

RESEARCH

Open Access



Facial expression recognition using optimized active regions

Ai Sun¹, Yingjian Li^{2*} , Yueh-Min Huang¹, Qiong Li³ and Guangming Lu²

*Correspondence:

hit_lyj@126.com

² School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, China
Full list of author information is available at the end of the article

Abstract

In this paper, we report an effective facial expression recognition system for classifying six or seven basic expressions accurately. Instead of using the whole face region, we define three kinds of active regions, i.e., left eye regions, right eye regions and mouth regions. We propose a method to search optimized active regions from the three kinds of active regions. A Convolutional Neural Network (CNN) is trained for each kind of optimized active regions to extract features and classify expressions. In order to get representable features, histogram equalization, rotation correction and spatial normalization are carried out on the expression images. A decision-level fusion method is applied, by which the final result of expression recognition is obtained via majority voting of the three CNNs' results. Experiments on both independent databases and fused database are carried out to evaluate the performance of the proposed system. Our novel method achieves higher accuracy compared to previous literature, with the added benefit of low latency for inference.

Keywords: Facial expression recognition, Optimized active regions, Convolutional Neural Network, Decision-level fusion

Introduction

Facial expression, which is a fundamental mode of transporting human's emotions, plays a significant role in our daily communication. Facial expression recognition is a complex and interesting problem, and finds its applications in driver safety, health-care, human-computer interaction etc. Due to its wide range of applications, facial expression recognition has received substantial attention among the researchers in the area of computer vision [1–3]. Although a number of novel methodologies have been proposed in recent years, recognizing facial expression with high accuracy and speed remains challenging due to the complexity and variability of facial expressions.

For facial expression recognition problems, the general recognition method appeared in previous work can be divided into two major steps, face representation and classifier construction. In the first step, features related to facial expression are extracted from images. Some of the features are hand-designed [4–6], whereas others are learnt from training images [7–9]. Then, the dimensionality of the features is reduced to facilitate an efficient classification and enhance the generalization capability. The universal expressions which are mentioned in the papers are usually anger, disgust, fear, joy (or happiness), sadness and surprise [1, 10], whereas some researchers add neutral as the seventh

expression [2, 3, 11, 12]. In the second step, classifiers are designed based on the simplified features to classify each expression as one of the six (or seven) expressions.

The success of facial expression recognition systems heavily rely on effective representation of the pixels stored within an image of a human face. Ekman et al. [13] proposed the Facial Action Coding System (FACS) to represent facial activity. They measure all observable facial motions in terms of Action Units (AUs). Each kind of expression can be decomposed into a group of AUs. In the behavioral sciences, this coding technique has become the leading method for the classification of facial expressions [1]. Accurate detection of AUs is necessary when using FACS. However, it is difficult to detect all the AUs. Therefore, some researchers decided to represent facial expressions by geometric-based methods ([14–19]) or appearance-based methods [20, 21]. In geometric-based methods, the location and shape of facial components, such as eye, eyebrows, mouth corner and so on, are extracted to form a feature vector that represents the face geometry. Although geometric-based methods can achieve similar performance as appearance-based method, they usually require more accurate and reliable facial components detection and tracking, which is difficult in many situations. Appearance-based methods apply appearance features to represent facial expression. Zhang et al. [22] demonstrated that Gabor-wavelet appearance features were more effective than geometric features. Nevertheless, this kind of feature is computationally expensive to extract. Shan et al. [21] introduced Local Binary Patterns (LBP) as low-cost discriminative features for facial expression recognition. Since then, LBP has been widely used to recognize facial expression due to its good performance [23, 24]. Generally speaking, the features extracted are high-dimensional and not efficient for expression classification. In order to reduce the dimensionality of the feature vector, some dimensionality reduction techniques are proposed. Principal Component Analysis (PCA) [25] and Linear Discriminant Analysis (LDA) [26] are two usually used methods for dimensionality reduction. The features mentioned above are all hand-designed. However, Convolutional Neural Network (CNN), a kind of deep learning method, can learn features from training data. CNN learns features through a mixture of convolutional layers and sub-sampling layers, and usually followed by a set of fully connected layers. CNN is getting increasingly popular in recent years because of its efficient performance [27, 28].

Classifier construction is the other significant step for facial expression recognition. Many methods were used to classify each expression as one of the six or seven kinds of expressions. Support Vector Machines (SVMs) are very popular for classification when the amount of data is small. Lee et al. [29] use an SVM to classify the facial expression images in CK+ database [30, 31] and JAFFE¹ database [32] with performance accuracies of 94.3% and 92.22%, respectively. A decision tree approach was applied to classify expressions in CK+ database by Salmam et al. [33] and the accuracy was 90%. In CNN, the fully connected layer is a classifier actually.

Some researchers divided the face area into several blocks and features are extracted from them. Better accuracy is obtained using these features [20, 21, 34]. In psychology, it is indicated that facial features of expressions are located around eyes, nose and mouth,

¹ CK+ and JAFFE are two widely used facial expression database. The details of them can be found in “[Experiments and discussion](#)” section.

and the locations of them are necessary for categorizing facial expression [35]. These works motivate us to recognize facial expression using several principal regions of the face.

In this paper, three kinds of square areas are explored, i.e. left eye regions, right eye regions and mouth regions. The centers of the three kinds of square areas are overlapped with the centers of left eye, right eye and mouth, respectively. All of these regions are called active regions in this paper. The size of active region affects the accuracy of facial expression recognition. We propose a method to search the proper size of active regions for facial expression classification. The active regions with the proper size are called optimized active regions. A decision-level fusion framework is designed for facial expression classification. For each of the three optimized active regions, a CNN is trained. The final classes of facial expressions are obtain by majority voting of the results of three CNNs. Experiments are carried out on CK+ database, JAFFE database and NVIE database [36–38]. The results indicate that the proposed method performs better than previous works in terms of accuracy. The main contributions of the paper are as follows.

1. Instead of using the whole face region, three kinds of active regions are applied to classify facial expressions. A method to search optimized active regions is proposed according to similarity of active regions.
2. We proposed a decision-level fusion framework, which is helpful to increase the accuracy of facial expression recognition.

The remainder of the paper is organized as follows. “[Related work](#)” section presents a review of previous works. In “[Facial expression recognition system](#)” section, the framework of facial expression recognition system is introduced. The method to search optimized active regions and the decision-level fusion method are proposed in “[Optimized active regions searching](#)” and “[Classification based on decision-level fusion](#)” sections, respectively. The experiments and result discussions are carried out in “[Experiments and discussion](#)” section. “[Conclusion](#)” section concludes the paper.

Related work

In order to represent the face accurately, detection of faces in images is necessary. Due to the importance of face detection, lots of researches have been done in recent years [39–41]. Dang et al. [39] discussed various face detection algorithms, such as Viola–Jones, SMQT features and SNOW classifier, neural network-based face detection and support vector machine-based face detection. These face detection methods were compared based on the precision and recall rate calculated using DetEval, a software for the evaluation of object detection algorithms. The experiment result showed that the best one among all of these algorithms is Viola–Jones in terms of precision and recall rate. Besides the algorithms mentioned above, there are other methods to detect face. Some real-world factors, such as viewpoint, extreme illuminations and expression changes, lead to the large intra-class variations and make the detection algorithm not robust enough. Tao et al. [40] proposed a locality-sensitive SVM using kernel combination algorithm to improve the robustness of face detection system. Paul et al. [41] proposed a new kind of feature called haar-like feature for face detection. The AdaBoost learning

algorithm was used for feature selection and an efficient classifier was built using cascade structure. This method was integrated into Open Source Computer Vision Library (OpenCV) [42]. OpenCV was designed for computational efficiency on real-time applications, with the added benefit of a easy to use interface. For these reasons, we choose OpenCV to detect face in images in this paper.

Face alignment is essential for better performance in facial expression recognition. The goal of face alignment is to localize a set of predefined facial landmarks (eye corners, eyebrow corners etc.) of the face in an image. Various methods have been proposed to detect facial landmarks, most of which were based on shape indexed features [43, 44]. Some companies, such as Microsoft, Face++ and so on, have implemented the state-of-the-art methods to detect facial landmarks. These companies provide facial landmarks detection service for researchers as Application Programming Interface (API), which is very convenient to invoke.

Most of the studies on facial expression recognition used the whole face area, whereas a few of them divided the face into several blocks and extract features from these blocks. In [21], the face area of every image was equally divided into 6×7 patches and then the LBP features were extracted from these empirically weighted patches to represent the facial expressions. After that, SVM was applied to classify facial expression using the LBP features. However, this method suffers from fixed region size and positions. As a result, in [45], Shan et al. proposed boosting LBP features to solve these problems. The boosting LBP features were obtained by scaling sub-window over face images and boosted by AdaBoost. Lin et al. [35] proposed an approach to learn the effective patches statistically. In [35], the face area was divided into 8×8 patches and a multi-task sparse learning method was applied to learn the active facial patches. The experiment result showed that the active patches were around eyes, nose and mouth, which confirmed the discovery in psychology. Moreover, three different scale sizes were used in [46]. Different from [21, 35, 46], Happy et al. [23] selected 19 active patches from face area, which was supported by [35]. LBP features were extracted from the 19 active patches and top 4 patches for classifying each pair of expressions were studied. However, the size of patches are fixed in [23]. In this paper, we use the active regions that defined in the introduction and investigate the proper size of optimized active regions.

