Similarity Features for Facial Event Analysis

Peng Yang[1], Qingshan Liu[1,2], and Dimitris Metaxas[1]

[1] Rutgers University, Piscataway NJ 08854, USA
peyang@cs.rutgers.edu
[2] National Laboratory of Pattern Recognition, Chinese Academy of Sciences
Beijing, 100080, China

Abstract. Each facial event will give rise to complex facial appearance variation. In this paper, we propose similarity features to describe the facial appearance for video-based facial event analysis. Inspired by the kernel features, for each sample, we compare it with the reference set with a similarity function, and we take the log-weighted summarization of the similarities as its similarity feature. Due to the distinctness of the apex images of facial events, we use their cluster-centers as the references. In order to capture the temporal dynamics, we use the K-means algorithm to divide the similarity features into several clusters in temporal domain, and each cluster is modeled by a Gaussian distribution. Based on the Gaussian models, we further map the similarity features into dynamic binary patterns to handle the issue of time resolution, which embed the time-warping operation implicitly. The haar-like descriptor is used to extract the visual features of facial appearance, and Adaboost is performed to learn the final classifiers. Extensive experiments carried on the Cohn-Kanade database show the promising performance of the proposed method.

1 Introduction

Automatic facial event analysis is a hot topic in the communities of computer vision and pattern recognition in recent years due to its potential applications in human-computer interface, biometrics, multimedia, and so on. Lots of methods have been proposed [1] [2] [3], and these methods can be categorized into two classes: image based methods and video based methods. The image based methods take only the mug shots (mostly the apexes) of the expressions into account [4] [5] [6] [7]. However, a natural facial event is dynamic, which evolves over time from the onset, the apex, to the offset, including facial expression. The image based methods ignore such dynamic characteristics, so it is hard for them to obtain good performances in a real world setting. Psychology studies also demonstrated the insufficiency of the image based methods [8] [9]. The video based methods attempt to analyze the facial event in the spatio-temporal domain [10] [11] [12] [13] [14] [15], and extensive experiments showed that they were better than the image based methods.

However, how to extract and represent the dynamics of facial is a key issue to the video based methods. The popular one is based on motion analysis. In [10],

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part I, LNCS 5302, pp. 685–696, 2008.

Black and Yacoob used the parametric motion models to describe the local facial dynamics, and took the parameters of local motion models as dynamic features. Torre [16] used condensation to track the local appearance dynamics with the help of subspace representation. In [17], the dynamics are represented by tracking the points of Active Shape Model [18]. Although the motion based methods are much intuitive, they are sensitive to image noise. Manifold learning was also employed to explore the intrinsic subspace of the facial expression events. [19] used the Leipschitz embedding to build a facial expression manifold, and [20] used multilinear models to construct a non-linear manifold model. How to find the intrinsic dimensions of the manifold is still an open problem. In addition, due to diversity of subjects, it is hard to obtain an efficient and general manifold structure. Recently, the volume features attracted much attention in dynamic event analysis [21] [13] [22], which embed the spatio and temporal variation together. The idea of the volume features is to regard the video data as a 3D volume and to extract the features directly from the volume data. Guoying [13] designed the volume local binary patterns (LBP) in the spatio-temporal domain to capture the dynamics of facial events. In [23], we developed the ensemble of Haar-like features with coding scheme for expression recognition.

Time resolution is another issue for video based method, especially in real environment, because there are many factors to make the data varied in different time resolutions. For example, different cameras have different capture speed; different subjects have different paces for the same expression; even the same person will have different response in different situation. Therefore, in practical systems, some pre-defined time-warping processing should be demanded. However, most previous works did not take this into account including recent volume features [13] [23], and they assumed the training and the testing data must have the same length and the same speed rate, i.e., the same time resolution.

