HMM-BASED SEGMENTATION AND RECOGNITION OF HUMAN ACTIVITIES FROM VIDEO SEQUENCES

Feng Niu *and* *Mohamed Abdel-Mottaleb*[*]

Department of Electrical and Computer Engineering, University of Miami
1251 Memorial Dr., Coral Gables, FL 33146
(fniu@umsis.miami.edu, mottaleb@miami.edu)

ABSTRACT

Recognizing human activities from image sequences is an active area of research in computer vision. Most of the previous work on activity recognition focuses on recognition from video clips that show only single activities. There are few published algorithms for segmenting and recognizing complex activities that are composed of more than one single activity. In this paper, we present a novel HMM-based approach that uses *threshold* and *voting to* automatically and effectively segment and recognize complex activities. Experiments on a database of video clips of different activities show that our method is effective.

1. INTRODUCTION

Automatically recognizing human activities from video is important for applications such as automated surveillance systems and smart home applications. Several human activity recognition methods [2][5][6][7][8][10] were proposed in the past few years to classify single human activities such as walking, skipping, sitting down, etc. These methods can not be directly used to recognize complex activities, which combine more than one activity. Algorithms for segmenting and recognizing continuous complex human activities are presented in [1] and [3]. In [1], from the lateral view they extract the angles subtended by three body major components (i.e., torso, upper leg and lower leg) with the vertical axis and use them as features to classify frames into breakpoints (an action's commencement or termination) and non-breakpoints to segment complex activities. This method can only be used for profile views. In [3], they obtained HMMs by minimizing the entropy of its component distributions, this enabled the HMMs' internal states to organize observed data into highly interpretable hidden states. Then, the transitions between these states were used to segment complex activities. In their approach training has to be performed on videos of complex activities.

In this paper, we present a HMM-based approach that uses *thresholding* and *voting* for activity segmentation and recognition. Individual video frames are represented using motion and shape features as in [5]. We represented each activity by a set of Hidden Markov Models, where each model represents an activity viewed from a specific direction (or viewing angle) to realize view-invariance. We trained the models on a database of four activities (walking, sitting down, standing up, and writing on a white board) and tested our approach on a database of video clips containing two or more of the four activities. The results show that the algorithm is robust and capable of recognizing complex activities from random viewing directions.

In Section 2, we introduce the single activity model. In Section 3 we present the *threshold-based* HMM*s* that were used. Our algorithm for segmenting and recognizing complex activities is described in section 4. In Section 5, we present the experimental results. Finally, we conclude the paper in Section 6.

2. SINGLE HUMAN ACTIVITY REPRESENTATION

We use the features and the model that we presented in [5] for representing single human activities. In this section we give a brief overview of the feature extraction and the modeling steps.

2.1 Feature extraction

2.1.1 Region of interest extraction

We start by extracting the area that contains the person performing the activity, i.e., region of interest (ROI),

0-7803-9332-5/05/$20.00 ©2005 IEEE

[*] Corresponding author.

using a background subtraction algorithm [5]. Figure 1 shows an example result of background subtraction.

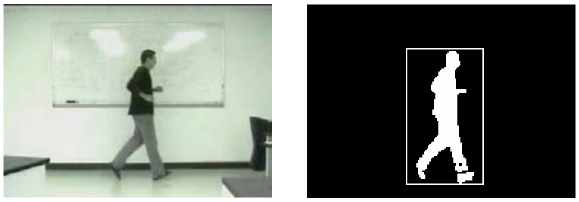
2.1.2 Motion features

A rectangular ROI is obtained from the result of background subtraction after noise removal as shown in Figure 1.b. Then, optical flow [11], $o(i,j)$, is calculated for each pixel in the ROI, and the optical flow values are normalized as follows:

$$\overline{o}(i,j) = o(i,j)/o_{\max} \qquad (1)$$

where

$$o_{\max} = \max\{o(i,j) \mid i,j \in ROI\} \qquad (2)$$



(a) **(b)**

Figure 1. a) A frame from a walking sequence, b) The ROI obtained after background subtraction.

Then, the ROI is partitioned into 64 blocks, $B(k)$, with equal sizes, where *k = 1, ..., 64.* The average optical flow vector for every block is then computed by:

$$\overline{O}_k = \begin{bmatrix} \overline{O}_{kx} \\ \overline{O}_{ky} \end{bmatrix} = \frac{1}{n} \sum_{i,j \in B(k)} \begin{bmatrix} \overline{o}_x(i,j) \\ \overline{o}_y(i,j) \end{bmatrix} \qquad (3)$$

where *n* is the number of pixels in a single block.

