



# Sample awareness-based personalized facial expression recognition

Huihui Li<sup>1</sup> · Guihua Wen<sup>1</sup>

Published online: 13 February 2019  
© The Author(s) 2019

## Abstract

The behavior of the current emotion classification model to recognize all test samples using the same method contradicts the cognition of human beings in the real world, who dynamically change the methods they use based on current test samples. To address this contradiction, this study proposes an individualized emotion recognition method based on context awareness. For a given test sample, a classifier that was deemed the most suitable for the current test sample was first selected from a set of candidate classifiers and then used to realize the individualized emotion recognition. The Bayesian learning method was applied to select the optimal classifier and then evaluate each candidate classifier from the global perspective to guarantee the optimality of each candidate classifier. The results of the study validated the effectiveness of the proposed method.

**Keywords** Facial expression recognition · Personalized classification · Dynamic selection · Bayesian

## 1 Introduction

Widely applied in mental health and human–computer interaction, emotion recognition is currently a popular research topic in the fields of computer vision and artificial intelligence [1–3] because it involves multiple disciplines, such as image processing, pattern recognition, and psychology. However, the diversity of facial expressions makes the emotion recognition difficult. For example, the collected facial images might be unidentifiable because of the lighting environment [4]. Moreover, the facial expressions of human beings are complicated and diverse, with fairly significant individual differences in skin color, age, and appearance. These differences place an added burden on machine learning.

Currently there are many emotion recognition methods, including deep learning and ensemble learning methods. They train an emotion classification model and then use this model to identify all test samples. This trained emotion classification model remains unchanged, without considering the practical conditions of each test sample. However, these methods are inconsistent with human cognition laws [5] in the real world. They model the inertial thinking and thus easily misclassify test samples [6]. Human beings change their methods

dynamically based on the current test samples, instead of identifying all test samples with the same method. For example, human thinking follows the principle of simplicity (the Gestalt principle) [7]. Simple object recognition only needs simple methods, while complex object recognition needs complex methods [8]. However, most of the existing machine learning methods only consider the complexity of the whole dataset [9], or the complexity of the local neighborhood [10], without distinguishing the complexity of the object to be identified. In addition, for the same test sample, each person’s emotional recognition ability is different, which is also true for classifiers. As the ensemble classifier emphasizes, the base classifiers should be diverse, indicating that many classifiers have different capabilities and complementarity [4, 10, 11]. In experiments, a classifier may work well for some test samples, but may often make mistakes for other test samples. In particular, when two classifiers are used to classify test samples, their classification ability may be totally opposite. Thus, it is rational to select the classifier dynamically in such circumstances [12, 13]. This can be implemented by first searching for the local neighborhood of each test sample, and then evaluating the classifier’s capability through the samples in the local neighborhood in order to choose the most suitable classifier by which to classify the test samples [14]. The key issue of this method is that a set of candidate classifiers should be generated with high accuracy and diversity. The diversity of two classifiers is reflected in terms of the ability of each to classify the different samples. Ideally, classifiers should complement each other so that the most appropriate classifier can be selected for

---

✉ Guihua Wen  
crghwen@scut.edu.cn

<sup>1</sup> School of Compute Science and Engineering, South China University of Technology, Guangzhou 510641, China

each new test sample [10, 11]. This is different from methods with static selection of classifiers, which occurs during model selection. During model selection, once the classifier is selected on the training set, it will classify all test samples without considering the differences among them. The study of dynamic classifier selection shows that it is a very effective tool for overcoming pathological classification problems, e.g., when the training data are too small or there are insufficient data by which to build the classification model [9].

The primary problem of dynamic classifier selection is measuring the ability of each classifier in classifying test samples. The most common methods for solving the problem are individual-based metrics and group-based metrics [13]. The former performs the measurement based on the classifier's individual information, such as rankings, accuracy, probability, and behavior, while the latter considers the relationship between the candidate classifiers. However, both measurement methods select the classifier according to the neighborhood of the test samples in the training set. It is difficult to obtain the globally considered performance using local estimation. Secondly, it is time-consuming to find the neighborhood of each test sample from a large training set. Cruz et al. proposed a method to dynamically select classifiers based on machine learning [14]. Using meta-features to describe the capabilities of each classifier in a local neighborhood, this method first dynamically selects classifiers for test samples through machine learning, and then uses the selected classifier to classify the test samples. The other type of methods not only consider the accuracy of the classifier but also the complexity of the problem, e.g., the complexity of the neighborhood of the test samples [9].

Based on the local neighborhood of the test samples, both aforementioned methods have two disadvantages. It is time-consuming to seek the neighborhood of a given test sample under large training data. Second, the performance of the classifier is limited to the local optimum rather than the global optimum. Hence, this paper proposes the sample awareness-based personalized (SAP) facial expression recognition method. SAP used the Bayesian learning method to select the optimal classifier from the global perspective, and then used the selected classifier to identify the emotional class of each test sample. The main contributions are that the idea of sample awareness is introduced to the field of emotion recognition, and a new emotion recognition method is proposed.

## 2 Related works

The SAP method proposed in this study is new in the field of emotion recognition. It selects the classifier dynamically for each test sample, which is different from the current dynamic classifier selection methods. The current dynamic classifier selection methods can be categorized into four types, which

will be compared and analyzed in this paper. The recently developed methods for facial expression recognition are also presented, such as those based on 3D information of face and ensemble learning methods.

### 2.1 Dynamic classifier selection methods

#### 2.1.1 Classification accuracy based on local neighborhood

These methods are based on the classification accuracy of the local neighborhood of the test sample, where the neighborhood is defined by the  $k$  nearest neighbors (KNN) algorithm [15] or the clustering algorithm [16]. For example, the overall local accuracy (OLA) selects the optimal classifier based on the accuracy of the classifier in the local neighborhood [17]. Another method is the local class accuracy (LCA), which uses posteriori information to calculate the performance of the base classifier for particular classes [18]. In addition, another method was proposed to sort the classifiers based on the number of consecutive correct classifications of samples in the local neighborhood. The larger the number, the higher the classifier is ranked to be selected [19].

There are two methods: A Priori (APRI), and A Posteriori (APOS) [20]. APRI selects the classifier based on the posterior probability of classes of the test sample in its neighborhood, which considers the distance from each neighborhood to the test sample. Unlike APRI, APOS considers each classifier's classification label for the current test sample. Based on these two methods, two new methods were proposed: KNORA-Eliminate (KE) and KNORA-Union (KU) [21]. KE only selects the classifier that correctly classifies all neighborhoods, whereas KU only selects the classifier that correctly classifies at least one neighborhood. Xiao et al. proposed a dynamic classifier ensemble model for customer classification with imbalanced class distribution. It utilizes the idea of LCA, but the prior probability of each class is used to deal with imbalanced data when calculating the classifier's performance [22]. The difference between these methods is that the local information is used in different ways, but they are both based on the local neighborhood of the test sample.

#### 2.1.2 Decision template methods

Decision template methods are also based on the local neighborhood, but the local neighborhood is defined in the decision space [23] rather than in the feature space. The decision space consists of the classifier output of each sample, where each classifier output vector is a template. The similarities between the output vectors are then compared. For example, the  $K$ -nearest output profile (KNOP) method first defines the local neighborhood of the test sample in the decision space, and then uses a method similar to that by KNORA-E to select the classifiers that correctly classified test samples in the

neighborhood in order to form an ensemble by voting [24]. The multiple classifier behavior (MCB) method also defines the neighborhood in the decision space, but the selection is determined based on a threshold. Classifiers larger than the given threshold are used for the ensemble [25]. Although such methods are defined in the decision space, they are still based on the local neighborhood of the test samples.

