3D Facial Expression Recognition Based on Histograms of Surface Differential Quantities

Huibin Li[1,2], Jean-Marie Morvan[1,3,4], and Liming Chen[1,2]

[1] Université de Lyon, CNRS
[2] Ecole Centrale de Lyon, LIRIS UMR5205, F-69134, Lyon, France
[3] Université Lyon 1, Institut Camille Jordan,
43 blvd du 11 Novembre 1918, F-69622 Villeurbanne - Cedex, France
[4] King Abdullah University of Science and Technology, GMSV Research Center,
Bldg 1, Thuwal 23955-6900, Saudi Arabia
{huibin.li,liming.chen}@ec-lyon.fr,
morvan@math.univ-lyon1.fr

Abstract. 3D face models accurately capture facial surfaces, making it possible for precise description of facial activities. In this paper, we present a novel mesh-based method for 3D facial expression recognition using two local shape descriptors. To characterize shape information of the local neighborhood of facial landmarks, we calculate the weighted statistical distributions of surface differential quantities, including histogram of mesh gradient (HoG) and histogram of shape index (HoS). Normal cycle theory based curvature estimation method is employed on 3D face models along with the common cubic fitting curvature estimation method for the purpose of comparison. Based on the basic fact that different expressions involve different local shape deformations, the SVM classifier with both linear and RBF kernels outperforms the state of the art results on the subset of the BU-3DFE database with the same experimental setting.

Keywords: 3D facial expression recognition, normal cycle theory, curvature tensor, histogram of surface differential quantities, SVM classifier.

1 Introduction

Facial expression recognition (FER) has been an intensive subject for the last three decades because of its usefulness in many applications such as human-computer interaction and the analysis of conversation structure [1]. Acknowledging that facial expressions are actuated by contraction of facial muscles, Ekman et al. [2] introduced the Facial Action Coding System (FACS) and evidenced six universal prototypic facial expressions, namely happiness, sadness, anger, fear, surprise and disgust along with neutral. There exists currently an impressive body of results on FER firstly in static 2D images, and then dynamic 2D videos [3] and more recently on static 3D scans [4,5,6,7,8,9,10,11] and dynamic 3D videos [12,13]. With 3D imaging systems readily available, the use of 3D

J. Blanc-Talon et al. (Eds.): ACIVS 2011, LNCS 6915, pp. 483–494, 2011.

facial data for FER has attracted increasing interest since 3D data are theoretically pose invariant and robust to illumination changes. Furthermore, they also capture accurate geometry information closely sensitive to expression variations.

Existing methods for FER based on static 3D data can be categorized into two streams, i.e. feature based and model based. The first category claims that the distributions of facial surface geometric information such as gradient, curvature [5], distances between pairs of interest landmarks [6] and local shapes near landmarks [10,11], are closely related to expression categories. These geometric information are then extracted as features and fed to various classifiers, such as linear discriminant analysis (LDA), support vector machine (SVM) or neural network etc., for FER. The main drawback of this kind of approaches is that they require a set of accurately located landmarks. This explains that most of these work made use of the 83 landmarks manually labeled on the BU-3DFE dataset. The second category tries to simulate the physical process of generating expression and explores a generic elastically deformable face model, which can generate universal expressions by adjusting parameters [7]. In general, this kind of methods needs alignment and normalization steps to find one-to-one correspondence among 3D faces, then the shape deformations between each pair of faces can be represented by model parameters [7] or feature vectors [9] etc, which are further used to perform FER. The downside of these approaches is first their computational cost which is expensive. Furthermore, the fitting process also require existence of some landmarks for initialization and hardly converge in case of opening of the mouth in facial expressions.

