dependence on the network's initial [19–23].

To fully utilize the structural features of the CNN and avoid its drawbacks, researchers integrate the feature extraction methods widely used in pattern recognition with the deep learning model, i.e, the deep subspace model. This model retains the deep learning idea, which can not only have powerful feature extraction ability but also reduce the computational burden. Liong et al. researched a deep learning model Deep PCA [24]. By using two-layer ZCA whitening (zero-phase component analysis whitening) and PCA networks, the new feature representation can be obtained by directly cascading the features learned at the first layer with those obtained at the second layer. The newly obtained features contain more discriminant information, which greatly improves the accuracy of image recognition. However, it shows sensitivity to a singular point, and the robustness of the algorithms for noise is poor. Chan et al. proposed PCANet structure [25], which cascades PCA, binary hash coding, and block histogram, to take the final output as the extracted deep feature. As an effective depth subspace model, the experimental results demonstrate that in terms of feature extraction, PCANet outperforms KPCA, PCA, and other methods and even AlexNet, ConvNet, and other CNN structures in some respects. Despite the demonstrated success, the model is restricted to the linear relation among the features and limits the use of features with nonlinear metrics.

Motivated by these successes, based on the current subspace learning and feature extraction algorithms, we propose a novel facial expression recognition algorithm using KECANet and DAGSVM. The convolutional kernel group of the expression sample set is obtained by KECA, which prevents CNN's excessive reliance on network initial value and requires a constant iterative solution. Then, the network output features are processed by binarization and block-histogram statistics to facilitate the extracted features to achieve better global discrimination. In addition, an effective classification method DAGSVM based on relative distance is applied to the classification.

The rest of this paper is organized as follows: In Section 2, an unsupervised DL model PCANet is introduced. Then, Section 3 describes the framework of KECANet and DAGSVM. Section 4 reports the performance of the method based on several experiments using facial expression images from multiple public datasets, including Extended Cohn-Kanade (CK+) database [26], Japanese Female Facial Expression (JAFFE) database [27], and CMU Multi-PIE face database [28]. Finally, main conclusions are given in Section 5.

## 2. Principal Component Analysis Network (PCANet)

PCANet is a simple unsupervised DL model based on the CNN for image recognition. In the PCANet [23], PCA replaces the convolution kernels of CNN to prevent learning filter kernels in each iteration like CNN. In addition, the network uses blockwise histograms to replace the pooling layer for down-sampling. Therefore, the PCANet model is simpler compared with some other convolutional network structure. Additionally, the parameter learning in PCANet requires no use of the backpropagation algorithm or pre-training convolution kernel through an automatic encoder network and deep confidence network. As a result, it saves plenty of computational complexity, and the training is highly effective.

Figure 1 shows the structure of the PCANet. It is comprised of two convolutional layers and one output layer. First, PCA filters are derived from a minimum reconstruction error for the input training sample. The filters convolve with the input image, and we input the convolved result into the second layer. Then, repeat the first layer in the second layer. Finally, hash coding and histogram block are applied to extract the ultimate features of images. The PCANet model relies on convolution to extract features, retaining spatial information of images. However, the method is restricted to the linear relation among the features, rather than the nonlinear relation when the image features are extracted. Therefore, the output image of PCANet will lose part of the original information. To resolve this problem, we propose the Kernel entropy component analysis network (KECANet) for image recognition in Section 3.

## 3. Overview of the Proposed Algorithm

For each facial expression image, we first preprocessed the image. Then, the features of expression images are extracted by using the KECANet algorithm. The Figure 2 shows the flowchart of facial expression recognition based on KECANet. Each face expression image has a uniform size after preprocessing. Next, in order to learn convolution kernels and obtain feature expression, the facial expression images were partitioned into training and testing sets.

For more efficient feature extraction, features are extracted by using the proposed method inspired by PCANet in the stage of feature extraction. A series of mapping features are produced in the first stage, serving as input to the second stage. Then, the output in the second stage is used as the input of the output stage. Next, binary hashing encoding and block histogram were used to compute the final features. Finally, we sent the final output features to the DAGSVM classifier for facial expression recognition. Figure 3 displays an illustration of KECANet.

*3.1. Kernel Entropy Component Analysis (KECA).* KECA is a data feature extraction method based on information entropy [29, 30]. The method was first proposed by Robert

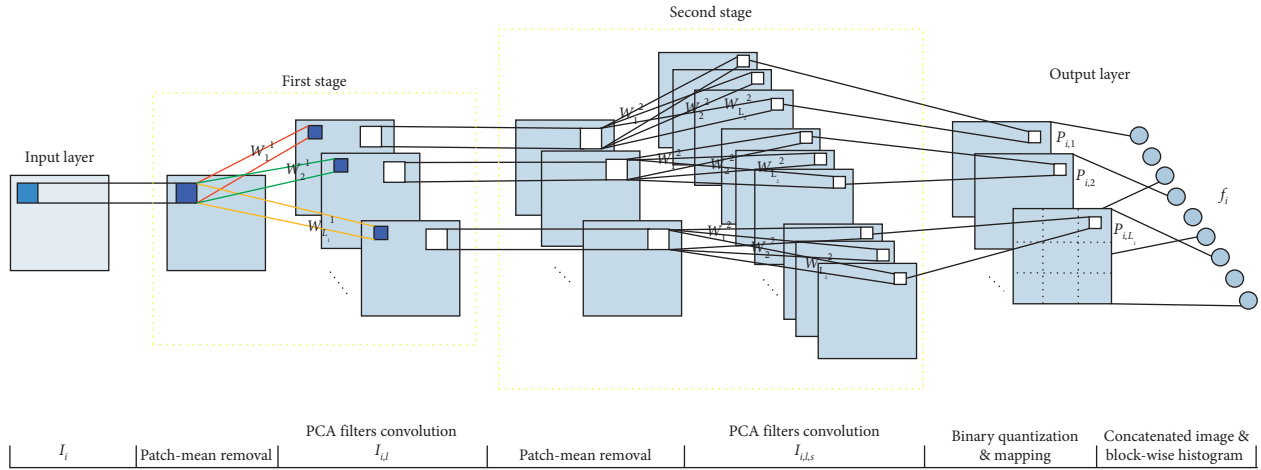


FIGURE 1: The structure of the PCANet model.

