COMPRESSED DOMAIN HUMAN MOTION RECOGNITION USING MOTION HISTORY INFORMATION

R. Venkatesh Babu and K.R. Ramakrishnan

Department of Electrical Engineering, Indian Institute of Science, Bangalore 560 012, India.
email addresses: {rvbabu,krr}@ee.iisc.ernet.in

ABSTRACT

In this paper we present a system for classifying various human actions in compressed domain video framework. We introduce the notion of quantifying the motion involved, through what we call "Motion Flow History" (MFH). The encoded motion information readily available in the compressed MPEG stream is used to construct the coarse Motion History Image (MHI) and the corresponding MFH. The features extracted from the static MHI and MFH compactly characterize the temporal and motion information of the action. Since the features are extracted from the partially decoded sparse motion data, the computational load is minimized to a great extent. The extracted features are used to train the KNN, Neural network, SVM and the Bayes classifiers for recognizing a set of seven human actions. Experimental results show that the proposed method efficiently recognizes the set of actions considered.

1. INTRODUCTION

Recognition of human actions and event detection has recently gained more interest among video processing community due to the automatic surveillance, monitoring systems [1], video indexing and retrieval, robot motion, human-computer interaction and segmentation [2, 3]. Most of the existing literature on action classification are based on pixel domain [4, 5, 6, 7, 8] and almost all the multimedia documents available nowadays are in the MPEG [9] compressed form to facilitate easy storage and transmission. Hence, it would be efficient if the classification is performed in the MPEG compressed domain without having to completely decode the bit-stream and subsequently perform classification in the pixel domain. This calls for techniques which can use information available in the compressed domain such as motion vectors and DCT coefficients.

Recently, we have developed a technique for recognizing human actions from the compressed video using Hidden Markov Model (HMM) [10], where the time-series features used for training the HMM are directly extracted from the motion vectors corresponding to each frame of the video. Though this approach has proven its ability to classify the video sequences, the extracted time series features are not suitable for other efficient classifiers such as Neural networks and Bayes.

The present work is motivated by a technique proposed by Davis *et al.,* [11] where a view-based approach is used to recognize actions. They presented a method for recognition of temporal templates. A temporal template is a static image where the value at each point is a function of the motion properties at the corresponding spatial location in an image sequence. The actions were represented by the cumulative motion images called Motion Energy Image (MEI) and MHI. The MEI represents where the motion has occurred in the image plane, whereas MHI represents the recency of motion using intensity. For recognition, the Hu moments [12], obtained from the templates are known to yield reasonable shape discrimination in a translation and scale invariant manner. Extracted Hu moments are matched using a nearest neighbor approach against the examples of given motions already learned. This work was extended by Rosales [6] using various classification approaches like KNN and Bayes with dimensionality-reduced representation of actions.

In this paper we propose a technique for building motion history images from the compressed video and extract features from the motion history information for action classification. The encoded motion information available in the MPEG video is exploited for constructing the coarse MHI and MFH. These MHI and MFH represents the human action in a very compact manner. Though the motion information extracted from each frame of the compressed video is very sparse, they are sufficient to construct the coarse MHI and MFH for representing the actions.

This paper is organized as follows: Section 2 describes the overview of the proposed work. Section 3 explains about the construction of coarse MHI and MFH. The feature extraction procedures are explained in Section 4. Section 5 presents the classification results and Section 6 concludes the paper.

2. SYSTEM OVERVIEW

The overview of the proposed system is shown in Fig. 1. First the motion vectors are extracted from the compressed video by partially decoding the MPEG video bit-stream. This partial decoding is very less expensive compared to the full decoding. Since the sampling rate of the video is normally very high (typically 25 frames/sec) compared to human motion dynamics, it is not necessary to extract the motion vectors from all the frames. So we have used only the motion vectors obtained from the predictive (P) frames for constructing the coarse MHI and MFH. As motion vectors are usually noisy, the coarse MHI and MFH are constructed after removing the noisy motion vectors. The constructed coarse MHI and MFH are at macroblock resolution not at pixel resolution. Hence the size of the MHI and MFH are sixteen times smaller than the original frame size. In feature extraction phase, various features are extracted from the constructed coarse MHI and MFH which hold the temporal and motion information of the video sequence. The features based on projection profiles and centroids are extracted from MHI. The affine features and motion vector histogram based features are obtained from the MFH. The features are finally fed to the classifiers such as KNN, Neural network, SVM