In this paper, we propose a new feature based method for FER using two local shape descriptors. To characterize shape information of the local neighborhood of facial landmarks, weighted statistical distributions of surface differential quantities, including histogram of mesh gradient (HoG) and histogram of shape index (HoS) are extracted as local shape descriptors from facial landmarks. These local shape descriptors can be considered as an extension of the popular 2D SIFT feature [14] (Scale Invariant Feature Transform) to 3D-mesh based discrete surfaces [15]. Normal cycle theory based curvature estimation method [17,18,19] is employed for the first time on 3D face models along with the popular cubic fitting curvature estimation method for the purpose of comparison. The local shape descriptors of 60 landmarks are then concatenated into a global shape descriptor according to the manual labeled order. Based on the basic fact that different expressions involve different local shape deformations, the global shape descriptors are fed to a SVM classifier using both linear and RBF kernels to perform facial expression recognition. The proposed approach was tested on the BU-3DFE database with the same experimental setting as its main competitors and the experimental results show that the proposed approach outperforms the state of the art.

The remainder of this paper is organized as follows: differential quantities estimated on mesh-based facial models are introduced in section 2. Section 3 presents the local shape descriptors. Experimental results are discussed in section 4. Section 5 concludes the paper.

2 Estimating Differential Quantities on Triangular Meshes

2.1 Estimating Curvature by Normal Cycle Theory Based Method

There are many approaches to calculate curvature on triangular meshes based on estimation of curvature tensor. Taubin [16] introduced a 3D curvature tensor from which the principal curvatures directions can be estimated by two of the three eigenvectors and the principal curvature can be computed by linear combinations of two of the three eigenvalues. Cohen-Steiner and Morvan introduced [17,18] a discrete definition for the mean, Gaussian curvature and the curvature tensor based on normal cycle theory, and proved that the estimated curvature tensors converge to the true ones of the smooth surface under specific sampling conditions. The basic idea and its discrete form can be carried out as follows [19]:

For every edge e of the mesh, there is an obvious minimum (i.e., along the edge) and maximum (i.e., across the edge) curvature. A nature curvature tensor can therefore be defined at each point along an edge, named as generalized curvatures [18]. This line density of tensors can now be integrated over the arbitrary region B by summing the different contributions from B, leading to the simple expression:

$$\mathcal{T}(v) = \frac{1}{B} \sum_{edges\ e} \beta(e)|e \cap B|\, \overline{e}\,\overline{e}^t \tag{1}$$

Where v is an arbitrary vertex on the mesh, $|B|$ is the surface area around v over which the tensor is estimated. $\beta(e)$ is the signed angle between the normals to the two oriented triangles incident to edge e (positive if convex, negative if cancave), $|e \cap B|$ is the length of $e \cap B$ (always between 0 and e), and $\overline{e}$ is a unit vector in the same direction as e. In our experiments, we estimate the tensor at every vertex location v, for a neighborhood B of 2-ring. The principal curvatures k_{min} and k_{max} at v can now be estimated by the two maximum eigenvalues of $\mathcal{T}(v)$. Fig. 1 shows the schematic of this method.

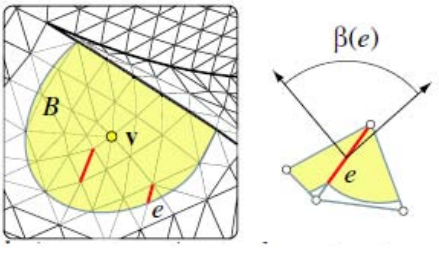

Fig. 1. Illustration of normal cycle theory based curvature estimation method (equation (1) [19])