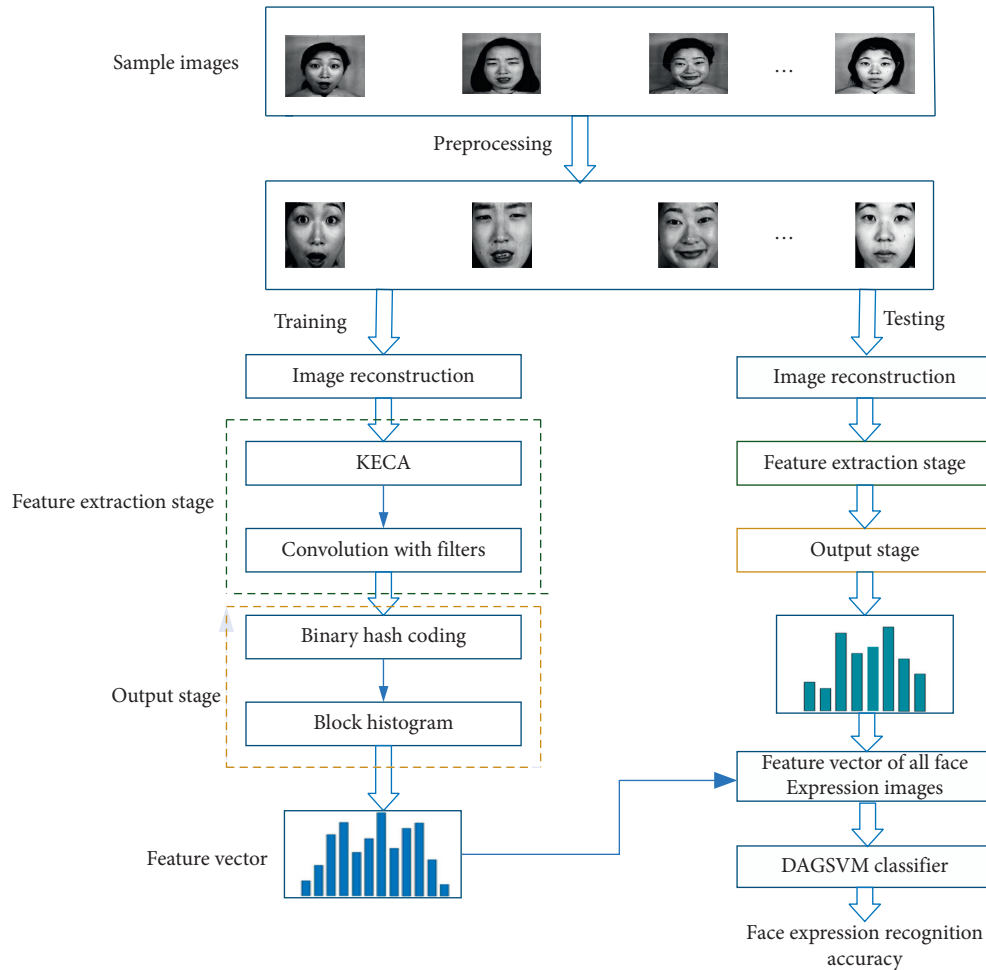


FIGURE 2: The flowchart of the FER based on KECANet.

Jensen in 2010. KECA works by projecting raw data into higher-dimensional space to eigen decomposing the kernel matrix. Then, the eigenvector with the maximum eigenvalue is selected to form a new data space. It is underpinned by Renyi entropy and Parzen window. KECA can resolve the problem of the linear inseparability of the other model and

enhances the separability between features. For the expression image training, sample  $X = [x_1, x_2, \dots, x_N]$ ,  $P(X)$  refers to the probability density function of  $X$ . The Renyi entropy can be written as  $H(p) = -\log \int p^2(x)dx$ . The parzen window can be written as  $\hat{p}(x) = 1/N \sum_{x_i \in D} K_\sigma(x, x_i)$ , where  $x$  denotes the sample