0-7803-7750-8/03/$17.00 ©2003 IEEE

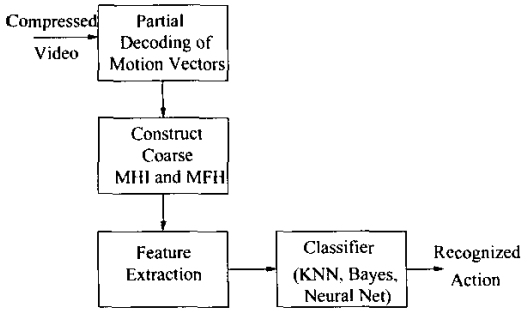

Fig. 1. Overview of the proposed system

and Bayes for recognizing the action.

3. REPRESENTATION OF ACTION USING MHI AND MFH

Since we are interested in analyzing the motion occurring in a given window of time, we need a method that allows us to capture and represent it directly from the video sequence. Such static representations are called Motion Energy Images (MEI), Motion History Images (MHI) and Motion Flow History (MFH). They are functions of the observed motion properties at the corresponding spatial image location in the video sequence.

MEI is basically a cumulative binary image with only spatial, and no temporal details of the motion involved. It answers the question 'where did the motion occur?'. MEI can be obtained by binarizing the MHI. The MHI is a cumulative gray scale image incorporating the spatial as well as the temporal information of the motion [11]. MHI points to, 'where and when did the motion occur?'. It does not indicate anything about the direction and magnitude of the motion. MFH gives the information about the extent of the motion at each macro block ('where and how much did the motion occur?'). In case of occlusion, the old motion information is over-written by the new reliable motion information.

Since it is computationally very expensive to decode the full video, we use the readily available encoded motion information in MPEG bit-stream for constructing the coarse MHI and MFH. In MPEG, the motion vectors are computed for each macroblock (of size 16×16 pixels) of (P) and (B) frames. The motion vectors not only indicate the blocks under motion but also gives the information regarding magnitude and direction of the block with respect to the reference frame. The spurious motion vectors which do not belong to the moving object are removed by connected component analysis before constructing MFH and MHI. The MFH and MHI are constructed from non zero P frame motion vectors according to the following:

$$MFH_d(k,l) = \begin{cases} m_d^{kl}(\tau) & \text{if } E\left(m_d^{kl}(\tau)\right) < T_r \\ M\left(m_d^{kl}(\tau)\right) & \text{otherwise} \end{cases} \quad (1)$$

where,
$E\left(m_d^{kl}(t)\right) = \| m_d^{kl}(\tau) - med\left(m_d^{kl}(\tau) \ldots m_d^{kl}(\tau-\alpha)\right) \|^2$ and $M\left(m_d^{kl}(\tau)\right) = med\left(m_d^{kl}(\tau) \ldots m_d^{kl}(\tau-\alpha)\right)$, here *med* refers to median filter, $m_d^{kl}(\tau)$ can be horizontal (m_x) component or vertical (m_y) component of motion vector located at kth row and lth column in frame τ and α indicates the number of previous P frames to be considered for median filtering. Typical range of α is 3-5. The function E checks the reliability of the current motion vector with respect to the past non-zero motion vectors at the same location against a predefined threshold T_r. This makes sure that no reliable motion vector will be replaced by a recent noisy motion vector. Such spurious motion vectors are replaced by the reliable median value.

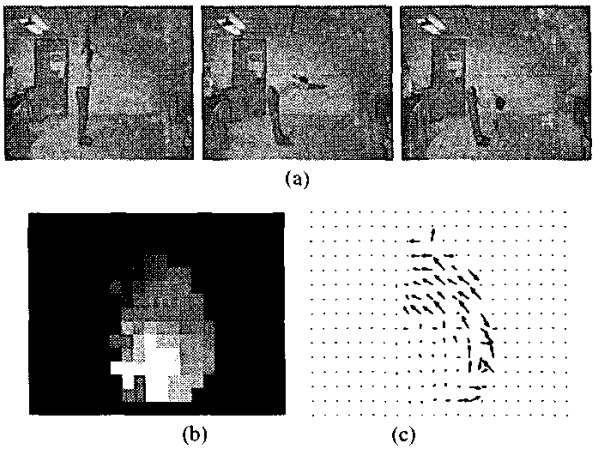

Fig. 2. (a) Key-frames of bend-down sequence and the corresponding coarse (b) MHI (c) MFH

$$MHI(k,l) = \begin{cases} \tau & \text{if } \psi\left(m^{kl}(\tau)\right) \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where, $\psi\left(m^{kl}(\tau)\right) = |m_x^{kl}(\tau)| + |m_y^{kl}(\tau)|$