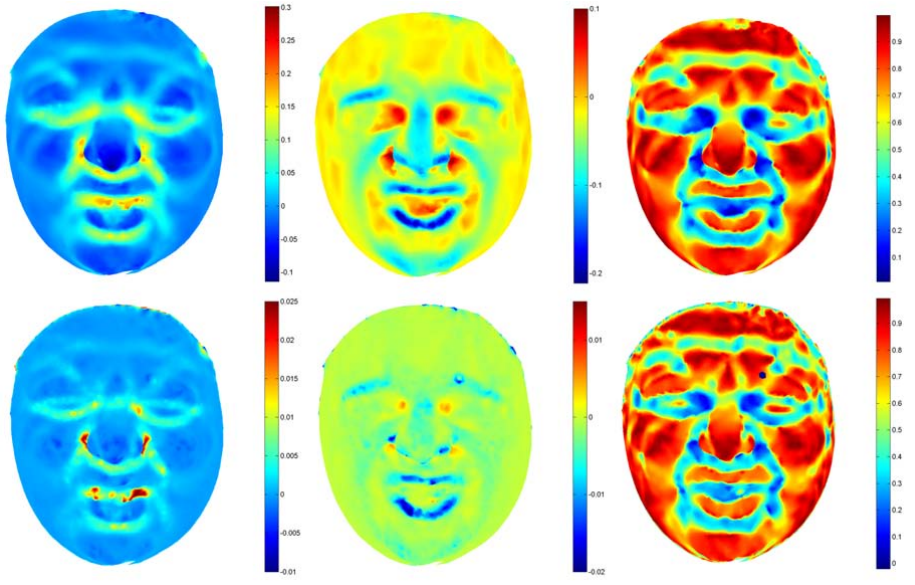

Fig. 2. First row, from left to right: κ_{max}, κ_{min} and shape index estimated by normal cycle theory based method. Second row, from left to right: κ_{max}, κ_{min} and shape index estimated by cubic fitting based method, (model M0044-DI03).

Shape index which expresses different shape classes by a single number ranging from 0 to 1 can then be estimated by the following equation:

$$\mathbf{S} = \frac{1}{2} - \frac{1}{\pi} arctan(\frac{\kappa_{max} + \kappa_{min}}{\kappa_{max} - \kappa_{min}}) \tag{2}$$

The first row of Fig.2 shows an example of maximum, minimum curvatures and shape index estimated by this method on a 3D face model.

2.2 Estimating Curvature by Local Cubic Fitting Based Method

In this paper, we also compare the performance of our shape descriptors using different curvature estimation methods. For this purpose, we also adopt local cubic fitting method [20] to estimate curvatures. The basic idea of this method is that: for each vertex p of the 3D mesh, a local coordinate system is defined by taking the vertex p as an origin and normal vector $n_p = (n_x, n_y, n_z)^T$ as the z axes. Two orthogonal axes, x and y, randomly chosen in the tangent plane perpendicular to the normal vector. The local neighborhood points (2-ring in our paper) and its corresponding normal vectors are first transformed to the local coordinate system, then used for fitting a cubic function and its normal respectively. The cubic function and its normal having the following forms:

$$z(x, y) = \frac{A}{2}x^2 + Bxy + \frac{C}{2}y^2 + Dx^3 + Ex^2y + Fxy^2 + Gy^3 \tag{3}$$

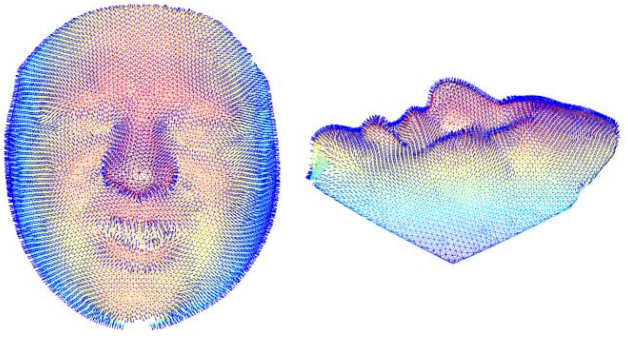

Fig. 3. Illustrated the estimated normal vectors (model M0044-DI03)

$$(z_x, z_y, -1) = (Ax + By + 3Dx^2 + 2Exy + Fy^2 + \\ + Bx + Cy + Ex^2 + 2Fxy + 3Gy^2, -1) \quad (4)$$

By using least-square fitting method to solve the fitting equations (3) and (4), the Weingarten matrix on a vertex can be computed as:

$$\mathbf{W} = \begin{pmatrix} \frac{\partial^2 z(x,y)}{\partial x^2} & \frac{\partial^2 z(x,y)}{\partial x \partial y} \\ \frac{\partial^2 z(x,y)}{\partial x \partial y} & \frac{\partial^2 z(x,y)}{\partial y^2} \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \quad (5)$$

The maximum curvature k_{max} and minimum curvature k_{min} then can be estimated as the eigenvalues of the Weingarten matrix. The second row of Fig.2 shows a example of maximum, minimum curvatures and shape index estimated by cubic fitting method on the same 3D face model.