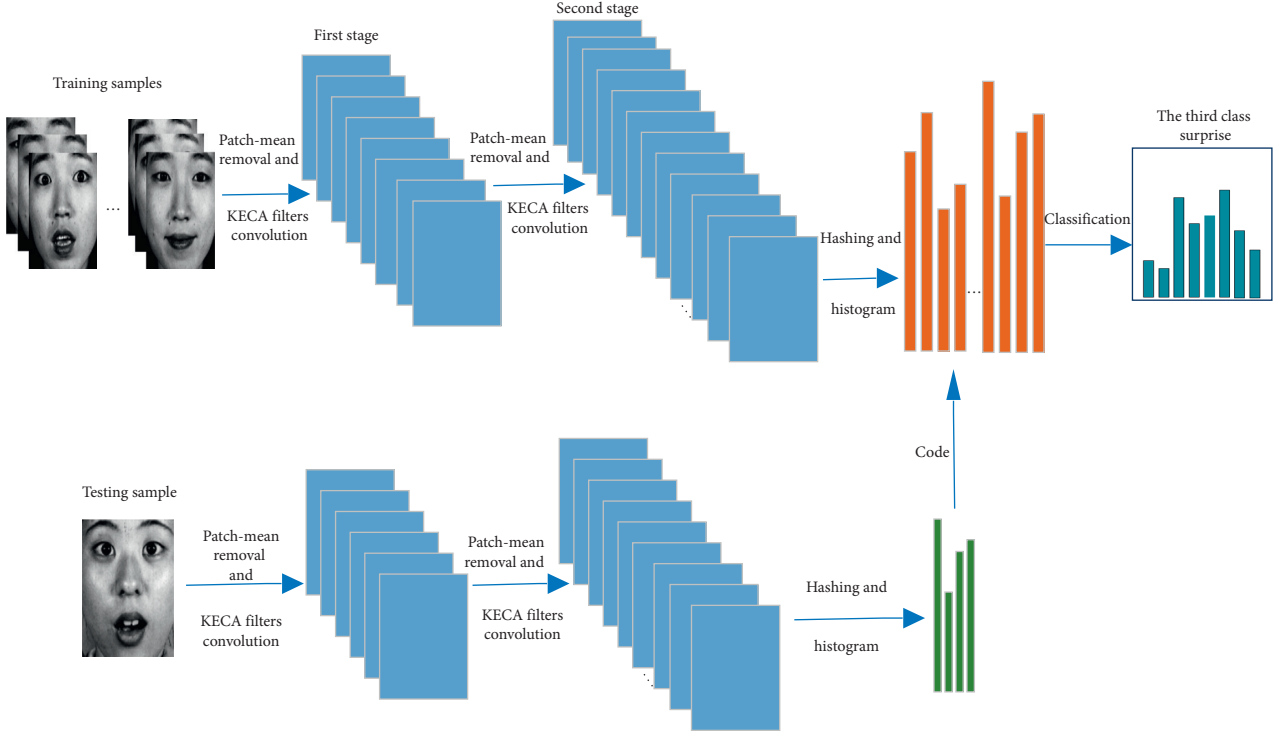


FIGURE 3: An illustration of the KECANet model.

$V(p) = \int p^2(x)dx$ . By approximating  $V(p)$  with the Parzen window, we have

$$\hat{V}(p) = \int p^2(x)dx = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N \int K_\sigma(x, x_i) K_\sigma(x, x_j) dx = \frac{1}{N^2} IKI^T, \quad (1)$$

where  $I$  denotes the  $N \times 1$  vector (every element is 1) and  $K$  denotes the  $N \times N$  kernel matrix,  $K(i, j) = K_\sigma(x_i, x_j)$ . We map the high-dimensional feature spaces  $\varphi$  to subspaces  $U_d$ . Then, the reordering of eigenvalues and eigenvectors is made according to the entropy. The principal component matrix  $\varphi_{eca}$  can be obtained:

$$\varphi_{eca} = U_d^T \varphi = D_d^{-1/2} E_d^T, \quad (2)$$

where  $X$  stands for a diagonal matrix composed of the first  $d$  eigenvalues of the matrix  $D$  and  $E$  represents a matrix with eigenvectors. The solution of the above equation is transformed into the solution of the minimum value problem, and we have

$$\varphi_{eca} = D_d^{-1/2} E_d^T: \min_{(\lambda_1, \alpha_1), \dots, (\lambda_d, \alpha_d)} \{\hat{V}(p) - \hat{V}_d(p)\}, \quad (3)$$

where  $\hat{V}_d(p) = 1/N^2 \sum_{i=1}^d (\sqrt{\lambda_i} \alpha_i^T I)^2$ . For a new sample  $x_{new}$ , the projection on the feature space can be expressed as follows:

$$W = U_d^T \varphi(x_{new}) = D_d^{-1/2} E_d^T \varphi(x_{new}) = D_d^{-1/2} E_d^T K(x, x_{new}). \quad (4)$$

### 3.2. Features Extraction with the Proposed KECANet

**3.2.1. Image Reconstruction.** Assume that we have  $N_1$  input training images of  $I_i$  ( $i = 1, 2, 3, \dots, N_1$ ). The size is  $m \times n$ . According to Figure 4, patches of size  $k_1 \times k_2$  are collected, where  $0 < k_1 \leq m$  and  $0 < k_2 \leq n$ . We collect all overlapping patches and vectorize them, i.e.,  $X_{i,1}, X_{i,2}, \dots, X_{i,\tilde{m}\tilde{n}} \in R^{k_1 k_2}$ , where  $\tilde{m} = m - k_1 + 1$  and  $\tilde{n} = n - k_2 + 1$ . Next, the patch mean removal is done on all overlapping patches and  $\bar{X}_i = [X_{i,1}, X_{i,2}, \dots, X_{i,\tilde{m}\tilde{n}}] \in R^{k_1 k_2}$ ,  $i = 1, 2, \dots, N_1$  is obtained. In this paper, we define  $\bar{X}_i = [\bar{X}_{i,1}, \bar{X}_{i,2}, \dots, \bar{X}_{i,\tilde{m}\tilde{n}}] \in R^{k_1 k_2}$ ,  $i = 1, 2, \dots, N_1$  as image reconstruction set (IRS).

Then, we do IRS on all training samples and obtain

$$X = [\bar{X}_1, \bar{X}_2, \dots, \bar{X}_{N_1}] \in R^{k_1 k_2 \times N_1 \tilde{m}\tilde{n}}. \quad (5)$$

**3.2.2. Image Convolution.** The key to feature extraction is to study the mapping matrix. To learn  $k_1 \times k_2$  convolution kernel, the KECANet is purposed to identify the eigenvectors and eigenvalues of the covariance matrix. Then, the optimal projection axis of the sample matrix is extracted by KECA as the convolution filter of the feature extraction stage  $W_l^1$ ,  $l = 1, 2, \dots, L_1$ .  $L_1$  denotes the number of principal component eigenvectors of the first feature extraction stage.  $W_l^1$  is used as the filters of the first layer. Then, the boundary  $I_i$  is zero-padded and convolved with  $W_l^1$ . Thus, we get the output of the first stage of feature extraction:



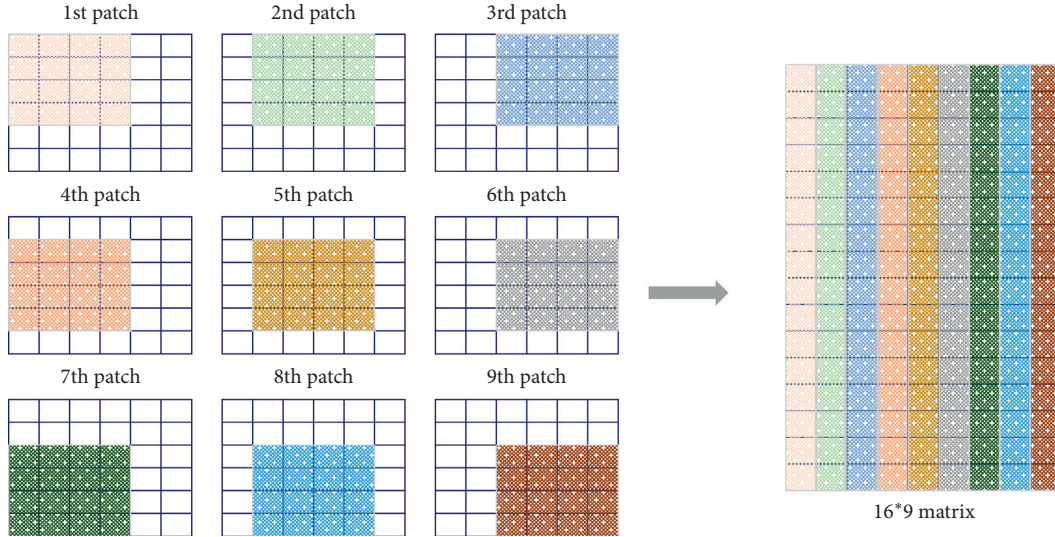


FIGURE 4: Illustration of  $4 \times 4$  patch size used for  $6 \times 6$  images.

$$O_{i,l} = I_i * W_l^1, \quad i = 1, 2, \dots, N_1; l = 1, 2, \dots, L_1. \quad (6)$$

The identical mapping process conducted in the first stage is repeated in the second stage. The output  $O_{i,l}$  in the first stage serves as the input in the second layer. Then, KECA is used to get  $L_2$  convolution kernel  $W_s^2$  of the second layer. So, the output of the second layer can be obtained after image convolution operation. The output can be expressed as

$$O_{i,l,s} = O_{i,l} * W_s^2, \quad i = 1, 2, \dots, N; l = 1, 2, \dots, L_1; s = 1, 2, \dots, L_2. \quad (7)$$

From the above,  $L_1$  matrices are produced from the first layer of KECA. For every output matrix of the first layer, the second layer will produce  $L_2$  corresponding feature matrix. When the  $i$ th image enters the KECA,  $L_1 L_2$  output matrix can be obtained.

**3.2.3. Output Stage.** In the output stage, binary hashing encoding and block histogram are adopted, which can reduce the number of the feature matrix and produce more desirable classification result. The outputs  $O_{i,l,s}$  are binarized to obtain the output  $H(O_{i,l} * W_s^2)$ . The binary vector is converted into a decimal number, and a single integer-valued image  $T_i^l$  is obtained:

$$T_i^l = \sum_{s=1}^{L_2} 2^{l-1} H(O_i^l * w_s^2). \quad (8)$$

Each of the images  $T_i^l$  computed by binary hashing is partitioned into  $B$  ( $0 < B < mn$ ) blocks. Then, the histogram of each block's decimal values is computed. Next, all histograms are concatenated into the vector  $\text{Bhist}(T_i^l)$ ,  $i = 1, 2, \dots, N_1; j = 1, 2, \dots, L_1$ . The feature vector  $f_i$  of the input expression image is obtained through the encoding process:

$$f_i = [\text{Bhist}(T_i^1), \dots, \text{Bhist}(T_i^{L_1})]^T. \quad (9)$$

**3.3. DAGSVM.** Based on the proposed KECA model, we present an effective classifier DAGSVM. The DAGSVM is based on SVM to solve multiclass problem. Assuming that there are  $K$  facial expressions, the method constructs a direct acyclic graph containing  $K(K-1)/2$  nodes and  $K$  leaves. Figure 5 shows a 4-layer DAGSVM classifier. For the  $i$ th and  $j$ th facial expression image, each node denotes a binary SVM classifier. Each leaf denotes a facial expression decision. The root node denotes the easiest classifier between two quite different expressions, whereas the node of the fourth stage denotes the most difficult classifier between two similar expressions.

The given test sample  $x$  is first put into the classifier of the root node to perform classification. The algorithm moves left or right for next computing based on the output value after the decision of the root classifier. The above steps are repeated until ending at the leaf node indicating the predicted facial expression.

Because the numbers of different facial expression images are always unbalanced, we need to adjust the penalty parameters of different binary classifiers. A larger penalty parameter is set for a smaller number of subjects, and a smaller penalty parameter is set for a larger number of subjects.  $n_k$  is the number of subjects, and  $c_k$  is a penalty parameter of the  $k$ th facial expression image ( $k = 1, 2, \dots, K$ ). The penalty parameter is set as

$$c_k = \frac{\alpha}{n_k}, \quad (10)$$

where  $\alpha$  denotes the penalty coefficient common for all the facial expression images.

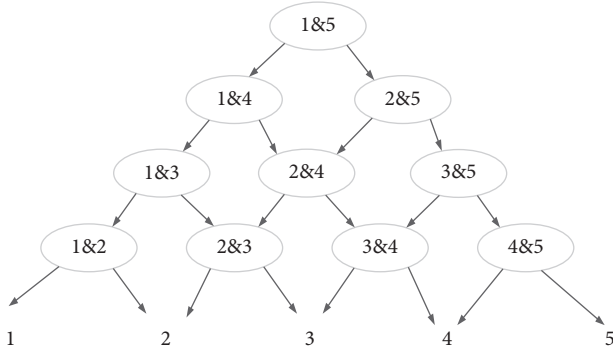


FIGURE 5: The architecture of 4-layer DAGSVM.

## 4. Experiments and Results

In this section, we performed experiments on facial expression databases CK+, JAFFE, and CMU Multi-PIE to verify the performance of our proposed method. We split input the facial expression images into the size of  $64 \times 64$  according to the two-eye positions. Then, we subjected the split expression images to down-sampling to  $48 \times 48$  pixels.

**4.1. Databases.** In CK+ database, there are 593 facial expression image sequences from 123 subjects, including men and women aged 18–30 years from different countries. The dataset contains seven facial expressions including anger (135 images), sadness (28 images), surprise (246 images), disgust (177 images), fear (75 images), happiness (207 images), and neutral (314 images). Figure 6 shows a proportion of exemplary images from the CK+ database.

The JAFFE database consists of 213 facial expression images, including sadness (31 images), fear (32 images), surprise (30 images), anger (30 images), happy (31 images), disgust (29 images), and neutral (30 images). Figure 7 presents some examples from JAFFE database.