Fig. 2 shows the key frames of the bend-down action and the corresponding MHI and MFH. The MHI is a function of the recency of the motion at every macroblock. The brightness of the macroblock is proportional to how recently the motion occurred. Whereas MFH describes the spatial distribution of motion over the video clip without temporal information. The MHI which has temporal information but no motion information is complemented by the MFH which has motion information without temporal information. Thus MHI and MFH together capture the temporal and motion information of the entire video sequence. The drawback of this representation is that, self occlusion or overlapping of motion on the image plane may result in the loss of a part of the motion information. However it might be representative enough for all human actions.

4. FEATURE EXTRACTION

Given the MHI and MFH of an action, it is essential to extract some useful features for classification. We have extracted features from MHI based on i) Projection profiles and ii) Centroid. The MFH based features are i) Affine motion model ii) Projected 1-D feature and iii) 2-D Polar feature [10].

4.1. MHI features

Projection profile based feature : Let N be the the number of rows and M be the number of columns of MHI. Then the vertical

profile is given by the vector P_v of size N and defined by: $P_v[i] = \sum_{j=1}^{M} MHI[i,j]$. The horizontal profile is represented by the vector P_h of size M and define by: $P_h[j] = \sum_{i=1}^{N} MHI[i,j]$. The features representing the distribution of projection profile with respect to the centroid are computed as follows:

$$\mathbf{F}_{pp} = \left[\frac{\sum_{i=1}^{h_c} P_h[i]}{\sum_{i=h_c+1}^{M} P_h[i]} \quad \frac{\sum_{i=1}^{v_c} P_v[i]}{\sum_{i=v_c+1}^{N} P_v[i]} \right] \quad (3)$$

where, h_c and v_c are the horizontal and vertical centroids of MEI.

Centroid based feature : The other feature is computed as the shift of centroids of MEI and MHI, which is given by:

$$\mathbf{F}_c = [MHI_{xc} - MEI_{xc} \quad MHI_{yc} - MEI_{yc}] \quad (4)$$

The above feature indicates the approximate direction of the centroid motion of the corresponding action.

4.2. MFH features

Three types of features are extracted from MFH. Since it holds the entire history of spatial motion information, many useful features are extracted from MFH.

Affine feature : An important feature is the six-parameter affine motion model 'a' corresponding to the MFH. This affine model is sufficient enough to capture the flow characteristics of MFH. The affine parameters **a** are computed as follows:

$$\mathbf{a}^T = \left[\sum \mathbf{\Pi}(\boldsymbol{p})^T \mathbf{\Pi}(\boldsymbol{p}) \right]^{-1} . \sum \mathbf{\Pi}^T(\mathbf{p}) \mathbf{v}(\mathbf{p}). \quad (5)$$

where,

$$\mathbf{\Pi}(\mathbf{p}) = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{bmatrix}$$

$\mathbf{p} = [x\ y]^T$ is the vector representing the position of pixel in the image plane. and $\mathbf{v}(\mathbf{p})$ is the motion at location $\mathbf{p}$. Here all the motion vectors are assigned to the center pixel of the corresponding macroblock.

Projected 1-D feature : This feature is computed from the histograms of horizontal and vertical components of motion vector. Let $f_x(m_x; r_i, r_j)$, $f_y(m_y; r_i, r_j)$ be the number of horizontal and vertical components of motion vectors in the range r_i and r_j, with non-overlapping intervals ($r_i < r_j$), then the combination $[f_x, f_y]$ forms the feature vector. The ranges used in the experiment for horizontal and vertical component to get a 10 dimensional feature vector are $[\min, -8, -3, 3, 8, \max]$.