2.3 Mesh Gradient Estimation

Let the normal vector at p as $n_p = (n_x, n_y, n_z)^T$, which can be estimated by averaging the normal vectors of one-ring faces. According to (3) and (4), the gradient direction and the gradient magnitude can be estimated as follows:

$$\theta = arctan(\frac{n_y}{n_x}) \quad (6)$$

$$||\nabla z(x,y)|| = \sqrt{(-\frac{n_x}{n_z})^2 + (-\frac{n_y}{n_z})^2} \quad (7)$$

Fig. 3 shows the estimated normal vectors on a 3D mesh-based face model.

3 Local Shape Descriptors

3.1 Landmarks Selection

A set of landmarks, e.g. eye corners, nose tip, etc. is used for the purpose of FER. For each landmark, two local shape descriptors, namely HoG and HoS, are

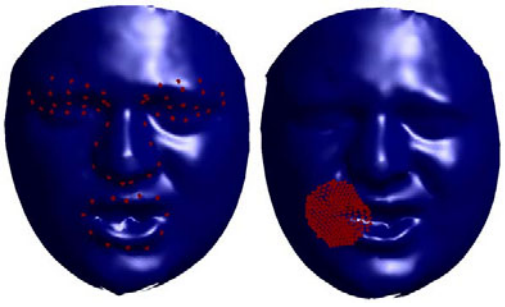

Fig. 4. From left to right, 60 selected manual landmarks, local neighborhood points of the left mouth corner (M0044-DI04 and M0044-DI03)

extracted from its neighborhood. In this work, a neighborhood with a geodesic disk is considered. The experiments were carried out on the BU-3DFE database using the first 60 landmarks defined on the regions of eyebrows, eyes, nose and mouth. These landmarks are a subset of 83 manually labeled landmarks with a specified order defining in the dataset. The radius of the disk is equal to 22 mm in our experiments. The 60 selected landmarks and an example of local neighborhood of the left mouth corner are shown in Fig. 4.

3.2 Local Coordinate System and Orientation Assignment

In practice, we first transform the points within a local neighborhood to a local coordinate system, in which the landmark point is the origin and its normal vector is along the positive z axis. Two perpendicular vectors x and y axis are randomly chosen in the tangent plane. In order to make the descriptor invariant to rotation, each landmark point is assigned one or several canonical orientations according to the dominant direction(s) of gradients in the local tangent plane with 360 bins. Once the canonical orientations are assigned, the local coordinate system rotates in the local tangent plane, making each canonical orientation as new x axis. Now y axis can be computed by cross product of z and x.

In this new local coordinate system, we project the neighbor points to the tangent plane of its corresponding landmark. Eight projected points along to eight quantized directions starting from canonical orientation with a distance of r_1 to the landmark point are fixed. Nine circles centered at the landmark point and its eight neighbors with a radius r_2 can be further located. Fig.5 shows this arrangement.

3.3 Feature Vectors Computed as Histograms of Surface Differential Quantities

In each circle, we calculate the histogram of surface gradient (hog^c) and histogram of shape index (hos^c). For hog^c, we compute the histogram of gradient angle weighted by gradient magnitude. This histogram has 8 bins representing

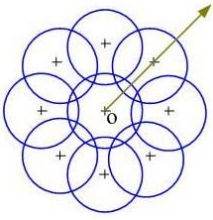

Fig. 5. Canonical orientation (arrow), landmark point (o) and its 8 neighborhood vertices (+) assigned with 9 circles