In this paper, we propose a new feature representation named similarity feature to address the above issues. It is well known that each facial event will give rise to complex appearance variation, and when it approximates to the apex, its discrimination become more distinct [24]. The kernel features achieved much success in describing the complex image variation [25], which are actually similarity representation against the training samples. Inspired by the kernel features, we measure each sample against the given references with a similarity function, and define the log-weighted summarization of the similarities as the feature to describe the complexity of facial appearance. The cluster-centers of the apex images are selected as the references. To capture the temporal dynamics, we perform the K-means clustering on the similarity features in the temporal domain, and each cluster is modeled as a Gaussian distribution. Based on the Gaussian models, we further map the similarity features into dynamic binary patterns to handle the issue of the time resolution, which involve the time-warping processing implicitly. The haar-like descriptor is used to extract the low-level visual features as in [23], and Adaboost learning is adopted to build the final classifier. Our experiments are conducted on the well-known Cohn-Kanade database, and the experimental results demonstrate that the proposed method has an

encouraging performance. The proposed feature representation is similar to the harr-like volume features [21] [22], but it can handle the data in various time resolution without any assumption. We will give detailed comparisons against the related work in the experiments.

The rest paper is organized as follows: We first give the definition of the similarity features in section 2, and describe how to map the similarity features into the dynamic binary patterns in Section 3. Section 4 addresses the classifier design, and the experiments are reported in Section 5, followed by conclusions.

2 Similarity Features

The kernel trick has attracted much attention, since the SVM achieved much success in the field of machine learning [26]. The kernel features are actually a kind of similarity representation, which are composed of the similarities between a given sample and all the training samples with a nonlinear kernel function, and they can describe the complex variations of images efficiently [25]. As we knew, each facial event is behaved by complex facial appearance variations. Inspired by the kernel features, we develop the similarity features to represent the complexity of facial appearance for facial event analysis. Different from the kernel features, we do not use all the training samples in computing similarities. We only take the apex images into account due to their distinctness. To avoid the influence of different subjects, we perform the K-Means clustering on the apex images to divide them into several clusters, and we take the cluster-centers as the references. The reference selection is also beneficial to computation cost reduction.

The similarity feature is calculated as follows: Given the references $\{r_i\}, i = 1, 2, ..., R$ and a given sample x, the similarities of x against the references are

$$S(x) = \{f(x, r_i\}, i = 1, 2, ..., R, \tag{1}$$

where $f(x, y)$ is the similarity function. In our experiments, we simply use the $L-2$ distance as the similarity function, $f(x, y) = \|x - y\|^2$. Now each sample is described by a R-dimensional similarity vector. Because the video data has both spatio and temporal information, it needs high computational cost if we directly use the above similarity vector. For example, if the number of the references is 100 and the video has 100 frames, then basically we need to do computation in a 10^4 dimensional space. To reduce the computational complexity, we convert the similarity vector into a log-weighted similarity as the final similarity feature,

$$F(x) = \sum_{i=1}^{R} \log(f(r_i, x)). \tag{2}$$

3 Dynamic Binary Coding

In practice, the video data we obtained often has different time resolution, so it is necessary to align the data into a same time scale by a pre-defined time-warping

operation. Most previous work did not discuss this issue and assumed the given data in the same time resolution including recent volume feature based methods. In this paper, we apply a coding scheme to handle this issue without any assumption in describing the dynamics of the similarity features.

Although facial event evolves over time from the onset, the apex, to the offset, we only take the process from the onset to the apex into account for simplicity, for this process is demonstrated to be enough for recognition in almost all the previous works. To describe the dynamics, we assume that the process from the onset to the apex is comprised of several intrinsic states (patterns) along the temporal domain. Correspondingly it means each kind of similarity feature can be divided into several patterns in temporal domain. In our experiments, we set the number of the intrinsic temporal patterns to 5 for all kinds of the similarity features. Without loss of generality, in the following we discuss how to build the five-level models for one event based on a similarity feature.

