

## Research Article

# Weighted Feature Gaussian Kernel SVM for Emotion Recognition

**Wei Wei and Qingxuan Jia**

*School of Automation, Beijing University of Posts and Telecommunications, Beijing 100876, China*

Correspondence should be addressed to Wei Wei; [wei\\_wei@bupt.edu.cn](mailto:wei_wei@bupt.edu.cn)

Received 26 June 2016; Revised 14 August 2016; Accepted 14 September 2016

Academic Editor: Francesco Camastra

Copyright © 2016 W. Wei and Q. Jia. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Emotion recognition with weighted feature based on facial expression is a challenging research topic and has attracted great attention in the past few years. This paper presents a novel method, utilizing subregion recognition rate to weight kernel function. First, we divide the facial expression image into some uniform subregions and calculate corresponding recognition rate and weight. Then, we get a weighted feature Gaussian kernel function and construct a classifier based on Support Vector Machine (SVM). At last, the experimental results suggest that the approach based on weighted feature Gaussian kernel function has good performance on the correct rate in emotion recognition. The experiments on the extended Cohn-Kanade (CK+) dataset show that our method has achieved encouraging recognition results compared to the state-of-the-art methods.

## 1. Introduction

Emotion recognition has necessary applications in the real world. Its applications include but are not limited to artificial intelligence and human computer interaction. It remains a challenging and attractive topic. There are many methods which have been proposed for handling problems in emotion recognition. Speech [1, 2], physiological [3–5], and visual signals have been explored for emotion recognition. Speech signals are discontinuous signals, since they can be captured only when people are talking. Acquisition of physiological signal needs some special physiological sensors. Visual signal is the best choice for emotion recognition based on the above reasons. Although the visual information provided is useful, there are challenges regarding how to utilize this information reliably and robustly. According to Albert Mehrabian's 7%–38%–55% rule, facial expression is an important mean of detecting emotions [6].

Further studies have been carried out on emotion recognition problems in facial expression images during the last decade [7, 8]. Given a facial expression image, estimate the correct emotional state, such as anger, happiness, sadness, and surprise. The general process has two steps: feature extraction and classification. For feature extraction, geometric feature, texture feature, motion feature, and statistical feature are in

common use. For classification, methods based on machine learning algorithm are frequently used. According to speciality of features, applying weighted features to machine learning algorithm has become an active research topic.

In recent years, emotion recognition with weighted feature based on facial expression has become a new research topic and received more and more attention [9, 10]. The aim is to estimate emotion type from a facial expression image captured during physical facial expression process of a subject. But the emotion features captured from the facial expression image are strongly linked to not the whole face but some specific regions in the face. For instance, features of eyebrow, eye, nose, and mouth areas are closely related to facial expression [11]. Besides, the effect of each feature on recognition result is different. In order to make the best of feature, using feature weighting technique can further enhance recognition performance. While there are several approaches of confirming weight, it remains an open issue on how to select feature and calculate corresponding weight effectively.

In this paper, a new emotion recognition method based on weighted feature facial expression is presented. It is motivated by the fact that emotion can be described by facial expression and each facial expression feature has different impact on recognition results. Different from previous works

by calculating weight of each feature directly, this method considers impact of feature by calculating subrecognition rate. Our method consists of two stages: weight calculation stage and recognition stage. In the weight calculation stage, we first divide face into 4 areas according to degree of facial behavior changes. Then, we use each area's features to calculate corresponding recognition rate. At last, we calculate weight of each area's features according to magnitude of recognition rate. In the recognition stage, we first use the above weight results to calculate weighted kernel function. Then, we obtain a new recognition model based on SVM with weighted kernel function.

For the proposed method, there are three main contributions and differences compared to the preliminary work. (1) A more advanced weight of feature method is used. In previous method, the weight of each feature was calculated individually without practical verification. To overcome this shortage, we group features and calculate corresponding subrecognition rate. Then we calculate weight of feature groups based on their respective subrecognition rate. (2) In the recognition stage, the previous method used the weight of features directly. In this paper, we use weight of feature groups to weight kernel function. Then we use new weighted kernel function in machine learning model. (3) The proposed method has been evaluated in a database which contains 7 kinds of emotions. Moreover, comparison results have been carefully analyzed and studied on whether to use weighted kernel function. The rest of the paper is organized as follows: Section 2 gives an overview of related works on feature extraction of facial expression, calculation of weight of feature, and classification of emotion. Section 3 describes the theorem in proposed method and proofs. Section 4 verifies the proposed method by experiment and analyzes experimental results. Section 5 concludes the paper.