### 2.1.3 Selection of candidate classifiers

The composition of candidate classifiers is very important for a dynamic classifier selection method since it must be accurate and diverse. In addition to methods that generate candidate classifiers using common ensemble classifier methods, there are also methods that focus on selecting training subsets for each candidate classifier [26]. For example, the particle swarm method directly selects a training set for each candidate classifier using the evolutionary algorithm [27]. The reason why a candidate classifier is generated by adopting different training subsets in the ensemble classifier is that it is easy to generate a large number of candidate classifiers that are likely to be similar rather than different. There are some methods that use heterogeneous candidate classifiers to make maintaining diversity easier.

### 2.1.4 Machine learning methods

The recently proposed method for dynamic selection of classifiers is based on machine learning and uses the local neighborhood features (such as meta-features of the test samples, the classification accuracy of the neighborhood samples, and the posterior probability of classes of the classified test samples) as the training samples for machine learning [14]. In the other method, the genetic algorithm was applied to divide training sets into subsets, each of which is used to train a classifier. The fitness function was defined as the accuracy of each classifier combined with the complexity of each training set [28]. Unlike these two methods, the method proposed in this study directly assigned each training sample to the classifier based on the Bayesian theorem. That is, the classifier was used as the class label of the training sample so that there was no need to calculate the neighborhood of the test sample and the machine learning could be global.

From the literatures mentioned above, it is discovered that dynamic classifier selection has not yet been applied to emotion recognition. The SAP proposed in this study is also different from currently available methods. It directly selected the candidate classifier according to the posterior classification accuracy calculated based on the Bayesian theorem. The evolutionary method was not used, and meta-features were not calculated. Instead, the proposed method directly endowed the training samples with classifier labels so that there was no need to calculate the neighborhood of the test samples. Since

the learning was conducted throughout the training set, it was also global in nature.

## 2.2 Face images for facial expression recognition

When facial images are transformed into feature vectors, any single classifier can be used for expression recognition, such as support vector machines and neural networks. One of the differences among these methods is the application of facial image information. Expression recognition can be performed based on 2D static images, or expression recognition can be performed based on 3D or 4D images. Because of the sensitivity to illumination and head posture changes, the use of 2D static images is unstable. By contrast, facial expressions are the result of facial muscle movement, resulting in different facial deformations that can be accurately captured in geometric channels [29, 30]. In such cases, using 3D or 4D images are the trend because they enable use of more image information.

Previous 3D expression recognition methods focus on the geometric representation of a single face image [31–34]. Currently, 3D video expression recognition methods emphasize modeling dynamic deformation patterns through facial scanning sequences. For example, a heuristic deformable model for static and motion information of the video was constructed, and then the hidden Markov model (HMM) was applied to recognize expressions [35]. Another method extracted motion features between adjacent 3D facial frames, and then utilized HMM to perform facial expression recognition [36]. Temporal deformation clues of 3D face scanning can also be captured using dynamic local binary pattern (LBP) descriptors, and then an SVM can be applied to perform the expression recognition [37]. Another novel method is the conditional random forest, which aims to capture low-level expression transition patterns [38]. When testing on a video frame, pairs are created between this current frame and previous ones, and predictions for each previous frame are applied to draw trees from pairwise conditional random forests (PCRF). The pairwise outputs of PCRF are averaged over time to produce robust estimates. A more complex approach is to use a set of radial curves to represent the face, to quantify the set using Riemann-based shape analysis tools, and to then classify the facial expressions using LDA and HMM [39, 40]. There are also methods for facial expression recognition using 4D face data. For example, scattering operators are expanded on key 2D and 3D frames to generate text and geometric facial representations, and then multi-kernel learning is applied to combine different channels of facial expression recognition to obtain the final expression label [41, 42].

Deep learning has also been applied to recognize facial expressions [43]. For example, a novel deep neural network-driven feature learning method was proposed and applied to multi-view facial expression recognition [44]. The input of the network is scale invariant feature transform (SIFT) features

that correspond to a set of landmark points in each facial image. There is a simple method to recognize facial expressions that uses a combination of a convolutional neural network and specific image preprocessing steps [45]. It extracts only expression-specific features from a face image, and explores the presentation order of the samples during training. A more powerful facial feature method called deep peak–neutral difference has also been proposed [46]. This difference is defined as the difference between two deep representations of the fully expressive (peak) and neutral facial expression frames, where unsupervised clustering and semi-supervised classification methods automatically obtain the neutral and peak frames from the expression sequence. With the development of deep learning, some studies emphasize the modeling of dynamic shape information of facial expression motion, and then adopt end-to-end deep learning [41, 42, 47–49], where a 4D face image network for expression recognition uses a number of generated geometric images. A hybrid method uses a contour model to implement face detection, uses a wavelet transform-based method to extract facial expression features, and uses a robust nonlinear method for feature selection; finally, the HMM is used to perform facial expression recognition [50].

The SAP method is different from the above expression recognition methods. These methods are thus taken as candidate classifiers for SAP so as to further improve SAP's performance. This also allows SAP to easily exceed them.

### 2.3 Ensemble learning for facial expression recognition

Ensemble learning is also used for facial expression recognition, which can be implemented by data integration, feature integration, and decision integration. Data fusion refers to the fusion of facial, voice, and text information. For example, the fusion of video and audio is applied to recognize emotions [51]. Meanwhile, the combination of facial expression data and voice data is utilized to identify emotions [52]. Another approach combines thermal infrared images and visible light images, using both feature fusion and decision fusion [53]. This approach extracts the active shape model features of the visible light image and the statistical features of the thermal infrared image model, and then uses a Bayesian network and support vector machine to make respective decisions. Finally, these decisions are fused in the decision layer to obtain the final emotion label. There is an automatic expression recognition system that extracts the geometric features and regional LBP features, and fuses them with self-coding. Finally, a self-organizing mapping network is used to perform expression recognition [54]. When the face image is divided into several regions, and the features of each region are extracted using the LBP method, the evidence theory can be used to fuse these features [55]. Furthermore, the fusion of both Gabor features

and LBP features can be applied to recognize expressions [56]. Some methods also use SIFT and deep convolution neural networks to extract features, and then use neural networks to fuse these features [57]. The decision level integrates the final decision information of multiple learning models. Each learning model participates in the processes of preprocessing, feature extraction, and decision-making. The fusion layer makes the final inference by evaluating the reliability of each member's decision-making information. For example, Wen et al. fused multiple convolutional neural network models by predicting the probability of each expression class for the test sample [4]. Zavaschi et al. extracted Gabor features and LBP features for facial images, and then generated a number of SVM classifiers. Finally, some classifiers were selected by a multi-objective genetic algorithm, and the final expression label was obtained by integrating these selected classifiers [58]. Moreover, Wen et al. proposed an integrated convolutional echo state network and a hybrid ensemble learning approach for facial expression classification [10, 11].

The SAP method is different from these ensemble learning methods for emotion recognition. SAP dynamically selects a classifier from multiple classifiers for the test sample. When a large number of candidate classifiers are available, SAP is more likely to find the most suitable classifier for the test sample. These aforementioned ensemble learning methods can be taken as candidate classifiers for SAP so that SAP's performance can be further improved and easily exceed that of the existing ensemble learning methods.

