Unsupervised Posture Modeling and Recognition based on Gaussian Mixture Model and EM Estimation

Xijun Zhu
College of Information Science and Technology, Qingdao University of Science & Technology, Qingdao China
Email:zhuxj990@163.com

Chuanxu Wang
College of Information Science and Technology, Qingdao University of Science & Technology, Qingdao China
Email:wangchuanxu_qd@163.com

***Abstract*—In this paper, we proposed an unsupervised posture modeling method based on Gaussian Mixture Model (GMM). Specifically, each learning posture is described based on its movement features by a set of spatial-temporal interest points (STIPs), salient postures are then clustered from these training samples by an unsupervised algorithm, here we give the comparison of four candidate classification methods and find the optimal one. Furthermore, each clustered posture type is modeled with GMM according to Expectation Maximization (EM) estimation. The experiment results proved that our method can effectively model postures and can be used for posture recognition in video.**

***Index Terms*—NERF C-means; posture modeling; posture recognition; GMM**

I. INTRODUCTION

Automatic recognition of actions is a challenging problem and is highly essential in an intelligent video surveillance system [1-3]. An action can be characterized by a sequence of salient postures, each posture representing a specific configuration of the body parts. Extraction and modeling of postures have been central to the human action recognition. Traditional methods[4-5] for posture extraction and modeling are often based on the extraction of the human body or silhouettes using background modeling techniques. However, these methods are usually subject to the interference of illuminance variation, shadow and occlusion.

As well-known, an action bears both spatial and temporal characteristics while silhouettes have been widely used to characterize the spatial information, spatial-temporal interest points (STIPs) [6] have recently emerged as an effective way to capture both local spatial and temporal information without a need for background modelling. This paper proposes to characterize the each posture using a set of the spatio-temporal interesting points (STIPs). Local features are extracted at each STIP and each learning posture is modelled with the histogram of these local features, then these learning samples are clustered with an optimal classification algorithm.

Clustering is a mathematical tool that attempts to discover structures or certain patterns in a data set, where the objects inside each cluster show a certain degree of similarity. Posture similarity is a kind of fussy measurement. In the framework of fuzzy clustering, it allows each feature vector to belong to more than one cluster with different membership degrees (between 0 and 1) and vague or fuzzy boundaries between clusters [7]. In fuzzy relational clustering, the problem of classifying data is solved by expressing a relation that quantifies the similarity, or dissimilarity, degree between pairs of objects. Based on such relation, objects very similar to each other, i.e., objects of the same type, will belong with high membership values to the same cluster [8], but this methods can be used to cluster a set of n objects described by pair-wise dissimilarity values if (and only if) there exist n points in Rn-1 whose squared Euclidean distances precisely match the given dissimilarity data. NERF C-means was designed to turn none Euclidean relational data into Euclidean relational with β-spread transform in order to get rid of the above constrains on traditional RF C-means [9]. And also an alternative way of NERF C-means is proposed recently for any relational data clustering [10].

In this paper, we proposed a posture modeling method with Gaussian Mixture Model based on EM estimation. It can effectively solve the problems caused by interferences from foreground segmentation, like illumination variation and camouflage. The orgnizition of this paper are as followings, part 2 introduces the extraction of STIPs from human action videos and gives the descriptor for each pose for learning. In part 3, we make the comparison the four kind of fuzzy clustering methods and also give the clustering results, which are salient postures. while part 4 gives discussion on how to model these clustering results. in part 5 we discuss the posture matching algorithm and in part 6 there is the test and conclusions..

II. STIP EXTRACTION AND POSTURE DESCRIPTOR MODELLING

doi:10.4304/jsw.6.8.1445-1451

Human action features are located at the spatial temporal neighborhood, where the image values have large variations in both the spatial and the temporal dimensions [11]. We can use fewer feature points to identify human's movement behavior, without the need of segmenting and tracking human any more. Points with such properties will be spatial-temporal points with a distinct location in time corresponding to the moments with non-constant motion of the image in a local spatial-temporal neighborhood. For example, during the walking process, feet lift and land, knees bend and so on

A. Extraction of STIPs

There are two methods to extract these spatial-temporal interesting points, which are proposed by Ivan Laptov and Dollár. The method of Ivan Laptov is to detect 3-dimensional Harris corner as STIPs, which are sparse and sensitive to scales, and it is not adaptive to posture modeling. So we choose the STIPs extraction method based on Dollár

Figure.1 STIP extraction based on Ivan Laptov

in this paper. The extracted STIPs based on Ivan Laptov [12] are shown in Figure 1.