## 2. Related Work

The recognition performance of motion based methods is highly dependent on the feature extraction methods. Many novel approaches have been proposed for feature extraction based on facial expression. They can be broadly classified into two categories: appearance-based methods and geometric-based methods. The appearance-based methods extract intensity or other texture features from facial expression images. The common methods of feature extraction include Local Binary Patterns (LBP) [12, 13], Histogram of Oriented Gradient (HOG) [14, 15], Gabor Wavelet [16, 17], and Scale-Invariant Feature Transform (SIFT) [18, 19]. These features can be used to extract Action Unit (AU) feature and recognize facial expression. The geometric-based methods describe facial component shapes based on key points of facial detected on images, such as eyebrows, eyes, nose, mouth, and contour line. The movement of these key points can be used for guiding the facial expression recognition process. For instance, the active appearance model (AAM) [20] or Active Shape Model (ASM) [21, 22] and the constrained local model (CLM) [23] are widely used to detect and trace these key points of face to record their displacement. However, the location accuracy of both ASM and AAM relies

on their geometric face models. And the model training phases sometimes need manual works and are usually time-consuming.

The recognition results obtained by classification algorithm are affected by all features. So the introduction of weight can distinguish the contribution of different features and improve classification performance. A variety of methods have been proposed to calculate the weight of every feature. Reference [24] presented Euclidean metric in the criterion extended to Minkowski metric to calculate weight of each feature directly. Some methods divided the facial image into some uniform subregions and calculated the weight of each subregion. Reference [25] introduced information entropy to distinguish the contribution of different partitions of the face. Reference [26] estimated the weight of each subregion by employing the local variance. For feature weighting in different ways, feature selection and weight calculation might be recognized as a latent problem. One effective method to solve this problem is to perform feature weighting based on the obtained feedback. Some methods [27, 28] divided the facial image into some uniform subregions and returned the subregion result for feature weighting. There is no restriction on each feature, which provides freedom on how the feature representations are structured.

Many machine learning methods have been proposed to classify facial expressions, such as SVM [29], Random Forest (RF) [30], Neural Network (NN) [31], and  $K$  nearest neighbor (KNN) [32]. Reference [33] presented the performance of RF and SVM in classification of facial recognition. Reference [34] used boosting technique for the construction of NNEs and the final prediction is made by Naive Bayes (NB) classifier. Reference [35] divided the region into different types and combined the characteristic of the Fuzzy Support Vector Machine (FSVM) with KNN, switching the classification methods to the different types. The studies show that these methods are extremely suitable for facial expression classification.

## 3. Support Vector Machine

*3.1. Linear Support Vector Machines.* SVM is a new supervised learning model with associated learning algorithm for classification problem of data whose ultimate aim is to find the optimal separating hyperplane. The mathematical model of SVM is shown below.

Given a training set  $\{y_i, \bar{x}_i\}_{i=1}^l$ , where  $\bar{x}_i \in R^n$  is input and  $y_i \in \{-1, +1\}$  is the corresponding output, if there is a hyperplane which can divide all the points  $\bar{x}_i$  into two groups correctly, we aim to find the "maximum-margin hyperplane" where the distance between the hyperplane and the nearest point  $\bar{x}_i$  from either group is maximized. By introducing the penalty parameter  $c > 0$  and the slack variable  $\bar{\xi} = (\xi_1, \xi_2, \dots, \xi_l)$ , the optimal hyperplane can be obtained by solving constraint optimization problem as follows:

$$\begin{aligned} \min \quad & \frac{1}{2} \|\bar{\omega}\|^2 + C \sum_{i=1}^l \xi_i \\ \text{s.t.} \quad & y_i [(\bar{\omega} \cdot \bar{x}_i) + b] \geq 1, \quad i = 1, 2, \dots, l \\ & \xi_i \geq 0, \quad i = 1, 2, \dots, l. \end{aligned} \quad (1)$$

