IJDCST @November Issue- V-1, I-7, SW-40
ISSN-2320-7884 (Online)
ISSN-2321-0257 (Print)

Perspective of Homogenous Motion Reincarnation Based on Artificial Intelligence

[1]Mallela.SivaNagaraju, [2]B. Srinivasa Rao, [3]V. Sowjanya, [4]V. Siva Parvathi

[1] Student, PVPSIT, KANURU, VIJAYAWADA.

[2] Senior Assistant Professor, PVPSIT, KANURU, VIJAYAWADA.

[3] Assistant Professor, PVPSIT, KANURU, VIJAYAWADA.

[4] Assistant Professor, PVPSIT, KANURU, VIJAYAWADA.

Abstract:-Learning-based approaches for homogenous motion reincarnation often rely on large training sets. Most of these approaches do not perform well when only a few training samples are available. In this paper, we consider the problem of homogenous motion reincarnation from a multi clip per action. Each clip contains at most 25 frames. Here in this paper we are using a novel technique of image reincarnation based on artificial intelligence. Algorithm for extracting and classifying multi clip motion in a sequence based on motion trajectories. First, a multiscale segmentation is performed to generate homogeneous regions in each frame. Regions between consecutive frames are then matched to obtain two-view correspondences. Affine transformations are computed from each pair of corresponding regions to define pixel matches. Pixels matches over consecutive clip pairs are concatenated to obtain pixel-level motion trajectories across the clip sequence. Motion patterns are learned from the extracted trajectories using a time-delay neural network. The novel representation of action videos is based on learning spatially related human body posture prototypes using Self Organizing Maps (SOM). Fuzzy distances from human body posture prototypes are used to produce a time invariant action representation. Multilayer perceptions are used for action classification. The algorithm is trained using data from a multi-camera setup. An arbitrary number of cameras can be used in order to recognize actions using a Bayesian framework. Some of the drawbacks are occurred in this frame work we are proposing the recognition of human motion into the following two: (i) state-space based and (ii) template matching based approaches.