Given the training similarity feature set $F = \{F_i\}, i = 1, 2, ..., N$, where N is the number of the training samples, and each sample F_i has different resolution in temporal domain, $F_i = \{F_i^t\}$, where t is the index of frames. We perform the K-Means algorithm on the feature set F to divide it into five clusters in the temporal domain, and each cluster is modeled by a Gaussian distribution, $N^k\{\mu^k, \sigma^k\}$, $k = 1, 2, ..., 5$, where μ and *sigma* represent the mean and the variance respectively. We take these five Gaussian models as the temporal patterns of the feature F. In this paper, we will use a lot of similarity features to represent one facial event, so the temporal patterns models of one facial event are an ensemble of these Gaussian models as:

$$E = \begin{cases} N_1^1(\mu_1^1, \sigma_1^1), N_1^2(\mu_1^2, \sigma_1^2), ..., N_1^5(\mu_1^5, \sigma_1^5) \\ N_2^1(\mu_2^1, \sigma_2^1), N_2^2(\mu_2^2, \sigma_2^2)..., N_2^5(\mu_2^5, \sigma_2^5) \\ \qquad\qquad \vdots \\ N_M^1(\mu_M^1, \sigma_M^1), N_M^2(\mu_M^2, \sigma_M^2)..., N_M^5(\mu_M^5, \sigma_M^5), \end{cases} \tag{3}$$

where the subscript is the index of the similarity feature, and M is the number of the similarity features.

As mentioned in Section 1, each sequence may have different time resolution and different number of frames t due to various reasons. In order to handle this issue, we adopt the coding scheme to further convert the similarity features to the dynamic binary patterns. Given a feature sequence F_i with t frames,$\{F_i^t\}$, based the temporal pattern models described above, we can first map each F_i^t into a five-dimensional binary vector, i.e., $F_i^t \longrightarrow b_i^t = \{v_c\}$, where $c = 1, 2, ..., 5$. v_c is binary, and it is computed by the Bayesian rule as:

$$v_c = \begin{cases} 1 \; if \;\; c = \underset{k}{\operatorname{argmax}} P(F_i^t | N^k), k = 1, 2, ..., 5; \\ 0 \; otherwise, \end{cases} \tag{4}$$

where $P(F_i^t|N^k)$ means the probability of F_i^t given the corresponding Gaussian model $\{N^k\}$. We can see that there has only one element is 1 and the other four are 0 in the 5-dimensional binary feature b_i^t. It means each feature in a temporal

point only belongs to one temporal pattern. We map all the $\{F_i^t\}$ in a sequence into the five-dimensional binary feature vectors, and compute their histogram and do normalization as in [27],

$$\varphi(F_i) = \frac{\sum_{i=1}^{t} b_i^t}{t}, \tag{5}$$

where $\varphi(F_i)$ is always a five-dimensional vector whatever the t is. Thus, $\varphi(F_i)$ is independent of the time resolution. We call $\varphi(F_i)$ the dynamic binary pattern, and we use it to represent the sequence. As in [13], the binary pattern is transferred into the decimal value for the final classifier design.

4 Classifier Design

4.1 Haar-Like Appearance Descriptor

Facial event is behaved by facial appearance variations. Besides taking the gray or color values as appearance descriptor directly, there have three popular local descriptors: Gabor [27] descriptor, haar-like descriptor [28], and LBP descriptor [29]. In this paper, we use the haar-like descriptor due to its simplicity, for it has obtained a good performance for face detection and expression recognition [28] [23]. Compared to the Gabor and LBP descriptors, the haar-like descriptor only needs simple add or minus operations, so its computation cost is much lower. We perform the harr-like descriptor on each frame to extract the visual appearance features, and based on each haar-like feature, we can obtain its corresponding similarity feature. Then we convert the similarity feature to the dynamic binary pattern for learning classifier. Figure 1 shows an example how to calculate the dynamic binary pattern for a haar-like descriptor.