Based on Lagrangian multiplier method, the problem is converted into a dual problem as follows:

$$\begin{aligned} \max \quad & \sum_{i=1}^l a_i - \frac{1}{2} \sum_{i,j=1}^l a_i a_j y_i y_j (\vec{x}_i \cdot \vec{x}_j) \\ \text{s.t.} \quad & \sum_{i=1}^l a_i y_i = 0, \\ & 0 \leq a_i \leq C, \quad i = 1, 2, \dots, l, \end{aligned} \quad (2)$$

where  $a_i > 0$  are the Lagrange multipliers of samples  $\vec{x}_i$ . Only a few  $a_i > 0$  are solutions of the problem of removing the parts of  $a_i = 0$ , so that we can get the classification decision function as follows:

$$f(x) = \text{sign} \left[ \sum_{i=1}^l a_i y_i (\vec{x}_i \cdot \vec{x}) + b \right]. \quad (3)$$

**3.2. Nonlinear Support Vector Machines.** For the linearly nonseparable problem, we first map the data to some other high-dimensional space  $H$ , using a nonlinear mapping which we call  $\Phi$ . Then we use linear model to achieve classification in new space  $H$ . Through defined "kernel function"  $k$ , (2) is converted as follows:

$$\begin{aligned} \max \quad & \sum_{i=1}^l a_i - \frac{1}{2} \sum_{i,j=1}^l a_i a_j y_i y_j k(\vec{x}_i \cdot \vec{x}_j) \\ \text{s.t.} \quad & \sum_{i=1}^l a_i y_i = 0, \\ & 0 \leq a_i \leq C, \quad i = 1, 2, \dots, l. \end{aligned} \quad (4)$$

And the corresponding classification decision function is converted as follows:

$$f(x) = \text{sign} \left[ \sum_{i=1}^l a_i y_i k(\vec{x}_i \cdot \vec{x}) + b \right]. \quad (5)$$

The selection of kernel function aims to take the place of inner product of basis function. The ordinary kernel functions investigated for linearly nonseparable problems are as follows:

(1)  $n$ th-degree polynomial kernel function

$$k(\vec{x}_i, \vec{x}_j) = (\vec{x}_i \cdot \vec{x}_j + 1)^d, \quad d = 1, 2, \dots \quad (6)$$

(2) (Gaussian) radial basis kernel function

$$k(\vec{x}_i, \vec{x}_j) = \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|^2) \quad (7)$$

(3) Sigmoid kernel function

$$k(\vec{x}_i, \vec{x}_j) = \tanh [b(\vec{x}_i \cdot \vec{x}_j) + c], \quad b > 0, c < 0. \quad (8)$$

**3.3. Weighted Feature SVM.** Weighted feature SVM is based on weighted kernel function of SVM, which is defined as Definition 1.

*Definition 1.* Let  $k$  be a kernel function defined in  $X * X$ ,  $X \subseteq R^n$ .  $P$  is a linear transformation square matrix of order  $n$  of given input space, where  $n$  is dimensionality of input space. Weighted feature kernel function  $k_P$  is defined as

$$k_P(\vec{x}_i, \vec{x}_j) = k(\vec{x}_i^T P, \vec{x}_j^T P), \quad (9)$$

where  $P$  is referred to as a weighted feature matrix. The different choices for  $P$  lead to different weight situation:

- (1)  $P$  is an identity matrix of order  $n$ , which is no weight situation.
- (2)  $P$  is a diagonal matrix of order  $n$ , where  $(P)_{ii} = \omega_i$  ( $1 \leq i \leq n$ ) is the weight of  $i$ th feature and not all  $\omega_i$  are equal to the others

$$P = \begin{bmatrix} \omega_1 & & & \\ & \omega_2 & & \\ & & \ddots & \\ & & & \omega_n \end{bmatrix}. \quad (10)$$

- (3)  $P$  is an arbitrary matrix of order  $n$ , which is full weight situation

$$P = \begin{bmatrix} \omega_{11} & \omega_{12} & \cdots & \omega_{1n} \\ \omega_{21} & \omega_{22} & \cdots & \omega_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_{n1} & \omega_{n2} & \cdots & \omega_{nn} \end{bmatrix}. \quad (11)$$

We only consider  $P$  is a diagonal matrix of order  $n$  in this paper.