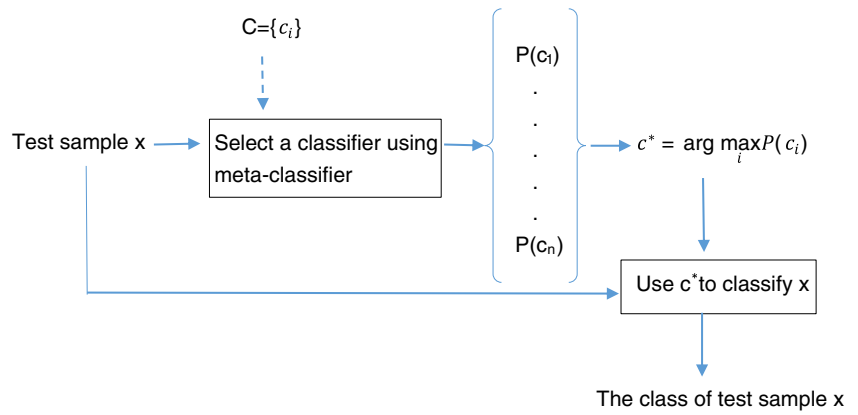
## 3 Proposed method

In the real world, different experts may have different abilities to identify the same sample. For example, it is justifiable to see the best doctor, but the “best doctor” is different for each disease. Similarly, each person wants to attend the best school, but different people have different definitions of the “best school.” Therefore, this study proposed the SAP method for facial expression recognition.

Figure 1 shows the structure of the method. The method differs from the ensemble method that averages all classifiers and weakens the strongest classifier so that it is theoretically inferior to the best classifier. SAP also differs from the model selection method that seeks the best classifier from all training samples rather than each individual sample. SAP considers each test sample to have its own optimal classifier because each expert has his own strengths.

The SAP method calculates the ability of each candidate classifier to classify each sample on the training set to find the most suitable classifier for each training sample based on the Bayesian theorem. Using this approach, a new training set,  $\Phi\{(x_i, c_i)\}$ ,  $c_i \in C$ , was constructed; that is, a label was assigned to each training sample as the optimal classifier by

**Fig. 1** Classification process of SAP



which to classify this sample. On this new training set, a new classifier was then trained to assign the most suitable classifier for each test sample.

**3.1 Labeling each sample with the classifier name**

$X = \{x_i | x_i \in \mathbb{R}^n\}$  is a training sample set,  $Y = \{y_i | y_i \in L\}$  is the corresponding label set, and  $L$  is the set of the labels of the samples. There is a classifier set  $C = \{c_i | c_i \in \mathbb{Z}\}$ , where classifier  $c \in C$  was used to classify sample  $x$  and calculate the probability that it would correctly classify  $x$ . The  $k$ -fold cross-validation method was applied to train the classifiers with some training samples, and then the classifiers were used to classify the test sample. If the test sample was classified correctly,  $P(x|c)$  could be easily calculated. The  $k$ -fold cross-validation method was used to divide the training set into subsets as follows:

$$X = X_1 \cup \dots \cup X_i \cup \dots \cup X_k, \tag{1}$$

$$X_i \cap X_j = \emptyset, |X_i| = |X_j|, \tag{2}$$

$$Y = Y_1 \cup \dots \cup Y_i \cup \dots \cup Y_k, \tag{3}$$

$$|X_i| = |Y_i|. \tag{4}$$

Suppose that the discriminant function of classifier  $c$  in the training set  $X_j$  is defined as  $g_{c, X_j \in X} : X_j \rightarrow Y_j$ . The prior probability of classifier  $c$  was calculated as follows. The higher the classification accuracy, the more likely it was to be selected as the optimal classifier:

$$P(c) = \frac{1}{k} \sum_{j=1}^k \sum_i^{|X_j|} 1_{g_{c, X_j \in X}(x_i) = y_i}. \tag{5}$$

The prior probability for classifier  $c$  to correctly classify  $x_i$  was calculated using the following equation:

$$P(x_i|c) = \frac{1}{k} \sum_j^k 1_{g_{c, X_j \in X}(x_i) = y_i}. \tag{6}$$

The goal was to calculate  $P(c|x)$ , which is the probability that each classifier will be selected based on the test sample. This allows us to select the most suitable classifier from the candidate classifier set to classify the test sample.

According to the Bayesian theorem, the following equation can be obtained:

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}. \tag{7}$$

This is similar to the assumption of the Naive Bayesian classifier,  $\text{all}p(x_i) = p(x_j)$ . According to the above formula, each training sample was labeled with the classifier name to construct a new training dataset. When the probability of the classifier chosen based on  $x$  is greater than a certain threshold,

$$D_i = \{(x, c_i) | P(c_i|x) > \delta_i, x \in X, c_i \in C\}, \tag{8}$$

$$S = \bigcup_{i=1}^{|C|} D_i. \tag{9}$$

The candidate classifiers were constructed by  $D$ :

$$D = \left\{ (x, c_i) | x \in S, c_i = \arg \max_i P(c_i|x) \right\} \tag{10}$$

Once the training sample set  $D$  was labeled with the classifier name, another classification algorithm,  $\varphi$ , was selected to be trained on this set so as to obtain a new classification function as follows:

$$h_{\varphi, D} : X \rightarrow 2^C, \tag{11}$$

$$c = \arg \max_i P(c_i \in h_{\varphi, D}(x)). \tag{12}$$

Given a test sample  $x$ , we selected a suitable classifier,  $c$ , to classify the test sample.

**3.2 SAP for emotion recognition**

Given the inputs of the training set  $X$ , the validation set  $X_v$ , the classifier set  $C = \{c_i\}$ , the threshold parameter  $\sigma$ , the test

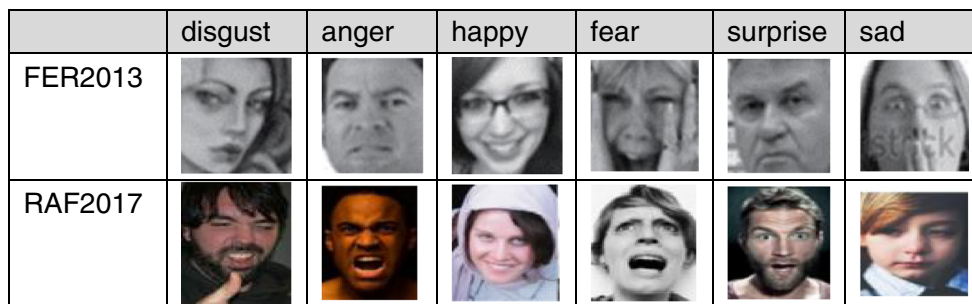


Fig. 2 Sample images from the experimental databases

sample  $x$ , as well as the output  $y$  (the label of the test sample), the SAP algorithm was described as follows:

1.  $|C|$  classifiers were trained on training set  $X$ .
  2. Training set  $X$  was divided into  $k$  groups using the  $k$ -fold cross-validation method.
  3. For  $j = 1$  to  $k$ :
    - (a) The  $j$ th fold of the training set was taken from the training set to train each classifier  $c$ .
    - (b) The classifier  $c$  was used to classify each sample  $x_i$  in the validation set  $X_v$ .
    - (c) The number of times that each sample in the validation set  $X_v$  was correctly classified in all folds was calculated, and then the probability  $p(x_i|c)$  was computed.
- End
4. The probability  $p(x_i|c)$  was normalized.
  5. The probability  $p(c|x_i)$  was calculated based on the Bayesian theorem so as to assign a classifier name to each training sample as the label.
  6. For  $i = 1$  to  $|C|$ :

$$D_i = \{(x, c_i) | P(c_i|x) > \sigma \ \& \ P(c_i|x) > P(c_j|x), x \in X_v\}.$$

End

7.  $S = \cup D_i$

8. The classification algorithm  $\varphi$  was used to train a meta-classifier  $h_{\varphi, D}: D \rightarrow 2^C$ .
9. The classifier  $c_i = \arg \max_i P(c_i \in h_{\varphi, D}(x))$  was selected.
10. The classifier  $c_i$  was used to classify the test samples  $x$  so as to obtain the class  $y$ .