Compared to Ivan Laptov, Dollár's [13] method considered that any region with spatially distinguishing characteristics undergoing a complex motion can induce a strong response. The response function can be calculated as: $R = (I * g * h_{ev})^2 + (I * g * h_{od})^2$ (1)

Where I is input gray video, and $g(x, y; \sigma)$ is the 2D Gaussian smoothing kernel, applied only along the spatial dimensions, h_{ev} and h_{od} are a quadrature pair of 1D Gabor filters applied temporally. Fig.2 shows STIPs

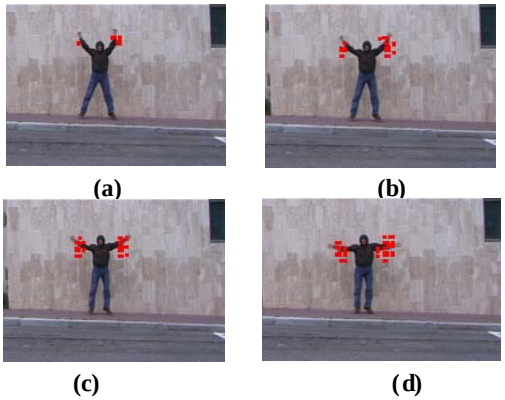

Figure.2 STIP extraction based on Dollár's method

Extraction method based on Dollár. From Fig.2, we can see that STIPs are located at the region of body's intense movement, so human behavior characteristics can be described by STIPs.

B. Descriptor of STIPs

In order to describe the distributions of STIPs for a posture, we need to design the descriptor of each STIP. Every STIP stands for a small area that is undergoing non-constant movement, so we choose an $5 \times 5 \times 5$ adjacent neighborhood which is called cubiod to model a STIP, where we calculate the 3D gradients (L_x, L_y, L_t) for each pixel, so each STIP can form a 375-dimensioned vector as its descriptor. Here, it is emphasized that 3D gradients (L_x, L_y, L_t) should be flattened as $(L_x/Norm, L_y/Norm, L_t/Norm)$, where $Norm$ is calculated as :

$$Norm = \sqrt{L_x^2 + L_y^2 + L_t^2}$$

C. HOG for a posture in a frame

Because the posture can be described by the statistic distribution of its STIPs, we can classify these postures via clustering their distributions of STIPs. In this paper, we model a single posture by calculating the histogram of all its STIPs. That is, a STIP descriptor is 375-dimension, which is composed 3 gradient sub-vectors, they are 2 spatial gradients on x and y directions and 1 temporal gradient on t direction respectively, each of them is 125-dimension. Suppose there are N STIPs in a posture frame, we calculate a histogram of 16 bins respectively for 3 types of gradient sub-vectors, each histogram is derived of N sub-vectors; and then we combine these 3 histograms as a 48-bined histogram, which is called HOG for a posture.

III. COMPARISON OF CLSTERING METHODS AND THE UNSUPERVISED CLUSTERING RESULTS

Unsupervised clustering is a mathematical tool that automatically discovers structures or certain patterns in a data set, where the objects inside each cluster show a certain degree of similarity. In this paper we need to classify N training pose samples into M salient postures, specifically we need to find an algorithm to measure the similarity for a pair of samples and a proper clustering method. Here we compared fuzzy C-means and its improved versions, and finally we get the optimal one. They are discussed in details as following.

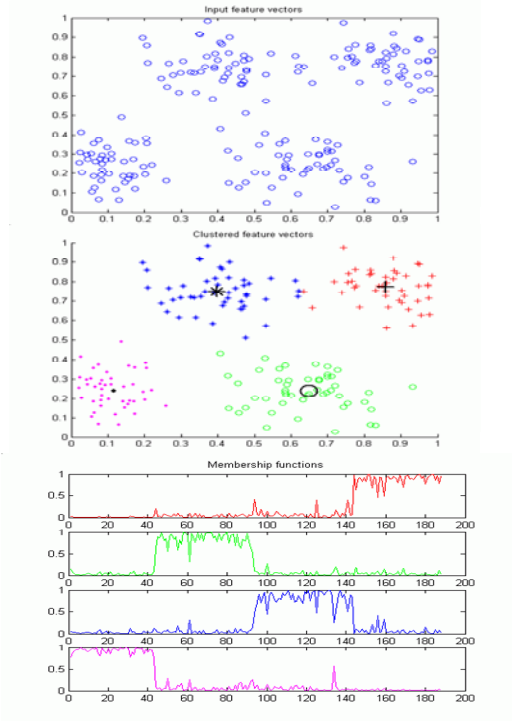

Figure.3 FCM clustering test
From up to bottom they are input data, clustering result and membership of functions U, Where n=188, p=2, C=4.