In addition to abovementioned facial expression databases, we also evaluate the algorithm proposed in this paper on CMU Multi-PIE database. The dataset involves 337 subjects, obtained under 15 viewing angles and 19 lighting conditions for totally over 750,000 images. We selected 1,251 emoticons from 71 people for the experiment in this paper, containing six different expressions, including disgust, screaming, smile, squint, surprise, and neutral. Figure 8 presents a proportion of expression image samples from the CMU Multi-PIE database.

**4.2. Comparing the Average Accuracy of Different FER Methods.** We compared the performance of KECANet with that of LDA, PCA, KPCA, as well as the LDANet and PCANet methods. The average recognition rates of different methods on three datasets are displayed in Table 1.

As shown in Table 1, for the CK+ and JAFFE database, the average accuracy rates produced by our proposed method KECANet are above 90%. The KECANet method produces the accuracy rate of 92.67% especially on the CK+ dataset, whereas the average accuracies produced by other methods are below

92%. Likewise, our proposed method can reach an accuracy of 91.19% on CMU Multi-PIE database, whereas the accuracy rates are 90% or below by other methods. By comparing the above data, it is concluded that the performance of the method KECANet is superior to that of the other FER methods.

Then, the proposed algorithm is tested on three datasets with the cross-validation method of Leaving one subject out (LOSO). We selected one subject out once as testing set and applied other subjects' expression images for training set. In this verification method, there is no intersection between training and testing sets, which is close to the actual situation. Figures 9–11 present the confusion matrices on three datasets in the LOSO scenario.

As revealed by the experimental data in Figures 9 and 10, compared with other expressions, anger and sadness expressions are more likely to misclassified as neutral expression. The major causes include two aspects. (1) The expressions of anger and sadness have no exaggerated expression features. They are particularly close to neutral expressions, which make them easy to confuse. (2) The number of expressions of anger and sadness is smaller relative to other expressions in the CK+ dataset, for which they are unable to produce positive training results. Figure 11 shows that disgust and smile expressions are more likely to misclassified as neutral expression compared with other expressions.

As revealed in Figures 9 and 10, the recognition results of the proposed algorithm on CK+ and JAFFE are different, and the recognition rate on the CK+ dataset is greater than that on the JAFFE dataset. The leading cause is that the number of expression images of CK+ and JAFFE dataset is different, and the number of samples on the CK+ dataset is approximately twice as large as the JAFFE dataset.

**4.3. Comparison of the Recognition Rates of Each Expression.** For further validation of the effectiveness of our proposed method, Figures 12–14 display the recognition results of each expression using different FER methods for three databases, and the performance of the proposed method precedes other FER methods in most cases.

**4.4. Effects of Filter.** In this subsection, we investigate the influence of the number of filters  $L_1$  for KECANet on CK+, JAFFE, and CMU Multi-PIE datasets. We fix  $L_2 = 8$  and change the filter numbers  $L_1$  of the first stage from 2 to 14. Figures 15–17 show the recognition results using PCANet, LDANet, and our proposed method KECANet with some filters in the first stage.

As shown in Figures 15–17, when the number of convolution in the first layer is comparatively low, PCANet and LDANet show a marginally greater recognition rate compared with KECANet. Nevertheless, with the growing number of convolution kernels in the first layer, the recognition rate of KECANet gradually exceeds PCANet and LDANet, thus showing an overall increasing trend.

The experimental results indicate that the number of filter  $L_1$  varies with each algorithm with the highest recognition rate. Therefore, we set  $L_1$  in each algorithm to the number corresponding to its highest recognition rate. The



FIGURE 6: Sample facial images from the CK+ dataset.



FIGURE 7: Sample facial images from the JAFFE dataset.

TABLE 1: Experiment results of different methods.

Methods	CK+	JAFFE	CMU Multi-PIE
LDA	73.23	69.17	70.15
PCA	72.21	70.02	72.19
KPCA	73.56	71.21	79.11
PCANet	90.82	87.32	90.72
LDANet	91.29	87.21	90.81
KECANet	92.67	88.21	91.19



FIGURE 8: Partial samples of CMU Multi-PIE database.

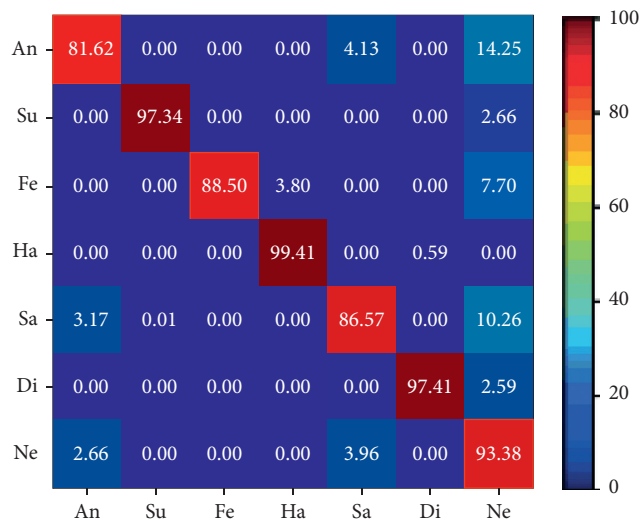


FIGURE 9: The confusion matrices on CK+ dataset.

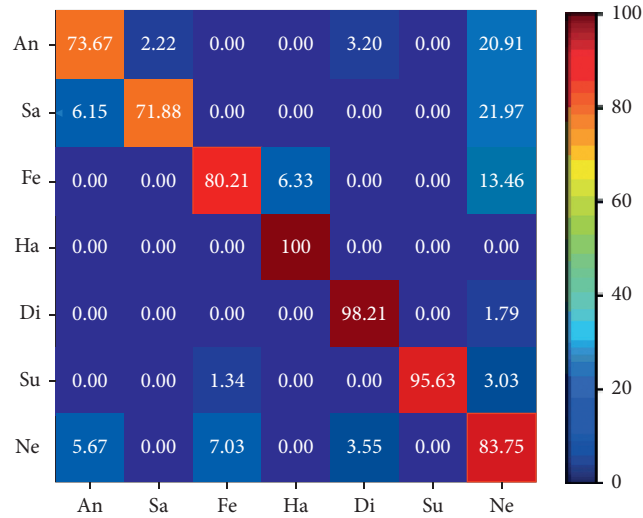


FIGURE 10: The confusion matrices on JAFFE dataset.

