Probabilistic Motion Parameter Models for Human Activity Recognition

Xinding Sun, Ching-Wei Chen, B. S. Manjunath
Department of Electrical and Computer Engineering,
University of California, Santa Barbara, CA 93106
{xdsun, cwei, manj}@ece.ucsb.edu

Abstract

A novel method for human activity recognition is presented. Given a video sequence containing human activity, the motion parameters of each frame are first computed using different motion parameter models. The likelihood of these observed motion parameters is optimally approximated, based directly on a multivariate Gaussian probabilistic model. The dynamic change of motion parameter likelihood in a video sequence is characterized using a continuous density hidden Markov model. Activity recognition is then posed as a motion parameter maximum likelihood estimation problem. Experimental results show that the method proposed here works well in recognizing such complex human activities as sitting, getting up from a chair, and some martial art actions.

1. Introduction

Human activity analysis in video has many applications in video surveillance, human computer interface etc. Typical activities include walking, running, jumping, turning around, sitting down on a chair, and getting up from a chair. In martial art activities, such activities can be more complex. These typical human activities usually involve changes in the environment, object occlusion etc. Therefore, feature point based or region-based techniques that work well on facial expression, lip reading, gesture recognition [7], cannot be directly applied to human activity recognition

Given the complexity of human body motion, techniques that do not require explicit image feature detection or segmentation are of much interest. Among the early work is Davis and Bobick [2], wherein they use temporal templates for human movement recognition. Their method requires less computation, but is sensitive to variances in the movement. Little and Boyd [3] use the moments of moving points to represent the optic flow for the purpose of periodic human gait recognition. Yacoob and Black [9] propose recognition of activities based on matching of principal component under global temporal change, in particular affine transforms.

Our proposed method, similar to those using global motion fields, does not require image feature tracking or segmentation. The work is motivated by two-dimensional object recognition. We introduce motion parameters to compute the likelihood of observed video sequences. It approximates the motion parameter likelihood of a video frame in an optimal way. The state transitions of the motion parameters are modeled using the continuous density hidden Markov model (HMM). Recognition of activity is then posed as a maximum likelihood parameter estimation problem.

2. Motion Parameter Estimation

The first step in activity detection is motion estimation. Here, we use a model-based approach posed in [1]. For a motion model, the motion parameter $\mathbf{P}_m$ for a given motion vector $\mathbf{V}$ at a given position $\mathbf{x} = (x, y)$ can be represented as:

$$\mathbf{V}(\mathbf{x}) = \mathbf{V}(\mathbf{x};\mathbf{P}_m) \quad (1)$$

This model can be projective, plannar, affine, optic flow, etc. In our experiment, we choose affine motion parameters and optic flow as features for activity recognition. The latter is the simplest case, i.e. the motion vector $\mathbf{V}$.

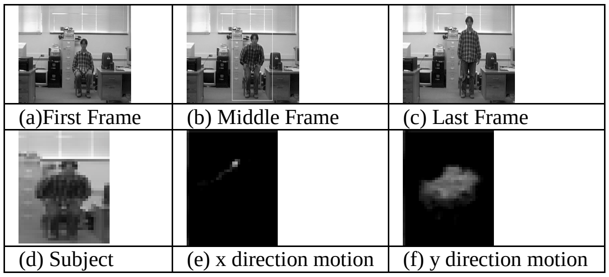

(a)First Frame (b) Middle Frame (c) Last Frame
(d) Subject (e) x direction motion (f) y direction motion

Figure 1. Video frames from a "su" Sequence and corresponding optic flow information.

It is generally not necessary to use the motion parameters of the whole video frame for activity recognition. Instead, a smaller region of interest is chosen in our experiment. To illustrate, figure 1(d) shows an example region of interest inside the window drawn in figure 1(b). Figure 1(e)-(f) show the corresponding normalized optic flow along the x- and y- directions respectively, for this region.

3. Computation of Motion Parameter Likelihood

Consider a motion parameter $\mathbf{P}_m = (p_1, p_2, ...p_d)$, computed at each pixel location, where d is the dimension of the parameter. For example, $\mathbf{P}_m$ could be the affine motion parameters, or a 2-D optic flow vector. These parameter values are then organized into a large vector by row scanning the image. Let L be the number of pixels in a video frame or a region of interest in a frame (ordered according to a row scan). Let

$$\mathbf{Z} = (p_1^1, p_1^2 ......p_1^L, p_2^1, p_2^2 ......p_{2,}^L, ...p_d^1, p_d^2 ......p_d^L)^T \quad (2)$$