Then, we compose the vector $O = [\overline{O}_1, \overline{O}_2, ..., \overline{O}_{64}]^T$ for every frame to represent its motion feature vector, where each element contains two components for the *x* and the *y* directions.

2.1.3 Shape features

The size of the ROI regions that result from background subtraction is normalized to 64 by 48 pixels. Each normalized ROI image is then represented as a vector by concatenating the rows in a raster scan fashion. Thus, all ROI images are mapped to a collection of points in a large dimensional feature space, i.e., 3072 dimensions. To efficiently use the shape information, principal component analysis (PCA) was used to reduce the 3072 dimensional feature space to 90 dimensions.

2.1.4 Feature combination

The motion and shape feature vectors are simply combined together in one feature vector

$$U_i = [\overline{O}_{1x}, \overline{O}_{2x}, ..., \overline{O}_{64x}, \overline{O}_{1y}, \overline{O}_{2y}, ..., \overline{O}_{64y}, f_1, f_2, ..., f_M]^T \quad (4)$$

where $\overline{O}_{ix}$ is the *x* component of $\overline{O}_i$,and $\overline{O}_{iy}$ is the *y* component of $\overline{O}_i$, and *f_i's* are the eigen shape components, where M is set to 90. Every video clip is then represented as a sequence, $U = \{U_{1,}U_{2,},...,U_L\}$, where *L* is the number of frames in the sequence.

2.2 Human activity model

For view independent recognition of activities, a set of models are built for each activity, where each model represents the activity from a different viewing direction to capture the variations arising from the changes in the view. For a given activity, *j*, through training, a set of HMMs is obtained:

$$A_j = \{A_{j1}, A_{j2}, ..., A_{jN}\} \qquad (5)$$

Each sub model, A_{ji}, represents the activity from a different viewing angle. All the HMMs that we used in our experiments have the same topology, *i.e.*, 6-state fully connected models. The number of states was empirically determined. Each observation was modeled as a mixture of Gaussians. Two mixtures per feature dimension were used in the experiments. We used the maximum-likelihood approach to classify each activity:

$$A = \underset{A_j \in \text{all activities}}{\arg\max} \; P(U \mid A_j) \qquad (6)$$

where $P(U \mid A_j)$ is the conditional probability for activity *j*, and is computed by:

$$P(U \mid A_j) = \max_i (P(U \mid A_{ji}), i = 1, ..., N) \qquad (7)$$

where U is a sequence of feature vectors of an unknown activity, and *N* is the number of different viewing directions, in this work *N* is set to eight.

3. THRESHOLD-BASED HMM*S*

To make sure that our single activity models do not mistakenly assign a single activity label to a clip with two activities, we determine a threshold T_j for the conditional probabilities of each activity *j*. Our algorithm uses these thresholds to reject assigning an activity label to a sequence U if all the conditional probabilities, $\overline{P}(U \mid A_j)$, for the different models fall below the corresponding thresholds T_j. $\overline{P}(U \mid A_j)$ is obtained from $P(U \mid A_j)$ by normalizing w.r.t. the number of frames. In order to determine the thresholds T_j, we use a set of single activity video clips and determine the conditional probabilities $\overline{P}(X \mid A_j)$ for each clip *X* based on the correct model *j*. These probability values represent values that we need to accept. We also use a set of video clips such that each clip has two activities. These are examples of cases that we do not want the system to classify. For each of these clips *Y*, we calculate two conditional probabilities $\overline{P}(Y \mid A_i)$ and $\overline{P}(Y \mid A_k)$ based on the models that correspond to the activities, *i* and *k,* in the clip. These probability values represent values that we need to reject. Then for each activity *j*, we select a threshold T_j that minimizes the number of misclassified cases. All the conditional probabilities used in this training are normalized w.r.t. the number of frames in the corresponding video clip as follows:

$$\overline{P}(X \mid A_j) = P(X \mid A_j) / length(X) \quad (8)$$

In the above discussion we used both *X* and *Y* to denote video clips, and in the conditional probability expressions they represent the corresponding feature vector sequences. Based on this idea, the recognition result can be obtained as follows:

$$A_{final} = \begin{cases} A & if \quad \overline{P}(U \mid A\ ) > T_J \\ reject & if \quad \overline{P}(U \mid A\ ) < T_J \end{cases} \quad (9)$$

where A is computed using equation (6) and $J = \arg\max_j P(U \mid A_j)$.