2-D polar feature : The motion vector direction and magnitude for each macroblock is obtained from both horizontal and vertical components of the corresponding motion vector. The number of motion vectors falling between the angle range θ_i and θ_j and having magnitude within the range r_i and r_j can be expressed as

$$f_\theta(m_\theta; r_i, r_j, \theta_i, \theta_j) = \#\{(k,l) : r_i \le |m_\theta| \le r_j \text{ and } \theta_i \le \angle m_\theta \le \theta_j\} \quad (6)$$

where m_θ is the motion vector (m_x, m_y) in polar co-ordinates with $|m_\theta| = \sqrt{m_x^2 + m_y^2}$ and $\angle m_\theta = \tan^{-1}\left(\frac{m_y}{m_x}\right)$

Here, (r_i, r_j) and (θ_i, θ_j) are chosen so as to cover the entire range of motion vectors and the angle ranges from $-\pi$ to π in a non-overlapping manner. In our experiment, the ranges of θ used are $[-\pi, -\pi/2, 0, \pi/2, \pi]$ and the ranges of r used are $[0, 5, 10, \max]$, which leads to a feature vector of 12 dimensions. The following table summarizes the features used in our experiment.

	Feature	Dimension
MHI Features	Proj. Profile	2
	Centroid	2
MFH Features	Affine	6
	1-D Projected	10
	2-D Polar	12
	Total	32

Table 1. Features extracted from MHI and MFH

5. CLASSIFICATION RESULTS AND DISCUSSION

We have used four types of classifiers for recognizing the action, namely Normalized K-nearest neighbors (KNN), Bayesian, Neural network (MLP) and SVM. Totally seven actions were considered for recognition namely walk, run, jump, bend down, bend up, twist left and twist right. In our experimental setup, we trained the system with 10 instances of each action performed by four to five different subjects. For testing, we have used at least five instances per action with the subjects that are not used for training phase. The total number of samples used for training is 70 (10 samples/action) and 51 samples for testing.

The KNN algorithm simply selects the k-closest samples from the training data to the new instance and the class with the highest number of votes is assigned to the test instance. An advantage of this technique is due to its non-parametric nature, because we do not make any assumptions on the parametric form of the underlying distribution of classes. In high dimensional spaces these distributions may be often erroneous. Even in situations where second order statistics can not be reliably computed due to limited training data, KNN performs very well, particularly in high dimensional feature spaces and on atypical samples. Table 2 shows the classification results of KNN classifier with all aforementioned features.

Class	Walk	Run	Jump	BD	BU	TWL	TWR	Error
Walk	**5**	0	0	0	0	0	0	0
Run	0	**7**	0	0	0	0	0	0
Jump	0	0	**7**	0	0	0	0	0
BD	0	0	0	**11**	0	0	0	0
BU	0	0	0	0	**8**	1	0	1
TWL	0	0	0	0	0	**6**	0	0
TWR	0	0	0	0	0	0	**6**	0
Error	0	0	0	0	0	1	0	**1**

Table 2. Confusion matrix for KNN Classifier (k=3)

The second classifier is Bayes classifier - a parametric classifier that assumes normal distribution for class (ω) conditional probability of feature vector $\mathbf{x}$, $P(\mathbf{x}|\omega_i)$. Though Bayes classifier is optimal, the performance degrades if the models used are erroneous. Hence, we added the features one by one as long as the classification result improves on the training data. Table 3 shows the performance of Bayes classifier with only 4 selected features out of total 32 features. Table 4 shows the classification results for a network trained with 2 hidden layers with 15 neurons in each layer using all the features. Table 5 shows SVM classification results with radial-based kernel.

