A Robust Learning Framework Using PSM and Ameliorated SVMs for Emotional Recognition

Jinhui Chen, Yosuke Kitano, Yiting Li, Tetsuya Takiguchi, Yasuo Ariki

Graduate School of System Informatics, Kobe University, Kobe, 657-8501, Japan
{ianchen, kitano, liyiting }@me.cs.scitec.kobe-u.ac.jp
{takigu, ariki}@kobe-u.ac.jp

Abstract. This paper proposes a novel machine-learning framework for facial-expression recognition, which is capable of processing images fast and accurately even without having to rely on a large-scale dataset. The framework is derived from Support Vector Machines (SVMs) but distinguishes itself in three key technical points. First, the measure of the samples normalization is based on the Perturbed Subspace Method (PSM), which is an effective way to improve the robustness of a training system. Second, the framework adopts SURF (Speeded Up Robust Features) as features, which is more suitable for dealing with real-time situations. Third, we use region attributes to revise incorrectly detected visual features (described by invisible image attributes at segmented regions of the image). Combining these approaches, the efficiency of machine learning can be improved. Experiments show that the proposed approach is capable of reducing the number of samples effectively, resulting in an obvious reduction in training time.

1 Introduction

Facial expressions recognition is a typical multi-class classification problem in computer vision. Furthermore, since it is one of the most significant technologies for auto-analyzing human behaviors, and it is able to be widely applied into many domains. Therefore, the need for this kind of technology in various different areas keeps pushing the research forward every year.

As the main detectors, Adaboost and SVMs, etc. are widely used in this field of research. In 1995, Freund and Schapire [1] supplied the AdaBoost algorithm for realizing the learning framework of Boosted Trees, which could be referred to Probably Approximately Correct (PAC) learning proposed by Valiant [2]. Since then great advances have been made based on AdaBoost, especially milestone work by Viola and Jones [3]. But some ideal strong classifiers are usually required a large number of training samples and very time-consuming training experiments. Even recently, many researchers are trying to solve these problems. Li *et al* [4] proposed a new learning SURF cascade for ameliorating boosting cascade frameworks. It improved the training efficiency, but the need for large-scale data gathering and extensive preparations create a critical bottleneck. On the other hand, similar problems also exist in methods based on SVMs, because of

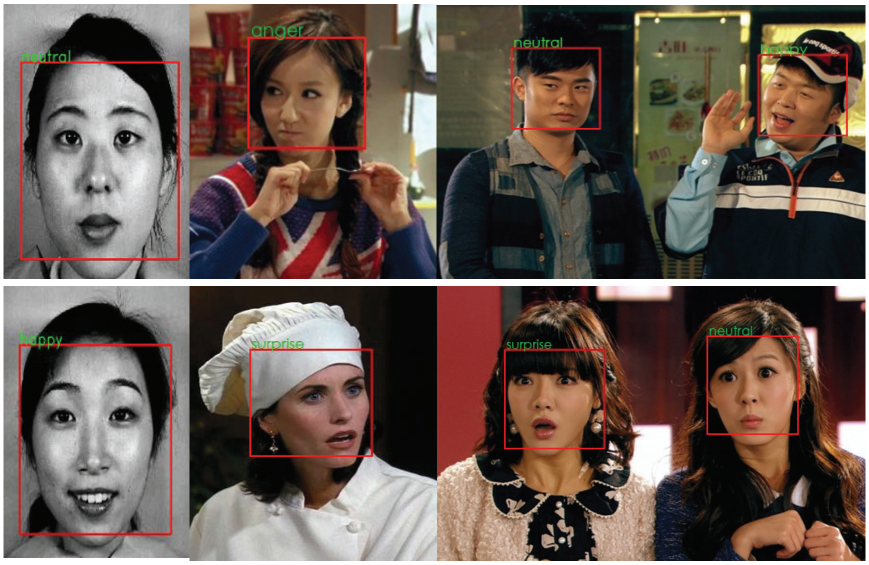


Fig. 1. Examples of Recognition Results

the limitation of length (they will be not enumerated here). Therefore, collecting many training samples and the long training time lead to considerable work and difficulty for researchers in the field of pattern recognition. Since training is a critical infrastructure for recognition engines, the research on training is significant for learning machines. Hence, there is a great need to solve the problem mentioned above.