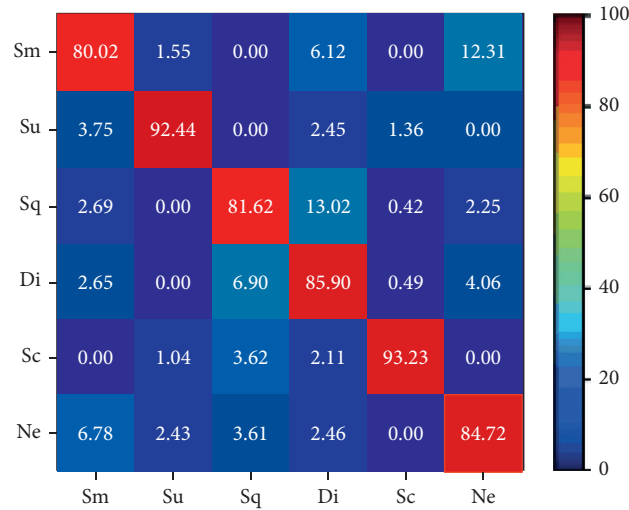


FIGURE 11: The confusion matrices on CMU Multi-PIE dataset.

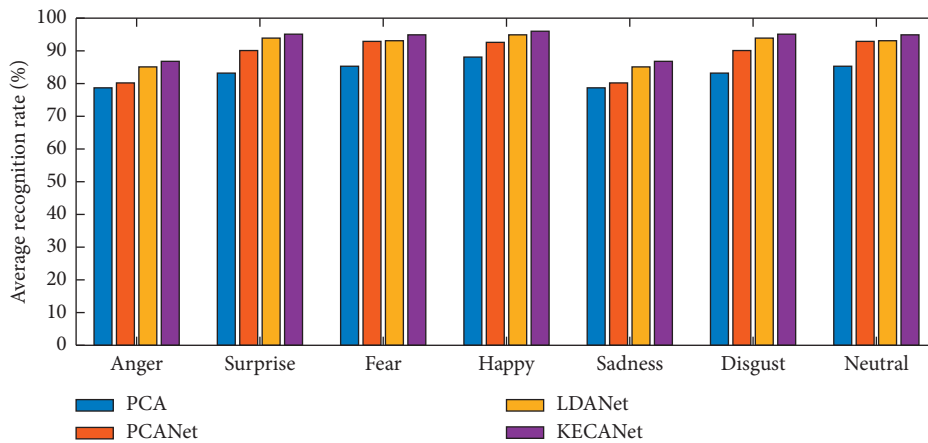


FIGURE 12: Comparing the test accuracies on the CK+ dataset.



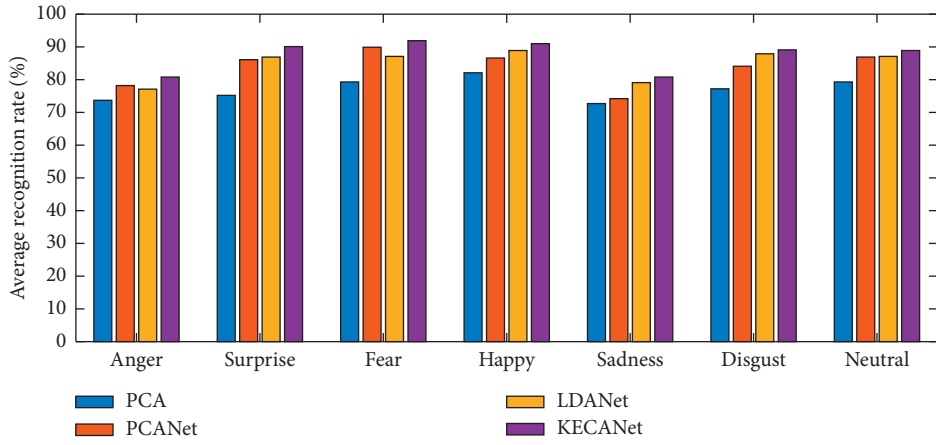


FIGURE 13: Comparing the test accuracies on the JAFFE dataset.

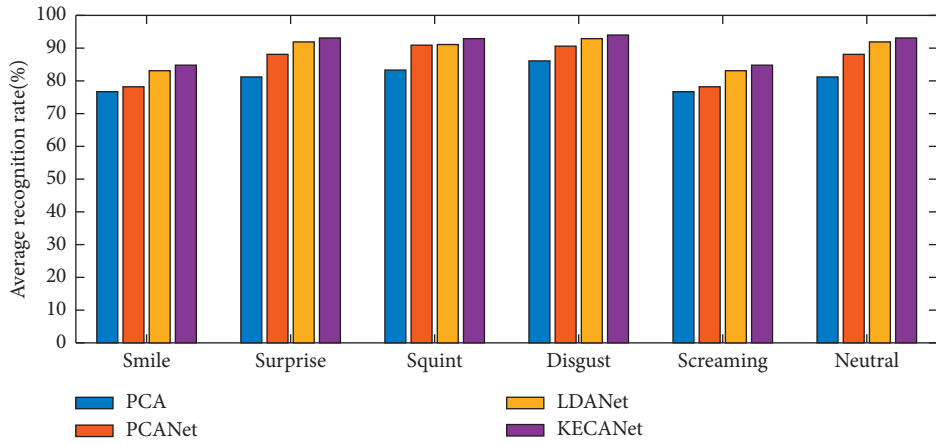


FIGURE 14: Comparing the test accuracies on the CMU Multi-PIE dataset.