*Definition 2.* The ordinary weighted feature kernel function can be got by (9), and the process is shown as follows:

(1) Weighted feature polynomial kernel function

$$k_P(\vec{x}_i, \vec{x}_j) = (\vec{x}_i^T P \cdot \vec{x}_j^T P + 1)^d = (\vec{x}_i^T P P^T \vec{x}_j^T + 1)^d, \quad (12)$$

$$d = 1, 2, \dots$$

(2) Weighted feature (Gaussian) radial basis kernel function

$$\begin{aligned} k_P(\vec{x}_i, \vec{x}_j) &= \exp(-\gamma \|\vec{x}_i^T P - \vec{x}_j^T P\|^2) \\ &= \exp(-\gamma ((\vec{x}_i - \vec{x}_j)^T P P^T (\vec{x}_i - \vec{x}_j))) \end{aligned} \quad (13)$$

(3) Weighted feature sigmoid kernel function

$$\begin{aligned} k_P(\vec{x}_i, \vec{x}_j) &= \tanh [b(\vec{x}_i^T P \cdot \vec{x}_j^T P) + c] \\ &= \tanh [b(\vec{x}_i^T P P^T \vec{x}_j^T) + c], \end{aligned} \quad (14)$$

$$b > 0, c < 0.$$

The motivation for introducing kernel function is to search nonlinear model in the new feature space which is obtained by using nonlinear mapping. Matrix  $P$  appears not to be related to the motivation, since it acts as linear mapping. However, it can be useful in practice, because it can change geometry shape of input space and feature space, thereby changing the weight of different functions in the feature space. And the weighted feature Gaussian basis kernel function is still a nonlinear model after using linear transformation. The conclusion can be proved by Theorem 3.

**Theorem 3.** *Let  $k$  be a kernel function defined in  $X * X$ ,  $X \subseteq \mathbb{R}^n$ .  $\phi : X \rightarrow F$  is a mapping from input space to feature space.  $P$  is a linear transformation square matrix and  $\tilde{x}_i = x_i^T P$ . Then it deduces  $\|\phi(\tilde{x}_i - \tilde{x}_j)\| \neq \|\phi(x_i - x_j)\|$ .*

*Proof.*  $k(\tilde{x}, \tilde{x}) = 1, \forall \tilde{x}$  acts as any of the three ordinary kernel functions in Definition 1; then it deduces

$$\begin{aligned}
& \|\phi(\tilde{x}_i) - \phi(\tilde{x}_j)\| \\
&= \sqrt{(\phi(\tilde{x}_i) - \phi(\tilde{x}_j)) \cdot (\phi(\tilde{x}_i) - \phi(\tilde{x}_j))} \\
&= \sqrt{\phi(\tilde{x}_i) \cdot \phi(\tilde{x}_i) - 2\phi(\tilde{x}_i) \cdot \phi(\tilde{x}_j) + \phi(\tilde{x}_j) \cdot \phi(\tilde{x}_j)} \\
&= \sqrt{k(\tilde{x}_i, \tilde{x}_i) - 2k(\tilde{x}_i, \tilde{x}_j) + k(\tilde{x}_j, \tilde{x}_j)} \\
&= \sqrt{2 - 2k(\tilde{x}_i, \tilde{x}_j)} = \sqrt{2 - 2k(x_i^T P, x_j^T P)} \\
&\neq \sqrt{2 - 2k(x_i, x_j)} = \|\phi(x_i) - \phi(x_j)\|.
\end{aligned} \tag{15}$$

□

**Theorem 4.** *When there is  $\omega_k = 0$  ( $1 \leq k \leq n$ ),  $k$ th feature of sample data is irrelevant to calculation of kernel functions and output of classifier. Furthermore, the smaller the value of  $\omega_k$  ( $1 \leq k \leq n$ ), the less the effect of calculation of kernel functions and output of classifier.*

*Proof.* From definition of weighted feature kernel function and classification decision function (5), the conclusion is straightforward. □

Theorem 3 indicates that changes of location relation between spot and spot lead to changes of geometry shape of feature space after linear transformation. And there may be better linear separating hyperplane in new feature space to improve the classification performance of SVM. Theorem 4 indicates that weighted kernel function can reduce the effects of weak correlation and no-correlative features, and we are looking forward to better classification results. The experiment results in the following section of this article demonstrate this conclusion.