8 main orientations ranging form 0 to 360 degree. For hos^c, the values of shape index ranging from 0 to 1 are also quantized to 8 bins. Then, all the values of histograms are weighted by a Gaussian kernel with the Euclidian distance to the center point of the circle as the standard deviation. Every histogram with the length of 72 is then normalized and form a feature vector for a single landmark point. Formally, hog^p and hos^p are defined as follows:

$$hog^p = (hog_1^c, hog_2^c, \ldots, hog_9^c) \tag{8}$$

$$hos^p = (hos_1^c, hos_2^c, \ldots, hos_9^c) \tag{9}$$

The final feature vectors of each face model, HoG and HoS, are then obtained respectively by simply concatenating the corresponding histograms of all 60 manual landmarks in a fixed order, giving birth to a $60 \times 72 = 4320$ length bins. They can be represented as follows:

$$HoG = (hog_1^p, hog_2^p, \cdots, hog_{60}^p) \tag{10}$$

$$HoS = (hos_1^p, hos_2^p, \cdots, hos_{60}^p) \tag{11}$$

Using an early fusion strategy, the fusion of these two local shape descriptors, HoG+HoS, is a simple concatenation of these two vectors.

4 Experimental Results

In our experiments, the BU-3DFE database [4] consisting of 2500 textured 3D face models of 100 subjects with different gender, race, and age is used. Each subject contains one neutral model and six universal non-neutral expressions: happiness, sadness, anger, fear, surprise and disgust. Also, each non-neutral expression is displayed at four different intensities: low, middle, high and highest. Fig.5 shows some samples of the database with six different expressions of high and highest intensities.

For fair comparison, we used the same experimental settings as [9]. A subset of 60 subjects was randomly selected with two high-intensity models for each of

Fig. 6. Examples of six universal expressions, from left to right: anger, disgust, fear, happiness, sadness, surprise. First row, high intensity, Second row, highest intensity.

the six facial expressions. In total, $60 \times 12 = 720$ 3D mesh-face models were selected. Then, 54 and 6 subjects were randomly selected respectively as training set (648 models) and test set (72 models). In practice, since the number of models in test set is very limited (only 12 models for each expression), and people of different race, gender and age may have different facial surface changes when they perform the same expression, the average recognition accuracy obtained by 10 or 20 random experiments varies greatly, from about 50% to more than 90% [9] with the same feature set, classifier and parameters setup. To obtain stable average recognition accuracies, we run all of our experiments 1000 times independently. To calculate descriptors, we set r_1 and r_2 equal to 15 mm and 7 mm respectively. With a dimension of 4320 for both HoG and HoS, it is hard to get the information of their distributions in such high dimension space. We made use of the SVM (LIBSVM [21]) classifier with both liner kernel and RBF (radial basis function) kernel for final FER. The parameter 'gamma' for RBF kernel was set to 0.04 by 8-fold cross-validation. The results are shown in table 1-5. AN, DI, FE, HA, SA, SU designate anger, disgust, fear, happiness, sadness, and surprise, respectively.

Table 1 shows the average confusion matrix obtained by the HoG feature using the SVM classifier with linear and RBF kernels. We can find that both linear and RBF kernels achieve relatively high recognition accuracies for happiness and surprise and lower accuracies for other expressions. We observe the same phenomenon already evidenced in other work: anger and fear have comparatively lower classification rates. Anger is confused mainly by sadness while fear is confused mainly by happiness and surprise. Linear kernel works slightly better for anger while slightly worse for disgust and happiness as compared to the RBF kernel. Both of the two kernels achieve almost the same average recognition rate about 76.5% for all six expressions.

Table 2 shows the average confusion matrix obtained by the HoS feature (estimated by normal cycle theory based method) and the SVM classifier of linear and RBF kernels. All the results are better than the accuracies in Table 1 except the one obtained by the RBF kernel for sadness (72%, vs 72.6%). Linear kernel