I. INTRODUCTION

Event detection and human action recognition have gained more interest of late, among video processing community because they find various applications in automatic surveillance, monitoring systems [2], video indexing and retrieval, robot motion, human – computer interaction and segmentation [28,30]. One of the important applications of human action recognition is the automatic indexing of video sequences, while most of the multimedia documents available nowadays are in the MPEG [21] compressed form, to facilitate easy storage and transmission, majority of the existing techniques for human action recognition are pixel domain based [35,8,27,5,32,13,1,26] which are computationally very expensive. 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. In the recent past, we reported a technique for recognizing human actions from compressed video using Hidden Markov Model (HMM)[3], 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 K-nearest neighbors (KNN), Neural networks, SVM and Bayes. In this paper we propose a technique for building coarse Motion History Image (MHI) and Motion Flow History (MFH) from the compressed video and extract features from these static motion history information for characterizing human actions. The MHI gives the temporal information of the motion at the image plane, whereas the MFH quantifies the motion at the image plane. The features extracted from MHI and MFH were used to train KNN, Bayes, Neural Network and SVM classifiers for recognizing a set of seven human actions. The encoded motion information available in the MPEG video is exploited for constructing the coarse MHI and MFH. These MHI and MFH represent 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 MHI and MFH for representing the actions. This work is motivated by a technique proposed by Davis and Bobick [11] where a view-based approach is used to recognize actions. They have 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. MEI indicates where the motion has occurred in the image plane, whereas MHI indicates the regency of motion using intensity. For recognition, the Humoments [16], 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 [27] using various classification approaches like KNN and Bayes with dimensionality-reduced represen-tation of actions. This paper is organized as follows: Section 2 gives a brief description of the proposed work from the literature. Section 3 describes the basics of MPEG video compression and the overview of the proposed work. Section 4 explains about the construction of MHI and MFH. Feature extraction procedures are explained in Section 5. Section 6 presents the Classification results. Human action recognition is an active research field, due to its importance in a wide range of applications, such as intelligent surveillance [1], human-computer interaction[2], content-based video compression and retrieval [3], augmented reality [4], etc. The term action is often confused with the terms activity and movement. An action (sometimes also called as movement) is referred to as a single period of a human motion pattern, such as a walking step. Activities consist of a number of actions/movements, i.e., dancing consists of successive repetitions of several actions, e.g. walk, jump, wave hand, etc. Actions are usually described by using either features based on motion information and optical flow [5], [6], or features devised mainly for action representation [7], [8]. Although the use of such features leads to satisfactory action recognition results, their computation expensive. Thus, in order to perform action recognition at high frame rates, the use of simpler action representations is required. Neurobiological studies [9] have concluded that the human brain can perceive actions by observing only the human body poses (postures) during action execution. Thus, actions can be described as sequences of consecutive human body poses, in terms of human body silhouettes [10], [11], [12]. After describing actions, action classes are, usually, learned by training pattern recognition algorithms, such as Artificial Neural Networks (ANNs) [13], [14], [15], Support Vector Machines (SVMs) [16], [17] and Discriminate dimensionality reduction techniques [18]. In most applications, the camera viewing angle is not fixed and human actions are observed from arbitrary camera viewpoints. Several researchers have highlighted the significant impact of the camera viewing angle variations on the action recognition performance [19],[20]. This is the so-called viewing angle effect. To provide view-independent methods, the use of multi-camera setups has been adopted [21], [22], [23]. By observing the human body from different viewing angles, a view-invariant action representation is obtained. This representation is subsequently used to describe and recognize actions. Although multi-view methods address the viewing angle effect properly, they set a restrictive recognition setup, which is difficult to be met in real systems [24]. Specifically, they assume the same camera setup in both training and recognition phases. Furthermore, the human under consideration must be visible from all synchronized cameras. However, an action recognition method should not be based on such assumptions, as several issues may arise in the recognition phase. Humans inside a scene may be visible from an arbitrary number of cameras and may be captured from an arbitrary viewing angle. Inspired from this setting, a novel approach in view-independent action recognition is proposed. Trying to solve the generic action recognition problem, a novel view-invariant action recognition method based on ANNs is proposed in this paper. The proposed approach does not require the use of the same number of cameras in the training and recognition phases. An action captured by an arbitrary number N of cameras, is described by a number of successive human body postures. The similarity of every human body posture to body posture prototypes, determined in the training phase by a self-organizing neural network, is used to provide a time invariant action representation. Action recognition is performed for each of the N cameras by using a Multi-Layer Perception (MLP), i.e., a feed-forward neural network. Action recognition results are subsequently combined to recognize the unknown action. The proposed method performs view-independent action recognition, using an uncalibrated multi-camera setup. The combination of the recognition outcomes that correspond to different viewing angles leads to action recognition with high recognition accuracy. The main contributions of this paper are: a) the use of Self Organizing Maps (SOM) for identifying the basic posture prototypes of all the actions, b) the use of cumulative fuzzy distances from the SOM in order to achieve time-invariant action representations, c) the use of a Bayesian framework to combine the recognition results produced for each camera, d) the solution of the camera viewing angle identification problem using combined neural networks. An overview of the recognition framework used in the existing approach and a small discussion concerning the action recognition task is given in Section 7. Section 8 presents details of the processing steps performed in the proposed method. Experiments for assessing the performance of the proposed method are described in Section 9. Finally, conclusions are drawn in Section 10.

II. PROPOSED WORK

In this section we will give a brief description of works related to human motion and gesture recognition. The recognition of human motion can be broadly classified into the following two: (i) state-space based and (ii) template matching based approaches.