Note that $\mathbf{Z}$ is a $N = d \times L$ dimensional vector. We model $\mathbf{Z}$ as a multivariate Gaussian. Let the mean of this Gaussian be $\mathbf{m}$ and the covariance be $\mathbf{Q}$. Then, given $\mathbf{Z}$ from an observation class Ω, we can write the conditional probability $P(\mathbf{Z} \mid \Omega)$ as:

$$P(\mathbf{Z} \mid \Omega) = \frac{\exp(-\frac{1}{2}(\mathbf{Z}-\mathbf{m})^{\mathbf{T}}\mathbf{Q}^{-1}(\mathbf{Z}-\mathbf{m}))}{(2\pi)^N \left|\mathbf{Q}\right|^{1/2}} \quad (3)$$

If we have activity in class Ω, then (3) gives the likelihood of the motion parameters for a given frame. This is essential in later statistic modeling of the activities using HMM. This approach to modeling the observation is similar to the work in [4], where the observation vector is the image intensity, and the application is object recognition. In the following discussion, we will refer to $\mathbf{Z}$ as the *parametric motion object (PMO).*

The Karhunen-Loeve transform (KLT) is used to simplify the computation of (3). Let $\tilde{\mathbf{Z}} = \mathbf{Z} - \mathbf{m}$. The covariance matrix can be decomposed as: $\mathbf{Q} = \mathbf{\Phi\Lambda\Phi}^{\mathbf{T}}$, where the columns of $\mathbf{\Phi}$ are the orthonormal eigenvectors of $\mathbf{Q}$, and $\mathbf{\Lambda}$ corresponds to the diagonal eigenvalue matrix of $\mathbf{Q}$. Let $\mathbf{Y} = \mathbf{\Phi}^{\mathbf{T}}\tilde{\mathbf{Z}}$, then (3) can be computed as:

$$P(\mathbf{Z} \mid \Omega) = P_P(\mathbf{Z} \mid \Omega) P_{\bar{p}}(\mathbf{Z} \mid \Omega)$$

$$= \left[\frac{\exp(-\frac{1}{2}\sum_{1}^{M} y_i^2/\alpha_i)}{(2\pi)^{M/2}\prod_{1}^{M}\alpha_i^{1/2}}\right]\left[\frac{\exp(-\frac{1}{2}\sum_{M+1}^{N} y_i^2/\alpha_i)}{(2\pi)^{(N-M)/2}\prod_{M+1}^{N}\alpha_i^{1/2}}\right] \quad (4)$$

Where M is the dimension of the principal subspace, y_i is the ith component of $\mathbf{Y}$, and α_i is the ith eigenvalue of $\mathbf{Q}$.

In (4), we divide the likelihood for a PMO into two parts. The first part, $P_P(\mathbf{Z} \mid \Omega)$, corresponds to the likelihood of the PMO in the principal subspace as used in principal component analysis (PCA). The second part $P_{\bar{P}}(\mathbf{Z} \mid \Omega)$ corresponds to the likelihood of the PMO in the complementary orthogonal subspace of the principal subspace. PCA has been successfully used for face recognition [8] and activity analysis [9]. The principal space is enough for general representation and approximation purposes. However, note that the likelihood in the principal space $P_P(\mathbf{Z} \mid \Omega)$ does not provide an optimal approximation of the likelihood $P(\mathbf{Z} \mid \Omega)$ in the whole space. The second part $P_{\bar{P}}(\mathbf{Z} \mid \Omega)$ plays an important role in the recognition process. This is also observed in our experiments, discussed in section 5.

Direct computation of $P_{\bar{P}}(\mathbf{Z} \mid \Omega)$ is too expensive for practical application, therefore, following [4] we use an optimal approximation of it:

$$P_{\bar{p}}(\mathbf{Z} \mid \Omega) \approx \left[\frac{\exp(-\frac{1}{2}\sum_{M+1}^{N} y_i^2/\rho)}{(2\pi\rho)^{(N-M)/2}}\right], \text{ where } \rho = \frac{1}{N-M}\sum_{M+1}^{N}\alpha_i .$$

4. Modeling Activity Using HMM

In the context of human motion recognition, promising results have been obtained using the HMM. The experiment in [10] is perhaps the first one. A generic HMM [5] can be represented as $\lambda = \{\Xi, A, B, \pi\}$, where $\Xi = \{q_1, q_1, ...q_{N'}\}$ denotes the N' possible states, $A = \{a_{ij}\}$ denotes the transition probabilities between the hidden states, $B = \{b_j(.)\}$ denotes the observation symbol probability corresponding to the state j, and π denotes the initial state distribution. Given a video sequence $\{O_1, O_2, ...O_T\}$, where T is the length of the sequence, we then want to find one model from a given dictionary $\{\lambda_1, \lambda_2, ...\lambda_E\}$ which maximizes the likelihood $P(O \mid \lambda)$.