Table 1. Average confusion matrix obtained by HoG

	linear kernel						RBF kernel					
%	AN	DI	FE	HA	SA	SU	AN	DI	FE	HA	SA	SU
AN	**74.0**	8.3	0.8	1.6	15.2	0.1	**66.0**	12.0	3.0	0.9	18.1	0
DI	5.9	**76.5**	10.4	3.7	2.6	0.9	2.3	**80.7**	8.5	3.8	3.2	1.5
FE	5.4	11.7	**63.6**	9.7	5.4	4.2	3.4	8.4	**63.2**	13.2	5.5	6.2
HA	1.1	0.9	16.0	**82.0**	0	0	0.1	1.4	12.6	**85.6**	0	0.3
SA	16.6	2.7	8.2	0	**72.2**	0.3	15.6	4.1	5.7	0	**72.6**	2.0
SU	1.0	2.3	4.9	0.3	1.5	**90.0**	0	2.6	4.1	1.6	0.9	**90.7**
Average	**76.4**						**76.5**					

Table 2. Average confusion matrix obtained by normal cycle based HoS

	linear kernel						RBF kernel					
%	AN	DI	FE	HA	SA	SU	AN	DI	FE	HA	SA	SU
AN	**77.0**	8.5	0.9	0	13.3	0.4	**71.5**	10.4	2.2	0	15.2	0.7
DI	7.6	**80.0**	6.2	2.8	3.4	0	3.6	**82.0**	6.5	3.4	4.4	0
FE	5.2	5.9	**70.9**	8.9	5.9	3.2	4.4	8.1	**65.6**	12.0	6.1	4.0
HA	0.4	0.9	4.8	**93.2**	0	0	0.6	1.0	6.5	**91.2**	0	0.7
SA	15.5	1.8	7.3	0	**74.4**	0.9	14.1	2.4	10.1	0	**72.0**	1.5
SU	0.2	1.9	1.9	0.4	0	**95.6**	0.1	1.6	1.4	0.2	0	**96.8**
Average	**81.9**						**79.9**					

works better than the RBF kernel except disgust (80.0% vs 82.0%) and surprise (95.6% vs 96.8%). The average recognition rates for all six expressions are 81.9% and 79.9% for linear and RBF kernels, respectively.

Table 3 shows the average confusion matrix obtained by the HoS feature (estimated by local cubic fitting method) and SVM classifier of linear and RBF kernels. All the results are better than the corresponding results in table 1 except the one obtained by linear kernel for anger (73.1%, vs 74.0%). For the case of linear kernel, the performances are worse than the ones in table 2 except for disgust (80.6% vs 80.0%), fear (72.3% vs 70.9%) and sadness (76.3% vs 74.4%); while for the case of RBF kernel, the performances are a little better than table 2 (80.6% vs 79.9%) especially for sadness (76.2% vs 72%). The two curvature estimation methods achieve comparative results which are in contrast to the conclusion made in [5]. On the other side, comparing the computational complexity of the two curvature estimating methods, the local cubic fitting method which needs to solve the linear system equation at each point takes more computation cost.

Table 4 presents the average confusion matrix obtained by fusing the features HoG and HoS into a single feature according to a simple early fusion scheme, denoted as HoG+HoS (estimated by normal cycle theory and cubic fitting), and a SVM classifier with a linear kernel. Compared to the results in left column of table 2 and the one of table 3 also achived by a SVM with a linear kernel, the fused feature HoG+HoS, when estimated using the normal cycle theory for the

Table 3. Average confusion matrix obtained by cubic fitting based HoS

	linear kernel						RBF kernel					
%	AN	DI	FE	HA	SA	SU	AN	DI	FE	HA	SA	SU
AN	**73.1**	7.7	2.7	0.2	16.2	0	**72.3**	9.9	2.6	0	15.2	0
DI	5.1	**80.6**	9.3	3.0	1.7	0.5	3.3	**81.3**	8.5	2.9	2.7	1.2
FE	5.4	4.8	**72.3**	8.7	6.0	2.7	5.5	73.3	**66.6**	9.5	6.7	4.4
HA	0.8	3.9	4.2	**90.5**	0	0.7	0.9	2.2	5.2	**91.1**	0	0.7
SA	16.3	0.6	5.9	0	**76.3**	0.9	12.8	1.5	8.6	0	**76.2**	0.9
SU	0.1	2.3	3.0	0	0.8	**93.9**	0.1	2.3	1.1	0	0.6	**95.9**
Average	**81.1**						**80.6**					