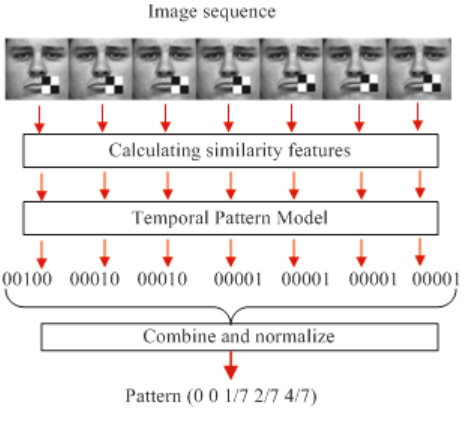

Fig. 1. Dynamic binary pattern calculation based on haar-like descriptor

4.2 Adaboost Learning

Since the number of the dynamic binary patterns is equal to the number of the haar-like features, each sequence has a large number of dynamic binary patterns. It is unrealistic to use all the dynamic binary patterns to design the classifier. Moreover, only parts of facial appearance are dominant in each facial event. Adaboost learning is a good tool to select some good features and combine them together to construct a strong classifier [28]. Therefore we adopt Adaboost to learn a set of discriminant dynamic binary patterns and use them to build the final classifier. In this paper, we take six basic facial expressions into account, i.e., happiness, sadness, angry, disgust, fear, and surprise, so it is a six-class recognition problem. We use the one-against-all strategy to decompose the six-class issue into multiple two-class issues. For each expression, we set its samples as the positive samples, and the samples of other expressions as the negative samples. Algorithm 1 summarizes the learning algorithm.

Algorithm 1. *Learning procedure*

1: Input the training image sequences (x_i, y_i),...,(x_n, y_n), $y_i \in \{+1, -1\}$.
2: Compute the similarity features for each image sequence.
3: Map the similarity features of each sequence into the dynamic binary patterns according to equation (4), and get $\varphi(F_{x_i})$.
4: Build one weak classifier on each $\varphi(F_{x_i})$.
5: Initialize weight $D_1(i) = 1/N$.
6: **for** $t = 1$ to T **do**
7: Find the classifier $h_t : \varphi(F_x) \rightarrow \{+1, -1\}$ that minimizes the error with respect to the distribution D_t. $h_t = \text{argmin}\, \varepsilon_j$, where $\varepsilon_j = \sum_{i=1}^{m} D_t(i)[y_i \neq h_j(\varphi(F_{x_i}))]$
8: Prerequisite: $\varepsilon_t < 0.5$, otherwise stop.
9: Choose $\alpha_t \in \mathbf{R}$, typically $\alpha_t = \frac{1}{2}\ln\frac{1-\epsilon_t}{\epsilon_t}$ where ε_t is the weighted error rate of classifier h_t.
10: Update: $D_{t+1}(i) = \frac{D_t(i)\, e^{-\alpha_t y_i h_t(\varphi(F_{x_i}))}}{Z_t}$, where Z_t is a normalization factor.
11: **end for**
12: Output the final classifier: $H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(\varphi(F_x))\right)$

5 Experiments

Our experiments are conducted on the Cohn-Kanade facial expression database [30], which is widely used to evaluate the facial expression recognition algorithms. This database consists of 100 students aged from 18 to 30 years old, of which 65% are female, 15% are African-American, and 3% are Asian or Latino. Subjects are instructed to perform a series of 23 facial displays, six of which are prototypic emotions mentioned above. For our experiments, we select 300 image sequences from 96 subjects. The selection criterion is that a sequence is labeled as one of the six basic emotions. We randomly select 60 subjects as the

training set, and the rest of subjects is for the testing set. The face is detected automatically by Viola's face detector [28], and it is normalized to 64×64 as in [14] based on the location of the eyes.

The proposed work is related to the haar-like volume features [22], so we first compare our work with it. For simplicity, we denote our method as the DSBP and the haar-like volume features as the 3D haar. We also investigate the robustness of the proposed method, if the training samples and the testing samples have different length and different time resolution. ROC curve is used as the measurement tool to evaluate the performance, because it is more general and reliable than the recognition rate. The number of references is set to 5 for all the haar-like features in all the experiments.