**3.4. Weight Estimation of Features.** Feature weighting technique based on certain principle gives a weight to various data features where calculating  $\tilde{\omega}$  is the key element. The changes in facial expression lead to slight different instant

changes in individual facial muscles in facial appearance. According to motion range of facial muscles, the whole face can be divided into three kinds of regions: rigid region (nose), semirigid region (eyes, forehead, and cheek), and on-rigid region (mouth). According to the principles above, we divide face into several areas and find out recognition rate  $p_i$  ( $1 \leq i \leq n$ ) of all the areas where the higher the recognition rates, the greater the influences. Otherwise, the lower the recognition rate, the smaller the influences. Regard weight determination as the base for calculating the value of weight, and the calculation formula is presented as follows:

$$\omega_i = \frac{p_i}{p_1 + p_2 + p_3 + p_4}. \tag{16}$$

This approach makes  $\omega_1 + \omega_2 + \omega_3 + \omega_4 = 1$ .

The area of the highest value of weight has the highest differentiation in the face, although it is also the largest contributor to classification results. Therefore, the higher the value of weight as a correlation measurement index, the stronger the correlation. The four constructing steps of weighted feature SVM are as follows:

- (1) Collecting origin facial expression image dataset  $O$  and extracting feature dataset  $S = \{d, \tilde{f}\}$ , where  $\tilde{f} = (\tilde{f}^1, \tilde{f}^2, \dots, \tilde{f}^n)$  is feature vector of facial expression,  $\tilde{f}^i$  ( $1 \leq i \leq n$ ) is the feature vector of  $i$ th region of face, and  $d$  is the corresponding class label of facial expression
- (2) Calculating recognition rates  $p_i$  ( $1 \leq i \leq n$ ) and corresponding value of weight  $\omega_i$  of each area. Constructing feature weight vector  $\tilde{\omega}$  and linear transformation square matrix  $P$ , where  $P = \text{diag}(\tilde{\omega})$
- (3) Replacing standard kernel function formulas with weighted feature kernel function formulas (12)–(14), and constructing a classifier based on sample dataset  $S$
- (4) Evaluating the performance of achieved classifier

## 4. Experiment

The experiments on the extended Cohn-Kanade (CK+) dataset show the effectiveness of the proposed method. In our experiments, we use python programs based on LIBSVM software packages, and the platform of data processing is a computer with Windows 7, Intel® Core™ i3-2120 CPU (3.30 GHz), 4.00 GB RAM.

**4.1. Extended Cohn-Kanade Dataset.** Lucey et al. [36] presented the CK+ dataset containing 593 sequences from 123 subjects. Each of the sequences incorporates images from onset (neutral frame) to peak expression (last frame). But, only 327 of the 593 sequences were found to meet criteria for one of seven discrete emotions. And, 327 peak frames have been selected and labeled which come together to compose origin facial expression image dataset  $O$ . The detailed number of images of each discrete emotion is shown in Table 1.



TABLE 1: The detailed number of images of each discrete emotion in dataset O.

	Sample set	Training set	Test set
Anger	45	13	32
Contempt	18	13	5
Disgust	59	13	46
Fear	25	13	12
Happiness	69	13	56
Sadness	28	13	15
Surprise	83	13	70



FIGURE 1: Partition and key points of human face.

**4.2. Facial Feature Extraction.** In the paper, we use facial key points of each image as feature points on emotion recognition based on facial expression. Each feature point is expressed as a 2-dimensional coordinate as follows:  $(x, y)$ . The resolution of each image of dataset O is  $640 \times 490$ ,  $640 \times 480$ , or  $720 \times 480$ . In order to unify the standard of coordinate system, image preprocessing is used to change the resolution of each image into  $640 \times 480$ . Reference [11] proposed the production of emotion, which has brought about facial behavior changes and is strongly linked to not the whole face but some specific areas, such as eyebrows, eyes, mouth, nose, and tissue textures. Besides, a face has different rigidness in different areas. According to the principles above, this paper divides face into 4 areas, which are shown in Figure 1 and corresponding feature vectors are listed as follows.

(1) *Eyebrows Area.* Select 8 key points from each eyebrow; their 2-dimensional coordinates  $(x_{1,k}, y_{1,k})$ ,  $k = 1, \dots, 16$ , work together to form a 32-dimensional feature vector  $\vec{f}^1 = (x_{1,1}, y_{1,1}, x_{1,2}, y_{1,2}, \dots, x_{1,16}, y_{1,16})$ .

(2) *Eyes.* Select 8 key points from each eye; their 2-dimensional coordinates  $(x_{2,k}, y_{2,k})$ ,  $k = 1, \dots, 16$ , work together to form a 32-dimensional feature vector  $\vec{f}^2 = (x_{2,1}, y_{2,1}, x_{2,2}, y_{2,2}, \dots, x_{2,16}, y_{2,16})$ .