A. Pros and cons of FCM

The popularity and usefulness of fuzzy C-means result from three facts. The algorithms are simple; they are very effective at efficiently finding minimisers of objective function J_m : give data set

$$X = \{x_1, \quad x_2, \quad \cdots \quad x_n\}$$

where n is the number of data points in X , $x_k \in R^p$, p is the number of features in each vector x_k ; in order to cluster X into C prototypes, J_m is sought as

$$\min_{(U,V)} \left\{ J_m(U,v) = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^m D_{ik}^2 \right\} \tag{2}$$

Constrain $\sum_{i=1}^{c} u_{ik} = 1, \forall k$, and distance $D_{ik}^2 = \|x_k - v_i\|_A^2$ (3)

A-norm $\|x_k\|_A = \sqrt{\langle x, x \rangle_A} = \sqrt{x^T A x}$ (4)

Degree of fuzzy $m \geq 1$, $v = (v_1, \quad v_2, \quad \cdots \quad v_C)^T$ (5)

And U is the membership functions, those minimizes usually represent the structure of X very well; test result is shown in Fig.3. Various theoretical properties of the algorithms are well understood, and are described in Refs [14]. Additionally, this method is unsupervised and always convergent.

Also this method does have some disadvantages, such as, long computational time, Sensitivity to the initial guess (speed, local minima), unable to handle noisy data and outliers, very large or very small values could skew the mean, not suitable to discover clusters with non-convex shapes.

B Out-performance of RFCM compared to FCM

1) Broaden the application domains of FCM

The RFCM classifier is useful when a feature space has an extremely high dimensionality that exceeds the number of objects and many of the feature values are missing, or when only relational data are available instead of the object data. The relational data is represented by a matrix in terms of distances (dissimilarity) between object data, and is not concerned with the relational database. Of course the pair wise relational matrix can be easily figured out while the data are given as data vector sets. So RFCM can deal with more than the problems that FCM can do.

2) Efficiency on computations

Whenever relational data are available that corresponds to measures of pair wise distances (actually, squared distances) between objects, RFCM can be used instead that rely on its computation efficiency. One of the advantages is that their driving criterion is "global", i.e. it assesses a property implicitly shared by all the objects even though the object data is not directly used. Another advantage is that these relational algorithms automatically inherit excellent numerical convergence properties of FCM, because they have a close relationship with the quickly convergent and reliable object-oriented algorithms.

Give matrix $R = [r_{ij}]$ for relational data, which is corresponding to pair wise distance between objects, different to FCM , its object function is defined as $K_m(U)$,

$$K_m(U) = \sum_{i=1}^{C} \left(\sum_{j=1}^{n} \sum_{k=1}^{n} (u_{ij}^m u_{ik}^m \delta_{jk}^2) / (2 \sum_{t=1}^{n} u_{it}^m) \right) \tag{6}$$

Where $m \geq 1$, and for $1 \leq j,k \leq n, \delta_{jk}^2 = r_{jk}$. Useful partitioning U of the data are sought as minimizers of $K_m(U)$.

The optimal partitioning U^* gives

$$K_m(U^*) = J_m(U^*, F_m(U^*)) = \min_v J_m(U,v) \tag{7}$$

from which it follows that U* is a minimizer of $K_m(U)$, if and only if U^* is a minimizer of min $J_m(U,v)$,

which is easily shown to be true if and only if (U^*, v^*) is a minimizer of $J_m(U, v)$. The above explanation proves FRCM has preserved the simplicity and convergence of FCM. Additionally, numerical experiments show the actual work done per iteration could be smaller for the RFCM algorithms than for the object data versions FCM,when the dimension p of the feature data is large.

3) Limitations of RFCM

RFCM has a strong restriction that restrains its applications. The relation matrix R must be Euclidean, i.e., there exists a set of N object data points in some p-space whose squared Euclidean distances match values in R. To ease the restrictions that RFCM imposes on the dissimilarity matrix, there are two improved versions of RFCM that are introduced in the following.