5.1 Comparison to 3D Haar-Like Features

In this subsection, we compare the DSBP with the 3D haar. As mentioned above, the 3D haar takes the video data as the 3D volume data, and performs the haar-like descriptors in the spatio-temporal domain directly on the volume data, so it needs all the input sequence with same time resolution, i.e., the data has the same length and the same motion speed. The DSBP does not make such assumption, for it embeds the time-warping process in the dynamic binary coding. For fair comparison, we compare them under the same framework, and the training samples and the testing samples have the same length, but we make the data with different time resolution. Since the sequences in the Cohn-Kanade facial database have different lengthes, we use a fixed-length window to slide over the sequences to produce the fix-length samples.

We fix the training samples with 7 frames and 9 frames respectively. Figure 2 reports the ROC curves of the comparison experiment, and table 1 reports the area below the ROC curves. We can see that the performance of the DSBP is better than that of the 3D haar. This because: 1) similarity features are able to efficiently describe complex facial appearance; 2) the dynamic binary patterns are encoded based on the statistics and the Bayesian rule, so it is robust to some noise; 3) the samples generated

from the fix-length window should have different active speeds, but the DSBP is insensitive to active speeds.

Table 1. The Area under the ROC curves (the 3D haar and the DSBP)

Expression	9(xxxxxxxxx) frames		7(xxxxxxx) frames	
	3D Haar	DSBP	3D Haar	DSBP
Angry	0.934	0.957	0.893	0.935
Disgust	0.822	0.941	0.769	0.952
Fear	0.697	0.935	0.830	0.952
Happiness	0.977	0.997	0.978	0.997
Sadness	0.758	0.963	0.875	0.917
Surprise	0.974	0.999	0.982	0.999

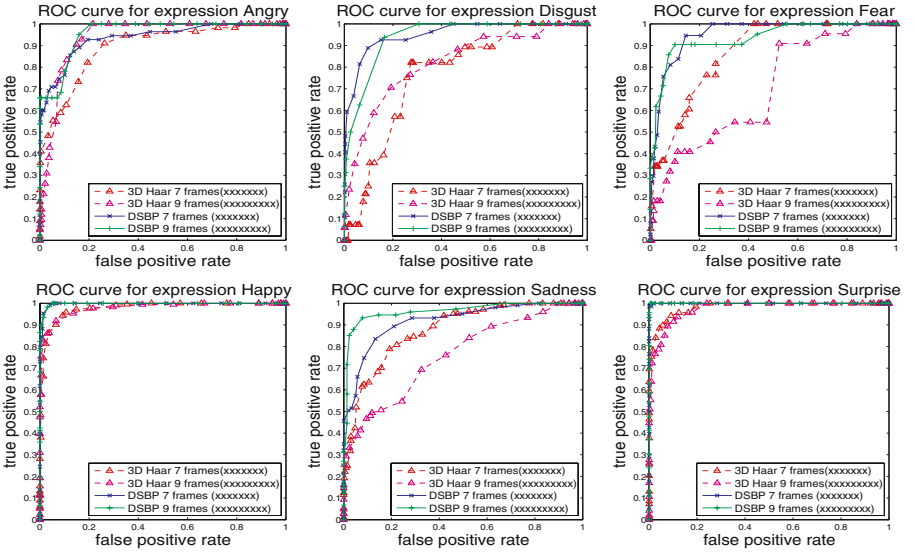

Fig. 2. ROC curves of six expressions in table 1

5.2 Robustness Analysis

The DSBP has another advantage against the 3D-haar: it has no requirement on the length of the samples. In the following, we will analyze its robustness if the training samples and the testing samples have different lengthes. We use sampling strategy to simulate this case. In following, the xxx0x0x means that we sample 5 frames from a sequence of 7 frames, where 0 means the corresponding frame is lost.