In the last decades, an increasing progress of performance has been made in facial expression recognition. An important part of this progress was achieved thanks to the emergence of deep learning methods. Liu et al. [11] proposed a novel Boosted Deep Belief Network (BDBN) for facial expression recognition. The BDBN was composed by a group of weak classifiers and each of them was responsible for classifying one expression. Experiments were carried out on CK+ database and JAFFE database, and achieved an accuracy of 96.7% and 91.8%, respectively. Although the accuracy was high, the BDBN needed a long time to train. In [11], the training time was about 8 days. The method used by Burkert et al. [47] was based on CNN. Their method was evaluated with the CK+ database and the MMI database [48] and achieved an accuracy of 99.6% and 98.63%, respectively. However, the training time of classifier was not given in [47]. Ding et al. [49] presented FaceNet2ExpNet and designed a two-stage algorithm. In the pre-training stage, face net was used to regularize expression net. In the refining stage, fully-connected layers are appended and the whole network was trained jointly. Zeng

et al. [50] used Deep Sparse Auto-Encoder (DSAE) to classify expressions. Three kinds of descriptors (LBP, Histogram of Oriented Gradient (HOG) and gray value) are used as the input of DSAE. Zeng et al. [51] proposed an Inconsistent Pseudo Annotations to Latent Truth (IPA2LT) framework to train a FER model from multiple inconsistently labeled data sets and large scale unlabeled data. Lopes et al. [28] proposed an approach for facial expression recognition by combining CNN with specific image preprocessing steps. These preprocessing steps include synthetic sample generation, rotation correction, intensity normalization, and so on. The method achieved an accuracy of 96.76% on CK+ database, and was fast to train. Similar to [28], we also combining CNN with some preprocessing steps in experiment to achieve a high accuracy with short training time. However, the preprocessing steps are different in our paper. Besides, we use the active regions to classify expressions instead of the whole face region in [28].

The represent ability of a single kind of features is weak. As a result, some researchers tried to fuse several kinds of features to achieve better performance in expression recognition [52, 53]. Fusion methods include feature-level fusion, classifier-level fusion and decision-level fusion [54], among which feature-level fusion and decision-level fusion are usually used. In [52], Chen et al. utilized feature-level fusion method to fuse visual features and acoustic features and the experiment accuracy on CK+ database was 95.7%. Kumari et al. [53] fused HOG features, Local Directional Pattern (LDP) features and LBP features in feature level to achieve effective facial expression recognition and the result showed the effectiveness of fusion. However, the feature-level fusion method suffers from curse of dimensionality. As a result, in this paper, we choose decision-level fusion method. Three classifiers are trained for left eye regions, right eye regions and mouth regions, respectively. The final classification result is obtained by majority voting of the three classifiers' results. Table 1 shows the comparison of the literatures in this section.

Facial expression recognition system

In this section, we introduce the framework of our facial expression recognition system. The proposed system can be divided into three parts: processing of data, training of classifier and testing of classifier. An overview of the facial expression recognition system is illustrated in Fig. 1.

The first part is processing of data, the images during the process are shown in Fig. 2. The images in different databases are with various background, which is noise for facial expression recognition. As a result, we first detect the face area in the image and crop it out using OpenCV for further processing. Because of the swing of subject or camera, some images in the databases are skewed. To address this problem, rotation correction according to the centers of both eyes is needed. Therefore, we need to detect some facial landmarks like the centers of eyes. The size of face varies a lot in the face area detected by OpenCV. To handle this problem, spatial normalization is carried out. After the rotation correction and spatial normalization, we search the optimized active regions using the proposed method, which is introduced in detail in "Optimized active regions searching" section. Note the active regions are related to the position of the both eyes and mouth. Although the facial landmarks have been detected, the locations of eyes and mouth were changed after rotation correction. Consequently, the landmarks are detected once more

Table 1 Comparison of literatures

Databases	Literatures	Features	Classifiers	Accuracy (%)
CK+	Liu et al. [11]		DBDN	96.7
	Shan et al. [21]	LBP of weighted patch	SVM	88.4
	Happy et al. [23]	LBP of active patch	SVM	94.09
	Lopes et al. [28]		CNN	96.76
	Zhong et al. [35]	LBP	Multi-task sparse learning	89.89
	Shan et al. [45]	Boosting LBP	SVM	92.6
	Burkert et al. [47]		CNN	99.6
	Ding et al. [49]		FaceNex2ExpNet	98.6
	Zeng et al. [50]	LBP, HOG, gray value	DSAE	95.79
	Zeng et al. [51]		LTNet	92.45
	Chen et al. [52]	HOG-TOP	SVM	89.6
JAFPE	Kumari et al. [53]	LBP, HOG, LDP	K-Nearest Neighbor	97.96
	Liu et al. [11]		DBDN	91.8
	Happy et al. [23]	LBP of active patch	SVM	92.63
	Lopes et al. [28]		CNN	53.57
	Shan et al. [45]	Boosting LBP	SVM	81
MMI	Kumari et al. [53]	LBP, HOG, LDP	K-Nearest Neighbor	95.31
	Zhong et al. [35]	LBP	Multi-task sparse learning	73.53
	Shan et al. [45]	Boosting LBP	SVM	86.9
	Burkert et al. [47]		CNN	98.63
Zeng et al. [51]		LTNet	65.61	

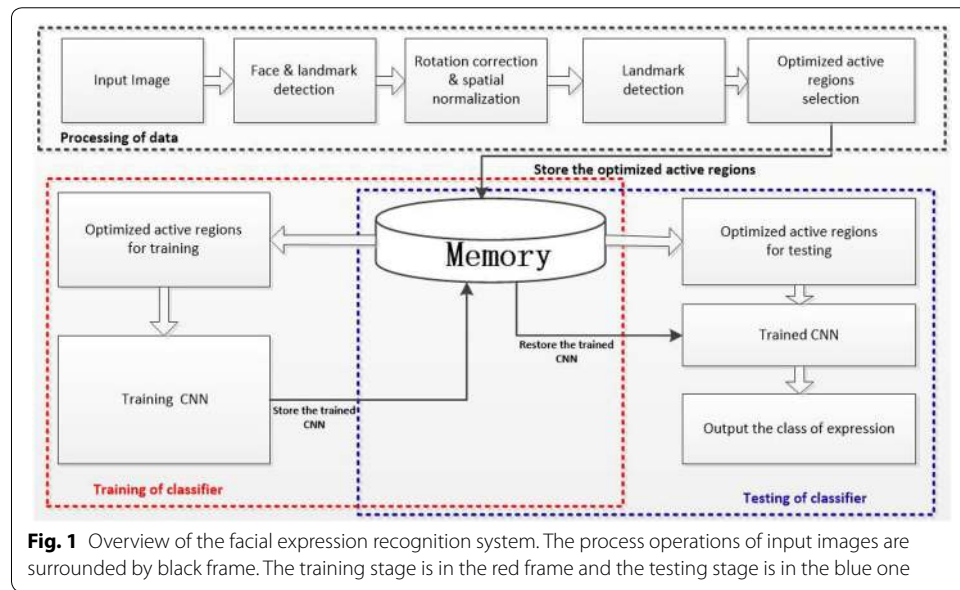
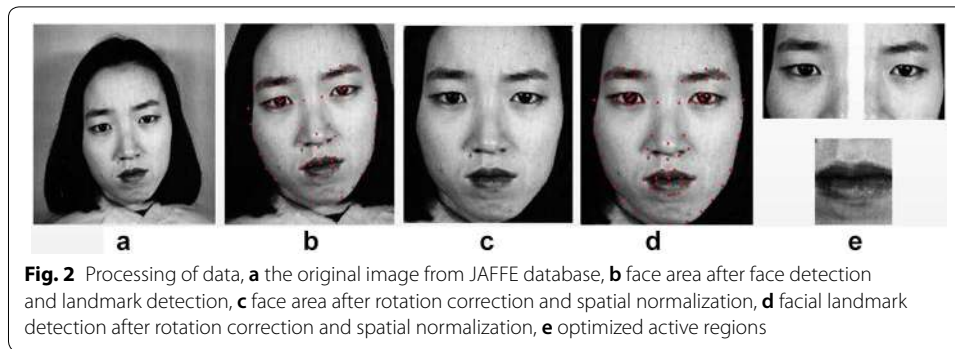


Fig. 1 Overview of the facial expression recognition system. The process operations of input images are surrounded by black frame. The training stage is in the red frame and the testing stage is in the blue one

before the search of optimized active regions. The optimized active regions are stored to the memory of computer before the training of classifier. The details for processing of data are shown in “Optimized active regions searching” section.

The second part of the system is training of classifier. In this paper, we train a CNN for each kind of optimized active regions. Three CNNs are trained for extracting features



and classification. The final classification result is obtained by majority voting of the three CNNs, which is illustrated in “[Classification based on decision-level fusion](#)” section in detail. In the training process, the optimized active regions are loaded from memory and used to train the CNNs. After that, the trained CNNs are stored to memory and wait for testing.

Testing of classifier is the last part of the system. In testing process, the optimized active regions for testing are loaded. Also, the trained CNNs are load to extract features and classify expressions for the images in testing set.