C. Analysis of two improved RFCM algorithms

NERFCM can transform the Euclidean relational matrix into Euclidean ones by using the β-spread transformation introduced in [15]. This transformation consists of adding a positive number β to all off-diagonal elements of R. As proved in [16], there exists a positive number β0 such that the β-spread transformed matrix Rβ is Euclidean for all β≥β0, and is not Euclidean for all β≤β0. The parameter β, which determines the amount of spreading, should be chosen as small as possible to avoid unnecessary spreads of data with consequent loss of cluster information.

On the other hand, the exact computation of β0 involves an expensive eigenvalue computation [16]. To reduce loss of information without decreasing performance dramatically, Hathaway and Bezdek [16] proposed an extension of RFCM, denoted non-Euclidean RFCM (NERFCM), in which the β-spread transformation is computed dynamically during the iteration process of RFCM. The βN computed by NERFCM is the minimum value which guarantees the convergence of RFCM. As RFCM can converge even if a relation is not Euclidean, it may happen that βN <β0. NERFCM has proved to be one of the most reliable fuzzy relational clustering algorithms; the performance of NERFCM depends, however, on the value of βN which could be so large that the structure inherent in R might not be mirrored by that in RβN.

ARCM represents a cluster in terms of a representative of the mutual relationships of the objects which belong to the cluster with a high membership value. Each object is represented by the vector of its relation strengths with the other objects in the data set, and a prototype is an object whose relationship with all the objects in the data set is representative of the mutual relationships of a group of similar objects. Like FCM, ARCM partitions the data set minimizing the Euclidean distance between each object (strongly) belonging to a cluster and the prototype of the cluster. ARCM determines the optimal partition minimizing the following objective function:

$$J_m(U, v) = \sum_{i=1}^{C} \sum_{k=1}^{n} u_{ik}^m \delta^2(x_k, v_i) \quad (8)$$

$\delta(x_k, v_i)$ is the deviation between, respectively, the relation between xk and all the other objects, and between vi and all the other objects.

Defining

$$\delta(x_k, v_i) = \sqrt{\sum_{s=1}^{n} (r_{ks} - v_{is})^2} \quad (9)$$

where r_{ks} is the relation between the pair of objects xk and xs, and v_{is} is the relation between the prototype vi and object xs, and applying the standard Lagrange multipliers minimization method, ARFCM algorithm can get the final convergence and give the clustering membership matrix U.

D. Posture clustering implementation

We take Weizmann databases as training samples for posture clustering, there are 8 types of action and each action has 9 action videos conducted by 9 different persons, we proposes to characterize the postures using a set of the spatial-temporal interesting points (STIPs).

Our observation on Fig.4 has shown that similar postures share a set of similar STIPs. Therefore, we propose to extract salient postures from example poses through clustering.

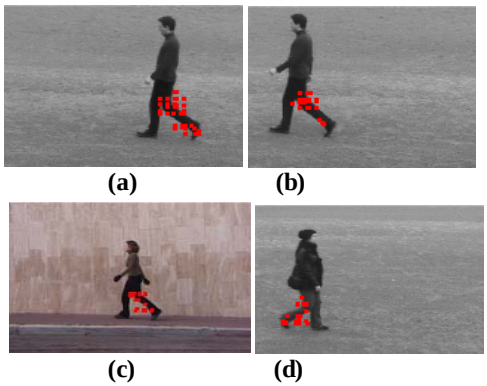

Figure.4 Similar postures and the distribution of their STIPs

We define the HOG similarity of two postures with histogram intersection method, which is:

$$S(p, q) = \sum_{u=1}^{B} \min\{p^{(u)}, q^{(u)}\} \quad (10)$$

Where p and q are 2 histograms with B bins, if they are the same, the similarity s is 1, so the dissimilarity can be defined as $d = 1 - s$. Consequently, the HOG dissimilarity of the total frame N can be calculated, and the whole HOG dissimilarity can form a dissimilarity matrix:

$$D=[d_{ij}]_N=\begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1N} \\ d_{21} & d_{22} & \cdots & d_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ d_{N1} & d_{N2} & \cdots & d_{NN} \end{bmatrix}$$

Where the value of diagonal elements d_{ii} is 0, the other elements value d_{ij} is the dissimilarity between i and j .The N frames are then clustered into M clusters by employing a pair-wise clustering algorithm which takes the dissimilarity matrix of every pair of samples to be clustered.