2.1. State-Space Based Approaches

State-space approach uses time-series features obtained from the image sequences for recognition. The widely used state-space model for activity recognition is HMM due to its success in the speech community. The first attempt to use HMM for activity recognition is done by Yamato et al.[35], where discrete HMMs are used for recognition of six tennis strokes. In their approach time sequential images expressing human actions are transformed to an image feature vector sequence by extracting mesh[33]feature vector from each image. The mesh features are extracted from a binarized

image obtained after subtracting the background image from original image by applying a suitable threshold. The drawbacks of this method are that it is sensitive to position displacement, noise, and also exhibits poor performance if the training and test subjects are different. The gesture recognition work by Darrell and Pentland[9] uses time-warping technique for recognition which is closely related to HMM. On similar lines, dynamic time warping is used in Ref.[6] to match an input signal to a deterministic sequence of states. Starner and Pentland[31] used HMMs to recognize a limited vocabulary of American Sign Language (ASL) sentences. Here, they used a view based approach with a single camera to extract two-dimensional (2D) features as input to HMMs. In the work by Bregler[8], this classification problem has been approached from a statistical view point. For each pixel in the image, the spatio-temporal image gradient and the color values are represented as random variables. Then the blob hypothesis is used wherein each blob is represented with a probability distribution over coherent motion, color and spatial support regions. Recently Ivanov and Bobick[17] proposed a method, which combines statistical techniques used for detecting primitive component of an activity with syntactic recognition of process structure. In this approach the recognition problem is divided into two levels: (i) the lower level detection of primitive components of activity followed by (ii) the syntactic recognition of the primitive features using a stochastic context-free grammar parsing mechanism. Another HMM based human activity recognition method is reported by Psarrou et al.[25]. Here the recognition is based on learning prior and continuous propagation of density models of behavior patterns. Ng et al. [22] proposed a real-time gesture recognition system

incorporating hand posture and hand motion. The recog-nition is done with HMM and recurrent neural networks (RNN). There are few works reported in literature which use neural networks for gesture recognition[19,7]. Boehm et al. [7]used Kohonen Feature Maps (KFM)[18]for recognizing dynamic gestures. Oliver et al.[23]proposed a system for modeling and recognizing human behaviors in a visual surveillance task. This system segments the moving objects from the background and a kalman filter tracks the object's features such as location, coarse shape, color and velocity. These features are used for modeling the behavior patterns through training HMMs and coupled HMMs (CHMM), which are used for classifying the perceived behaviors. Based on the above-mentioned work Madabhushi and Aggarwal[20] presented a system for recognition of human action by tracking the head of the subject in an image sequence. The difference in centroids of the head over successive frames form their feature vector. The human actions are modeled based on the mean and covariance of the feature vector. Here detection and segmentation of the head is done manually. Apart from the above mentioned pixel domain state-space based approaches, recently we have proposed a technique for recognizing human actions using HMM in compressed domain framework [3]. Here the time series features from the MPEG video are extracted from the readily available motion vectors of each inter coded frame. Totally seven actions were considered for recognition (walk, run, jump, bend up, bend down, twist right and twist left). A discrete HMM for each action is trained with the corresponding MPEG video sequences. The recognition of a given action is achieved by feeding the test sequence to all the trained HMMs and employing a likelihood-based measure. The performance of the system for three types of

motion-based features were compared. 2.2. Template matching based approaches One of the earlier works using this approach is found in the work done by Polana and Nelson [24], where the flow information is used as feature. They compute the optical flow fields[15]between consecutive frames and divide each frame into a spatial grid and sum the motion magnitude to get the high dimension feature. Here they assume that the human motion is periodic. The final recognition is performed using nearest neighbor algorithm. Davis and Bobick [11,5] presented a real-time approach for representing human motion using compact MHIs in pixel domain. Here, the recognition of 18 exercises was achieved by statistically matching the higher order moment based feature extracted from the MHI. The limitations of the above method are related to the 'global image' feature calculations and specific label based recognition. To overcome these limitations the author extended the previous approach with a mechanism to compute dense local motion vector field directly from the MHI for describing the movement [10]. For obtaining the dense motion, the MHI is represented at various pyramid levels to tackle multiple speeds of motion. These hierarchical MHIs are not directly created from the original MHI, but through the pyramid representation of the silhouette images. This indirect way of generating MHI pyramid increases the computational load. The resulting motion is character-ized by a polar histogram of motion orientation. Rosales [27]use these motion energy and MHIs[11]for obtaining the spatial location and the temporal properties of human actions from raw video sequences. From these motion energy and MHIs, a set of Hu-moment[16]features that are invariant to translation, rotation and scaling are generated. Using principal component analysis, the dimension of the Hu-moment space is reduced in a statistically optimal

way. The recognition performances were evaluated for the following three classifiers namely KNN, Gaussian and mixtures of Gaussian. All the above mentioned techniques process the data in the pixel domain, which is computationally very expensive.