Optimized active regions searching

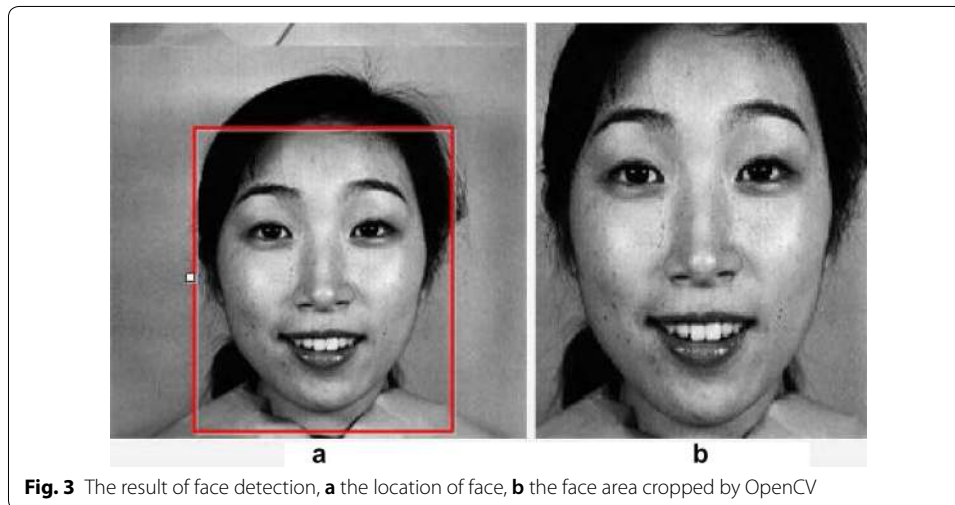
Some researchers present that not all the regions of face are active during different expressions. In [35], Zhong et al. divided the face area into 8×8 patches first and learned the active ones from the 64 patches. They reported that the active patches were around eyes, nose and mouth. This is a kind of patch-based method. In this paper, we use region-based method. The main difference between the two methods is that the region-based method select the active regions directly, instead of dividing the face area into patches. In this section, the processing of data and the method to search optimized active regions are illustrated in detail. JAFFE database is widely used in facial expression recognition area because the expression images in it are very typical. To study the proper size of active patches for expression recognition, we choose 70 images of the 10 subjects in JAFFE database. For each subject, 7 images with different expressions (happiness, surprise, fear, anger, disgust, sadness, neutral) are selected. There are many symbols in this section. To help the readers read quickly, we first show a symbol table, i.e. Table 2, and the details of the symbols will be described later.

Face detection and facial landmarks detection

It is the face area that contains the information for facial expression recognition. As a result, we detect the face area first and crop it out of image for further processing. The method proposed by Paul et al. [41] is taken for face detection, which has been implemented in OpenCV. For convenience, we detect face using OpenCV rather than implement the method by ourselves. The location of face and the cropped face area are shown in Fig. 3. However, the face area cropped by OpenCV still contains lots of background area. This problem is mitigated after the spatial normalization.

Table 2 The symbol table

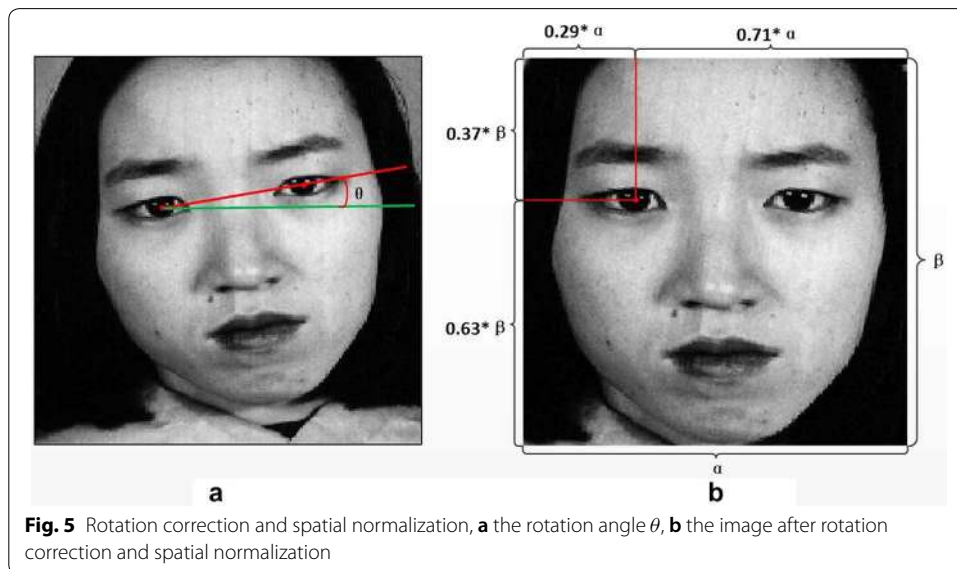
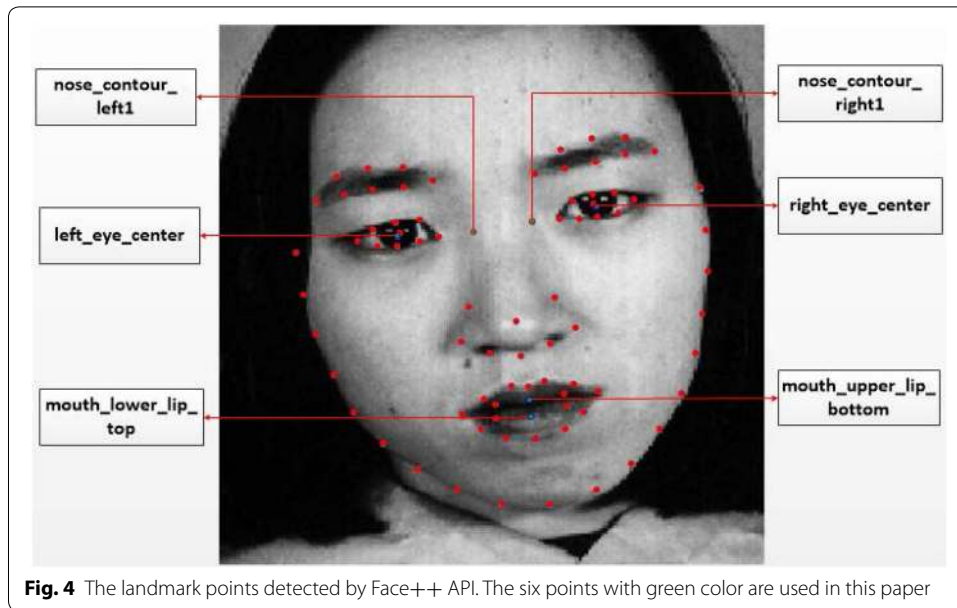
Symbols	Meanings
θ	The angle that need to rotate
I	The set of subjects in JAFFE database
J	The set of seven expressions
E_{ij}	Image of the i th subject and the j th expression
C	The set of classes of active regions
L_c	The range of the active regions size of class c
$bound_c$	The max size of active regions of class c
D	The set of four directions
Dis_{ij}^{cd}	The distance between the center of c and the limitation in direction d of E_{ij}
R_{ij}^{cl}	The region of with size l
$Hash_i^{cl}$	The sum of the distance among all c regions of subject i with a size of l
Sim_i^{cl}	The similarity of c regions of subject i with a size of l
Sim^{cl}	The similarity among c regions with size l



Face++ API is used to detect facial landmarks. Two kinds of landmark points are defined by Face++. The number of landmark points are 83 and 106, respectively. In this paper, we choose the 83 landmark points to select active regions. In fact, we only use six landmark points in this paper. The distribution of the landmark points are shown in Fig. 4, in which the six landmark points that used are highlighted. The centers of eyes are located by `left_eye_center` and `right_eye_center`, respectively. The center of mouth is defined as the middle of `mouth_upper_lip_bottom` and `mouth_lower_lip_top`. The `nose_contour_left1` and `nose_contour_right1` are used to restrict the size of active regions.

Rotation correction and spatial normalization

Some images in databases are skewed because of the moving of subject or camera, which can result in low accuracy in facial expression recognition. To solve this problem, we carry out the rotation correction according to the position of eye centers. In Fig. 5a, the red line connects the centers of two eyes and the green one is a horizontal



axis. The angle between the two lines needs to be corrected. Define the angle as θ . Suppose the centers of left eye and right eye are (x_1, y_1) , (x_2, y_2) , respectively. we calculate θ is by (1). The image is rotated according to θ .

$$\theta = \arctan \left(\frac{y_2 - y_1}{x_2 - x_1} \right) \tag{1}$$

The size of face in the cropped face area varies among subjects. This make it difficult to classify expressions with high accuracy. Another problem came with the relatively large background area in the cropped face area. However, the background area

is not useful for expression recognition. As a result, we apply spatial normalization to normalize the cropped face area. The spatial normalization is carried out according to the location of the center of left eye. Suppose that the length of face area is α after spatial normalization and the width is β . In our paper, we set $\alpha = \beta = 400$ pixels. The relationship between the center of left eye and the edges of the normalized face area is shown in Fig. 5b. After the spatial normalization, facial landmarks are detected once more.

Searching of optimized active regions

In this paper, we use three kinds of active regions to classify expressions. We call these active regions left eye regions, right eye regions and mouth regions. Some examples of active regions are shown in Fig. 6. The regions in red frame are smaller active regions and those in green are bigger ones. By experiment, we find that the similarity among the same kind of active regions from different expressions varies non-monotonously with the size of them. We define the active regions with the size that maximizes the similarity as optimized active regions. In this part, we would like to search the optimized active regions. The similarity that we use is defined via the hash distance, which is inspired by the work of Haviana et al. [55]. The smaller the hash distance is, the bigger the similarity will be. The algorithm steps to calculate hash distance of two images are shown in Algorithm 1. It should be claimed that Algorithm 1 is not our original work.