(3) *Nose.* Select 10 key points from nose; their 2-dimensional coordinates  $(x_{3,k}, y_{3,k})$ ,  $k = 1, \dots, 10$ , work

together to form a 20-dimensional feature vector  $\vec{f}^3 = (x_{3,1}, y_{3,1}, x_{3,2}, y_{3,2}, \dots, x_{3,10}, y_{3,10})$ .

(4) *Mouth.* Select 17 key points from mouth; their 2-dimensional coordinates  $(x_{4,k}, y_{4,k})$ ,  $k = 1, \dots, 17$ , work together to form a 34-dimensional feature vector  $\vec{f}^4 = (x_{4,1}, y_{4,1}, x_{4,2}, y_{4,2}, \dots, x_{4,17}, y_{4,17})$ .

Above all, we select 59 key points from the eyebrows, eyes, nose, and mouth. Therefore, 118-dimensional facial feature vector  $\vec{f}$  can be got from each frame where  $\vec{f} = (\vec{f}^1, \vec{f}^2, \vec{f}^3, \vec{f}^4)$ .

**4.3. Experiment Contrast with Different Feature.** Sample set S contains 327 feature vectors of facial images of seven discrete emotions. We use the method of stratification sampling to get training set and test set. First, we treat the sample set S in 7 disjoint layers on the basis of certain emotions. Then, we select a fixed number of feature vectors from each layer independently and randomly. The number is determined by the smallest size of 7 facial expression sample sets, which is 70% of the size of contempt sample set in this article. At last, all these selected feature vectors come together to compose training set T, while the rest of feature vectors come together to compose test set V. The detailed number of feature vectors of each emotion is shown in Table 1.

Select the  $\vec{f}^i$  component of feature vector  $\vec{f}$  to compose training set  $T_i$  ( $1 \leq i \leq 4$ ) and test set  $V_i$  ( $1 \leq i \leq 4$ ). Thus we experiment four times under different facial area features, respectively. The detailed recognition accuracy of each facial area feature is shown in Table 2. According to the analysis of experimental results in four feature areas, the influence of features of three types of region is different. The nonrigid region has the biggest impact; rigid region has the least while semirigid region has an impact at a fair level.

**4.4. Experiment Contrast with Different Kernel Function.** We use the previous experiment results and (10) and (16) to obtain the weight  $\omega_i$  of each area and corresponding linear transformation square matrix P as follows:

$$\vec{\omega} = (\omega_1, \omega_2, \omega_3, \omega_4) = (0.24, 0.25, 0.15, 0.36),$$

$$P = \begin{bmatrix} P_1 & 0 & 0 & 0 \\ 0 & P_2 & 0 & 0 \\ 0 & 0 & P_3 & 0 \\ 0 & 0 & 0 & P_4 \end{bmatrix} \quad (17)$$

$$= \begin{bmatrix} \omega_1 E_{32 \times 32} & 0 & 0 & 0 \\ 0 & \omega_2 E_{32 \times 32} & 0 & 0 \\ 0 & 0 & \omega_3 E_{20 \times 20} & 0 \\ 0 & 0 & 0 & \omega_4 E_{34 \times 34} \end{bmatrix}.$$

Standard Gaussian kernel function  $k(\vec{f}_i, \vec{f}_j)$  and weighted feature Gaussian kernel function  $k_p(\vec{f}_i, \vec{f}_j)$  can be

TABLE 2: The recognition accuracy of each facial area feature.

Subregion	Eyebrows	Eyes	Nose	Mouth
Recognition rate	40.55%	41.94%	25.45%	60.37%

TABLE 3: The number and average precision of correctly recognized facial expressions under two kernel functions.

Emotion	Test set	SVM	WF-SVM
Anger	32	24	28
Contempt	5	2	4
Disgust	46	40	42
Fear	12	9	11
Happiness	56	50	54
Sadness	15	10	12
Surprise	70	62	69
Average precision	—	83%	93%

got by (7) and (13) for the 118-dimensional facial feature vector  $\vec{f}$

$$\begin{aligned}
 k(\vec{f}_i, \vec{f}_j) &= \exp\left(-\gamma \|\vec{f}_i - \vec{f}_j\|^2\right), \\
 k_P(\vec{f}_i, \vec{f}_j) &= \exp\left(-\gamma \left\| \vec{f}_i^T P - \vec{f}_j^T P \right\|^2\right) \\
 &= \exp\left(-\gamma \left( (\vec{f}_i - \vec{f}_j)^T P P^T (\vec{f}_i - \vec{f}_j) \right)\right).
 \end{aligned} \tag{18}$$

Thus we experiment twice under training set  $T$  and test set  $V$  with different kernel function, respectively. The number of correctly recognized facial expressions under two kernel functions is shown in Table 3.