This paper brings together new normalization measures, visual features and image attributes to construct a framework for facial expressions recognition. As almost all of our approaches relate to vectors processing, the classifier of our proposed method is built based on SVMs. There are three main approaches with emphasis on reducing training samples and improving the efficiency of learning machines. First, PSM is used to extend the training data, which allows for the generation of ideal strong classifiers without having to collect a large number of training samples. Second, the features are described by local multi-dimensional SURF descriptors [5], which are spatial regions with windows and are good at processing real-time scenes. Moreover, SURF is much faster and more efficient than most of the existing local features algorithms [4], such as SIFT [6], HoG [7] *etc.* Third, the region attributes of images are adopted to revise incorrect detection of classifiers relying on visual features, which are represented by feature vectors in a segmented region. Therefore, the discriminative capability can guarantee the proposed framework will be more robust.

In summary, there are these main contributions in this paper. The first is that the efficiency of machine learning can be improved. Experiments show that the proposed approach is capable of reducing the number of samples effectively, resulting in an obvious reduction in training time. The second is that the recognition accuracy is comparable to the state-of-the-art algorithms. In experiments, the results show that although using a mini-sized database of training samples, our approaches can also construct a robust facial expressions recognition system, which is comparable to the state-of-the-art methods. Some examples of recognition result are shown in Fig. 1. We believe the proposed method is a good try for machine learning, because recognition accuracy plays a very significant role in machine learning, but without doubt, the training efficiency and is also equally important.

The rest of this paper is organized as follows: we will first revisit related works in section 2, then we describe the samples normalization in section 3 and classifying framework in section 4 respectively. Section 5 elaborates on region attributes estimation. Section 6 shows the experiments and conclusions are drawn in section 7.

2 Related Work

Facial-expression recognition is a hot research topic in computer vision due to its many applications, and many researchers attach great importance to this field. For instance, Lyons *et al* [8] adopted PCA and LDA to analyze facial expressions through closed experiments, and they achieved 92% accuracy on JAFFE [9,10]; Bartlett *et al* [11] proposed a Gabor feature based AdaSVM method for expression recognition, obtaining a good performance based on the use of the Cohen-Kanade expression database [12]. However, it has been shown that the processing speed of these approaches is too slow to deal with real-time scenes. More recently, Anderson *et al* [13] and Chen *et al* [14] proposed their approaches for real-time expression recognition separately, but their methods required a large amount of data for training.

Our approach enables competition that completes a recent line of papers that use third-party software tools to obtain mirror images of samples for training in their object/facial-expression recognition systems, which we briefly review here. To the best of our knowledge, our approach is the first to apply the proposed method into both of object recognition and facial expression recognition. Meanwhile, it is the first to employ the PSM directly for detectors training, but not use any tool. Experiments show it has the greatest impact on the performance of training efficiency, because time can be saved, which would be spent on collecting vast amounts of data from the Internet or using third-party software to deal with the samples for getting mirror images of these samples. In addition, in our framework, the classifier is based on SVMs, furthermore, its function is ameliorated, which can guarantee the results will be more reliable.

3 PSM for Samples Normalization

PSM is applied to the normalization of samples, but different from previous versions (many-to-one mapping), in this paper, PSM is a one-to-many mapping, we use it to extend the subspace of samples. There are many existing methods based on virtual images, which seem similar with ours, but most of them rested on pixel-level transformation (such as [15] and [16] *etc.*), so that after the calculation, some features data would be damaged easily. Moreover,they required some handwork, and in the training period, the program had to read a large number of virtual image files again, these lead to time waste. We don't have these problems. Therefore, it is an effective way to improve the robustness of the training system.

3.1 Training-sample Normalization

In order to reduce the noise, the size of the images is unified by $m \times n$ pixels, and the original samples are normalized by the mean value and variance of pixels transformation. Therefore, the images after the normalization can be obtained according to the following equation:

$$I'(x,y) = a\frac{I(x,y)-\mu}{2\sqrt{2}\sigma} + b. \tag{1}$$

Here σ is the standard deviation, and

$$\sigma = \sqrt{\frac{1}{mn}\sum_{x=1}^{m}\sum_{y=1}^{n}(I(x,y)-\mu)^2}. \tag{2}$$

(a, b) is used to adjust the value of pixels (In this paper, we used regular samples in experiments, therefore, a was set as 1, b was set as 0). μ is the mean value of pixels, and it can be computed through image traversal by the equation:

$$\mu = \frac{1}{mn}\sum_{x=1}^{m}\sum_{y=1}^{n} I(x,y). \tag{3}$$

3.2 Changing Facial Orientations

After the calculation of subsection 3.1, we can thus extend the subspace of samples through changing the facial directions of the images. In this paper, we use the method proposed by Chen *et al* [14] to reconstruct three-dimensional faces and obtain three-dimensional data, and we indicate it in Algorithm 1.