4. SEGMENTATION AND RECOGNITION

In our algorithm, activity segmentation and recognition are combined in one process. During training, we train the *threshold-based* HMM*s* for each single activity separately. Then, during recognition we slide a window of length *N* over the sequence of frame features and classify the activity represented by the sequence in the window, see Figure 3. For a video clip with *M* frames we obtain a set of results r_i, *i= 1, 2, ..., M-N+1,* where resul*t* r_i is the activity assigned to window w_i. The result is used as a vote assigned to each frame in this window. We shift the window frame by frame and repeat the classification process. This will result in obtaining *N* results, r_j, for frame f_i, where *i-N+1 < j <i+1*. These classification results are considered as votes and we classify the activity of a frame by the activity that has maximum votes.

A low-pass filter was applied to smooth the voting curves as shown in Figure 3 in order to obtain the final segmentation and recognition results. Figure 3 shows two examples of voting results (after being filtered). Four curves (solid, dashed, point, star) represent votes for four activities (walking, standing up, sitting down, and writing on a white board) obtained separately for each frame. Sometimes the frames in a window contain frames from two different activities. The recognition results for these clips can be inaccurate and can induce errors in the final segmentation and recognition results. This is the reason for using *threshold-based* HMM*s* in our work.

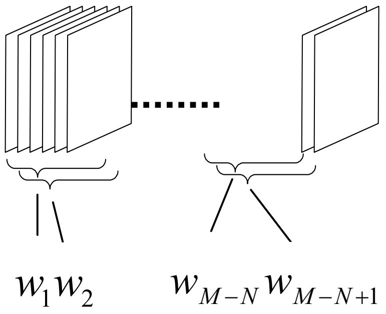


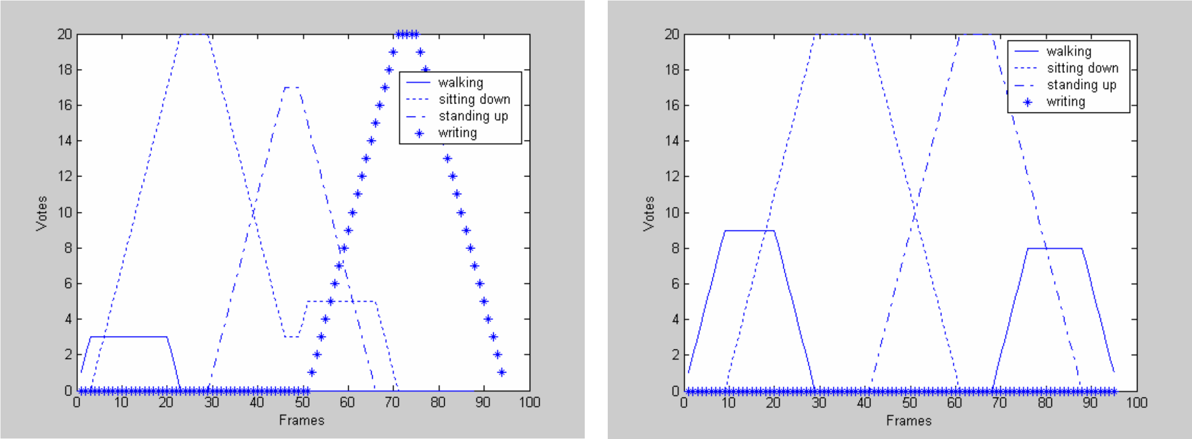
Figure 2. Sliding windows through the sequence of frames.

5. EXPERIMENTAL RESULTS

Experiments were performed on video clips that have 352x240 pixel resolution and 30 frames per second. Four sets of models were trained for walking, sitting down, standing up and writing on a white board. We collected 81 video sequences of single activities for training. The length of the sliding window was set to 20 based on the results from one video clip. The algorithm has been tested on a set of sixteen sequences of continuous complex activities consisting of totally 66 single activities. Each test sequence consisted of a set of activities performed in a continuous manner with no pauses. In our experiment, 59 single activities were properly segmented and recognized from the sixteen video clips. This means that our system has an accuracy rate of 89%. The results demonstrate the efficiency of the algorithm with respect to segmentation and recognition. Figure 4 contains 27 sample frames from one test sequence with frame numbers shown below. Figure 3.b gives the results for this sequence.