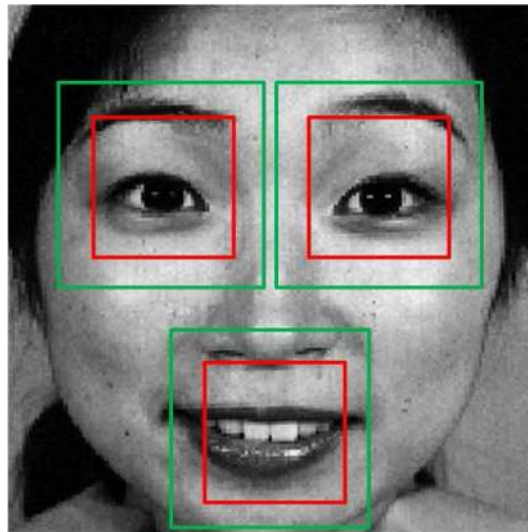
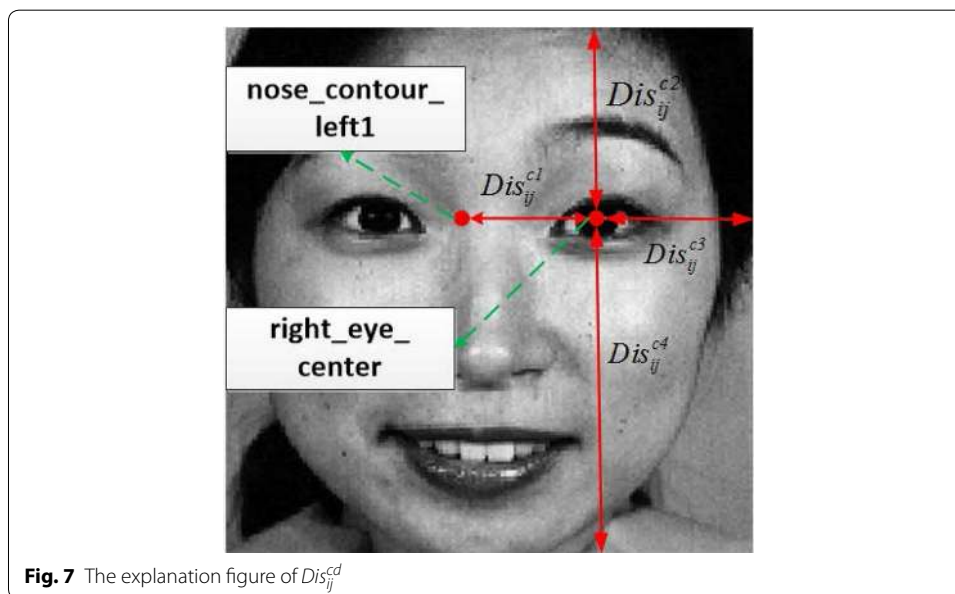


Fig. 6 Examples of active regions with two scales, surrounded by red squares and green squares, respectively

Algorithm 1. hash distance calculation**Input:** two images**Output:** hash distance between input images

- 1: Resize the input images to 8*8.
- 2: Convert the images to grayscale images.
- 3: Calculate the average grayscale value of each image.
- 4: For each resized images, set the grayscale value above its average grayscale value as 1, the rest are set as 0.
- 5: Output the Hamming distance of the resized images according to the new grayscale value (0 or 1).

To describe the searching method clearly, we introduce some useful symbols. Suppose $I = \{1, 2, 3, 4, \dots, 10\}$ is the set of subjects in JAFFE database and $J = \{1, 2, 3, \dots, 7\}$ is on behalf of the expressions (1-happy, 2-surprise, 3-fear, 4-anger, 5-disgust, 6-sadness, 7-neutral). Set E_{ij} , $i \in I$, $j \in J$, as the image of the i th subject and the j th expression. $C = \{\text{left eye}, \text{right eye}, \text{mouth}\}$ is defined as the classes of active regions. $L_c = \{1, 2, 3, \dots, \text{bound}_c\}$, $c \in C$, is the range of the active regions' size of class c . Next, we introduce the way to calculate bound_c . Define $D = \{1, 2, 3, 4\}$ as the four directions (1-left, 2-top, 3-right, 4-bottom). Write Dis_{ij}^{cd} as the distance between the center of c and the limitation in direction d of E_{ij} . For better explanation the meaning of Dis_{ij}^{cd} , we take $c = \text{right eye}$ as an example and show it in Fig. 7. For $c = \text{right eye}$, Dis_{ij}^{c1} is the distance between the center of right eye and the landmark point called nose_contour_left1. Dis_{ij}^{c2} , Dis_{ij}^{c3} , Dis_{ij}^{c4} are the distances between the center of right eye and the top edge, right edge, and bottom edge, respectively. For $c = \text{left eye}$, Dis_{ij}^{c3} is the distance between the center of left eye and the landmark point called nose_contour_right1. Dis_{ij}^{c1} , Dis_{ij}^{c2} , Dis_{ij}^{c4} , are the distances between the left eye center and the left edge, top edge, and bottom edge, respectively. For $c = \text{mouth}$, Dis_{ij}^{cd} , $d \in D$, are the distances between the center of mouth and the four edges of the image E_{ij} . bound_c is calculated by (2).



$$bound_c = \min_{i,j,d} (Dis_{ij}^{cd}), \quad i \in I, j \in J, d \in D \quad (2)$$

Define R_{ij}^{cl} , $c \in C, l \in L_c$, as the region of with size l . $Hash_i^{cl}$, $c \in C, l \in L_c$, is the sum of the distance among all c regions of subject i with a size of l and is calculated by (3).

$$Hash_i^{cl} = \sum_{x \neq y} Algorithm1(R_{ix}^{cl}, R_{iy}^{cl}) \quad (3)$$

where $x, y \in J$. In this paper, the similarity of c regions of subject i with a size of l is defined as below.

$$Sim_i^{cl} = \frac{1}{Hash_i^{cl} + 1}, \quad i \in I, c \in C, l \in L_c \quad (4)$$

Set Sim^{cl} as the average of Sim_i^{cl} among the 10 subjects. Sim^{cl} is on behalf of the similarity among c regions with size l . For each $c \in C$, we want to choose a size l maximizing Sim^{cl} .

The detailed method to search the proper size of active regions is shown in Algorithm 2. The input is 70 images from JAFFE database, i.e. E_{ij} , $i \in I, j \in J$. These images are pre-processed in the data processing stage. The output is the proper size of each kind of active region.

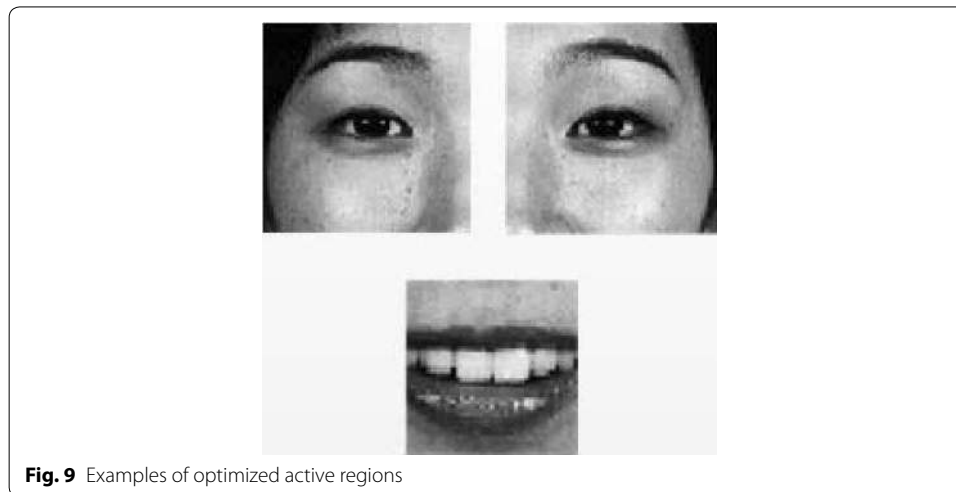
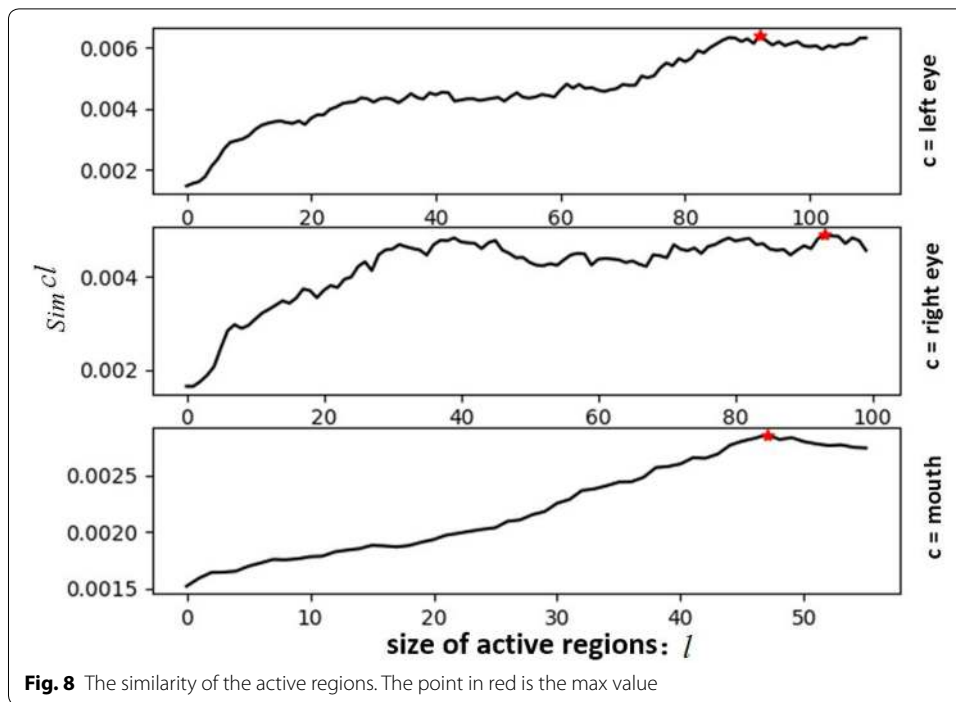
Algorithm 2. optimized active regions' sizes searching