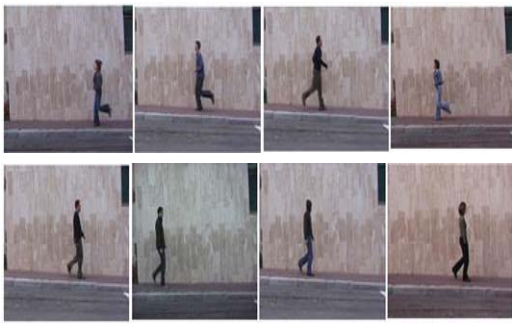

Figure.5 Similar postures from walking and running conducted by different persons as the same cluster

Figure.6 Similar postures from pjump conducted by different persons as the same cluster

In this paper training samples number N is 3314 and cluster number M is 37. Some clustering results are show in Fig.5 and Fig.6.It intuitively satisfies human senses that sample frames in Fig.4 and Fig.5 are classified into the same cluster, because they are indeed of similar postures.

IV. POSTURE MODELING BASED ON GMM

A Gaussian Mixture Model

In this paper, Gaussian Mixture Model is regarded as using several distributions to describe each type of posture [17]. In other words, we use K weighted sum of Gaussian distribution functions to close in the distribution function of each posture's observed values.

For a single sample x_i in each type of posture observation data set $X=\{x_1,x_2,\ldots x_N\}$, the Gaussian mixture distribution density function is:

$$P(x_i|\Theta)=\sum_{i=1}^{K}\omega_k p_k(x_i|\theta_k) \tag{11}$$

Where K is the number of Gaussian distributions ω_k is the weight estimation of k th Gaussian in the mixture, and it satisfied with: $\sum_{k=1}^{K}\omega_k=1$. p_k is the Gaussian probability density function, and $\Theta=(\theta_1,\theta_2,\ldots,\theta_K)$ is the parameter vector of mixture composition. $\theta_k=(\mu_k,\sum_k)$ is the Gaussian distribution parameter, that is, the mean value and covariance matrix respectively.

There are two main methods to estimate the parameters of GMM, which are based on the online updating and EM algorithm. The principle of the online updating method can be described as follows [18]. Every new observation x_i is checked with each of K current Gaussian distributions. If they match, the parameters $\mu_{j,t}$ and $\sigma^2_{j,t}$ for the matching distribution are updated as:

$$\begin{cases}\mu_{j,t}=(1-\alpha)\cdot\mu_{j,t\text{-}1}+\alpha\cdot I_t \\ \sigma^2_{j,t}=(1-\alpha)\cdot\sigma^2_{j,t\text{-}1}+\alpha\cdot(\mu_{j,t}-I_t)^2\end{cases} \tag{12}$$

$$\omega_{n,t}=(1-\alpha)\cdot\omega_{n,t\text{-}1}+\alpha\cdot M_{n,t} \qquad n\in[1,K] \tag{13}$$

Where α is the Gaussian adaptation-learning rate. $M_{n,t}$ is 1 for the model which matched the pixel and 0 for the none matched models.

The other method of GMM parameter estimation is based on EM. Our sample data is incomplete, and EM algorithm is capable of parameter estimation with MLE (Maximum likelihood Estimation) under insufficient samples. So we choose EM to estimate the parameters of GMM.

B. EM algorithm

It is an iterative algorithm to get the maximum likelihood estimation of distribution density function, when the observation data is incomplete. It can significantly reduce computational complexity, but the performance is similar with the maximum likelihood estimation, so it has a good practical application value. In this paper, the observation data X of each posture is incomplete, so the missing data Y is introduced, and the complete data is $Z=\{X,Y\}$, where y_i is the class

that x_i belongs to. If x_i comes from the k th Gaussian component, we can obtain $y_i = k$. So the likelihood function of complete data is:

$$L(\Theta|Z) = L(\Theta|X,Y) = P(X,Y|\Theta)$$

There are two steps in EM algorithm: Expectation step and Maximization step. When the observation data and the current parameter is known, we can get the Expectation Maximization of complete likelihood function $L(\Theta|Z)$ according to the missing data Y. So the E-step and M-step are:

E-step: $$Q(\Theta,\Theta^t) = E\left[\log P(X,Y|\Theta)\middle|X,\Theta^t\right]$$

M-step: $$\Theta^{t+1} = \arg\max Q(\Theta,\Theta^t)$$

Formulas (7) and (8) can ensure to get the maximum after the iterative computation E-step and M-step.