### 3.3 Time complexity analysis

As in Step 3 of SAP training  $k \times |C|$  classifiers, which involved a complexity of  $k \times \max(O(c_i))$ , the other steps of SAP were linear. The greatest complexity of the algorithm laid in training or testing a classifier, and therefore the complexity of the entire algorithm was  $\max(O(c_i))$ . SAP spent the most time on training the classifiers using the  $k$ -fold cross-validation method. However, this calculation was only performed once during the training. The trained model was used to directly classify the test samples, and there was no need for a recalculation. Therefore, SAP was less complex than all dynamic algorithms based on the local neighborhoods.

## 4 Experimental results

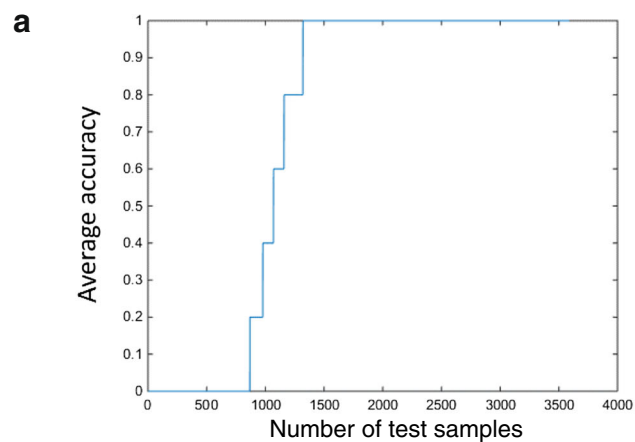
### 4.1 Objective

The effectiveness of the proposed method was demonstrated by conducting experiments on two standard datasets. In principle, there are many alternative classifiers for the

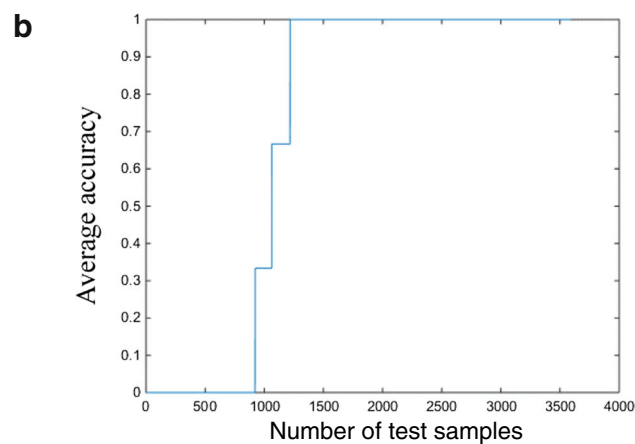
**Table 1** The distribution of samples in the two experimental databases

Emotions Databases	Angry	Disgust	Fear	Happy	Sad	Surprised	Neutral	Total
FER2013-TRAIN	3995	436	4097	7215	4830	3171	4965	28,709
FER2013-PUBLIC	467	56	496	895	653	415	607	3589
FER2013-TEST	491	55	528	879	594	416	626	3589
RAF2017-TRAIN	705	717	281	4772	1982	1290	2524	12,271
RAF2017-TEST	162	160	74	1185	478	329	680	3068

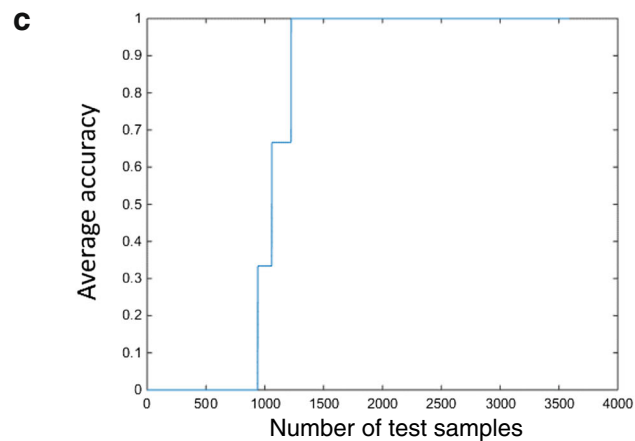
**Fig. 3** Distribution of test samples against the classification satisfiability



Candidate classifiers are SOFTMAX, SVM, LDA, QDA, and RF



Candidate classifiers are SOFTMAX, SVM, and RF



Candidate classifiers are SOFTMAX, SVM, and LDA

proposed method. However, in the experiments, the most representative methods were chosen, i.e., SOFTMAX [4, 59], SVM [60], LDA [60], QDA [60], and RF [61]. Since the SOFTMAX classifier is a widely used classifier for deep learning, SAP can be applied to deep learning with

the SOFTMAX classifier chosen. SVM is one of the best classifiers for small training samples. LDA and QDA are the simplest linear classifiers, whereas RF is the most representative ensemble classifier. For these candidate classifiers, default parameters were used in the experiments. The

LDA algorithm was used as the meta-classifier because it is simple and fast. In this way, two objectives will be obtained. One is to prove that the dynamic selection of classifiers is superior to the constant use of a single classifier. The other is to illustrate that the proposed method outperforms some ensemble algorithms.

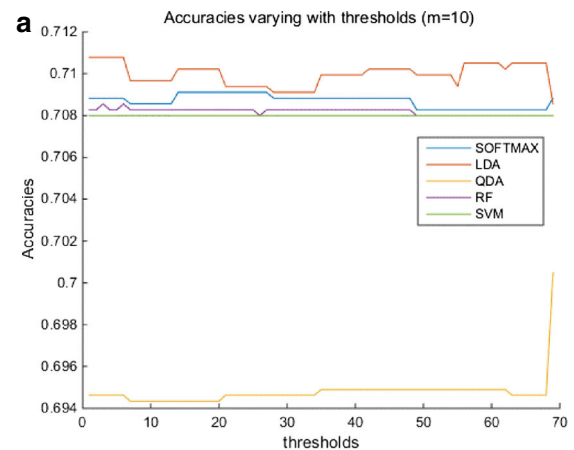
## 4.2 Experimental data

The deep neural network is currently the most effective approach for extracting the features of images, but it requires a large amount of training data. Therefore, FER2013 [62] and RAF [63] are selected as the experimental data. They are generally recognized as benchmark databases. Sample images from these databases are shown in Fig. 2.