In Algorithm 1, when E_r is upper a threshold ε, or K landmarks are processed over, the while loop would be stopped, and the three-dimensional data will be output. Here $\beta = (\beta_1, \beta_2, \cdots, \beta_m)^T$ is the shape parameter and m is the dimensionality of the shape parameter, which is used to adjust three-dimensional shape data. S_{3D} is a $3 \times n$ matrix, P is a 2×3 orthographic projection matrix, T is a $3 \times n$ translation matrix consisting of n translation vectors $t = [t_x, t_y, t_z]^T$, and R_θ is a 3×3 rotation matrix where the yaw angle is θ. In this paper, θ is

Algorithm 1 Reconstruct Three-dimensional Face

```
Require:
  Input: two-dimensional shape vector: $S_{2D} \in R^2$
  Output: three-dimensional shape vector: $S_{3D} \in R^3$
  Initialization: set $\beta_0 = 0$, $i = 0$
  while $i < K$ or $E_r \leq \varepsilon$ do
    1. Let
          $S_{3D} \Leftarrow s_0 + \sum_{i=1}^{m} \beta_i s_i$
    2. Alignment: $S_{2D}$ is aligned with the two-dimensional shape, which is obtained
       by projecting the frontal three-dimensional shape ($s_i$) onto the $x - y$ plane.
    3. Minimize
          $\|P(R_\theta S_{3D} + T) - S_{2D}\|^2$
    4. Reconstruct $(S_{3D})_i$ using the shape parameter $\beta_i$.
    5. Update $R_\theta$ and $T$ with the fixed shape parameter and
          $E_r \Leftarrow \|P(R_\theta S_{3D} + T) - S_{2D}\|^2$
    6. Let
          $i \Leftarrow i + 1$
  end while
  7. Reconstruct three-dimensional shape using the final shape parameters.
  8. Output $S_{3D}$.
```

set as $\pm 15^\circ$, $\pm 30^\circ$, $\pm 60^\circ$. Thus, through Algorithm 1, we can obtain the three-dimensional data $X = (x, y, z)^T$ from the original images. Hence, according to the transformation matrix formula:

$$X' = T_z \cdot T_y \cdot T_x \cdot S \cdot R_z \cdot R_y \cdot R_x \cdot X. \tag{4}$$

we can convert the facial directions to extend the subspace of the training samples. Here T and R are the shear mapping transformation matrix and the rotation matrix respectively, and S is represented by the scaling matrix.

3.3 Changing Illumination Attributes

The illuminative change is conducted according to the following equation:

$$V_2^{(n)} = V_1^{(n)} + \sum_{m=1}^{K} w_m \cdot e_m^{(n)}. \tag{5}$$

where V_1 is the changing feature, V_2 is the result after the changes, n is the dimensionality of the feature vector, w is the weight coefficient, and e is the basis of illumination-change-factor vectors.

In this paper, e is obtained through processing the luminance normalized rendering images by principal component analysis (PCA), wherein, m is the

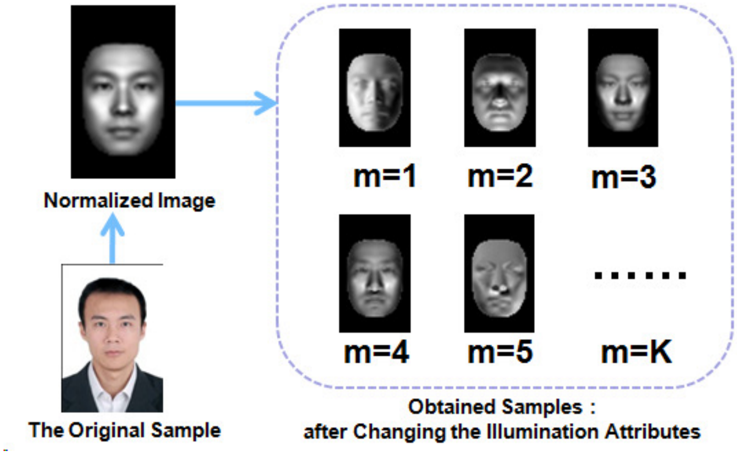


Fig. 2. Illumination-attribute Changes Used for Extending Training Samples

principal component ($m = 1, \cdots, 8$). The rendering images are gained by the treatment of three-dimensional images obtained in subsection 2.2. An example of results is shown in Fig. 2.

4 Classifying Framework

This section will provide the framework used for SVM machine learning through adopting SURF features. Moreover we will also employ the region attributes of image to revise miss-detection of classifier relying on visual features. We will describe them separately in this section.