III. SYSTEM OVERVIEW

The objective of our work is to rapidly process the video stored in MPEG format, without full-frame decompression, for recognizing human actions. Here we are using the motion vector data, which is easily extractable from the MPEG video bit-stream for our recognition task. Though we have used MPEG-1 video, our algorithm is easily extendable to MPEG-2 or the recent MPEG-4 video streams. To begin with, we briefly describe the relevant parts of MPEG video compression standard. The MPEG-1 video defines three types of coded pictures namely: interceded (I-frames); predicted (P-frames); and bidirectional predicted (B-frames). These pictures are organized into sequences of groups of pictures (GOP) in MPEG video streams. A GOP must start with anI-frame, followed by any number of P- and B-frames. The I- and P-frames are referred as anchor frames. The B-frames appear between each pair of consecutive anchor frames in the GOP and before the I-frame of the next GOP. Fig. 1 shows the typical GOP structure that is used in our work with 12 frames in a GOP. Each frame of the video is divided into non-overlapping macroblocks. For video coding in 4:2:0 format[29], each macroblock consists of six 8×8 pixel blocks: four luminance(Y)blocks and two chrominance(Cb;Cr)blocks. Each macroblock is either intra coded or inter coded. AnI-frame is completely intra coded. Here every 8×8 pixel block in the macroblock is transformed to frequency domain using the discrete cosine transformation (DCT). The resulting 64 DCT coefficients are then quantized (lossy) and entropy (run length and Huffman, lossless) encoded to achieve compression. Since coding of I-frame does not refer to any other video frames, it can be decoded independently and thus provides access points for fast random access to the compressed video.

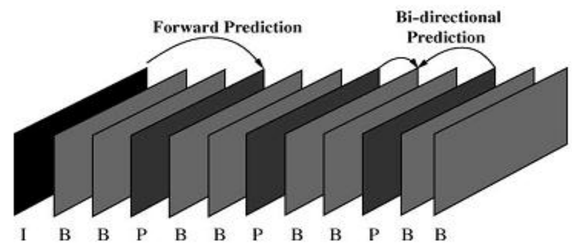

Figure 1: Data representation for input operations.

Each-frame is predicatively encoded with reference to its previous anchor frame (i.e. previous I-or P-frame). For each macroblock (16£16 pixel block) inP-frame, a local region in the anchor frame is searched for a good match in terms of difference in intensity. If a good match is found, the macroblock is represented by a motion vector to the position of the match together with the DCT encoding of the residue (i.e. difference) between the macro block and its match. The DCT coefficients of the residue are quantized and entropy coded while the motion vector is differentially and entropy coded with respect to its neighboring motion vector. This is known as forward motion compensation, and such macro blocks are referred a sinter coded macro blocks. If good match cannot be found, the macro block is intra coded like the macro blocks of I-frame. Since the residue of an inter coded macro block can be coded with fewer bits, it has better compression gain compared to the intra coded macro block. In our work the motion vectors extracted from these P-frames are used for

ISSN-2320-7884 (Online)
ISSN-2321-0257 (Print)

recognizing human actions. The coding of P-frame is illustrated in Fig. 2. To achieve further compression, B-frames are bidirectional-ally predicatively encoded with forward and/or backward motion compensation referenced to its closest past and/or future I- and/or P-frames.

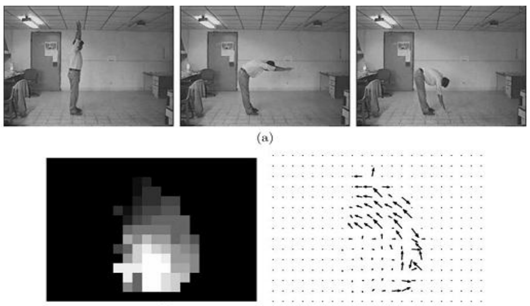

Figure 2: (a) key frames of band down sequence and corresponding coarse (b) MHI (c) MFH.