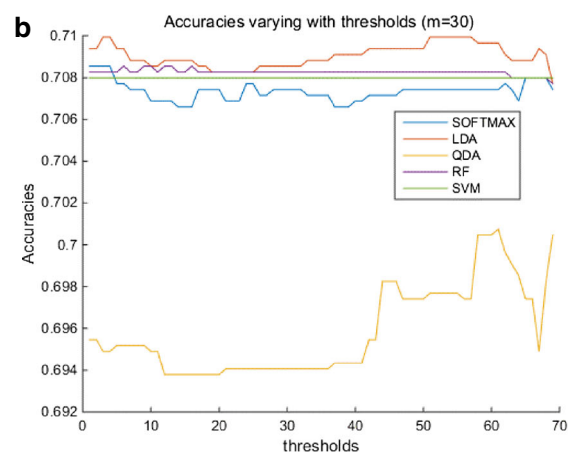
FER2013 has the larger amount of data and its images are the most difficult to distinguish. Each sample in the database has great differences in age, facial orientation, and so on. It is also closest to real world data, with the human emotion recognition rate in this database is  $65 \pm 5\%$ . At the same time, the images in the database are all gray-scale images with a size of  $48 \times 48$  pixels. The samples are divided into seven categories: anger, disgust, fear, happiness, neutral, sadness, and surprise. This database consists of three parts: FER2013-TRAIN for training a deep neural network, FER2013-PUBLIC as the validation set, and FER2013-TEST as the test set. Their sample distributions are shown in Table 1.

The Real-world Affective Faces Database (RAF 2017) was constructed by analyzing 1.2 million labels of 29,672 greatly diverse facial images downloaded from the Internet. Images in this database vary greatly in subject age, gender, ethnicity, head poses, lighting conditions, and occlusions. For example, the subjects in the database range in age from 0 to 70 years old. Fifty two percent are female, 43% are male, and 5% ambiguous; meanwhile, 77% are Caucasian, 8% are African-American, and 15% are Asian [62]. Therefore, it has large diversity across a total of 29,672 real-world images, with seven classes of basic emotions and 12 classes of compound emotions. To be able to objectively measure the performance for the following testing. In our experiments, the database with seven basic emotions is considered; these emotions are anger, disgust, fear, happiness, neutral, sadness, and surprise. This database is split into a training set RAF2017-TRAIN with 12,271 samples and a test set RAF2017-TEST with 3068 samples.

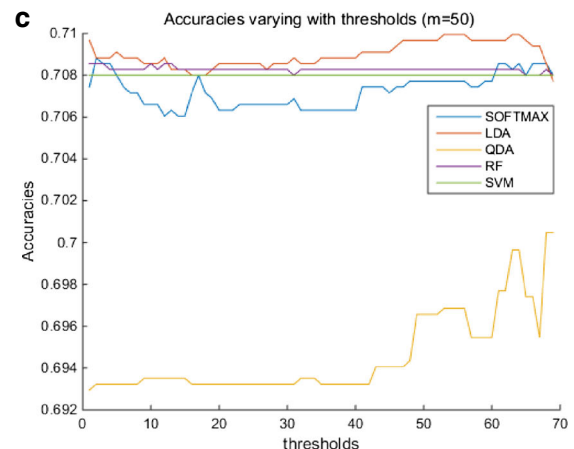
The features of all datasets were extracted using the deep neural network model [59]. Parameter analysis and time complexity analysis were performed on FER2013 since it is harder to be classified. In SAP, the  $j$ -th fold of training samples was



The average of ten test accuracies on the validation set ( $m = 10$ ), which varies with the threshold



The average of 30 test accuracies on the validation set ( $m = 30$ ), which varies with the threshold



The average of 50 test accuracies on the validation set ( $m = 50$ ), which varies with the threshold

**Fig. 4** Relationship between classification accuracy of SAP and the threshold  $\sigma$

taken from the training set to train the classifier, and FER2013-PUBLIC was taken as the validation set.



**Table 2** The number of samples assigned to each classifier with the optimal threshold ( $m = 10$ )

Meta-classifier	Number of samples assigned to be classified to each candidate classifier					Accuracy (%)
	SOFTMAX	LDA	QDA	RF	SVM	
SOFTMAX	50	15	42	36	3446	70.91
<b>LDA</b>	133	75	112	167	3102	<b>71.08</b>
QDA	562	0	0	3027	0	70.05
RF	2		1	2	3584	70.86
SVM	0	0	0	0	3589	70.80

Bold data indicates the best meta classifier with the best accuracy

### 4.3 Evaluation on complementarity among candidate classifiers

The key to SAP is the complementarity among the candidate classifiers. To objectively evaluate the complementarity among the candidate classifiers, the concept of classification satisfiability was proposed. The probability measure for any sample to be correctly classified is referred to as classification satisfiability, which can be calculated using the following equation:

$$\mu(x) = \frac{\sum_1^n f_i(x)}{n} \quad (13)$$

where  $n$  is the number of classifiers. If classifier  $f_i$  can correctly classify  $x$ , then  $f_i(x) = 1$ ; otherwise  $f_i(x) = 0$ . The greater the classification satisfiability, the more likely the sample is to be correctly classified.

Figure 3 shows the distribution of the classification satisfiability of the test samples for a given set of candidate classifiers, where FER2013 was used. The samples were ranked according to classification satisfiability from high to low. In Fig. 3a, when the candidate classifiers SOFTMAX, SVM, LDA, QDA, and RF were used, 868 samples were classified completely incorrectly, 2270 samples were correctly classified, and 451 samples were correctly classified by at least one classifier. Figure 3b shows that when the candidate classifiers SOFTMAX, SVM, and RF were used, 922 samples were classified completely incorrectly, 2371 samples were correctly classified, and 296 samples were correctly classified

by at least one classifier. Figure 3c illustrates that when the candidate classifiers SOFTMAX, SVM, and LDA were used, 939 samples were classified completely incorrectly, 2366 samples were correctly classified, and 284 samples were correctly classified by at least one classifier.

In Fig. 3, there were approximately 900 samples whose classification satisfiability was 0, indicating that these samples could not be correctly classified by any classifier. It was inevitable for them to be misclassified. This indicated that the candidate classifier set is incomplete and needs to be extended so as to reduce the occurrence of such situations. As shown in Fig. 3, the number of erroneously classified samples was different for different sets of candidate classifiers. Since there was a maximum number of candidate classifiers in Fig. 3, a minimum number of misclassified samples was expected. Moreover, the greater the number of candidate classifier sets, the greater the number of samples whose classification satisfiability was greater than zero. This indicates that some candidate classifiers can correctly classify these samples. In these cases, the accuracy of the meta-classifier is extremely important. Ideally, the meta-classifier should be able to select the candidate classifier that can correctly classify these samples.