4.1 Feature Description

SURF is a scale- and rotation-invariant interest point detector and descriptor. It is faster than SIFT [6] and more robust against different image transformations. In this paper, we adopt an 8-bin T2 descriptor to describe the local feature, which is inspired by [4]. Different from [17], we further allow different aspect ratio for each patch (the ratio of width and height), because this can make the speed of image traversal become quicker. Meanwhile we imported diagonal and anti-diagonal filters, this can improve description capability of SURF descriptors.

Given a detection window, we define rectangular local patches within it, each patch with 4 spatial cells and allow the patch size ranging from 12×12 pixels to 40×40 pixels. Each patch is represented by a 32-dimensional SURF descriptor. The descriptor can be computed quickly based on sums of two-dimensional Haar wavelet responses and we can make an efficient use of Integral Images [3]. Suppose d_x as the horizontal gradient image, which can be obtained using the filter kernel $[-1, 0, 1]$, and d_y is the vertical gradient image, which can be obtained using the

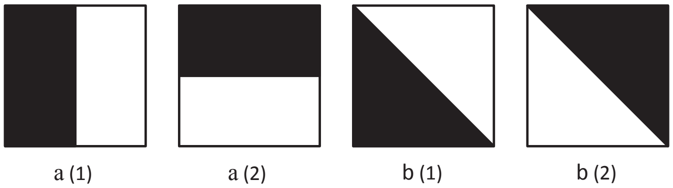


Fig. 3. Haar-type Filter Used for Computing SURF Descriptor

filter kernel $[-1, 0, 1]^T$; Define d_D as the diagonal image and d_{AD} as the anti-diagonal image, both of which can be computed using two-dimensional filter kernels *diag* $(-1, 0, 1)$ and *antidiag* $(-1, 0, 1)$. Therefore, 8-bin T2 is able to be defined as $v = (\sum(|d_x| + d_x), \sum(|d_x| - d_x), \sum(|d_y| + d_y), \sum(|d_y| - d_y), \sum(|d_D| + d_D), \sum(|d_D| - d_D), \sum(|d_{AD}| + d_{AD}), \sum(|d_{AD}| - d_{AD}))$. Here, d_x, d_y, d_D, and d_{AD} can be computed individually by filters shown in Fig 3 in use of Integral Images, the details about how to compute two-dimensional Haar responses with Integral Images, please refer to [3].

The detection template for SURF is 40 × 40 with 4 spatial cells, and allow the patch size ranging from 12 × 12 pixels to 40 × 40 pixels. We slide the patch over the detection template with 4 pixels forward to ensure enough feature-level difference. We further allow different aspect ratio for each patch (the ratio of width and height). The local candidate region of the features is divided into 4 cells. The descriptor is extracted in each cell. Hence, concatenating features in 4 cells together yields a 32-dimensional feature vector. About feature normalization, in practice, L_2 normalization followed by clipping and renormalization (L_2Hys) [18] is shown working best.

4.2 Classifier Construction

The classifier of our framework is built based on One-Versus-Rest SVMs (OVR-SVMs). OVR strategy consists of constructing one SVM per class, which is trained to distinguish the samples of one class from the samples of all remaining classes. Normally, classification of an unknown object is carried out by adopting the maximum output among all SVMs. The proposed method is based on OVR-SVMs classifier, and implemented by re-developing liblinear SDK [19].

Usually, most of researchers estimate posterior probability by mapping the outputs of each SVM into probability separately. The method was proposed by Platt [20] . It applies an additional sigmoid function:

$$H(\omega_j | f_j(x)) = \frac{1}{1 + exp\ (c_j f_j(x) + d_j)}. \quad (6)$$

$f_j(x)$ denotes the output of the SVM trained to separate the class ω_j from the other classes (total samples are M). Then, for each sigmoid the parameters c_j and d_j are optimized by minimizing the local negative log-likelihood:

$$-\sum_{k=1}^{N}\{p_k log(h_k) + (1-p_k)log(1-h_k)\}. \tag{7}$$

Here are N outputs of the sigmoid function, where h_k is the output of the sigmoid function with the probability p_k event. In order to solve this optimization problem, [20] applied a model-trust minimization algorithm based on the Levenberg-Marquardt algorithm. But in [21], Lin *et al* pointed out that there are some problems in this method, meanwhile they proposed another minimization algorithm based on Newton's method with backtracking line search.