C. EM estimation on GMM

To complete the algorithm mentioned above, the key step is to get the probability density of the missing data Y. We can get the probability density of Y according to

$$p(Y|X,\Theta^t) = \prod_{i=1}^{N} p(y_i|x_i,\Theta^t)$$

Bays:. So the iterative functions based on EM estimation are:

$$\left.\begin{aligned}
&\alpha_k^t = \sum_{i=1}^{N} p(k|x_i,\Theta^t)\\
&\pi_k^t = \frac{1}{N}\alpha_k^t\\
&\mu_k^{t+1} = \frac{1}{\alpha_k^t}\sum_{i=1}^{N} x_i p(k|x_i,\Theta^t)\\
&\sum\nolimits_k^{t+1} = \frac{1}{\alpha_k^t}\sum_{i=1}^{N} x_i p(k|x_i,\Theta^t)(x_i-\mu_v^{t+1})(x_i-\mu_v^{t+1})^T
\end{aligned}\right\} \quad (14)$$

The following data is the GMM information for cluster 5, there are 95 frames in cluster 5, and it is modeled with 3 components of Gaussians:

Nin:48

Ncentres: 3

Cover_type: ”spherical”

Priors:[0.3765 0.2616 0.3618]

Centres: [3x48 double]

Covars: [3.9156e-004 7.2282e-004 5.0298e-004]

The first item “nin:48” stands for its dimension is 48; the second item”ncentres:3” stands for there are 3 components of Gaussians; the third stands for its covariance shape is ‘spherical’; the fourth one is its weights, and so on.

V. POSTURE RECOGNITION

The key question of posture is how to measure the similarity between the sample sequence and the test sequence. Give a frame, its STIPs can first be extracted, then we calculate the 375-D descriptor for each STIP and the posture histogram f can be figured out. The posture recognition is to find the best matching for f among all the cluster GMM models. That is :

$$p(f) = \arg\max_{\varphi\in\psi} \sum_{i=1}^{K} \omega_{i,\varphi} \cdot \eta_{i,\varphi}(f \cdot \mu_i \cdot \sigma_i) \quad (15)$$

Where ψ is the collection of all cluster models, φ is one of ψ, and φ has K sub-Gaussian models of GMM, $\eta_{i,\varphi}$ is the i th Gaussian probability density function, μ_i, σ_i are its mean and variance respectively, $\omega_{i,\varphi}$ is the weight of i th Gaussian model in cluster φ.

If Maximum similarity ration larger than a threshold, then the input frame can be judged as one specific a cluster, otherwise it is a posture of a new action.

VI. EXPERIMENT RESULTS AND ANALYSIS

We take Blank Action databases as training samples, there are 8 types of action and each action has 9 action videos conducted by 9 different persons, in all there are 3314 STIP detected frames in our training experiments. The 8 types of action videos include walk, run, bend, jack, jump, skip, wave by one hand, wave by two hands. After clustering, 3314 frames can be clustered into 37 posture class.

In order to verify the overall performance of the proposed model, we adopt Leave One Sample Out Test. In the Leave One Sample Out Test, each of the 37 samples was taken as the test sample and the residual samples were used as training samples. In table I, we list 18 cluster testing results.

TABLE I.
POSTURE RECOGNITION EXPERIMENT RESULT S

Sequence number	Test frame number After NERF	Frame number After GMM	Shared Frame number	Correct percentage
1	107	111	100	0.93458
2	60	64	56	0.93333
3	63	64	59	0.93651
4	95	96	89	0.93684
5	46	42	42	0.91304
6	60	65	55	0.91667
7	85	87	81	0.95294
8	90	91	81	0.90000
9	93	89	88	0.94624
10	33	32	30	0.90909
11	99	109	93	0.93939
12	92	89	85	0.92391
13	147	139	134	0.91156
14	171	182	155	0.90643
15	60	64	54	0.90000
16	30	31	29	0.96667
17	43	44	42	0.97674
18	76	75	72	0.94737

VII. CONCLUSIONS

Posture modeling is critical for Human behavior recognition. In this paper, we have proposed an effective algorithm, which is posture modeling base on EM estimation on GMM, and it can obtain a high recognition rate. The experiments prove that our method is accurate and effective, which is robust to the interferences caused by video segmentation, such as, illumination variation and camouflage, and so on.