### 4.4 Parameter performance analysis

Since the SAP algorithm used the machine learning method (meta-classifiers) to assign classifiers to test samples, the meta-classifiers needed to be trained by the samples

**Table 3** The number of samples assigned to each classifier with the optimal threshold ( $m = 30$ )

Meta-classifier	Number of samples assigned to be classified to each candidate classifier					Accuracy (%)
	SOFTMAX	LDA	QDA	RF	SVM	
SOFTMAX	58	31	42	156	3302	70.86
<b>LDA</b>	135	105	77	176	3096	<b>70.99</b>
QDA	35	357	98	0	3099	70.08
RF	1	1	1	2	3584	70.86
SVM	0	0	0	0	3589	70.80

Bold data indicates the best meta classifier with the best accuracy

**Table 4** The number of samples assigned to each classifier with the optimal threshold ( $m = 50$ )

Meta-classifier	Number of samples assigned to be classified to each candidate classifier					Accuracy (%)
	SOFTMAX	LDA	QDA	RF	SVM	
SOFTMAX	43	34	49	131	3332	70.88
<b>LDA</b>	110	124	98	54	3203	<b>70.99</b>
QDA	2	418	39	3130	0	70.05
RF	1	1	0	2	3585	70.86
SVM	0	0	0	0	3589	70.80

Bold data indicates the best meta classifier with the best accuracy

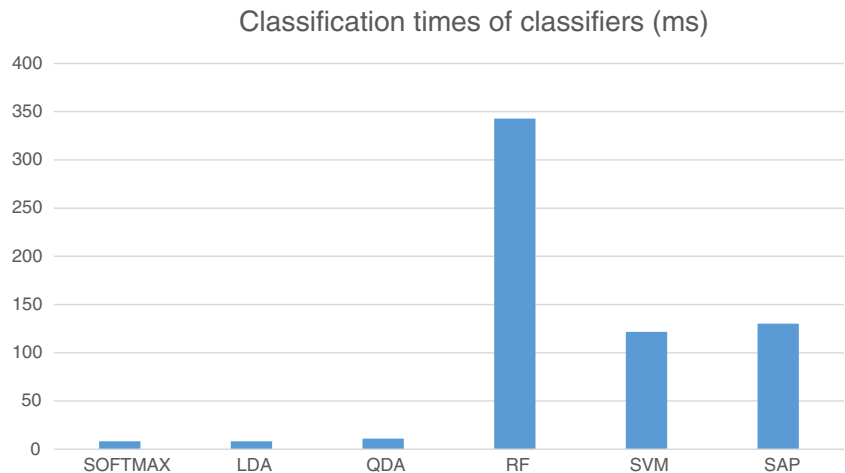
whose labels were candidate classifier names. The labels for these samples were automatically completed on the training and verification sets, and their classification satisfiability was found to be the average of the test accuracy on the cross-validation set. The greater the classification satisfiability, the more reliable the classifier name that was labeled on the test sample. Therefore, a classification satisfiability below the threshold may have been wrong and therefore should be removed from the training samples of the meta-classifier.

FER2013-TRAIN was divided into 100 pieces for cross-validation, 99 of which were used as the training set each time. FER2013-PUBLIC was used as the validation set, with the validation results taken out  $m$  times. For example,  $m = 10$  means that the validation results obtained for the first ten times were taken out, and then the average of the test accuracy on the validation set was calculated to obtain the classification satisfiability for each sample on the validation set. Based on the given threshold parameters, the samples in the validation set with values larger than the threshold were selected as the training samples of the meta-classifier. After the meta-classifier was trained, each test sample in FER2013-TEST would be assigned a candidate classifier.

The classification effect of SAP was related to  $m$  and the threshold  $\sigma$  of the classification satisfiability. The results in Fig. 4 demonstrate that different thresholds affected the classification accuracy of SAP. However, the range of the best results was relatively large and stable. This indicated that the optimal threshold  $\sigma$  could be easily obtained experimentally. Secondly, the optimal thresholds corresponding to different meta-classifiers were different. Although the classification accuracy of SAP varied with different values of  $m$ , its change with threshold  $\sigma$  was similar, which indicated that a relatively small  $m$  could be selected as the threshold parameter to reduce the time cost of the experiment.

Figure 4 also shows that the effectiveness of different meta-classifiers was different because the number of test samples assigned to each candidate classifier was different. As shown in Tables 2, 3 and 4, the more dispersed the assigned test samples, the more complementary they were and the more effective the classification. Additionally, the assignments were unbalanced. Effective candidate classifiers were in the majority. However, when all were assigned to the majority, the classification became ineffective. This behavior was associated with unbalanced data, which could be further improved with methods that are good at dealing with classification of unbalanced data.

**Fig. 5** Comparison of candidate classifiers and SAP in terms of classification time



**Table 5** Recognition rates of SAP and the candidate classifiers on the three test sets

Data	Candidate classifiers					Ensemble			SAP
	SOFTMAX	LDA	QDA	RF	SVM	Ens1	Ens2	Ens3	
FER2013	0.6996	0.6999	0.6949	0.6941	0.7080	0.7052	0.7035	0.7069	<b>0.7108</b>
RAF2017	0.8165	0.8132	0.8145	0.8136	0.8132	<b>0.8184</b>	0.8158	0.8171	<b>0.8181</b>

The bold entry shows that it is the best result in the compared methods

The experimental results show that LDA as the optimal meta-classifier was not only effective but also fast. In later experiments, only LDA was used as the meta-classifier. SVM as the meta-classifier led to the worst effect since it assigned all the test samples to itself.

#### 4.5 Time complexity analysis

When classifying the test samples, SAP first used a meta-classifier to assign a candidate classifier to each test sample, and then used the selected candidate classifier to classify the test sample, which added to the classification time. However, LDA was applied as meta-classifier in this study. Since it worked quickly, the time it added to classification was negligible. As shown in Fig. 5, it was much smaller than the maximum RF but larger than the minimum LDA and QDA. This is because SAP assigned many samples to SVM and RF, which thereby improved the emotion recognition accuracy. Among all the candidate classifiers, SVM had the highest accuracy; however, SAP was more accurate than SVM, and its classification time was only slightly bigger. Therefore, the comprehensive advantages of SAP are noteworthy.

#### 4.6 Comparison of standard datasets

SAP only selected the optimal classifier from the candidate classifiers. We addressed the question of whether it was better

**Table 6** Recognition results obtained by the selective ensemble methods on FER2013

Selective integration algorithm	Accuracy (%)
Kappa [64]	68.74
QSEP [65]	68.49
DFEP [65]	68.82
Inconsistent EP [65]	69.38
DREP [66]	70.05
Complementarity method [67]	68.82
OO [68]	70.52
MRMREP [59]	70.66
ECNN [4]	69.96
<b>SAP</b>	<b>71.08</b>

The bold entry shows that it is the best result in the compared methods

than the single and ensemble versions of these candidate classifiers. For FER2013, each method adopts FER2013-TRAIN as the training set and FER2013-TEST as the test set. For RAF2017, each method adopts RAF2017-TRAIN as the training set and RAF2017-TEST as the test set.

All the results are shown in Table 5, where Ens1 denotes the combination of SOFTMAX, LDA, QDA, RF, and SVM; Ens2 indicates the combination of SOFTMAX, RF, and SVM; and Ens3 denotes the combination of SOFTMAX, LDA, and SVM. It can be observed that SAP is better than both the ensemble classifier and single candidate classifier for the FER2013 database. The ensemble classifier is not better than the best candidate classifier SVM, but it is more stable. Besides, the ensemble method and selective ensemble method were relatively effective in emotion recognition; however, as shown as in Table 6, the SAP method was shown to be superior to some ensemble methods, where the accuracy rate of ensemble methods comes directly from the original literature. Due to different techniques used in ensemble methods, such as feature extraction, the comparison of effectiveness here should only be used as a reference.

On RAF2017, SAP still outperforms any single candidate classifier. However, it seems that SAP is slightly worse than Ensemble 1, which contains all candidate classifiers, but it works faster.