But unfortunately, there is nothing to guarantee that:

$$\sum_{j=1}^{M} H(\omega_j|f_j(x)) = 1. \tag{8}$$

For this reason, it is necessary to normalize the probabilities as following:

$$H(\omega_j|x) = \frac{H(\omega_j|f_j(x))}{\sum_{j'=1}^{M} H(\omega_{j'}|f_{j'}(x))}. \tag{9}$$

Thus, we use another approach to estimate posterior probability, using OVR-SVMs to exploit the outputs of all SVMs to estimate overall probabilities. In order to achieve this goal, we apply the softmax function, regarding it as a generalization of sigmoid function for the multi-SVMs case. Thus, in the spirit of the improved Platt's algorithm [22], this paper applies a parametric form of the softmax function to normalize the probabilities by:

$$H(\omega_j|x) = \frac{exp\ (c_j f_j(x) + d_j)}{\sum_{j'=1}^{M} exp\ (c_{j'} f_{j'}(x) + d_{j'})}. \tag{10}$$

And here the parameters c_j and d_j are optimized by minimizing the global negative log-likelihood

$$-\sum_{k=1}^{N} log(H(\omega_k|x_k)). \tag{11}$$

Optimizing the parameters c_j and d_j are done with intention of obtaining the lowest error rate on testing dataset. The reason of why we use the negative log-likelihood is not only because it can optimize the parameters c_j and d_j , but also because it can be used for comparing the various probability estimates, in other words, it can evaluate the error rate on machine learning and reject some of unsatisfactory candidate expression regions described by SURF features.

5 Region Attributes Estimation

After classifying the OVR-SVMs model, we can obtain the recognition result classified by classifiers based on visual features. However, it is necessary to make further efforts on reducing misrecognition; thus, we use invisible image attributes to realize this purpose.

The detected face region is divided into 9×10 blocks, and the feature vector of each block is computed. We call this region attributes. It can be obtained after normalization by equalizing the value and variance of the luminance, while the norm is set as 1. The region attribute is estimated using the following score equation.

$$d = \left\|X - \bar{X}\right\|^2 - \sum_{i=1}^{N} \frac{\lambda_i}{\lambda_i + \delta^2} (\varphi_i(X - \bar{X}))^2. \tag{12}$$

here φ is eigenvector and λ is eigenvalue, δ^2 is the image noise correct divisor. When $\delta^2 = 0$, it means that the distances of all feature vectors of the current image projecting into subspace are unified, in the other words, the noise is negligible. X is estimated image region attributes, and $\bar{X}$ is the average feature vector of samples. The value of distance is smaller, the score is higher, namely, the probability of miss-detection is lower.

In this paper, the most significant way to ameliorate OVR-SVMs is based on two conflicted criteria, inspired by Boosting cascade: A error rate evaluating threshold e ($e^n = (1 - d)$), its function is similar with false-positive-rate in Boosting cascade [2]. And recognition rate evaluating threshold d, its function is similar with hit-rate in Boosting cascade. They are used for the detection-error tradeoff: $e < 0.5$ the classifying result will be considered as miss-detection and OVR-SVMs classifying model is executed repeatedly until a given Boolean condition $d \leq 0.2$ is met.

6 Experiments

In this section we will show the details of implementation, dataset, and evaluation results. The proposed method is designed for Neutral-, Happy-, Anger- and Surprise-expression recognition. And the recognition result examples are shown in Fig. 1.

We implemented all training and detection programs in C++ on RHEL (Red Hat Enterprise Linux) 6.5 OS. The facial recognition part used the source code of Open CV, which was based on Viola and Jones framework [3]. The expressional recognition part was implemented based on the proposed framework. The experiments were done on the PC with Core i7-2600 3.40 GHz CPU and 8 GB RAM, the training procedure was fully automatic. For SURF extraction, we adopted Integral Image to speedup the computation as described in section 3.1. For machine learning, we built the OVR-SVMs through re-developing liblinear software [19].

6.1 Experimental Dataset

Training Database Set We used Cohn-Kanade expression database (CK+) [12] as training database, which is a set of front face images posed by 123 posers, but not all of posers posed each type of expressions what we need. Therefore, we also collected some samples by online image search engine, fianlly we obtained 240 initial facial samples for each type of emotion. All of facial samples were normalized to 90 × 100-pixel patches and processed by histogram equalization, no color information was used.