Finally, we compare our results with the experiments of two kernel functions, which are all image-based framework and tested on the CK+ dataset. The average precision of WF-SVM which uses weighted feature Gaussian kernel function is 93%, which is higher than SVM that uses standard Gaussian kernel function whose average precision is 83%, as is shown in Table 3. And the recognition rate is better than the previous method for the seven emotions. These confirm the effectiveness of our method. After investigating the reason, we find it can be explained from robustness of machine learning algorithm. This method reduces the influence of weak correlation feature by weighted feature, thus improving the robustness of algorithm.

## 5. Conclusion and Future Work

In this paper, we propose an approach of emotion recognition based on facial expression. In our approach, we propose a feature weighting technique since the effect of each feature on recognition result is different. Different from previous works by calculating weight of each feature directly, the facial expression images are divided into some uniform subregions and weight of subregion features is calculated based on

their respective subrecognition rate. The experimental results suggest that the approach based on weighted feature Gaussian kernel function has good performance on the correct rate in emotion recognition. But our approach shows a pretty good performance for the dataset with limited head motion. Emotion recognition based on facial expression is still full of challenges in the future.

## Competing Interests

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

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (no. 61573066).

## References

- [1] J. B. Alonso, J. Cabrera, C. M. Travieso, and K. López-de-Ipiña, “New approach in quantification of emotional intensity from the speech signal: emotional temperature,” *Expert Systems with Applications*, vol. 42, no. 4, pp. 9554–9564, 2015.
- [2] W. H. Dai, D. M. Han, Y. H. Dai, and D. R. Xu, “Emotion recognition and affective computing on vocal social media,” *Information and Management*, vol. 52, no. 7, pp. 777–788, 2015.
- [3] W.-L. Zheng and B.-L. Lu, “Investigating critical frequency bands and channels for eeg-based emotion recognition with deep neural networks,” *IEEE Transactions on Autonomous Mental Development*, vol. 7, no. 3, pp. 162–175, 2015.
- [4] G. Chanel and C. Mühl, “Connecting brains and bodies: applying physiological computing to support social interaction,” *Interacting with Computers*, vol. 27, no. 5, pp. 534–550, 2015.
- [5] N. Jatupaiboon, S. Pan-Ngum, and P. Israsena, “Subject-dependent and subject-independent emotion classification using unimodal and multimodal physiological signals,” *Journal of Medical Imaging and Health Informatics*, vol. 5, no. 5, pp. 1020–1027, 2015.
- [6] A. Mehrabian, *Silent Messages*, Wadsworth Publishing Company, Belmont, Calif, USA, 1971.
- [7] Y. Guo, G. Zhao, and M. Pietikäinen, “Dynamic facial expression recognition with atlas construction and sparse representation,” *IEEE Transactions on Image Processing*, vol. 25, no. 5, pp. 1977–1992, 2016.
- [8] E. Barroso, G. Santos, L. Cardoso, C. Padole, and H. Proenca, “Periocular recognition: how much facial expressions affect performance?” *Pattern Analysis and Applications*, vol. 19, no. 2, pp. 517–530, 2016.
- [9] Y. J. Lei, Y. L. Guo, M. Hayat, M. Bennamoun, and X. Z. Zhou, “A Two-Phase Weighted Collaborative Representation for 3D partial face recognition with single sample,” *Pattern Recognition*, vol. 52, pp. 218–237, 2016.
- [10] X. Zhang and M. H. Mahoor, “Task-dependent multi-task multiple kernel learning for facial action unit detection,” *Pattern Recognition*, vol. 51, pp. 187–196, 2016.
- [11] B. Fasel and J. Luetttin, “Automatic facial expression analysis: a survey,” *Pattern Recognition*, vol. 36, no. 1, pp. 259–275, 2003.