## 5 Conclusion

The SAP method proposed in this study is innovative because it adopts a global approach to dynamically selecting the optimal classifier for each test sample. It used the Bayesian theorem to calculate the posterior probability of each sample, and then labeled the candidate classifier name to each sample according to its posterior probability. As a global method, SAP can be used to avoid the effects of noise and to reduce the time it takes to search for local neighborhoods when classifying the test samples. The meta-classifier, which was linear, was shown to be efficient and fast.

Although SAP requires a large number of basic classifiers, it is different from ensemble learning. The ensemble classification method needs to run multiple classifiers simultaneously to classify the test samples, which makes their work comparatively slower. It is the same for all test samples. SAP selects

the classifier most suitable to classify a given test sample from the given basic classifiers. This is more consistent with human cognition laws. In experiments, SAP's effectiveness in emotion recognition was shown to be significantly better than that of any candidate classifier, and the same was nearly true for the recognition effect of the ensemble of these candidate classifiers. Secondly, SAP is different from the traditional model selection method. Model selection involves selecting a suitable model by testing on the training data, and then this model is used to classify all test samples. In the process of classification, this model is unchanged. SAP changes dynamically according to the test sample, and therefore has a personalized classification ability.

The key technique of SAP is that it requires a method to select a suitable classifier for any given test sample. This classifier is critical for ensuring the accuracy of SAP. At present, a linear classifier is selected. In the future, we will choose a more suitable classifier to finish this task, and nonlinear classifiers may be considered. Secondly, SAP depends on a large number of candidate classifiers being available. The more candidate classifiers available, the more suitable a classifier can be selected for the given test samples, thus leading to greater classification accuracy. In the future, more candidate classifiers will be considered, and these candidate classifiers should be diverse. Finally, the advantage of SAP is that it makes full use of global information, but the disadvantage is that it fails to utilize local information. In the future, we will consider both global and local information simultaneously so as to select a more accurate classifier to classify a given test sample. Therefore, the accuracy of SAP can be further improved.

**Acknowledgments** This study was supported by China National Science Foundation (Grant Nos. 60973083 and 61273363), Science and Technology Planning Project of Guangdong Province (Grant Nos. 2014A010103009 and 2015A020217002), and Guangzhou Science and Technology Planning Project (Grant No. 201504291154480, 201604020179, 201803010088).

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## References

- Zhang KH, Huang YZ, Du Y, Wang L (2017) Facial expression recognition based on deep evolutionary spatial-temporal networks. *IEEE Trans Image Process* 26(9):4193–4203
- Zeng NY, Zhang H, Song BY, Liu WB, Li YR, Dobaie AM (2018) Facial expression recognition via learning deep sparse autoencoders. *Neurocomputing* 273:643–649
- Choi I, Ahn H, Yoo J (2018) Facial expression classification using deep convolutional neural network. *J Electr Eng Technol* 13(1): 485–492
- Wen GH, Hou Z, Li HH, Li DY, Jiang LJ, Xun EY (2017) Ensemble of deep neural networks with probability-based fusion for facial expression recognition. *Cogn Comput* 9(5): 597–610
- Wen G, Wei J, Wang J, Zhou T, Chen L (2013) Cognitive gravitation model for classification on small noisy data. *Neurocomputing* 118:245–252
- Corcoran K, Hundhammer T, Mussweiler T (2009) A tool for thought! When comparative thinking reduces stereotyping effects. *J Exp Soc Psychol* 45:1008–1011
- Baruchello G (2015) A classification of classic, gestalt psychology and the tropes of rhetoric. *New Ideas Psychol* 26:10–24
- Smith MR, Martinez T, Giraud-Carrier C (2014) An instance level analysis of data complexity. *Mach Learn* 95:7225–7256
- Brun AL, AlceuS B Jr, Oliveira LS, Enembreck F, Sabourin R (2018) A framework for dynamic classifier selection oriented by the classification problem difficulty. *Pattern Recogn* 76:175–190
- Wen GH, Li HH, Li DY (2015) An ensemble convolutional echo state networks for facial expression recognition. In: 2015 International Conference on Affective Computing and Intelligent Interaction (ACII), Xian, China, pp 873–878
- Li D, Wen G, Hou Z, Huan E, Hu Y, Li H (2018) RTCRelief-F: an effective clustering and ordering-based ensemble pruning algorithm for facial expression recognition. *Knowl Inf Syst*:1–32
- Krawczyk B (2016) Dynamic classifier selection for one-class classification. *Knowl-Based Syst* 1307:43–53
- Britto AS Jr, Sabourin R, Oliveira LES (2014) Dynamic selection of classifiers—a comprehensive review. *Pattern Recogn* 47:3665–3680
- Cruz RMO, Sabourin R, Cavalcanti GDC, Ren TI (2015) META-DES: a dynamic ensemble selection framework using META-learning. *Pattern Recogn* 48:1925–1935
- Ko AHR, Sabourin R, Britto Jr AS (2008) From dynamic classifier selection to dynamic ensemble selection. *Pattern Recogn* 41:1735–1748
- Kuncheva L (2002) Switching between selection and fusion in combining classifiers: an experiment. *IEEE Trans Syst Man Cybern* 32(2):146–156
- Mendialdua I, Martínez-Otzeta JM, Rodríguez-Rodríguez I, Ruiz-Vázquez T, Sierra B (2015) Dynamic selection of the best base classifier in one versus one. *Knowl-Based Syst* 85:298–310
- Didaci L, Giacinto G, Roli F, Marcialis GL (2005) A study on the performances of dynamic classifier selection based on local accuracy estimation. *Pattern Recogn* 38(11):2188–2191
- Sabourin M, Mitiche A, Thomas D, Nagy G (1993) Classifier combination for handprinted digit recognition. In: Second International Conference on Document Analysis and Recognition, pp 163–166
- Giacinto G, Roli F (1999) Methods for dynamic classifier selection. In: 10th International Conference on Image Analysis and Processing, pp 659–664
- Ko AHR, Sabourin R, Britto AS Jr (2008) From dynamic classifier selection to dynamic ensemble selection. *Pattern Recogn* 41:1735–1748
- Xiao J, Xie L, He C, Jiang X (2012) Dynamic classifier ensemble model for customer classification with imbalanced class distribution. *Expert Syst Appl* 39:3668–3675