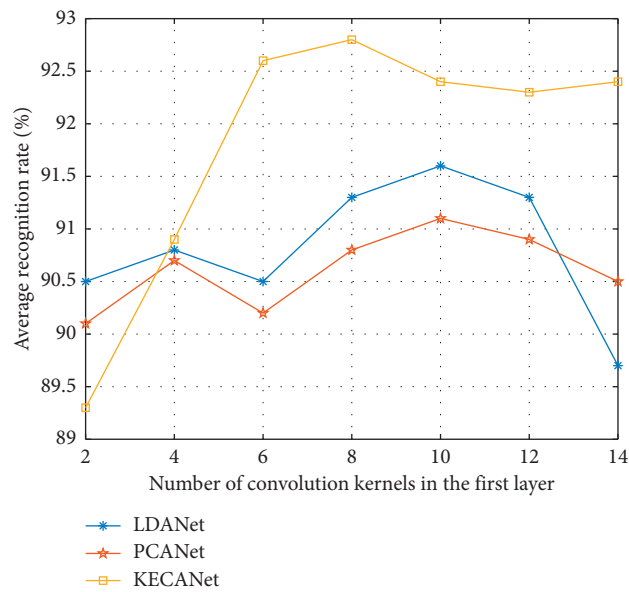


FIGURE 15: The average recognition rate corresponding to different values of  $L_1$  using CK+ dataset.

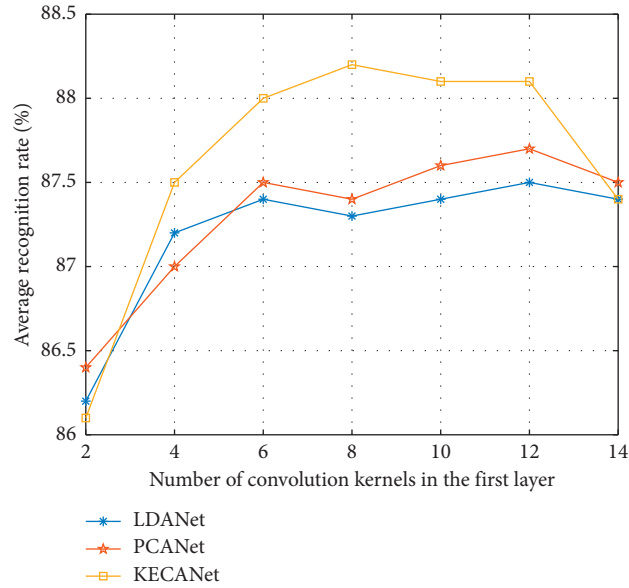


FIGURE 16: The average recognition rate corresponding to different values of  $L_1$  using JAFFE dataset.

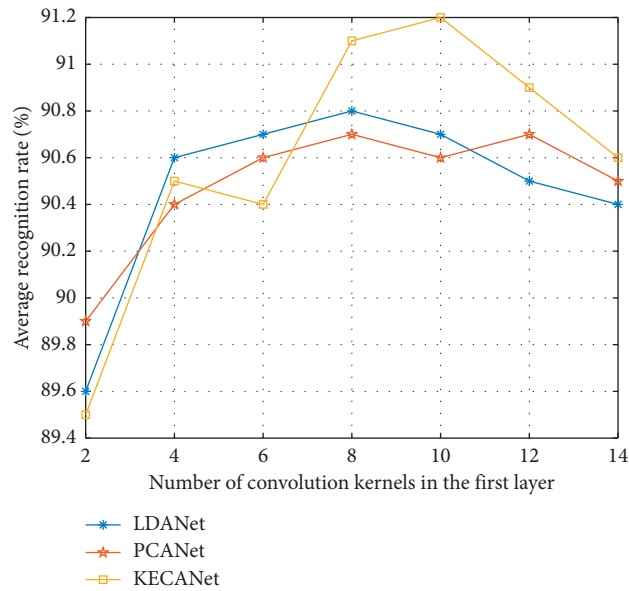


FIGURE 17: The average recognition rate corresponding to different values of  $L_1$  using CMU Multi-PIE dataset.

allocation of the training sets and testing sets is identical to the prior experiment. In addition, to validate the robustness of each algorithm, we repeated each experiment 8 times. Tables 2–4 show the experimental results where  $L_1$  represents the filter numbers in the first stage and mean denotes the average recognition rate of 8 groups of experiments.

TABLE 2: The recognition rate of CK+ database.

Algorithm	$L_1$	Mean
PCANet	10	91.19
LDANet	10	91.51
KECCANet	8	92.57

TABLE 3: The recognition rate of JAFFE database.

Algorithm	$L_1$	Mean
PCANet	12	87.51
LDANet	12	87.47
KECANet	8	88.35

TABLE 4: The recognition rate of CMU Multi-PIE database.

Algorithm	$L_1$	Mean
PCANet	8	90.67
LDANet	8	90.81
KECANet	10	91.22

Tables 2–4 show that the average recognition rate of the KECCANet algorithm is noticeably greater than those of PCANet and LDANet.

## 5. Conclusions

In this paper, we propose a novel facial expression recognition method using the Kernel entropy component analysis network (KECCANet) and DAGSVM. First, the deep features of training samples and testing samples are acquired based on the KECCANet model. Secondly, the final features of images are extracted by the binary hashing coding and histogram block. Then, the features extracted from the KECCANet model are classified by using the DAGSVM classifier. Finally, the experiments are carried out on facial expression databases CK+, JAFFE, and CMU Multi-PIE, and the experimental results verify the superiority of our proposed method.

It is seen that our proposed method consolidates the idea of deep learning. It resolves the problem of the linear inseparability of the PCANet model and enhances the separability between features. Our future work will focus on how to reduce the runtime of the deep learning algorithm, integrate the high-level expression features, and apply the proposed method to other recognition problems.

## 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.

## Authors' Contributions

X. C. wrote the preliminary draft of this paper, studied the code, and carried out the simulations of this proposed algorithm. L. K. and Q. D. helped to check the coding and the simulation results. J. L. and X. D. provided some analysis DAGSVM algorithm. All the authors wrote this paper together, and they have read and approved the final manuscript.

## Acknowledgments

This work was supported by the National Nature Science Foundation of China under grant no. 51377109, the Nature Science Foundation of Liaoning Province of China under grant no. 2019-ZD-0204, and the Major Project of Liaoning Province of China under grant no. 20201362106.