However, there are still some disadvantages. For example, it is only effective for stable camera environment and simple background. Our next step work is to improve our algorithm to adapt to the dynamic camera environment and complex background. In addition, we will make action recognition with posture transitional graphic.

ACKNOWLEDGMENT

Our research is supported by Natural Science Fund of Shandong（ZR2009GM007,Y2008G09), and Doctorial Fund of QingDao University of Science & Technology，
We would like to thank M.Blank at the Weizmann Institute for sharing their datasets.

REFERENCES

[1] Robertson, N.,Reid,I.D.. A general method for human action recognition in video. Computer vision and Image Understanding,2006, 104(2):232~248
[2] Ahmad M.,Seong-Whan Lee.Human action recognition using multi-view image sequences. Proceedings of the international Conference on Automatic Face and Gesture Recognition, 2006,523~528
[3] Christoph Bregler. Learnig and Recognizing Human Dynamics in Video Sequences. IEEE CVPR'97,1997,568~574
[4] P. Kakumanu, S. Makrogiannis, N. Bourbakis, "Asurvey of skin-color modeling and detection methods". Pattern Recognition 40 (2007) 1106 – 1122.
[5] Cheng duansheng, Liukaisheng, Summarization of skin detection techeniques, journal of computer, Vol.29 No.2 Feb，2006 194-207.
[6] Ivan Laptev and Tony Lindeberg,Space-Time Interest Points. In Proc. ICCV 2003, Nice, France, pp.I:432-439.
[7] Bezdek JC, Hathaway RJ, Sabin MJ, Tucker WT (1987) Convergence theory for fuzzy C-means: counterexamples and repairs. IEEE Trans Syst, Man, and Cybern SMC-17(5):873－877.
[8] Mingzhou (Joe) Song and Lin Zhang, 2008 Eighth IEEE International Conference on Data Mining, 560-570.
[9] RICHARD J. HATHAWAY and JOHN W. DAVENPORT, Pattern Recognition, Vol. 22, No. 2, pp. 205 212, 1989.
[10] P. Corsini B. Lazzerini F. Marcelloni, Soft Comput (2005) 9: 439－447
[11] A. Oikonomopoulos,I.Patras, and M.Pantic, "Spatio-temporal salient points for visual recognition of human actions," IEEE Trans. SMC-B, vol.36,pp.710-719, 2006.
[12] Ivan Laptev and Tony Lindeberg. Space-Time Interest Points. In Proc. ICCV 2003, Nice, France, pp.I:432-439.
[13] P.Dollar, V.Rabaud, G.Cottrell, S.Belongie. "Behavior recognition via sparse spatio-temporal features," IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance.,432-439, 2005.
[14] P. Corsini B. Lazzerini F. Marcelloni, Soft Comput (2005) 9: 439－447
[15] Setnes M, Babuska R (1999) Fuzzy relational classifier trained by fuzzy clustering. IEEE Trans Syst Man Cybern 29(5):619－625.
[16] RICHARD J. HATHAWAY and JAMES C. BEZDEK Pattern Recognition, Vol. 27, No. 3, pp. 429 437, 1994.
[17] Wanqing Li, zhengyou Zhang, and ZiLiu. Expandable Data-Driven Graphical Modeling of Human Action Based on Salient Postures. IEEE Transactions on Circuits and Systems for Video Technology. Vol.18, No.11, Nov.2008. pp.1499-1510.
[18] C. Stauffer, W.EL Grimson, Adaptive background Mixture Models for Real-Time Tracking. In CVPR'99. vol.2.pp:246-252.

Dr. **Xijun Zhu** is a associate professor of College of Information Science and Technology at Qingdao University of Science & Technology,China. His current research interests are in the scope of Signal and Image Processing. He received his Ph.D. degree in 2006 from the Geomatics College at Shandong University of Science & Technology,China.

Chuanxu Wang received his B.S in Electronic engineering in 1990 and M.S. in Industry Automation in 2000 respectively in China Petroleum University and PhD in Ocean information Processing in China Ocean University in 2007. He is currently an Associate Professor in the Qingdao University of Science and Technology, China. His interesting research direction is computer vision.