Since B-frames are not used as reference frames, they can accommodate more distortion, and thus, higher compression gain compared to I-or P-frames. The overview of the proposed system is shown inFig. 3. First the motion vectors are extracted from the compressed video by partially decoding the MPEG video bit-stream. This partial decoding is far less expensive compared to the full decoding. Since the sampling rate of the video is normally very high (typically 25 frames/s) compared to human motion dynamics, it is not necessary to extract the motion vectors from all the frames. Hence we have used

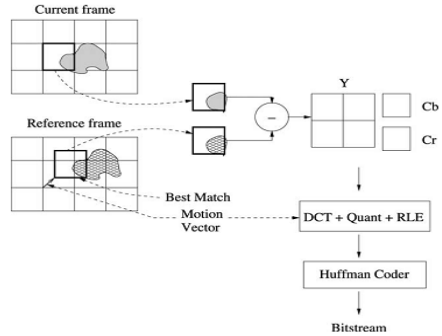

Figure 3: Coding of P-frame.

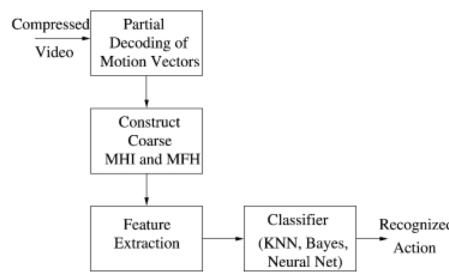

Figure 4: Overview of the proposed work.

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 macro block resolution and not at pixel resolution. Hence the size of the MHI and MFH are sixteen times smaller than the original frame size i.e. 162 times smaller in terms of number of pixels. 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. Affine features and motion vector histogram based features are obtained from the MFH. These features are finally fed to the classifiers such as KNN, Neural network and Bayes for recognizing the action.

IV. 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 motion directly from the video sequence. Such static representations are called MEIs, MHIs and MFH. They are functions of the observed motion parameters 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 convey any information about the direction and magnitude of the motion. MFH gives the information about the extent of the motion at each macroblock ('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.

Posture images of eight actions taken from various viewing angles.

Figure 5: Image actions in different angles.

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. The motion vectors not only indicate the blocks under motion but also give 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. To remove the spurious motion vectors, first a binary image of the frame is generated from the motion vector magnitude with a threshold of 0.5 to retain the half-pel motion values. Then a simple morphological clean operation is employed to remove isolated motion vectors (1's surrounded by 0's). The MFH is constructed from non-zero P-frame motion vectors according to the following:

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

where $E(m_d^{kl}(\tau)) = \|m_d^{kl}(\tau) - \text{med}(m_d^{kl}(\tau)\ldots m_d^{kl}(\tau-\alpha))\|^2$
and $M(m_d^{kl}(\tau)) = \text{med}(m_d^{kl}(\tau)\ldots m_d^{kl}(\tau-\alpha))$.

Here med refers to median filter, $m_d^{kl}(\tau)$ can be horizontal(mx) component or vertical (m_y) component of motion vector located atkth row and lth column in frame tanda indicates the number of previous P-frames to be considered for median filtering. Typical range ofais 3 – 5 for various kinds of noise. Since the correlation of the frames decreases with the temporal distance between them, it is not advisable to increase the avalue beyond 5. The functionEchecks the reliability of the current motion vector with respect to the past non-zero motion vectors at the same location against a predefined threshold Tr : The purpose of this thresholdTr is to check the reliability of each newly arriving motion vector. Considering the human motion dynamics, the motion vectors of current P-frame cannot change

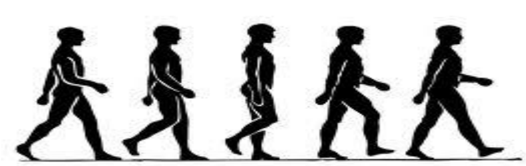

Figure 6: Image formation with different image extraction processes.