Our initial experiments consist of two sets. To motivate discussion, we introduce the first set here. In this part, eight office activities are to be recognized. We separate these activities into two groups. In the first group, we have: turning of the body from left to front ("l2f"), front to left ("f2l"), front to right ("f2r") and right to front ("r2f"). In the second group we have: standing up ("su"), sitting down (“sd”), starting to sit down but returning to the standing position without sitting down (“bu”), and starting to get up (from a sitting position) but returning to the sitting position without getting up (“bd”). The second group is designed in such a way that the sequences have similar sub-processes. Figure 1(a-c) shows three frames from a “su” sequence.

4.1. Model

We choose a four-state continuous density HMM for activity recognition here. The number of states is empirically determined and we observed that an increase to a larger number of states did not result in any performance gains on our initial data sets. An example of the HMM structure before and after training for a “bd” sequence is shown in Figure 2. Note that our HMM has a typical left to right graph structure.

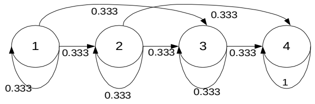
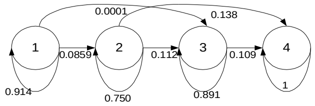
 1 2 3 4 0.333 0.333 0.333 0.333 0.333 0.333 0.333 0.333 1	(a). HMM for “bd” before training
 1 2 3 4 0.0001 0.138 0.0859 0.112 0.109 0.914 0.750 0.891 1	(b). HMM for “bd” after training

Figure 2. An example HMM for the “bd” sequence.

4.2. Training

The first step of our HMM training is to obtain the observation model B. Since the motion pattern at any given short interval can be regarded as unchanged, we can divide the sequence into temporal segments where each segment corresponds to a state. We uniformly segment each training sequence into four segments before clustering. Each segment is assigned a state number that is the same as its segment order in the sequence. As in speech recognition, this method provides a good initial clustering of states. The position of the PMO in each frame is manually selected around the moving subject. Then, we compute $\mathbf{m}$ and $\mathbf{Q}$, and consequently $\mathbf{\Phi}$ and $\mathbf{\Lambda}$, for each state. After this step, we follow the conventional *K-means* clustering method to iteratively classify the frames based on their likelihood computed using (4). Any misclassified initial segmentation can be corrected in the clustering process. Note that we have one set of bases for each hidden state, unlike PCA based applications that have only a single set of bases for a whole database.

At this stage we have the observation model B, with $\mathbf{m}$ and $\mathbf{Q}$ computed. The next step is to obtain the state transition matrix A. This is done using the EM algorithm as proposed in [6]. A is initialized as shown in **Figure 2**(a). Note that we do not need to compute π, as in our model we always start in state 1. The trained HMM structure for the “bd” activity is shown in **Figure 2**(b). **Figure 3** shows the normalized likelihood of each frame from one of the “sd” sequences, based on four different “sd” state models. The transition from one state to the next is clearly evident, and it confirms our initial segmentation assumption (see the regions labeled state 1 to state 4).

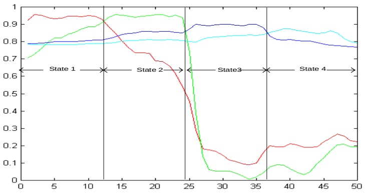


Figure 3. Normalized likelihood of one “sd” sequence.

4.3. Recognition

Given a test video sequence O, we first compute its motion parameters. A window of the same size as the training PMO is moved around the video frame to find the position where the maximum likelihood for a state model is obtained using (4). The likelihood is used as $b_j(.)$ for a given state model j. A Kalman filter can be applied to track the window in order to speed the searching process. The recognition of the activity λ_{i*} follows from the maximum likelihood estimate:

$$i* = \underset{1\le i\le E}{\arg\max}[P(O/\lambda_i)] \qquad (5)$$

5. Experimental Results

Experiments are performed using 352x240 pixel resolution video, captured at 30 frames per second. For simplicity, the cameras are put in front of the subjects at a constant distance. We collect 20 sequences for each activity. There are a total of 160 training sequences and 160 test sequences. Each sequence contains 20 to 56 frames. Half of the video sequences are used for training, while the other half are used for evaluation. A window of fixed size that covers the human body is used in computing the PMO. Affine motion parameters and optic flow vectors are used to to compute the PMO. The PMOs are normalized to a zero-mean unit-norm.

In the first set of experiments, we use office activity video for testing. The fixed window size is 128x224. The sequences are introduced in section 4. In the second set of experiments, we test eight martial art activities. The fixed window size is 160x224. A subject stands in front of the camera to perform martial arts. During the performance, he plants one foot on the ground, and follows a set of motions closely. The images in **Figure 4** show the representative frames from each of the eight activities. In this case, we put all the activities in a single group, regardless of their complexity. Table 1 summarizes these results.