23. Kuncheva LI, Bezdek JC, Duin RPW (2001) Decision templates for multiple classifier fusion: an experimental comparison. *Pattern Recogn* 34:299–314
24. Cavalin PR, Sabourin R, Suen CY (2012) Logid: an adaptive framework combining local and global incremental learning for dynamic selection of ensembles of HMMs. *Pattern Recogn* 45(9): 3544–3556
25. Giacinto G, Roli F (2001) Dynamic classifier selection based on multiple classifier behavior. *Pattern Recogn* 34:1879–1881
26. Szepeannek G, Bischl B, Weihs C (2009) On the combination of locally optimal pairwise classifiers. *Eng Appl Artif Intell* 22:79–85
27. de Souza BF, de Carvalho A, Calvo R, Ishii RP (2006) Multiclass svm model selection using particle swarm optimization. In: Sixth International Conference on Hybrid Intelligent Systems, IEEE, p 31
28. Brun AL, AlceuS B Jr, Oliveira LS, Enembreck F, Sabourin R (2018) A framework for dynamic classifier selection oriented by the classification problem difficulty. *Pattern Recogn* 76:175–190
29. Fang T, Zhao X, Ocegueda O, Shah SK, Kakadiaris IA (2011) 3D facial expression recognition: a perspective on promises and challenges. In: IEEE International Conference on Automatic Face and Gesture Recognition, vol 28, pp 603–610
30. Zhen Q, Huang D, Wang Y, Chen L (2016) Muscular movement model-based automatic 3D/4D facial expression recognition. *IEEE Trans Multimedia* 18(7):1438–1450
31. Zhao X, Huang D, Dellandra E, Chen L (2010) Automatic 3D facial expression recognition based on a Bayesian belief net and a statistical facial feature model. In: IEEE/IAPR International Conference on Pattern Recognition
32. Li H, Chen L, Huang D, Wang Y, Morvan J-M (2012) 3D facial expression recognition via multiple kernel learning of multi-scale local Normal patterns. In: IEEE/IAPR International Conference on Pattern Recognition
33. Zhen Q, Huang D, Wang Y, Chen L (2015) Muscular movement model based automatic 3D facial expression recognition. In: International conference on MultiMedia Modeling
34. Li H, Ding H, Huang D, Wang Y, Zhao X, Morvan J-M, Chen L (2015) An efficient multimodal 2D + 3D feature-based approach to automatic facial expression recognition. *Comput Vis Image Underst* 140:83–92
35. Yin L, Chen X, Sun Y, Worm T, Reale M (2008) A high-resolution 3D dynamic facial expression database. In: IEEE International Conference on Automatic Face and Gesture Recognition
36. Sandbach G, Zafeiriou S, Pantic M, Rueckert D (2012) Recognition of 3D facial expression dynamics. *Image Vis Comput* 30(10):762–773
37. Fang T, Zhao X, Shah SK, Kakadiaris IA (2011) 4D facial expression recognition. In: IEEE International Conference on Computer Vision Workshops, pp 1594–1601
38. Dapogny A, Bailly K, Dubuisson S (2017) Dynamic pose-robust facial expression recognition by multi-view pairwise conditional random forests. *IEEE Trans on Affect Comput* 99:1–14
39. Drira H, Ben Amor B, Daoudi M, Srivastava A, Berretti S (2012) 3D dynamic expression recognition based on a novel deformation vector field and random forest. In: IEEE International Conference on Pattern Recognition, pp 1104–1107
40. Ben Amor B, Drira H, Berretti S, Daoudi M, Srivastava A (2017) 4D facial expression recognition by learning geometric deformations. *IEEE Trans Cybern* 44(12):2443–2457
41. Yao Y, Huang D, Yang X, Wang Y, Chen L (2018) Texture and geometry scattering representation based facial expression recognition in 2D+3D videos. In: ACM Transactions on Multimedia Computing and Applications
42. Joan B, Stephane M (2013) Invariant scattering nonvolution networks. *IEEE Trans Pattern Anal Mach Intell* 35(8):1872–1886
43. Ding H, Zhou SK, Chellappa R (2017) Facenet2expnet: regularizing a deep face recognition net for expression recognition. In: 12th IEEE International Conference on Automatic Face & Gesture Recognition, pp 118–126
44. Zhang T, Zheng W, Cui Z, Zong Y, Yan J (2016) A deep neural network-driven feature learning method for multi-view facial expression recognition. *IEEE Trans Multimedia* 18(12):2528–2536
45. Lopes AT, Aguiar ED, Souza AFD, 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
46. Chen J, Ruyi X, Liu L (2018) Deep peak-neutral difference feature for facial expression recognition. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-018-5909-5>
47. Yang X, Huang D, Wang Y, Chen L (2015) Automatic 3D facial expression recognition using geometric scattering representation. In: IEEE International Conference on Automatic Face and Gesture Recognition
48. Liu Y, Zeng J, Shan S, Zheng Z (2018) Multi-channel pose-aware convolution neural networks for multi-view facial expression recognition. In: 13th IEEE International Conference on Automatic Face & Gesture Recognition
49. Li W, Huang D, Li H, Wang Y (2018) Automatic 4D facial expression recognition using dynamic geometrical image network. In: 13th IEEE International Conference on Automatic Face & Gesture Recognition
50. Siddiqi MH (September 2018) Accurate and robust facial expression recognition system using real-time YouTube-based datasets. *Appl Intell* 48(9):2912–2929
51. Xu C, Du PF, Feng ZY, Meng ZP, Cao TY, Dong CC (2013) Multimodal emotion recognition fusing video and audio. *Appl Math Inform Sci* 7(2):455–462
52. Wang Y, Yang X, Zou J (2013) Research of emotion recognition based on speech and facial expression. *Institute of Advanced Engineering & Science* 11(1):83–90
53. Wang SF, He S, Wu Y, He MH, Ji Q (2014) Fusion of visible and thermal images for facial expression recognition. *Front Comput Sci-Chi* 8(2):232–242
54. Majumder A, Behera L, Subramanian VK (2018) Automatic facial expression recognition system using deep network-based data fusion. *IEEE Trans Cybern* 48(1):103–114
55. Wang WC, Chang FL, Liu YL, Wu XJ (2017) Expression recognition method based on evidence theory and local texture. *Multimed Tools Appl* 76(5):7365–7379
56. Sun YC, Yu J (2017) Facial expression recognition by fusing Gabor and local binary pattern features. In: International Conference on Multimedia Modeling, MMM, vol 10133. Springer, Cham, pp 209–220
57. Sun B, Li LD, Zhou GY, He J (2016) Facial expression recognition in the wild based on multimodal texture features. *J Electron Imaging* 25(6):061407
58. Zavaschi THH, Britto AS, Oliveira LES, Koerich AL (2013) Fusion of feature sets and classifiers for facial expression recognition. *Expert Syst Appl* 40(2):646–655
59. Li D, Wen G (2017) MRMR-based ensemble pruning for facial expression recognition. *Multimed Tools Appl* 10:1–22
60. Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning: data mining, inference, and prediction, 2nd edn. Springer, Berlin
61. Ho TK (1998) The random subspace method for constructing decision forests. *IEEE Trans Pattern Anal Mach Intell* 20(8):832–844
62. Goodfellow LJ, Erhan D, Carrier PL, Courville A, Mirza M, Hamner B, Cukierski W, Tang YC, Thaler D, Lee DH (2015)

- Challenges in representation learning: a report on three machine learning contests. *Neural Netw* 64:59–63
63. Li S, Deng W, Junping D (2017) Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild, CVPR
  64. Kuncheva LI (2013) A bound on kappa-error diagrams for analysis of classifier ensembles. *IEEE Trans Knowl Data Eng* 25(3):494–501
  65. Kunchava LI, Whitaker CJ (2003) Measures of diversity in classifier ensemble and their relationship with the ensemble accuracy. *Mach Learn* 51(2):181–207
  66. Li N, Yu Y, Zhou ZH (2012) Diversity regularized ensemble pruning. In: *Machine Learning and Knowledge Discovery in Databases, Proceedings of the European Conference (ECML PKDD 2012)*. Springer Verlag, Bristol, pp 330–345
  67. Dai Q, Han XM (2016) An efficient ordering-based ensemble pruning algorithm via dynamic programming. *Appl Intell* 44(4):816–830
  68. Oleg O Giorgio V (2009) *Applications of supervised and unsupervised ensemble methods [M]*. Springer Berlin Heidelberg

**Huihui Li** received the M.S degree in South China University of Technology. She is currently working towards the Ph. D degree from the Department of Computer Science and Technology of South China University of Technology. Her research area includes facial expression recognition, artificial intelligence and Machine Learning in Traditional Chinese Medicine.