Testing Database Set In order to evaluate both of the real life and ideal situations, we used two parts of testing sets. One part was obtained from soap operas, because many public databases were processed by providers in advance, or the other reasons, such as the images cannot represent real-life scene, because they are not continuous images *etc.*, hence, we had to use some video clips of comedy dramas, which had the total of 10 persons whose facial expressions were similar to the training samples. These images of these actors and actresses are on 8 video clips having a length of 120 seconds. We marked this set as Test Set A. The other testing set was the JAFFE database [9,10], whose facial samples are totally different from CK+ database. 213 images of JAFFE were mixed randomly and one image can be used repeatedly (ensure that there are enough images for different videos making), these images were also made into 8 120-second-long videos, and we marked this set as Test Set B. All of the test videos have speed of 60 FPS (Frames Per Second)

6.2 Experimental Evaluation

Training Experiments The training database of all methods was mentioned above, but only the proposed method did not adopt any process to obtain plenty of mirror samples. Hence, it reduced a mass of samples and took only 49.8 min to complete the whole process. Besides, the training procedure was fully automatic. The relative data are shown in Table 1.

Table 1. Training Efficiency Evaluation Results

Method	Proposed	K-means [23]	LUT_Ada [14]
Time cost	49.8 min	1,589 min	172.5 min

However, in order to enhance the generalization performance of comparison method [23] and comparison method [14], we had to deal with the images by some transformations (mirror reflection and rotate the images by horizontal and vertical angles ±15°, ± 30°, ± 60° *etc.*), finally, we obtained each class 30,960, total 123,840 facial samples for training classifiers. Therefore, they are very time-consuming tasks.

Testing Experiments In order to be evaluated easily, CSV files were created automatically by the experimental program, and the tested results for each frame image of test videos were stored in these files. After doing test experiments, tested

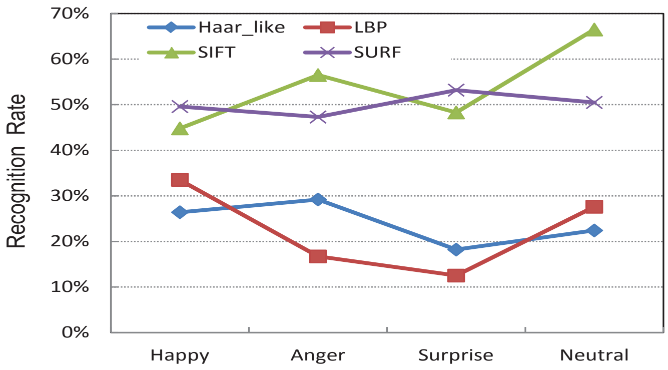


Fig. 4. Green: Recognition rate for OVR-SVMs with SIFT. Purple: Recognition rate for OVR-SVMs with SURF. Blue: Recognition rate for OVR-SVMs with Haar-like. Red: Recognition rate for OVR-SVMs with LBP. Features using SURF and SIFT obtained the more accurate results, but the feature extraction speed of SIFT was low.

videos were divided into images, the tested results of CSV files were checked one by one with these images. These measures can guarantee the validity of experimental data. All of approaches of this paper were evaluated though testing experiments, the details are indicated as follows:

Fig. 4 indicates the expression-recognition rate for different feature detectors based on our ameliorated SVMs detector. The aim of this experiment was evaluating the performance of the proposed detector using different methods of feature extraction. Hence, this experiment was done without a PSM model. Feature detectors using SURF and SIFT obtained the more accurate recognition rates, but the average speed of the SIFT detectors version was only 16.8 FPS. In comparison, the speed of the SURFs version reached 39.4 FPS. Theoretically, 16.8 FPS is also too slow to deal with complex scenes, such as real-time scenes. Thus, SURF was selected as the feature detector.

In this paper, We placed 450 local patches on the 40×40-pixel size detection template with 4 spatial cells, and allowed the patch size ranging from 12×12 pixels to 40×40 pixels. We slide the patch over the detection template with 4 pixels forward to ensure enough feature-level difference. Different from [17], we further allow different aspect ratio for each patch (the ratio of width and height). Our framework adopted 8-bin T2 descriptor as descriptor. It obtained similar precision of recognition results to the accuracy of original SURF's version and even SIFT's one, but dominated others on feature extraction speed. In fact, in our experiments, 8-bin T2 descriptors had almost the same accuracy as the original SURF; however, its extraction speed can reach more than 39.4 FPS, while the speed of original SURF version had only about 19 FPS, which was also too slow. Therefore, the feature descriptor based on 8-bin T2 SURF is the best choice for our framework.