## References

- [1] S. Bagherzadeh, K. Maghooli, J. Farhadi, and M. Zangeneh Soroush, "Emotion recognition from physiological signals using parallel stacked autoencoders," *Neurophysiology*, vol. 50, pp. 428–435, 2018.
- [2] R. Vishnu Priya, "Emotion recognition from geometric fuzzy membership functions," *Multimedia Tools and Applications*, vol. 78, pp. 17847–17878, 2019.
- [3] W. Zheng, Y. Zong, X. Zhou, and M. Xin, "Cross-domain color facial expression recognition using transductive transfer subspace learning," *IEEE Transactions on Affective Computing*, vol. 9, no. 1, pp. 21–37, 2018.
- [4] M. H. Siddiqi, "Accurate and robust facial expression recognition system using real-time YouTube-based datasets," *Applied Intelligence*, vol. 48, pp. 2912–2929, 2018.
- [5] C. Xu, Y. Cui, Y. Zhang, P. Gao, and J. Xu, "Person-independent facial expression recognition method based on improved Wasserstein generative adversarial networks in combination with identity aware," *Multimedia System*, vol. 26, pp. 53–61, 2020.
- [6] H. Li, J. Sun, Z. Xu, and L. Chen, "Multimodal 2D + 3D facial expression recognition with deep fusion convolutional neural network," *IEEE Transactions on Multimedia*, vol. 19, no. 12, pp. 2816–2831, 2017.
- [7] M. Wu, W. Su, L. Chen, Z. Liu, W. Cao, and K. Hirota, "Weight-adapted convolution neural network for facial expression recognition in human-robot interaction," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 99, pp. 1–12, 2019.
- [8] J. Chen, R. Xu, and L. Liu, "Deep peak-neutral difference feature for facial expression recognition," *Multimedia Tools and Applications*, vol. 77, no. 22, pp. 29871–29887, 2018.
- [9] C. L. P. Chen and Z. Liu, "Broad learning system: an effective and efficient incremental learning system without the need for deep architecture," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 1, pp. 10–24, 2018.
- [10] K. Gabriels, "Response to "uncertainty in emotion recognition"" *Journal of Information, Communication and Ethics in Society*, vol. 17, no. 3, pp. 295–298, 2019.
- [11] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.
- [12] P. K. Sathish and S. Balaji, "A complete person re-identification model using kernel-PCA-based Gabor-filtered hybrid descriptors," *International Journal of Multimedia Information Retrieval*, vol. 7, pp. 221–229, 2018.
- [13] Y. Yao, J. Bao, and M. Skyllas-Kazacos, "Fault detection and diagnosis in Hall-Héroult cells based on individual anode current measurements using dynamic kernel PCA," *Metallurgical and Materials Transactions B*, vol. 49, pp. 1–12, 2018.
- [14] G. Tanisik, C. Zalluhoglu, and N. Ikizler-Cinbis, "Facial descriptors for human interaction recognition in still images," *Pattern Recognition Letters*, vol. 73, pp. 44–51, 2016.

- [15] Q. Zhou, W. Yang, G. Gao et al., "Multi-scale deep context convolutional neural networks for semantic segmentation," *World Wide Web*, vol. 22, no. 2, pp. 555–570, 2018.
- [16] K. Li, Y. Jin, M. W. Akram, R. Han, and J. Chen, "Facial expression recognition with convolutional neural networks via a new face cropping and rotation strategy," *The Visual Computer*, vol. 36, pp. 391–404, 2019.
- [17] H. Ma and T. Celik, "FER-Net: facial expression recognition using densely connected convolutional network," *Electronics Letters*, vol. 55, no. 4, pp. 184–186, 2019.
- [18] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 10, pp. 1533–1545, 2014.
- [19] S. I. Venieris and B. Christos-Savvas, "fpgaConvNet: mapping regular and irregular convolutional neural networks on FPGAs," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 2, pp. 326–342, 2019.
- [20] D. Peralta, I. Triguero, S. García, Y. Saeys, J. M. Benitez, and F. Herrera, "On the use of convolutional neural networks for robust classification of multiple fingerprint captures," *International Journal of Intelligent Systems*, vol. 33, no. 1, pp. 213–230, 2018.
- [21] S. Basu, M. Karki, S. Ganguly et al., "Learning sparse feature representations using probabilistic quadrees and deep belief nets," *Neural Processing Letters*, vol. 45, no. 3, pp. 855–867, 2017.
- [22] C.-Y. Zhang, C. L. P. Chen, D. Chen, and K. T. Ng, "Map-Reduce based distributed learning algorithm for Restricted Boltzmann Machine," *Neurocomputing*, vol. 198, pp. 4–11, 2016.
- [23] L. Fu, Y. Feng, T. Elkamil, Z. Liu, R. Li, and Y. Cui, "Image recognition method of multi-cluster kiwifruit in field based on convolutional neural networks," *Transactions of the Chinese Society of Agricultural Engineering*, vol. 34, no. 2, pp. 205–211, 2018.
- [24] V. E. Liong, J. Lu, and G. Wang, "Face recognition using deep pca," in *Proceedings of the 2013 9th International Conference on Information Communications and Signal Processing*, pp. 1–5, Tainan, Taiwan, December 2013.
- [25] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet: a simple deep learning baseline for image classification?" *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5017–5032, 2015.
- [26] X. Huang, G. Zhao, W. Zheng, and M. Pietikäinen, "Towards a dynamic expression recognition system under facial occlusion," *Pattern Recognition Letters*, vol. 33, no. 16, pp. 2181–2191, 2012.
- [27] F. Cheng, J. Yu, and H. Xiong, "Facial expression recognition in JAFFE dataset based on Gaussian process classification," *IEEE Transactions on Neural Networks*, vol. 21, no. 10, pp. 1685–1690, 2010.
- [28] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-PIE," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.
- [29] B. Scholkopf, J. Platt, and T. Hofmann, "Kernel maximum entropy data transformation and an enhanced spectral clustering algorithm," in *Conference of Advances in Neural Information Processing Systems*, pp. 633–640, MIT Press, Cambridge, MA, USA, 2006.
- [30] Y. S. Qi, H. L. Zhang, X. J. Gao, and P. Wang, "Novel fault monitoring strategy for chemical process based on KECA," *CIESC Journal*, vol. 67, pp. 1063–1069, 2016.