Input: 70 images selected from JAFFE database

Output: the sizes of optimized active regions

- 1: For c in C :
 - 2: Calculate $bound_c$.
 - 3: For i in I :
 - 4: For l in L :
 - 5: Calculate $Hash_i^{cl}$ by (3).
 - 6: Calculate Sim_i^{cl} by (4).
 - 7: Calculate the the average value of Sim_i^{cl} among the ten subjects and get Sim^{cl} .
 - 8: Draw the picture of Sim^{cl} , take l as the X-axis and Sim^{cl} as the Y-axis.
 - 9: Find the size l maximizing Sim^{cl} and output it.
-

The picture drawn in step 8 in Algorithm 2 is shown in Fig. 8. The proper sizes for the left eye regions, right eye regions and mouth regions are 93 pixels, 94 pixels and 48 pixels. Note that the size obtained from Algorithm 2 is half of the edge length of optimized active regions. As a result, the edge length of the three kinds of optimized active regions are 186 pixels, 188 pixels, and 96 pixels, respectively. An example of each kind of optimized active region is shown in Fig. 9. As we can see, the background area is abandoned and the area with hair is small. It should be noticed that most of the active patches around nose in [35] are included in the optimized active regions. That is why we do not consider the nose regions as independent active regions.



Classification based on decision-level fusion

Effective facial expression recognition hugely depends upon the accurate representation of facial features. In this section, CNN is used to learn the features of facial expressions and classification. There are three kinds of optimized active regions and we train a CNN for each of them. The final classification result is obtained using a decision-level fusion method. The overview of these processes is shown in Fig. 10.

For convenience, the structures of the three CNNs are the same in this paper. However, the sizes of optimize active regions, i.e. the inputs of CNN, are different. As a result, All optimized active regions need to be resized as 64*64 pixels initially. The features of the input images are learnt by the convolution layers and the sub-sampling layers of CNN.

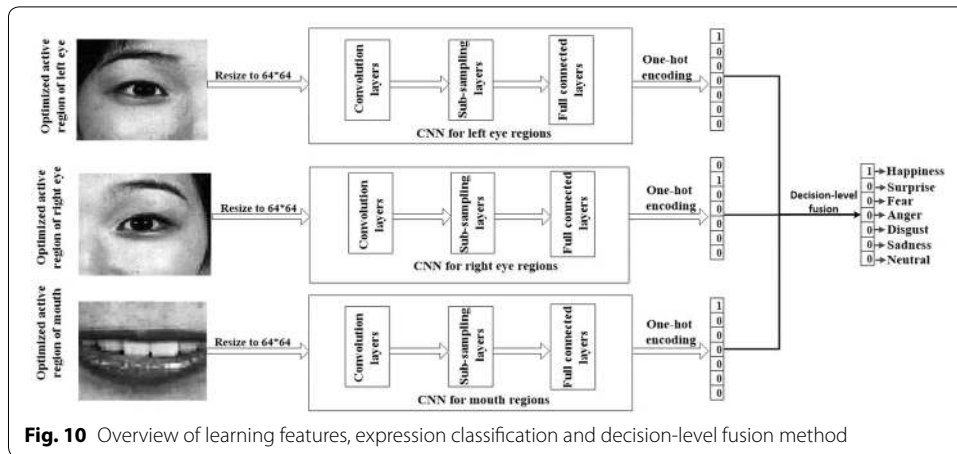


Fig. 10 Overview of learning features, expression classification and decision-level fusion method

The learnt features are applied by the fully connected layers to classify expressions. One-hot code is used for the labels of expressions. There are seven classes of expressions in this paper and the size of one-hot code is seven. Each bit of the one-hot code is on behalf of a class of expression. The index of one-hot code corresponding with facial expressions is shown in Fig. 10 as well. The bit “0” means no and “1” means yes. For example, the result after fusion in Fig. 10 is “Happiness”.

Decision-level fusion is designed to get the final result of expression recognition. The fusion method is carried out as follow.

$$final\ result = \begin{cases} j, & \text{if two or more CNNs} \\ & \text{classify the expression} \\ & \text{as } j, j \in J \\ \text{the result of} \\ \text{the CNN for} \\ \text{left eye region, otherwise} \end{cases}$$

If there are two or more classifiers classify the expression as the same class $j, j \in J$, then the final result of the classification is j . For example, the classification results of the three CNNs in Fig. 10 are “Happiness”, “Surprise” and “Happines”, since two CNNs classify the expression as “Happiness”, the fusion result is set as “Happiness”. When the results of the three CNNs are all different, we choose to believe the result of the CNN for left eye region. This is because the CNN for left eye region has the highest accuracy, as will be shown in “Experiment” section.

Experiments and discussion

We evaluated the proposed method using public available and widely used databases in facial expression recognition research area: the CK+ database, the JAFFE database and the NVIE database. The structure of our CNN is introduced in this section. A tenfold cross validation is executed for each of the three independent databases to test the performance of proposed method. In addition, we also fused the samples of three databases

together to train the classifiers, which was inspired by Happy et al. [23]. The computational complexity of the proposed method is analyzed as well in this section.

Database

The CK+ database consists of 327 facial expression sequences, and each of them was labeled as one facial expression of seven. Due to the expression image with contempt are not widely used, we eliminated the sequence which are labeled as contempt. According to Liu et al. [11], only the last frame in each sequence is provide with an expression label. However, in order to collect more image samples, we selected the last five frames from each labeled sequence in our experiment.

The JAFFE database contains 213 facial expression images from 10 Japanese female subjects. There are seven different labels in this database, i.e. happiness, surprise, angry, disgust, fear, sadness and neutral. For each kind of expression, there are about 3 images from each subject. The emotions expressed by the expression images are intense in this database, which makes it easier to classify the expression. All the images in JAFFE database were used in our experiments.

The NVIE database consists of natural visible and infrared facial expression images from 100 subjects, all of which are from China. In this database, the images are collected under different conditions, i.e. with or without glasses, different illumination conditions. These differences increased the difficulty of facial expression recognition. In this paper, we choose the natural visible images for our experiment.

Some samples from the three databases are shown in Fig. 11. It is well known that deep learning methods need a lot of data for training to achieve wonderful performance. Unfortunately, the amount of data in the available databases are not enough for the training of CNNs. To generate more data, we first flipped the images horizontally. Then the histogram equalization was carried out on the original and flipped images. After that, the gray scales of each image are slightly changed randomly by the help of OpenCV. After the three operations, each original image can generate images (including the original one). As a result, the amount of the data was increased to 8 times of the original amount. The optimized active regions of all the images are saved in memory for training and testing the classifiers.

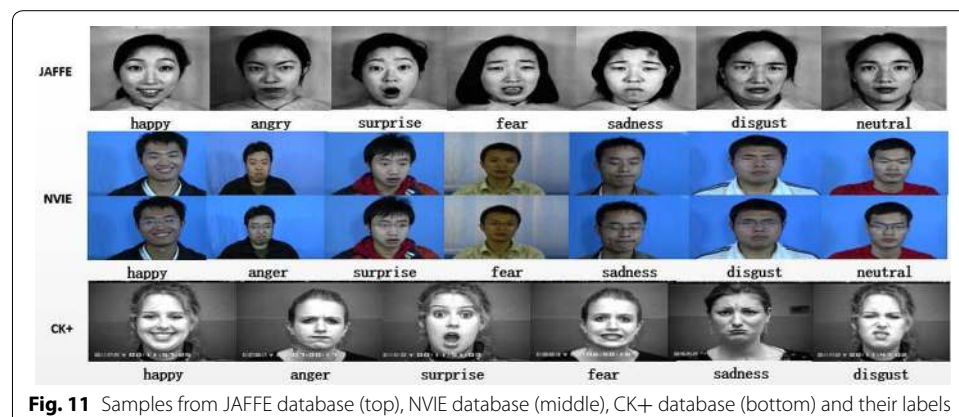


Fig. 11 Samples from JAFFE database (top), NVIE database (middle), CK+ database (bottom) and their labels

Structure of CNN

The structure of the proposed CNN is represented in Fig. 12. The CNN receives 64*64 grayscale images as input. There are three convolution layers with ReLU active functions in our CNN, each of which is followed by a sub-sampling layer. The kernel sizes in convolution layers and sub-sampling layers are set as 5*5 and 2*2, respectively. In each sub-sampling layer, the stride step is set as 2. The size and number of feature maps after the first convolution are 64*64 and 32, respectively, which is written as “64*64*32” in Fig. 12. The same writing method is also used in the rest layers in Fig. 12. There are 1024 neurons in both of the two fully connected layers. Finally, the CNN outputs the confidence score of different expressions. The number of the output, i.e. *N* in Fig. 12, depends on the number of different expressions in the database.