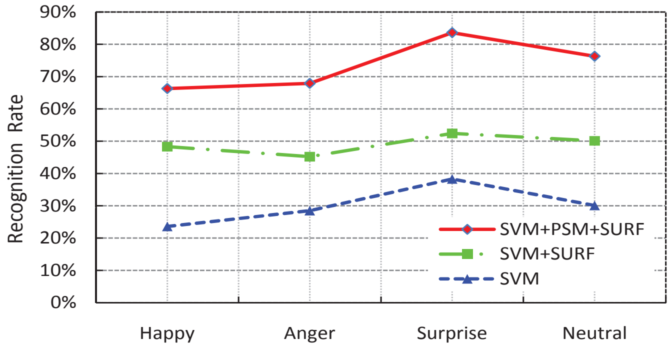
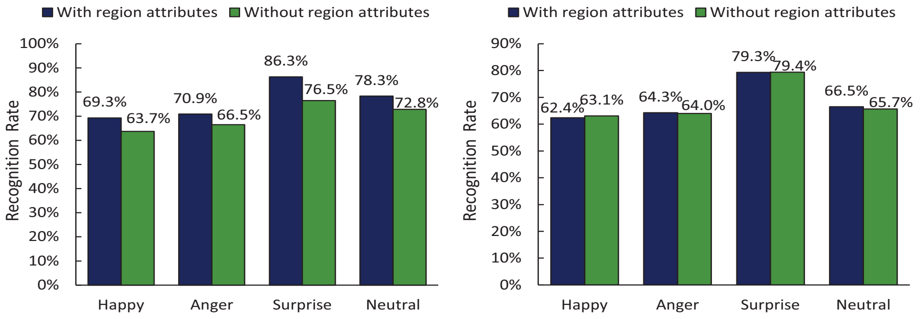


Fig. 5. Top: Recognition rate for proposed method. Middle: Proposed method without PSM; *i.e.*, the OVR-SVMs+SURF model. Bottom: Only OVR-SVMs. OVR-SVMs+PSM+SURF (proposed method) is the most accurate version of our detector.

In Fig. 5, the component selection of the proposed method was carried out to investigate how each component contributes to the recognition rate. In fact, in our experiments, OVR-SVMs had almost the same convergence speed as the original SVM and non-linear SVMs with the RBF kernel. However, its accuracy was the best one. As we known, the original SVM cannot be applied to the continuous classification, if it includes more than 2-classe categories. Meanwhile, we also tried the non-linear SVMs using the RBF kernel, but its average precision of 4-type emotional classification was only approximately 33.6%. As a result, the OVR-SVMs+PSM+SURF model was the most accurate version of our classifier.



(a) Region Attributes Estimation for Test Set A

(b) Region Attributes Estimation for Test Set B

Fig. 6. Evaluation Results for Expressional Region Attributes

Fig. 6 shows the results of the evaluation experiments for expressional region attributes. Fig. 6(a) shows the results for Test Set A videos, and Fig. 6(b) shows the results for Test Set B. In the experiments, we found that after introducing the region attributes model, the recognition accuracy of Test Set A improved approximately by 7%. On the other hand, the results of Test Set B were almost unchanged, since the videos in Test Set B consisted of JAFFE images, and these images had been normalized by the supplier [9]. But the videos of Test Set A were used without any normalization. Therefore, this approach is capable of dealing with original images better; *i.e.*, it is good at processing real-life videos.

Table 2. Experimental Results for Test Set A

	Proposed	K-means [23]	LUT_Ada [14]
Happy	69.3%	52.0%	61.2%
Anger	70.9%	64.5%	50.9%
Suprise	86.3%	42.8%	68.6%
Neutral	78.3%	37.1%	65.6%

Table 2 and Table 3 indicate the recognition accuracies, and they show the performance of the proposed method compared to other classifiers ([14] is one of the latest methods for facial expressions recognition, and it was based on AdaBoost; [23] is a typical expressions recognition method using K-means). Table 2 shows the recognition rate of evaluation experiments for Test Set A. Since the races and facial expressions of Test Set As people were similar to those of the training samples, the region attributes model was effective for Test Set A in which there are videos from real life. Consequently, its accuracy was quite better than the Test Set B's. The maximum recognition precision of the proposed method was 86.3%, and the worst result was 69.3%.

Table 3. Experimental Results for Test Set B

	Proposed	K-means [23]	LUT_Ada [14]
Happy	62.4%	55.3%	57.7%
Anger	64.2%	59.5%	48.2%
Suprise	79.3%	44.8%	68.4%
Neutral	66.5%	32.6%	71.6%

On the other hand, Table 3 shows the recognition accuracies for Test Set B. Due to the variation and complexity of facial expressions across different cultures and races, the region attributes model was not effective for facial recognition. The results for this test set were not better than Test Set A's. But on the whole, the results of both test sets show that the proposed method was the more accurate version of these methods. Note that the proposed method used training samples without any image-mirror process here. Namely, based on the mini-size training set, the proposed method can also obtain a better result; thus, this model allows for generating ideal strong classifiers without the need for large volumes of training samples. Hence, under these experimental conditions, the validity of the proposed methods was proved.