Class	Walk	Run	Jump	BD	BU	TWL	TWR	Error
Walk	**3**	2	0	0	0	0	0	2
Run	0	**7**	0	0	0	0	0	0
Jump	0	0	**7**	0	0	0	0	0
BD	0	0	0	**11**	0	0	0	0
BU	0	0	0	0	**9**	0	0	0
TWL	0	0	0	0	0	**6**	0	0
TWR	0	0	0	0	0	1	**5**	1
Error	0	2	0	0	0	1	**0**	**3**

Table 3. Confusion matrix for Bayes Classifier

Class	Walk	Run	Jump	BD	BU	TWL	TWR	Error
Walk	**4**	1	0	0	0	0	0	1
Run	0	**7**	0	0	0	0	0	0
Jump	0	0	**7**	0	0	0	0	0
BD	0	0	0	**11**	0	0	0	0
BU	0	0	0	0	**9**	0	0	0
TWL	0	0	0	0	0	**6**	0	0
TWR	0	0	0	0	0	0	**6**	0
Error	0	1	0	0	0	0	0	**1**

Table 4. Confusion matrix for Neural Net Classifier

Comparing the results of the classifiers, the results obtained by KNN, Neural Net and SVM show excellent performance. Bayes classifier recognizes most of the actions, but fails to discriminate between 'walk' and 'run' actions. The parametrization of the underlying feature distribution could have caused it. Moreover the Bayes result is obtained only with the selected 4 features, whereas the other classifiers use all features. Table 6 summarizes the recognition results for various classifiers.

6. CONCLUSION

In this paper we have proposed the method for constructing coarse Motion History Image (MHI) and Motion Flow History (MFH) from compressed MPEG video with minimal decoding. Various useful features are extracted from the above mentioned two motion representations for human action recognition. We have shown the recognition results for four classification paradigms. Though the test instances are from entirely different subjects that are used for training the classifiers, the results show excellent recognition accuracy. Since the data is handled at macroblock level, the computational cost is extremely less compared to the pixel domain processing.

Class	Walk	Run	Jump	BD	BU	TWL	TWR	Error
Walk	**5**	0	0	0	0	0	0	0
Run	0	**7**	0	0	0	0	0	0
Jump	0	0	**7**	0	0	0	0	0
BD	0	0	0	**11**	0	0	0	0
BU	0	0	0	0	**8**	1	0	1
TWL	0	0	0	0	0	**6**	0	0
TWR	0	0	0	0	0	0	**6**	0
Error	0	0	0	0	0	1	0	**1**

Table 5. Confusion matrix for SVM classifier (RBF-kernel)

Classifier	No. of Features used	Classification Accuracy
KNN (k=3)	32	98.0%
Neural Net	32	98.0%
Bayes	4	94.1%
SVM (RBF-kernel)	32	98.0%

Table 6. Comparison of various Classifiers

7. REFERENCES

[1] Douglas Ayers and Mubarak Shah, "Monitoring human behavior from video taken in an office environment," *Image and Vision Computing*, vol. 19, no. 12-1, pp. 833–846, Oct. 2001.

[2] M. Shah and R. Jain, *Motion based Recognition*, Kluwer Academic, 1997.

[3] T. Starner and A. Pentland, "Real-time american sign language recognition from video using hidden markov models," Tech. Rep. TR-375, MIT Media Lab, 1995.

[4] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden markov model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1992, pp. 379–385.

[5] C. Bregler, "Learning and recognizing human dynamics in video sequences," in *Proceedings of the IEEE CVPR*, 1997, pp. 568–574.

[6] R. Rosales, "Recognition of human action based on moment based features," Tech. Rep. BU 98-020, Boston University, Computer Science, Nov. 1998.

[7] Aaron F. Bobick and James W. Davis, "The recognition of human movement using temporal templates," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 257–267, Mar. 2001.

[8] Xinding Sun and B. S. Manjunath, "Panoramic capturing and recognition of human activity," in *IEEE International Conference on Image Processing*, Sept. 2002, vol. 2, pp. 813–816.

[9] J.L. Mitchell, W.B. Pennebaker, C.E. Fogg, and D.J. LeGall, *MPEG Video Compression Standard*, International Thomson Publishing, 1996.

[10] R. Venkatesh Babu, B. Anantharaman, K. R. Ramakrishnan, and S. H. Srinivasan, "Compressed domain action classification using HMM," *Pattern Recognition Letters*, vol. 23, no. 10, pp. 1203–1213, Aug. 2002.

[11] J. Davis and A.F. Bobick, "The representation and recognition of human movements using temporal templates," in *Proceedings of the IEEE CVPR*, 1997, pp. 928–934.

[12] M.K. Hu, "Visual pattern recognition by moment invariants," *IRE Trans. on Information Theory*, vol. 8, no. 2, pp. 179–187, Feb. 1962.