We first fix the training samples with the same length, but the length of the testing samples is variable. First, we fix the training samples with 7 frames, and the testing samples are with different number of frames. Table 2 reports a group

Table 2. The Area under the ROC curves (Training on 7(xxxxxxx) frames)

	Angry	Disgust	Fear	Happiness	Sadness	Surprisee
xxx00xxxx0xx	0.9758	0.9283	0.9623	0.9992	0.9455	1.0000
x0xxx0xxxxx0	0.9756	0.9407	0.9768	0.9991	0.9412	1.0000
xxxxxxxxxx	0.9313	0.9374	0.9401	0.9954	0.9086	0.9998
xxxxxxxxx	0.9314	0.9610	0.9471	0.9948	0.9185	0.9993
x000xxx	0.9277	0.9356	0.9537	0.9966	0.9223	0.9995
xx0xxxx	0.9499	0.9509	0.9366	0.9957	0.9136	1.0000
x0x0xxx	0.9369	0.9494	0.9486	0.9971	0.9204	0.9992
0xx0x	0.9422	0.9455	0.9503	0.9961	0.9167	0.9999
mean	0.9463	0.9436	0.9519	0.9968	0.9233	0.9997
standard variance	0.0194	0.0103	0.0128	0.0016	0.0131	0.0003

Table 3. The Area under the ROC curves (Training on 9(xxxxxxxxx) frames)

	Angry	Disgust	Fear	Happiness	Sadness	Surprise
xxx00xxxx0xx	0.9809	0.9361	0.9677	0.9985	0.9825	1.0000
x0xxx0xxxxx0	0.9780	0.9378	0.9706	0.9991	0.9713	1.0000
xxxxxxxxxx	0.9344	0.9053	0.8848	0.9970	0.9528	0.9988
xxxxxxxxx	0.9374	0.9392	0.9004	0.9951	0.9502	0.9985
x000xxx	0.9306	0.9129	0.9206	0.9969	0.9548	0.9991
xx0xxxx	0.9454	0.9217	0.9062	0.9970	0.9530	0.9991
x0x0xxx	0.9401	0.9218	0.9167	0.9969	0.9536	0.9990
0xx0x	0.9451	0.9048	0.9285	0.9965	0.9531	0.9996
mean	0.9490	0.9224	0.9244	0.9971	0.9589	0.9993
standard variance	0.0195	0.0141	0.0306	0.0012	0.0116	0.0006

of experimental results, where the length of testing samples from 12 to 5 and the sampling ratio is variant, the corresponding ROC curves are shown in 3. We can see that the DSBP is basically not influenced by the length variation of the testing data. We extend the length of the training samples to 9 frames, and use the same testing samples. Table 3 shows the experiment results and the corresponding ROC curves are displayed in 4. We can see that results are similar to those in Table 2 and 3. It means the DSBP is insensitive to the length variance and resolution variance of the testing samples. The large window size has a little better performance, because the large window captures much dynamics of the expressions.