7 Conclusions

This paper brings together new normalization measures, visual features and image attributes to construct a novel framework, which minimizes the training data but improves the training efficiency. It may well have broader application in machine learning.

PSM is an effective approach for alleviating suffering of collecting plenty of training samples. Then, through doing a great many of experiments, we find SURF is the most suitable feature descriptor for our detector, and the region attributes of images can revise some miss-detection caused by visual features. Combining these approaches together, a robust expression recognition framework can be constructed, but due to the variation and complexity of the facial expression across different cultures and human races, using mini-size training set to obtain high recognition precision, there are many difficult challenges have to be overcome.

About future plan, considering a possible implementation in a real scenario, we are inclined to consider these points: 1) Try to use region attributes as binary latent variables, which are incorporated into the SVMs model for inference. 2) Ameliorate approaches on the construction of SVMs to improve accuracy and to make it be qualified for more complex tasks. 3) applying the approach to object recognition such as human detection, car detection, events understanding *etc.*

References

1. Freund, Y., Schapire, R.: A Desicion-theoretic Generalization of On-line Learning and an application to Boosting. In: Computational learning theory. (1995) 23–37
2. Valiant, L.G.: A Theory of the Learnable. Communications of the ACM **27** (1984) 1134–1142
3. Viola, P., Jones, M.: Rapid Object Detection Using a Boosted Cascade of Simple Features. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). Volume 1. (2001) I–511–I–518 vol.1
4. Li, J., Zhang, Y.: Learning SURF Cascade for Fast and Accurate Object Detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). (2013) 3468–3475
5. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded up Robust Features. In: Computer Vision–ECCV 2006. (2006) 404–417
6. Lowe, D.G.: Object Recognition from Local Scale-invariant Features. (In: The proceedings of the seventh IEEE International Conference on Computer Vision (ICCV))
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). Volume 1. (2005) 886–893

8. Dailey, M.N., Joyce, C., Lyons, M.J., Kamachi, M., Ishi, H., Gyoba, J., Cottrell, G.W.: Evidence and a Computational Explanation of Cultural Differences in Facial Expression Recognition. Emotion **10** (2010) 874–893
9. Kamachi, M., Lyons, M., Gyoba, J.: The Japanese Female Facial Expression (JAFFE) Database. URL http://www. kasrl. org/jaffe. html **21** (1998)
10. Pantic, M., Rothkrantz, L.J.M.: Automatic analysis of facial expressions: The state of the art. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 1424–1445
11. Bartlett, M., Littlewort, G., Fasel, I., Movellan, J.: Real Time Face Detection and Facial Expression Recognition: Development and Applications to Human Computer Interaction. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops). Volume 5. (2003) 53–53
12. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops). (2010) 94–101
13. Anderson, K., McOwan, P.W.: A Real-time Automated System for the Recognition of Human Facial Expressions. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics **36** (2006) 96–105
14. Chen, J., Ariki, Y., Takiguchi, T.: Robust Facial Expressions Recognition Using 3D Average Face and Ameliorated Adaboost. In: Proceedings of the 21st ACM International Conference on Multimedia (ACM MM). (2013) 661–664
15. Chu, W.S., Huang, C.R., Chen, C.S.: Gender classification from unaligned facial images using support subspaces. Information Sciences **221** (2013) 98–109
16. Decoste, D., Schölkopf, B.: Training invariant support vector machines. Machine Learning **46** (2002) 161–190
17. Li, J., Wang, T., Zhang, Y.: Face detection using surf cascade. In: IEEE International Conference on Computer Vision Workshops (ICCV Workshops). (2011) 2183–2190
18. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). Volume 1. (2005) 886–893 vol. 1
19. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A Library for Large Linear Classification. The Journal of Machine Learning Research **9** (2008) 1871–1874
20. Platt, J.: Probabilities for SV Machines. In: Advances in Large Margin Classifiers. (2000)
21. Lin, H.T., Lin, C.J., Weng, R.C.: A Note on Platts Probabilistic Outputs for Support Vector Machines. Machine learning **68** (2007) 267–276
22. Sun, Z., Ampornpunt, N., Varma, M., Vishwanathan, S.: Multiple Kernel Learning and the SMO Algorithm. In: Advances in Neural Information Processing Systems (NIPS). (2010) 2361–2369
23. Alldrin, N., Smith, A., Turnbull, D.: Classifying Facial Expression with Radial Basis Function Networks, Using Gradient Descent and K-means. CSE253 (2003)