Experiments on independent database

We ran tenfold cross validation to evaluated the performance of the proposed method. Not only the accuracy after decision-level fusion was tested, but also the accuracy of CNN for each kind of optimized active regions. Figure 13 reports the accuracy of

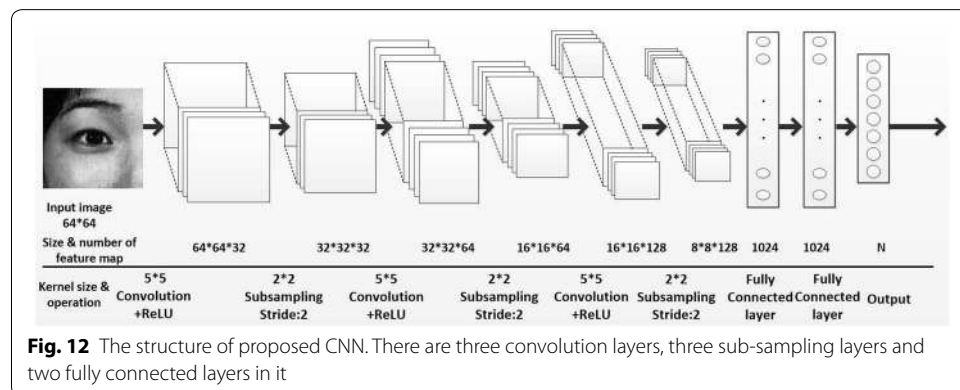


Fig. 12 The structure of proposed CNN. There are three convolution layers, three sub-sampling layers and two fully connected layers in it

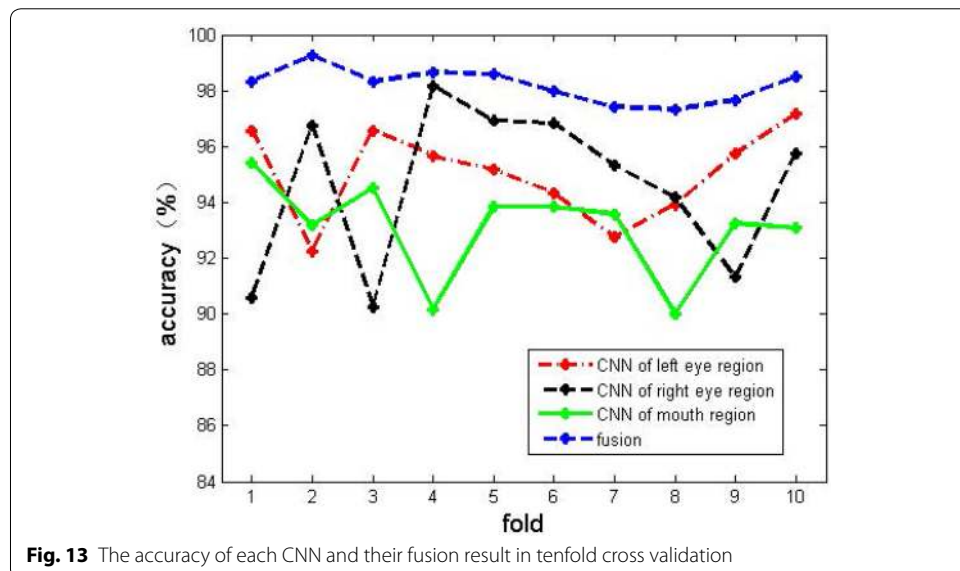


Fig. 13 The accuracy of each CNN and their fusion result in tenfold cross validation

each fold in tenfold cross validation. The mean accuracy of tenfold cross validation is reported in Table 3. On one hand, Table 3 shows that the performance of the three CNNs is excellent and even the worst one can achieve an accuracy of 93.08%. This result illustrate that the optimized active regions maintain most information for facial expression recognition. On the other hand, compared with each CNN, the accuracy was increased by 3–5% by decision-level fusion, which shows the effectiveness and feasibility of the fusion method.

In the decision-level fusion method proposed in “Classification based on decision-level fusion” section, when the results of the three CNNs are all different, we need to choose one result to believe. It seems that different choices would result in different fusion accuracy. From Table 3, we draw a conclusion that the CNN for left eye regions achieved the highest accuracy. This is the reason why we choose to believe the result of CNN for left eye regions when the result of the three CNNs are all different. However, we do not know if this choice is really helpful. In other words, what the accuracy will be if we choose the CNNs for right eye region or mouth region? To answer this question, tenfold cross validation experiments using different choices of CNNs were carried out.

The fusion accuracy using different results of CNNs is shown in Fig. 14. The mean fusion accuracies using CNN of left eye regions, right eye regions and mouth regions are

Table 3 Mean accuracy on CK+ database

Optimized active regions	Mean accuracy in tenfold cross validation (%)
CNN of left eye region	95.02
CNN of right eye region	94.61
CNN of mouth region	93.08
Fusion	98.21

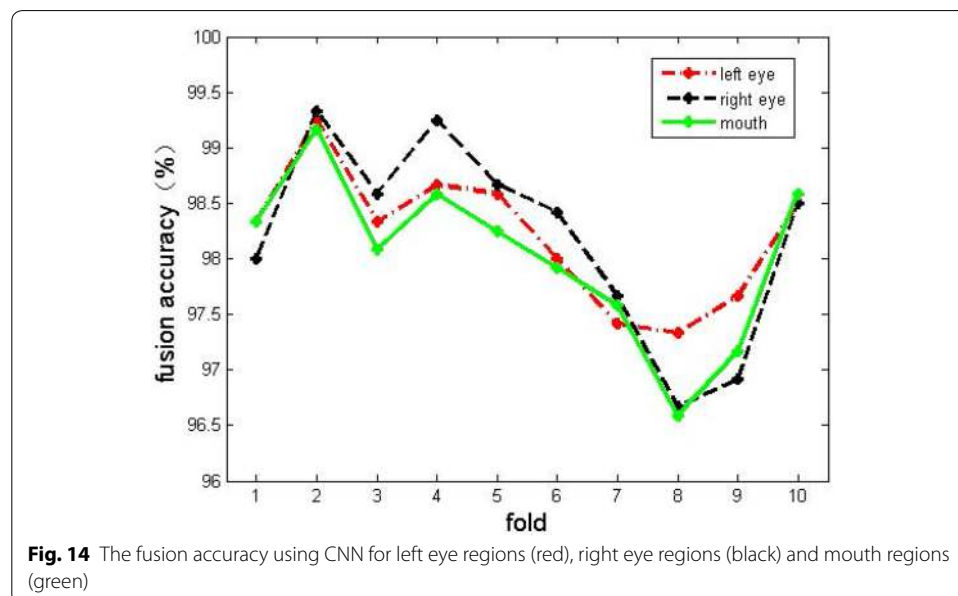


Table 4 Confusion matrix on CK+ database

	Ha	Su	Fe	An	Di	Sa
Ha	99.62	0	0	0	0.38	0
Su	0	98.74	0	0.63	0	0.63
Fe	0	1.26	92.41	3.80	0	2.53
An	0	0	0	99.45	0.55	0
Di	0	0	0	0.86	99.14	0
Sa	0	0	0	0.83	0	99.17

Italic values indicate the recognition accuracy of each expression

98.21%, 98.20% and 98.02%, respectively. The results obtained by using left eye regions and right eye regions are really closed. We assume this is caused by the symmetry of the two eye regions. Though the results are closed, the fusion accuracy using left eye regions is the highest. These results confirmed the properness of our choice.

In this paper, the expressions of happiness, surprise, fear, anger, disgust, sadness, neutral are denoted as Ha, Su, Fe, An, Di, Sa and Ne for simplicity. Table 4 shows the confusion matrix of six expressions obtained by proposed method. As observed in Table 4, most results of the six expressions are higher than 98%. The best result was achieved by the happiness expression. In our opinion, this is because the happiness expressions can be characterized by rising mouth corner and narrowing eyes easily. There are 3.8% of the fear expressions are classified as anger expression. We assume this is because fear and anger expressions involve similar and subtle facial movements. The results of the three CNNs and their fusion on different expressions are shown in Fig. 15.

A comparison of proposed method with the state-of-the-art methods is reported in Table 5. Although the accuracies of surprise, fear and disgust expressions obtained by proposed method are not as good as that of other method, we achieve better performance on happiness, anger and sadness expressions. What is more, the average accuracy is the highest among these methods.

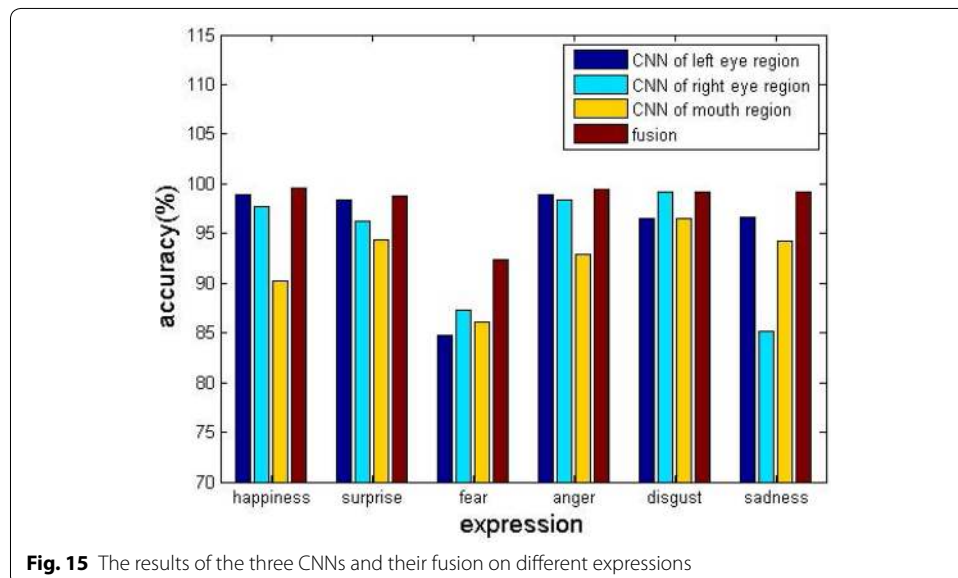


Fig. 15 The results of the three CNNs and their fusion on different expressions

Table 5 Accuracy (%) comparison on CK+ database

	[35]	[56]	[23]	[28]	Proposed
Ha	95.42	98.07	94.20	98.55	<i>99.62</i>
Su	98.27	<i>100</i>	98.46	99.20	98.74
Fe	81.11	92.00	94.33	<i>96.00</i>	92.41
An	71.39	87.10	97.80	93.33	<i>99.45</i>
Di	95.33	90.20	93.33	<i>100</i>	99.14
Sa	88.01	91.47	96.42	84.52	<i>99.17</i>
Average	88.26	93.14	94.09	96.76	<i>98.50</i>