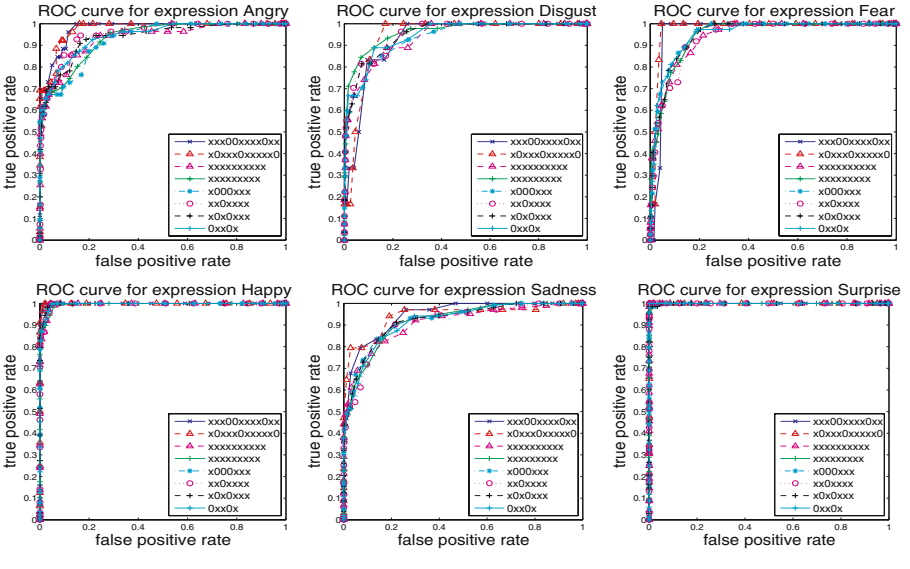

Fig. 3. ROC curves of six expressions in table 2

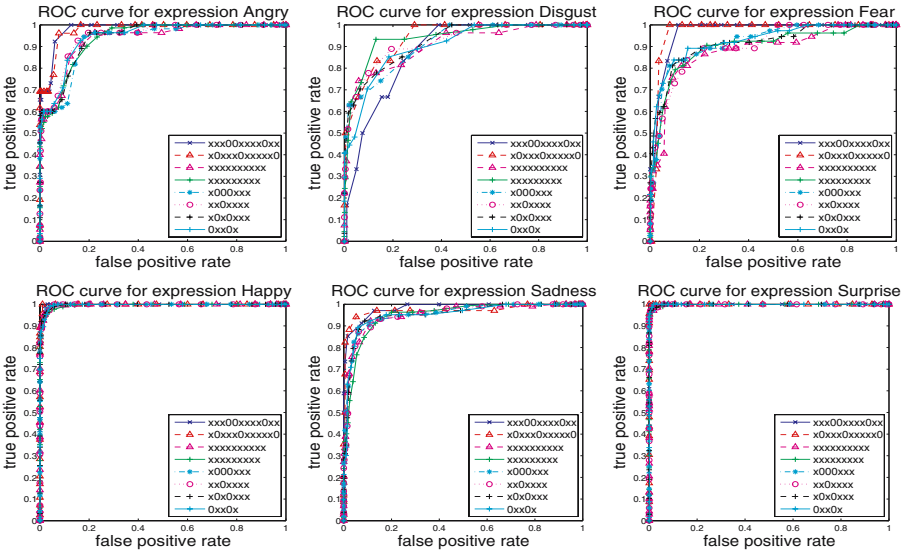

Fig. 4. ROC curves of six expressions in table 3

6 Conclusions

In this paper, we designed a novel similarity feature to describe the facial appearance for facial event analysis, which is inspired by the kernel features. The similarity feature is defined as the log-weighted summarization of the similarities between the given sample and the reference samples. We selected the references from the apexes of facial events due to their distinctness. In order to capture the dynamics of facial event, we divided the similarity features into several clusters in the temporal domain, and used the Gaussian distribution to model each cluster. Then we further mapped the similarity features into dynamic binary patterns to handle the issue of time-resolution, for this mapping processing involved the time-warping operation implicitly. The haar-like descriptor was used to extract the low-level visual features, and Adaboost was adopted to learn the final classifier. Experiments on the well-known Cohn-Kanade facial expression database showed the power of the propose method.

References

1. Fasel, B., Luettin, J.: Automatic Facial Expression Analysis: A Survey. Pattern Recognition 36, 259–275 (2003)
2. 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, 1424–1445 (2000)
3. Zeng, Z., Pantic, M., Roisman, G.I., Huang, T.S.: A survey of affect recognition methods: audio, visual and spontaneous expressions. In: Int. Conf. on Multimodal interfaces (2007)