- [12] G. Zhao and M. Pietikäinen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 915–928, 2007.
- [13] C. F. Shan, S. G. Gong, and P. W. McOwan, "Facial expression recognition based on Local Binary Patterns: a comprehensive study," *Image and Vision Computing*, vol. 27, no. 6, pp. 803–816, 2009.
- [14] A. Dhall, A. Asthana, R. Goecke, and T. Gedeon, "Emotion recognition using PHOG and LPQ features," in *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition and Workshops (FG '11)*, pp. 878–883, IEEE, Santa Barbara, Calif, USA, March 2011.
- [15] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 886–893, San Diego, Calif, USA, June 2005.
- [16] J. P. Jones and L. A. Palmer, "An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex," *Journal of Neurophysiology*, vol. 58, no. 6, pp. 1233–1258, 1987.
- [17] Y.-Z. Zhan, J.-F. Ye, D.-J. Niu, and P. Cao, "Facial expression recognition based on gabor wavelet transformation and elastic templates matching," in *Proceedings of the 3rd IEEE International Conference on Image and Graphics (ICIG '04)*, pp. 254–257, Hong Kong, December 2004.
- [18] F. Ren and Z. Huang, "Facial expression recognition based on AAM-SIFT and adaptive regional weighting," *IEEE Transactions on Electrical and Electronic Engineering*, vol. 10, no. 6, pp. 713–722, 2015.
- [19] H. Soyel and H. Demirel, "Localized discriminative scale invariant feature transform based facial expression recognition," *Computers & Electrical Engineering*, vol. 38, no. 5, pp. 1299–1309, 2012.
- [20] X. B. Gao, Y. Su, X. Li, and D. Tao, "A review of active appearance models," *IEEE Transactions on Systems, Man and Cybernetics—Part C: Applications and Reviews*, vol. 40, no. 2, pp. 145–158, 2010.
- [21] J. W. Sung, T. Kanada, and D. J. Kim, "A unified gradient-based approach for combining ASM into AAM," *International Journal of Computer Vision*, vol. 75, no. 2, pp. 297–310, 2007.
- [22] K.-W. Wan, K.-M. Lam, and K.-C. Ng, "An accurate active shape model for facial feature extraction," *Pattern Recognition Letters*, vol. 26, no. 15, pp. 2409–2423, 2005.
- [23] S. Lucey, Y. Wang, J. Saragih, and J. F. Cohn, "Non-rigid face tracking with enforced convexity and local appearance consistency constraint," *Image and Vision Computing*, vol. 28, no. 5, pp. 781–789, 2010.
- [24] R. Cordeiro de Amorim and B. Mirkin, "Minkowski metric, feature weighting and anomalous cluster initializing in K-Means clustering," *Pattern Recognition*, vol. 45, no. 3, pp. 1061–1075, 2012.
- [25] M. Hu, K. Li, X. H. Wang, and F. J. Ren, "Facial expression recognition based on histogram weighted HCBP," *Journal of Electronic Measurement and Instrument*, vol. 29, no. 7, pp. 953–960, 2015.
- [26] C. Cui and V. K. Asari, "Adaptive weighted local textural features for illumination, expression, and occlusion invariant face recognition," in *Imaging and Multimedia Analytics in a Web and Mobile World*, vol. 9027 of *Proceedings of SPIE*, San Francisco, Calif, USA, February 2014.
- [27] R. S. Lee, C.-W. Chung, S.-L. Lee, and S.-H. Kim, "Confidence interval approach to feature re-weighting," *Multimedia Tools and Applications*, vol. 40, no. 3, pp. 385–407, 2008.
- [28] U.-D. Jang, "Facial expression recognition by using efficient regional feature," *Journal of Korean Institute of Information Technology*, vol. 11, no. 1, pp. 217–222, 2013.
- [29] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.
- [30] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [31] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.
- [32] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997.
- [33] E. Kremic and A. Subasi, "Performance of random forest and SVM in face recognition," *International Arab Journal of Information Technology*, vol. 13, no. 2, pp. 287–293, 2016.
- [34] G. Ali, M. A. Iqbal, and T.-S. Choi, "Boosted NNE collections for multicultural facial expression recognition," *Pattern Recognition*, vol. 55, pp. 14–27, 2016.
- [35] X.-H. Wang, A. Liu, and S.-Q. Zhang, "New facial expression recognition based on FSVM and KNN," *Optik*, vol. 126, no. 21, pp. 3132–3134, 2015.
- [36] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '10)*, pp. 94–101, IEEE, San Francisco, Calif, USA, June 2010.