Italic values indicate the highest accuracy

Table 6 Confusion matrix on JAFFE database

	Ha	Su	Fe	An	Di	Sa	Ne
Ha	<i>96.43</i>	0	0	0	0	0	3.57
Su	0	<i>100</i>	0	0	0	0	0
Fe	0	4.00	<i>96.00</i>	0	0	0	0
An	0	0	0	<i>95.00</i>	0	0	5.00
Di	4.35	0	0	0	<i>91.30</i>	0	4.35
Sa	4.17	0	0	0	0	<i>91.66</i>	4.17
Ne	0	0	0	0	0	0	<i>100</i>

Italic values indicate the recognition accuracy of each expression

Table 7 Confusion matrix on NVIE database

	Ha	Su	Fe	An	Di	Sa	Ne
Ha	<i>97.76</i>	0.82	0	0.20	0.82	0	0.41
Su	0.86	<i>98.27</i>	0.22	0	0.65	0	0
Fe	4.26	1.07	<i>92.75</i>	0	0.43	1.07	0.43
An	0.41	0	0	<i>99.19</i>	0	0.20	0.20
Di	1.15	1.83	0	0	<i>96.56</i>	0.23	0.23
Sa	1.01	0.61	1.62	1.42	1.01	<i>94.13</i>	0.20
Ne	0	0	0	0	1.36	0	<i>98.64</i>

Italic values indicate the recognition accuracy of each expression

In addition to CK+ database, we also carried out tenfold cross validation on JAFFE database and NVIE database. Both of the two database contain images with neutral expression. The average accuracies on JAFFE database and NVIE database are 98.41% and 96.51%, respectively. The confuse matrices are reported in Tables 6 and 7. It should be noticed that the conditions of the expression images in NVIE database are complex. About half of the expression images are with glasses. Besides, the illumination condition varies among these images. However, the accuracy is still high, which shows the robustness of the proposed method.

Experiments on fused database

As generalization, we also trained the classifiers, i.e. the three CNNs, using the images gathered from the three databases. We selected the images randomly and put them together as a new database, i.e., the fused database. The fused database is used to train

Table 8 The results on different databases

Database	Accuracy (%)
CK+	95.36
JAFFE	96.57
NVIE	89.38

the classifiers. For each database, 90% of the images are selected and the rest images are regarded as testing set. In each database, the images were chosen to train or test the classifiers with equal probability. The testing results on the three database is shown in Table 8. Note that the neutral expressions are not used because there is no such expression in CK+ database.

The images in the fused database are different in many aspects: the subjects may come from eastern countries or western countries, the illumination may be in different conditions, and the subjects may wear glasses or not. All of these differences result in the difficulty of facial expression recognition. Therefore, the accuracy on fused database is lower than that on the independent database. There is an obvious difference between the NVIE database and the other two databases: the NVIE database contains images from the subjects who wear glasses. When the classifier are trained on the fused database, they may learn more information about the expressions without glasses. We assume this is the reason why the accuracy on NVIE declined the most. Happy et al. [23] also carried out experiments on fused data of CK+ database and JAFFE database. The accuracies are 89.64% and 85.06% respectively, which are lower than our results.

Computational complexity

Most of the experiment time was spent on searching optimized active regions and training the CNNs. The searching of optimized active regions was carried out on a computer with a intel core i7-4790K CPU, the frequency of which is 4.00 GHz. The Random Access Memory (RAM) of the computer is 8 Gb. It took about 5 min 19 s to run Algorithm 2, i.e., the searching process of optimized active regions. It seems that the searching time is too long. However, we can get all of the optimized active regions by running Algorithm 2 only one time. As a result, the running time is acceptable.

It is well known that training deep models ordinarily needs a long time. The training time is related to many factors, such as the amount of data, the structure of model, the computing power and so on. In this paper, our models are trained on a computer with high computing power, the experiment environment of which is shown in Table 9.

Table 9 The environment of our experiments

Operator system	Ubuntu 17.04
Language	Python 3.6
Framework	Tensorflow 1.4
CPU	Intel Core i7-7700K @4.20 GHz
GPU	NVIDIA GeForce GTX 1080TI
CUDA	Version 8.0
CuDNN	Version 5.1

Table 10 Average training time on different databases

Database	Time
CK+	3 min 48 s
JAFFE	1 min 40 s
NVIE	27 min 57 s
Fused database	32 min 43 s

As reported in Fig. 12, the structure of the CNN is simple. As a result, training such model does not need lots of time. In this paper, we have trained the CNNs on different databases, and both of the training time and testing time have been recorded. In tenfold cross validation, the average training time of each fold, i.e., the average time to train the three CNNs, is shown in Table 10. The training time on the fused database is reported in the table as well.

As we can see from Table 10, it takes more time to train CNNs on NVIE database than on CK+ and JAFFE database. This is because there are more images in NVIE database and the condition of them are complex. For the same reason, the training time on fused database took more time than that on independent database. Our proposed system takes about 0.01 s to recognize an expression. Due to the different experiment environment, it is difficult to make a fair comparison for the efficiency of different methods. Still, here we list some reported results and their hardware conditions. The training time of CNN which was applied by Lopes et al. [28] was about 20 min on CK+ database. The result was obtained using Intel Core i7 CPU with a frequency of 3.4 GHz and a NVIDIA GeForce GTX 660 GPU. Liu et al. [11] trained BDBN on 6-core 2.4 GHz PC and the training time was about 8 days for eightfold cross validation.

Conclusion

In this paper, we reported an effective facial expression recognition system for classifying six or seven basic expressions accurately. Instead of using the whole face region, we defined three kinds of active regions, i.e., left eye regions, right eye regions and mouth regions. Method to search optimized active regions from the three kinds of active regions was proposed. We trained a CNN for each kind of optimized active regions to extract features and classify expressions. A decision-level fusion method was applied, by which the final result of expression recognition was obtained via majority voting of the three CNNs. Experiments on independent databases and fused database were carried out to evaluate the performance of the proposed system.

According to the similarity of active regions, we proposed the method to search optimized active regions. The edge length of optimized active regions for left eye regions, right eye regions and mouth regions are 186 pixels, 188 pixels and 96 pixels respectively. In order to find out which kind of regions is best for expression recognition, we carried tenfold cross validation on CK+ database. The result indicated that the left eye region was the best optimized region. The accuracies using left eye regions, right eye regions and mouth regions are 95.02%, 94.61% and 93.08%, respectively. The high accuracies illustrate that the optimized active regions maintain most information for facial expression recognition. As a result, our proposed method is effective. It needs to be addressing

that the accuracies of CNNs for left eye regions and right eye regions are very closed. We applied a decision-level fusion strategy, by which the three CNNs vote to derive the final label of expressions. The effect of decision-level confusion was obvious. After the fusion, the accuracy was increased by 3–5%. Experiments on JAFFE database, NVIE database and the fused database are also carried out in this paper. Compared with the earlier works with similar approach, our proposed system achieved better performance.

Searching of optimized active regions and training the CNNs are two steps that take the longest time. Although it takes about 5 min 19 s to search the optimized active regions, we can get all of the optimized active regions by running the searching algorithm just one time. The training time varies among databases due to the amount and characteristics of images. For each fold of the tenfold cross validation, it took about 1 min 40s to train the CNNs on JAFFE database whereas 27 min 57 s was required on NVIE database. The principle is that the more complex the database is, the more time will be taken to train the CNNs. The recognition speed of our proposed system is fast. It takes about 0.01 s to recognize an expression.

Although the performance of the proposed system is good, it has limitations. First, the time to search optimized active regions is long. We would like to explore some other methods that need less time. Second, the facial landmarks are detected using API of Face++ company with the limitation of accessibility unless there is the internet. Last but not least, the shape of active regions in this paper is square. As we all know, the shape of eyes and mouth are like rectangle more. What will the performance of the system be if we use rectangle regions? In the future, we are gonging to devote ourselves to address these problems.

Authors' contributions

All authors read and approved the final manuscript.

Author details

¹ Department of Engineering Science, National Cheng Kung University, Tainan, Taiwan. ² School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, China. ³ School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China.