6. CONCLUSIONS

In this paper, we proposed an algorithm for activity segmentation and recognition from video clips containing complex activities. Both motion and shape features were used to represent human activities. We used threshold based HMMs to reject classifying the activity in a given sequence of frames if the evidence is not strong. We used a voting based algorithm for segmentation and recognition of activities. In our experiments, we experimented with videos that contain two or more activities. The activities included walking, sitting down, standing up, and wiring on a white board. The results showed that our algorithm is effective for segmenting and recognizing complex activities independent of the viewing direction.



(a) **(b)**

Figure 3. Two voting results for two complex activity.

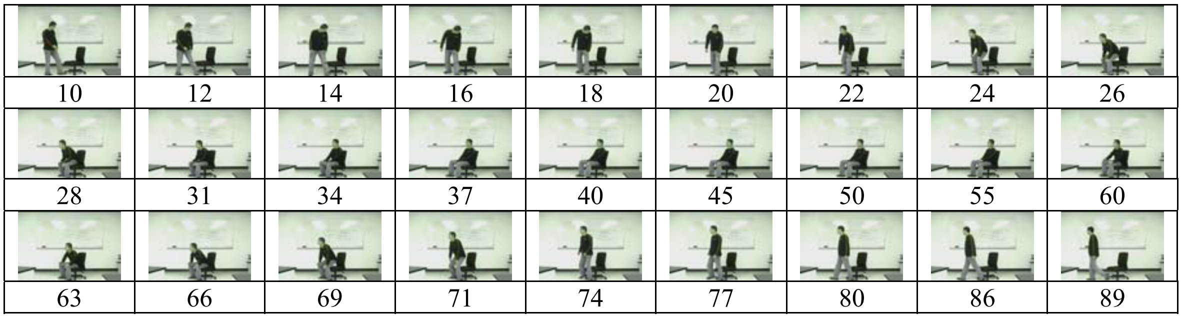


Figure 4. Sampled frames from a sequence with several activities.

7. REFERENCES

[1] A. Ali, J. K. Aggarwal, "Segmentation and Recognition of Continuous Human Activity", *IEEE Workshop on Detection and Recognition of Events in Video,* PP. 28-35, Vancouver, Canada, July 08, 2001.

[2] O. Masound, N. Papanikolopoulos, "Recognizing Human activities", *IEEE Conference on Advanced Video and Signal Based Surveillance*, PP. 157-162, Miami, Florida, July 21-22, 2003.

[3] M. Brand, V. Kettnaker, "Discovery and segmentation of activities in video" *IEEE Trans. Pattern Analysis and Machine Intelligence*., vol. 22, PP. 844—851, Aug, 2000.

[4] S. Luhr, H. H. Bui, S. Venkatesh, G. A. W. West, "Recognition of Human Activity through Hierarchical Stochastic Learning", *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, PP. 416-425, Fort worth, Texas, March 23 – 26, 2003.

[5] F. Niu, M. Abdel-Mottaleb. "View-Invariant Human Activity Recognition Basec on Shape and Motion Features", *IEEE Sixth International Symposium on Multimedia Software Engineering, pp. 546-556, Miami, FL, Dec.13-15, 2004.*

[6] N. Oliver, E. Horvitz and A. Garg, "Layered Representation for Human Activity Recognition", Proceedings *Ninth IEEE ICCV*, PP. 641-648, 2003.

[7] R. Hamid, Y. Huang, I. Essa, "ARGMode–Activity Recognition using Graphical Models", *Conference on Computer Vision and Pattern Recognition Workshop*, Volume 4, PP. 38-45, Madison, Wisconsin, June 16-22, 2003.

[8] J. Ben-Arie, Z. Wang, P. Pandit, S. Rajaram, "Human Activity Recognition Using Multidimensional Indexing", *IEEE Trans. on PAMI,* Volume 24 , Issue 8, PP. 1091-1104, August 2002.

[9] D. Gavrila, "The visual analysis of human movement: A survey," *Computer Vision and Image Understanding*, vol. 73, PP. 82--98, 1999.

[10]Y. Yacoob, M. J. Black, "Parameterized modeling and recognition of activities," J*ournal of Computer Vision and Image Understanding*, vol. 73, no. 2, PP. 232-247, 1999.

[11] B. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision", *International Joint Conference on Artificial Intelligence*, PP. 674-679, 1981.