much with respect to the neighboring P-frame motion vectors. At the same time the threshold should not be too tight since most of the recent motion vectors would then be ignored. In our system the threshold Tr is set at 4 for generating MFH. In other words, this threshold Tr makes sure that no reliable motion vector of MFH will be replaced by a recent noisy motion vector. Such spurious motion vectors are replaced by the reliable median value. The MHI is constructed as given by Eq. (2),

$$MHI(k,l) = \begin{cases} \tau & \text{if } (|m_x^{kl}(\tau)| + |m_y^{kl}(\tau)|) \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Figs. 4 and 5show the key frames of the bend-down and twist-left actions and the corresponding coarse MHI and MFH 2 . The coarse MHI and MFH of other actions are shown inFig. 6. 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. The MFH describes the spatial distribution of motion vectors over the video clip. In other words MFH quantifies the motion at spatial locations through horizontal and vertical components of the motion. The MHI, which has spatio-temporal information but no motion vector infor-mation, is complemented by the MFH. Thus MHI and MFH together capture the temporal and motion vector (mx,my) 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 proposed work compared to efficient results.

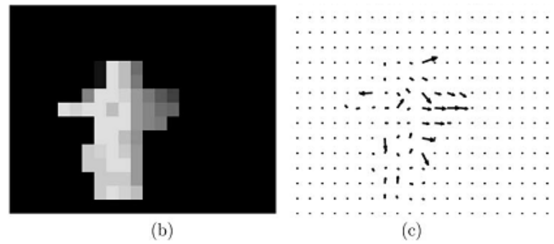

Figure 7: Image extraction with different rotations.

the motion information. However, it might be representative enough for the considered human actions.

V. 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 1D feature and (iii) 2D polar feature[3].

5.1. MHI features

Projection profile based feature. Let N be the number of rows and M be the number of columns of MHI. Then the vertical profile is given by the vector Pv of size N and defined as Pv[i] $=\sum^{N}_{j=1}$ MHI[i,j]. The horizontal profile is

Represented by the vector P_h of size M and define as $P_h[j]= =\sum^{N}_{i=1}$ MHI[i,j]: The features representing the distribution of projection profile with respect to the centroid are computed as

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

wherehc andvc are the horizontal and vertical centroids of MEI. The above featureðFpp Þ indicates the bias of the MHI along horizontal and vertical direction with respect to the centroid of MEI. This indirectly conveys the temporal information of motion along horizontal and vertical direction. Centroid based feature. This feature is computed as the shift of centroids of MEI and MHI, which is given by the 2D vector

$$\mathbf{F}_c = [\,\mathrm{MHI}_{xc} - \mathrm{MEI}_{xc} \quad \mathrm{MHI}_{yc} - \mathrm{MEI}_{yc}\,] \tag{4}$$

5.2. 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. Though it is difficult to capture some complex motion, affine model gives a good approximation to the actual optical flow of the planar surface under orthographic projection[12]. An affine model requires six basic flow fields as shown inFig. 7. The affine parameters are estimated by standard linear regression techniques. The regression is applied separately on each motion vector component since thexaffine parameter depends only on horizontal component of motion vector andy parameter depends only on the vertical component of motion vector. Let c=[c1…c6] be the 6D affine parameter vector. Then the linear least squares estimate of c is given by:

$$\mathbf{c}^{\mathrm{T}} = \left[\sum \mathbf{\Pi}(\mathbf{p})^{\mathrm{T}}\mathbf{\Pi}(\mathbf{p})\right]^{-1} \cdot \sum \mathbf{\Pi}^{\mathrm{T}}(\mathbf{p})\mathbf{v}(\mathbf{p}) \tag{5}$$

where

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

Table 1
Features extracted from MHI and MFH

	Feature	Dimension
MHI based	Proj. profile	2
	Centroid	2
MFH based	Affine	6
	1D projected	10
	2D polar	12
	Total	32

Table 2
KNN classification result for $k = 3$

Input class	Result							
	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**