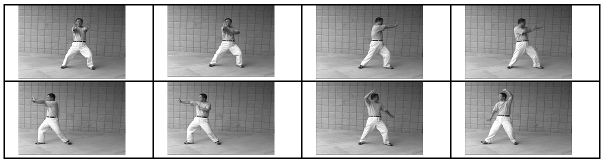


Figure 4. Representative frames from martial art activity sequences.

Table 1. Experimental results on the test sequences.

Activity			Office		Martial Art
			Group1	Group2	
Affine model	PCA	6 bases	40%	30%	30%
		10 bases	45%	40%	35%
	PMO	6 bases	50%	50%	45%
		10 bases	60%	50%	47%
Optic Flow	PCA	6 bases	70%	55%	77%
		10 bases	70%	60%	81%
	PMO	6 bases	100%	90%	89%
		10 bases	100%	95%	91%

Results for group 1 are better than those for group 2 activities in the office set. This is partly due to the fact that group 2 activities share similar sub-processes, making their estimation more difficult. Also, group 2 activities are more complex. For example, the first state of “su” is the same as the first state of “bu”. In addition, the transitions in “bu” and “bd” are also more complicated than those in group 1. The martial art activities are in general more complex than the office activities. There are occlusions among different parts of the body. This can be very difficult for feature tracking based methods. However, our motion parameter based method still achieves a recognition rate of about 90%.

Note that the optic flow based modeling performs better than the more informative affine model. One possible explanation is that the affine motion parameters are more sensitive than the optic flow, and variations are not well captured within the four-state HMM used in our experiments. Increasing in the training dataset perhaps helps to improve its accuracy. Two different numbers of principal subspace dimensions are also tested. In general, larger dimensions of principal subspaces perform better than smaller ones, but we did not observe significant differences here between six and ten dimensions.

PCA based method is also tested here. It is done by taking $P_{\bar{P}}(\mathbf{Z}\,|\,\Omega)$ out of computation in (4). It is essentially the same feature used in [4]. It can be seen from experiment that in general PMO method outperforms the PCA method.

6. Discussion and Conclusion

We have presented a general method for complex human activity recognition. The likelihood of the observed motion parameters is computed based on a multivariate Gaussian probabilistic model. The temporal change of the likelihood is modeled using HMM. Our initial test results containing activities such as sitting, getting up from a chair, and martial arts appear quite promising. The framework proposed here can be fit in a more general Bayesian network [4] for human activity understanding.

While the proposed method has been investigated in two different settings, more work is needed to investigate how this method scales to different environments. The experiments have been carried out on sequences that have approximately the same spatial resolution. Our preliminary experiments indicate that scaling can be handled by re-normalizing the motion field appropriately, but more investigation is needed.

Acknowledgement: This research is in part supported by the following grants/awards: Office of Naval Research grant #N00014-01-1-0391, an NSF instrumentation award #EIA-9986057, an NSF infrastructure award #EIA-0080134, and a matching grant from CalTrans #65V250A3-TO-3022.

7. References

[1] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, "Hierarchical Model-Based Motion Estimation,” in *ECCV’92*, pp.237-252, 1992.

[2] J. W. Davis and A. F. Bobick, " The representation and recognition of human movement using temporal templates,” in *CVPR’97*, pp.928-34, 1997.

[3] J. J. Little and J. Boyd, “Recognizing People by Their Gait: the Shape of Motion, ” Videre, 1(2), the MIT press, 1998.

[4] B. Moghaddam and A. Pentland, “Probabilistic Visual Learning for Object Representation,” *IEEE PAMI*, 19(7), pp.696-710, 1997

[5] D.J Moore, I. A. Essa, and M. H. Hayes III, **“**Exploiting human actions and object context for recognition tasks,” in *ICCV’99*, pp. 80-86, 1999.

[6] L. R. Rabiner,"A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, 77(2), pp. 257-286, 1989.

[7] M. Shah, and R. Jain, “Motion-based Recognition,” *Kluwer-Academic Publishers, Computational Imaging and Vision Series*, 1997

[8] M. Turk and A. Pentland, “Eigenfaces for Recognition,” *Journal of Cognitive Neuroscience*, 3(1), pp. 71-86, 1991.

[9] Y. Yacoob, M.J. Black, **“**Parameterized modeling and recognition of activities,**”** in *ICCV’98*, pp.120-127, 1999.

[10] I. Yamato, I.Ohya, and K. Ishii, “Recognizing Human Action in Time-Sequential Images Using Hidden Markov Model,” in *CVPR’92*, pp. 379-385, 1992.