Table 4. Average confusion matrix obtained by HoG+HoS descriptor using a linear kernel with the SVM

	normal cycle based						cubic fitting based					
%	AN	DI	FE	HA	SA	SU	AN	DI	FE	HA	SA	SU
AN	**76.8**	7.6	2.1	0	13.5	0	**76.4**	8.0	1.8	0	13.7	0
DI	7.6	**78.1**	6.6	2.1	5.0	0.7	4.4	**80.2**	10.2	2.8	2.0	0.5
FE	4.6	7.6	**73.2**	7.3	5.1	2.3	5.1	6.2	**73.6**	8.0	5.3	1.7
HA	0.5	0.5	6.8	**91.4**	0	0.8	0.8	2.2	6.5	**90.4**	0	0.8
SA	14.5	1.1	8.3	0	**75.5**	0.6	14.7	0.5	6.2	0	**77.8**	0.8
SU	0	1.7	2.0	0.9	0.8	**94.5**	0	2.0	3.3	0.1	1.0	**93.6**
Average	**81.6**						**82.0**					

computation of curvature, ameliorates the recognition accuracy for sadness and fear while recording a slight performance drop for other expressions. Now when estimated using the cubic fitting technique for the computation of curvature, the fused feature HoG+HoS improves the recognition accuracies for sadness, fear along with anger this time and also records a slight drop for the remaining expressions. Overall, there are no significant increase of recognition accuracies either using normal cycle or cubic fitting in the case of the early fused feature vector HoG+HoS. This may be caused by the huge dimension (8640×1) of the fused descriptor on the one hand and the different nature of geometric information captured by HoG and HoS. It may be interesting to study some other fusion strategies, for instance late fusion strategy in combining similarity scores.

Table 5 compares the experimental results achieved by the proposed method (HoG by RBF kernel, HoS by normal cycle and linear kernel, HoG+HoS by cubic fitting and linear kernel) with the ones reported in [9] and [11]. In fact, the results of the approaches proposed by Gong et al. (Gong) [9], Wang et al. (Wang) [5], Soyel et al. (Soyel) [6], and Tang et al. (Tang) [8], were obtained using the same experimental setting. In Berretti et al. (Berretti) [11], 60 subjects were selected randomly from experiment to experiment. It can be seen from the table that the proposed approach using HoG feature achieves comparable result as the others while HoS and HoG+HoS features outperform all the other methods.

Table 5. Comparison of the proposed method with the state of the art [11], [9], [5], [6], [8]

	HoG	HoS	HoG+HoS	Berretti	Gong	Wang	Soyel	Tang
AVE	**76.48%**	**81.86%**	**82.01%**	**77.54%**	**76.22%**	**61.79%**	**67.52%**	**74.51%**

5 Conclusion and Future Work

In this paper, we have developed a mesh-based 3D facial expression recognition approach and evaluated it on the BU-3DFE database. The proposed approach is based on two local shape descriptors, namely HoG and HoS, achieved by computing the histograms of normal vector and shape index within the local neighborhood of landmarks respectively. In this work, we selected the first 60 of the 83 ordering manually labeled landmarks in the BU3D-FE database. Curvatures are estimated using normal cycle theory based method and cubic fitting method. Both linear and RBF kernels of SVM are employed for classification. The experimental results show that the proposed approach outperforms the state of the art and demonstrate its effectiveness.

In our future work, we will investigate alternative fusion schemes than the simple early fusion scheme used in this work. Furthermore, we also want to study the stability of the proposed approach when landmarks are automatically located for instance by our statistical technique SFAM as proposed in [22].