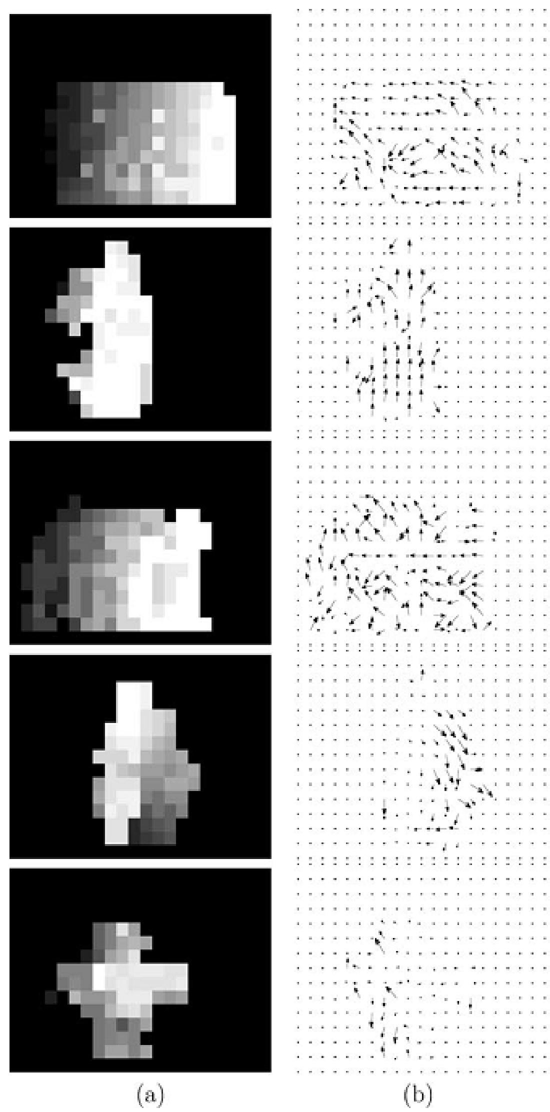

Figure 8: Video attracting with different image processing intervals.

is the regressor and p = $[x,y]^T$ is the vector representing the position of pixel in the image plane and v(p) is the motion vector at location p (here the spatial location of motion vectors are assigned to the center of the corresponding macroblock).Projected 1D feature. Here horizontal and vertical components of the motion vectors are considered separately. The histogram values are quantized into five bins to cover the entire range in the following intervals: [MIN,-8),[-8,-3),[-3,3],(3,8],(8,MAX].The bins are chosen in such a way so as to capture the low, medium and higher speeds. The distance between the centers of low and medium speeds is set apart by 5 peels approximately. The motion vector magnitude exceeding 8 are considered as high speed. 2D polar feature. The angular direction and magnitude of motion vectors are considered together to quantize the polar plane into histogram bins. Each bin is defined by the angular range as well as the magnitude (radius) range. Here angular range is quantized into four intervals of length π/2 from-π to +π: The magnitude range is quantized into the following intervals: (0,5](5,10],(10,MAX)This leads to a feature vector of 12 dimension. Table 1summarizes the features used in our experiment.

VI. CLASSIFICATION RESULTS AND DISCUSSION

The following seven actions were considered for recognition: walk, run, jump, bend up, bend down, twist right and twist left. For collecting the database, each subject was asked to perform each action many times in front of the fixed camera inside the laboratory. The actions were captured at the angle at which the camera could view the motion with minimal occlusion. The subjects are given freedom to perform the actions at their own pace at any distance in front of the camera. We have used four types of classifiers for recognizing the action, namely Normalized KNN, Bayesian, Neural net-work: Multi-Layer feed forward Perceptron (MLP) and Support Vector Machines (SVM). As in the previous paper, seven actions (walk, run, jump, bend down, bend up, twist left and twist right) were considered for recognition. 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.

6.1. K-Nearest Neighbors Classifier

The KNN algorithm simply selectsk-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 higher dimensional spaces these distributions may be often erroneous. Even in situations where second order

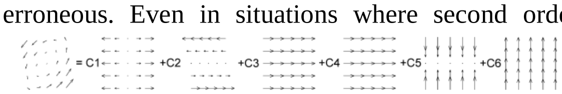

Fig. 7. Affine flow model expressed as a linear sum of basis flows.

statistics

cannot 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 aforemen-tioned features.