Acknowledgements

The authors wish to thank Lucey et al. for the use of the CK+ database, Lyons et al. for the use of the JAFFE database, and Wang et al. for the use of NVIE database.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 26 August 2018 Accepted: 21 October 2018

Published online: 09 November 2018

References

1. Calvo RA, D'Mello S (2010) Affect detection: an interdisciplinary review of models, methods, and their applications. *IEEE Trans Affect Comput* 1(1):18–37
2. Mollahosseini A, Chan D, Mahoor MH. Going deeper in facial expression recognition using deep neural networks. In: 2016 IEEE winter conference on applications of computer vision (WACV); 2016. p. 1–10.
3. Zavaschi TH, Britto AS Jr, Oliveira LE, Koerich AL (2013) Fusion of feature sets and classifiers for facial expression recognition. *Expert Syst Appl* 40(2):646–655
4. Zhang Y, Ji Q (2005) Active and dynamic information fusion for facial expression understanding from image sequences. *IEEE Trans Pattern Anal Mach Intell* 27(5):699–714
5. Zhao G, Pietikainen M (2007) Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Trans Pattern Anal Mach Intell* 29(6):915–928

6. Bartlett MS, Littlewort G, Frank M, Lainscsek C, Fasel I, Movellan J. Recognizing facial expression: machine learning and application to spontaneous behavior. In: CVPR 2005. IEEE computer society conference on computer vision and pattern recognition, vol. 2. 2005. p. 568–73.
7. Buciu I, et al. A new sparse image representation algorithm applied to facial expression recognition. In: Proceedings of the 2004 14th IEEE signal processing society workshop. Machine learning for signal processing. 2004. p. 539–48.
8. Lin Y, Song M, Quynh DTP, He Y, Chen C (2012) Sparse coding for flexible, robust 3d facial-expression synthesis. *IEEE Comput Graph Appl* 32(2):76–88
9. Liu W, Song C, Wang Y. Facial expression recognition based on discriminative dictionary learning. In: 21st international conference on pattern recognition (ICPR). 2012. p. 1839–42.
10. Izard CE (1994) Innate and universal facial expressions: evidence from developmental and cross-cultural research. *Psychol Bull* 115(2):288–99
11. Liu P, Han S, Meng Z, Tong Y. Facial expression recognition via a boosted deep belief network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2014. p. 1805–12.
12. Rivera AR, Castillo JR, Chae OO (2013) Local directional number pattern for face analysis: face and expression recognition. *IEEE Trans Image Proces* 22(5):1740–1752
13. Ekman P, Friesen W (1978) Facial action coding system: a technique for the measurement of facial movement. Consulting Psychologists, San Francisco
14. Chang Y, Hu C, Turk M. Manifold of facial expression. In: IEEE. 2003. p. 28.
15. Pantic M, Patras I (2006) Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences. *IEEE Trans Syst Man Cybern* 36(2):433–449
16. Rothkrantz LJ, Pantic M (2004) Facial action recognition for facial expression analysis from static face images. *IEEE Trans Syst Man Cybern* 34(3):1449–1461
17. Cohen I, Sebe N, Garg A, Chen LS, Huang TS (2003) Facial expression recognition from video sequences: temporal and static modeling. *Computer Vision Image Understand* 91(1–2):160–187
18. Valstar MF, Patras I, Pantic M. Facial action unit detection using probabilistic actively learned support vector machines on tracked facial point data. In: IEEE computer society conference on computer vision and pattern recognition workshops. 2005. CVPR Workshops. p. 76.
19. Valstar M, Pantic M. Fully automatic facial action unit detection and temporal analysis. In: Computer vision and pattern recognition workshop, 2006 conference on CVPRW'06. 2006. p. 149.
20. Lajevardi SM, Hussain ZM (2012) Automatic facial expression recognition: feature extraction and selection. *Signal Image Video Process* 6(1):159–169
21. Shan C, Gong S, McOwan PW. Robust facial expression recognition using local binary patterns. In: IEEE international conference on image processing. ICIP 2005, vol. 2. 2005. p. 370.
22. Zhang Z, Lyons M, Schuster M, Akamatsu S. Comparison between geometry-based and gabor-wavelets-based facial expression recognition using multi-layer perceptron. In: Third IEEE international conference on automatic face and gesture recognition. 1998. p. 454–9.
23. Happy S, Routray A (2015) Automatic facial expression recognition using features of salient facial patches. *IEEE Trans Affect Comput* 6(1):1–12
24. Liu Y, Cao Y, Li Y, Liu M, Song R, Wang Y, Xu Z, Ma X. Facial expression recognition with pca and lbp features extracting from active facial patches. In: IEEE third international conference on real-time computing and robotics (RCAR). 2016. p. 368–73.
25. Wold S, Esbensen K, Geladi P (1987) Principal component analysis. *Chemometr Intell Lab Syst* 2(1–3):37–52
26. Wang Z, Ruan Q, An G. Facial expression recognition based on tensor local linear discriminant analysis. In: IEEE 11th IEEE international conference on signal processing (ICSP), vol. 2. 2012. p. 1226–9.
27. Jung H, Lee S, Park S, Kim B, Kim J, Lee I, Ahn C. Development of deep learning-based facial expression recognition system. In: 21st Korea–Japan joint workshop on frontiers of computer vision (FCV). 2015. p. 1–4.
28. Lopes AT, de Aguiar E, De Souza AF, Oliveira-Santos T (2017) Facial expression recognition with convolutional neural networks: coping with few data and the training sample order. *Pattern Recogn* 61:610–628
29. Lee SH, Plataniotis KN, Ro YM (2014) Intra-class variation reduction using training expression images for sparse representation based facial expression recognition. *IEEE Trans Affect Comput* 5(3):340–351
30. Lucey P, Cohn JF, Kanade T, Saragih J, Ambadar Z, Matthews I. The extended cohn-kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression. In: IEEE computer society conference on computer vision and pattern recognition workshops (CVPRW). 2010. p. 94–101.
31. Kanade T, Tian Y, Cohn JF. Comprehensive database for facial expression analysis. In: IEEE Fg. 2000. p. 46.
32. Lyons M, Akamatsu S, Kamachi M, Gyoba J. Coding facial expressions with gabor wavelets. In: Third IEEE international conference on automatic face and gesture recognition. 1998. p. 200–5.
33. Salmam FZ, Madani A, Kissi M. Facial expression recognition using decision trees. In: 13th international conference on computer graphics, imaging and visualization (CGIV). 2016. p. 125–30.
34. Jabid T, Kabir MH, Chae O (2010) Robust facial expression recognition based on local directional pattern. *ETRI J* 32(5):784–794
35. Zhong L, Liu Q, Yang P, Liu B, Huang J, Metaxas DN. Learning active facial patches for expression analysis. In: IEEE conference on computer vision and pattern recognition (CVPR). 2012. p. 2562–9.
36. Wang S, Liu Z, Lv S, Lv Y, Wu G, Peng P, Chen F, Wang X (2010) A natural visible and infrared facial expression database for expression recognition and emotion inference. *IEEE Trans Multimedia* 12(7):682–691
37. Lv Y, Wang S, Shen P. A real-time attitude recognition by eye-tracking. In: Proceedings of the third international conference on internet multimedia computing and service. New York: ACM. 2011. p. 170–3.
38. Tong Y, Wang Y, Zhu Z, Ji Q (2007) Robust facial feature tracking under varying face pose and facial expression. *Pattern Recogn* 40(11):3195–3208
39. Dang K, Sharma S. Review and comparison of face detection algorithms. In: 7th international conference on cloud computing, data science & engineering-confluence. 2017. p. 629–33.

40. Tao Q-Q, Zhan S, Li X-H, Kurihara T (2016) Robust face detection using local cnn and svm based on Kernel combination. *Neurocomputing* 211:98–105
41. Viola P, Jones MJ (2004) Robust real-time face detection. *Int J Comput Vision* 57(2):137–154
42. Fan X, Zhang F, Wang H, Lu X. The system of face detection based on opencv. In: 24th Chinese control and decision conference (CCDC), 2012. p. 648–51.
43. Xiong X, De la Torre F. Supervised descent method and its applications to face alignment. In: Proceedings of the IEEE conference on computer vision and pattern recognition. p. 532–9.
44. Zhu S, Li C, Change Loy C, Tang X. Face alignment by coarse-to-fine shape searching. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2015. p. 4998–5006.
45. Shan C, Gong S, McOwan PW (2009) Facial expression recognition based on local binary patterns: a comprehensive study. *Image Vision Comput* 27(6):803–816
46. Zhong L, Liu Q, Yang P, Huang J, Metaxas DN (2015) Learning multiscale active facial patches for expression analysis. *IEEE Trans Cybern* 45(8):1499–1510
47. Burkert P, Trier F, Afzal MZ, Dengel A, Liwicki M. Dexpression: deep convolutional neural network for expression recognition. 2015. arXiv preprint [arXiv:1509.05371](https://arxiv.org/abs/1509.05371).
48. Pantic M, Valstar M, Rademaker R, Maat L. Web-based database for facial expression analysis. In: 2005 IEEE international conference on multimedia and expo. 2005. p. 5.
49. Ding H, Zhou SK, Chellappa R. Facenet2expnet: regularizing a deep face recognition net for expression recognition. In: 12th IEEE international conference on automatic face & gesture recognition (FG 2017). 2017. p. 118–26.
50. Zeng N, Zhang H, Song B, Liu W, Li Y, Dobaie AM (2018) Facial expression recognition via learning deep sparse autoencoders. *Neurocomputing* 273:643–649
51. Zeng J, Shan S, Chen X. Facial expression recognition with inconsistently annotated datasets. In: Proceedings of the European conference on computer vision (ECCV). 2018. p. 222–37.
52. Chen J, Chen Z, Chi Z, Fu H (2018) Facial expression recognition in video with multiple feature fusion. *IEEE Trans Affect Comput* 9(1):38–50
53. Kumari J, Rajesh R, Kumar A. Fusion of features for the effective facial expression recognition. In: 2016 international conference on communication and signal processing (ICCSPP). 2016. p. 0457–61.
54. Petridis S, Pantic M (2016) Prediction-based audiovisual fusion for classification of non-linguistic vocalisations. *IEEE Trans Affect Comput* 7(1):45–58
55. Haviana SFC, Kurniadi D (2016) Average hashing for perceptual image similarity in mobile phone application. *J Telemat Inform* 4(1):12–18
56. Zhang L, Tjondronegoro D (2011) Facial expression recognition using facial movement features. *IEEE Trans Affect Comput* 2(4):219–229

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