References

1. Otsuka, K., Sawada, H., Yamato, J.: Automatic inference of cross-modal nonverbal interactions in multiparty conversations: "who responds to whom, when, and how?" from gaze, head gestures, and utterances. In: ICMI, pp. 255–262. ACM Press, New York (2007)
2. Ekman, P.: Universals and cultural differences in facial expressions of emotion. In: Nebraska Symposium on Motivation, Lincoln, NE, pp. 207–283 (1972)
3. Fasel, B., Luettin, J.: Automatic facial expression analysis: a survey. In: Pattern Recognition, pp. 259–275 (2003)
4. Yin, L., Wei, X., Sun, Y., Wang, J., Rosato, M.: A 3D Facial Expression Database For Facial Behavior Research. In: The 7th International Conference on Automatic Face and Gesture Recognition (FG), pp. 211–216. IEEE Computer Society, Los Alamitos (2006); TC PAMI
5. Wang, J., Yin, L., Wei, X., Sun, Y.: 3d facial expression recognition based on primitive surface feature distribution. In: CVPR, pp. 1399–1406 (2006)
6. Soyel, H., Demirel, H.: Facial expression recognition using 3d facial feature distances. In: Int. Conf. on Image Analysis and Recognition, pp. 831–838 (2007)
7. Mpiperis, I., Malassiotis, S., Strintzis, M.G.: Bilinear models for 3-d face and facial expression recognition. IEEE Transactions on Information Forensics and Security 3(3), 498–511 (2008)
8. Tang, H., Huang, T.S.: 3d facial expression recognition based on automatically selected features. In: Int. Conf. on Computer Vision and Pattern Recognition, pp. 1–8 (2008)

9. Gong, B., Wang, Y., Liu, J., Tang, X.: Automatic facial expression recognition on a single 3d face by exploring shape deformation. In: Int. Conf. on Multimedia, pp. 569–572 (2009)
10. Maalej, A., Ben Amor, B., Daoudi, M., Srivastava, A., Berretti, S.: Local 3D Shape Analysis for Facial Expression Recognition. In: ICPR, pp. 4129–4132. IEEE, Los Alamitos (2010)
11. Berretti, S., Bimbo, A.D., Pala, P., Ben Amor, B., Daoudi, M.: A Set of Selected SIFT Features for 3D Facial Expression Recognition. In: ICPR, pp. 4125–4128. IEEE, Los Alamitos (2010)
12. Yin, L., Chen, X., Sun, Y., Worm, T., Reale, M.: A High-Resolution 3D Dynamic Facial Expression Database. In: IEEE Int. Conference on Automatic Face Gesture Recognition, pp. 1–6 (2008)
13. Sun, Y., Yin, L.: Facial Expression Recognition Based on 3D Dynamic Range Model Sequences. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 58–71. Springer, Heidelberg (2008)
14. Lowe, D.G.: Distinctive image features from scale invariant keypoints. IJCV, 91–110 (2004)
15. Fabry, M.C., et al.: Feature detection on 3d face surfaces for pose normalisation and recognition. In: BTAS (2010)
16. Taubin, G.: Estimating the tensor of curvature of a surface from a polyhedral approximation. In: ICCV, p. 902. IEEE Computer Society, USA (1995)
17. Cohen-Steiner, D., Morvan, J.M.: Restricted delaunay triangulations and normal cycle. In: Proceedings of the Nineteenth Annual Symposium on Computational Geometry, pp. 312–321. ACM, New York (2003)
18. Morvan, J.M.: Generalized Curvatures. Springer, Berlin (2008)
19. Alliez, P., Cohen-Steiner, D., Devillers, O., Lévy, B., Desbrun, M.: Anisotropic polygonal remeshing. ACM Trans. Graph. 22(3), 485–493 (2003)
20. Goldfeather, J., Interrante, V.: A novel cubic-order algorithm for approximating principal direction vectors. ACM Trans. Graph., 45–63 (2004)
21. Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines (2001), `http://www.csie.ntu.edu.tw/~cjlin/libsvm`
22. Zhao, X., Szeptycki, P., Dellandrea, E., Chen, L.: Precise 2.5D Facial Landmarking via an Analysis by Synthesis approach. In: 2009 IEEE Workshop on Applications of Computer Vision (WACV 2009), Snowbird, Utah (2009)