4. Shan, C., Gong, S., McOwan, P.W.: Conditional mutual information based boosting for facial expression recognition. In: British Machine Vision Conference (2005)
5. Bartlett, M., Littlewort, G., Fasel, I., Movellan, J.: Real time face detection and facial expression recognition: Development and applications to human computer interaction. In: Computer Vision and Pattern Recognition Workshop on Human-Computer Interaction (2003)
6. Pantic, M., Rothkrantz, J.: Facial action recognition for facial expression analysis from static face images. IEEE Transactions on Systems, Man and Cybernetics (2004)
7. Shan, C., Gong, S., McOwan, P.W.: Robust facial expression recognition using local binary patterns. In: IEEE Int. Conf. on Image Processing (2005)
8. Bassili, J.: Emotion recognition: the role of facial movement and the relative importance of upper and lower areas of the face. J. Personality Socical Psychol. 37 (1979)
9. Ambadar, Z., Schooler, J., Cohn, J.F.: Deciphering the enigmatic face The importance of facial dynamics in interpreting subtle facial expression. Psychological Science (2005)
10. Black, M.J., Yacoob, Y.: Recognizing facial expressions in image sequences using local parameterized models of image motion. Int. J. Computer Vision 25, 23–48 (1997)
11. Yacoob, Y., Davis, L.: Computing spatio-temporal representations of human faces. Computer Vision and Pattern Recognition (1994)
12. Cohen, I., Sebe, N., Chen, L., Garg, A., Huang, T.: Facial expression recognition from video sequences Temporal and static modeling. Computer Vision and Image Understanding 91, 160–187 (2003)
13. Zhao, G., Pietikainen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans. Pattern Anal. Mach. Intell. 29, 915–928 (2007)
14. Tian, Y.: Evaluation of face resolution for expression analysis. In: Computer Vision and Pattern Recognition Workshop on Face Processing in Video (2004)
15. Yeasin, M., Bullot, B., Sharma, R.: From facial expression to level of interest: A spatio-temporal approach. Computer Vision and Pattern Recognition (2004)
16. Torre, F., Yacoob, Y., Davis, L.: A probabilistic framework for rigid and non-rigid appearance based tracking and recognition. In: The Fourth IEEE Int. Conf. on Automatic Face and Gesture Recognition (2001)
17. Cohn, J.: Automated analysis of the configuration and timing of facial expression. What the face reveals (2nd edition): Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS), 388 – 392 (2005)
18. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models: their training and application. Comput. Vis. Image Underst. 61, 38–59 (1995)
19. Hu, C., Chang, Y., Feris, R., Turk, M.: Manifold based analysis of facial expression. In: Computer Vision and Pattern Recognition Workshop (2004)
20. Lee, C.S., Elgammal, A.: Facial expression analysis using nonlinear decomposable generative models. In: Zhao, W., Gong, S., Tang, X. (eds.) AMFG 2005. LNCS, vol. 3723, pp. 17–31. Springer, Heidelberg (2005)
21. Ke, Y., Sukthankar, R., Hebert, M.: Efficient visual event detection using volumetric features. In: IEEE International Conference on Computer Vision (2005)
22. Cui, X., Liu, Y., Shan, S., Chen, X., Gao, W.: 3d haar-like features for pedestrian detection. In: IEEE International Conference on Multimedia and Expo. (2007)

23. Yang, P., Liu, Q., Metaxas, D.N.: Boosting coded dynamic features for facial action units and facial expression recognition. Computer Vision and Pattern Recognition (2007)
24. Tversky, A.: Features of similarity. Psychological Review (1977)
25. Liu, Q., Jin, H., Tang, X., Lu, H., Ma, S.: A new extension of kernel feature and its application for visual recognition. Neurocomput. 71, 1850–1856 (2008)
26. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 2, 121–167 (1998)
27. Daugman, J.: Demodulation by complex-valued wavelets for stochastic pattern recognition. Int'l J. Wavelets, Multiresolution and Information Processing (2003)
28. Viola, P., Jones, M.: Robust real-time object detection. Int. J. Computer Vision 57, 137–154
29. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. 24, 971–987 (2002)
30. Kanade, T., Cohn, J., Tian, Y.L.: Comprehensive database for facial expression analysis. In: Proceedings of the 4th IEEE Int. Conf. on Automatic Face and Gesture Recognition (FG 2000) (2000)