IJDCST @November Issue- V-1, I-7, SW-40
ISSN-2320-7884 (Online)
ISSN-2321-0257 (Print)

6.2. Bayes Classifier

The second classifier used is Bayes—a parametric classifier that assumes normal distribution for class (w) conditional probability of feature vector x, $p(x|w_i)$ Though Bayes classifier is optimal, the performance degrades if the models used are erroneous. Since erroneous models degrade classification performance, starting with the first feature, we have added subsequent features only if they improved the classification performance.

Table 3shows the performance

Table 3
Bayes classification result

Input class	Result							
	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 4
Neural net classification result

Input class	Result							
	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**

of Bayes classifier with only four selected features out of total 32 features. The selected feature numbers are 1, 3, 6 and 9, i.e. three from the affine feature and one from MHI centroid-based feature. With all the features Bayes classifier gives a performance of 92.1% (seeTable 8). 6.3. Neural network classifier MLP is a supervised neural network. It can have multiple inputs and outputs and multiple hidden layers with arbitrary number of neurons (nodes). In our network, the commonly

used sigmoid function is used as the activation function for nodes in the hidden layer. The MLP utilizes the back-propagation (BP) algorithm for determining suitable weights and biases of the network using supervised training [14]. Table 4shows the classification results obtained with an MPL trained with two hidden layers with 15 neurons in each layer using all the features.

6.3. SVM Classifier

SVM[34]are powerful tools for data classification. SVM is based on the idea of hyper plane classifier, that achieves classification by a separating surface (linear or nonlinear) in the input space of the data set. SVMs are modeled as optimization problems with quadratic objective functions and linear constraints.

Table 5
Linear SVM classifier result

Input class	Result							
	Walk	Run	Jump	BD	BU	TWL	TWR	Error
Walk	**5**	0	0	0	0	0	0	0
Run	0	**6**	1	0	0	0	0	1
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	1	0	0	1	0	**2**

Tables 5 and 6show the classification results of SVM classifier with linear kernel and radial-based kernel. Comparing the results of the classifiers, the results obtained by KNN, Neural Net and SVM (with RBF-kernel) show excellent performance. Bayes classifier recognizes most of the actions, but is relatively less successful in discriminating between ‘walk’ and ‘run’ actions. This could be due to the parameterization of the underlying feature distribution. Moreover the Bayes result is obtained only with the selected four features, whereas the other classifiers use all features.

ISSN-2320-7884 (Online)
ISSN-2321-0257 (Print)

Table 7 summarizes the recognition results for various classifiers. Consistency in results obtained using various classifiers, proves the credibility of features used. The uniform

Results obtained using SVM, KNN and Neural Nets points to the fact that the system has very few outliers. It must be noted that the performance of the Bayesian classifier is only marginally lesser, in spite of drastically reducing the number of features to only four, compared to 32 used in the other cases. The deterioration in the performance of the Bayesian classifier on using all features may be attributed to 'curse of dimensionality'.

Table 6
Classification result using non-linear SVM classifier with a radial based kernel

Input class	Result							
	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 7
Comparison of various classifiers

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

The remaining sections of section(7,8,9&10) concluded .

VII. CONCLUSION

Event detection and human action interaction have gained more interest of late among video processing. For implementing course motion history image on compressed video examples of the motion trajectory include tracking results from video trackers, sign language data measurements gathered from wired glove interfaces fitted with sensors, Global Positioning System (GPS) coordinates of satellite phones, cars using Car Navigation Systems (CNS), animal mobility experiments, etc. This spatio-

ISSN-2320-7884 (Online)
ISSN-2321-0257 (Print)

temporal data embodies semantically rich information about the behavior of the object of interest, the action performed and the interaction among groups of objects. In this paper we propose to extend our existing work video buffering, building coarse Motion History Image (MHI) and Motion Flow History (MFH) from the compressed video and extract features from these static motion history information for characterizing human actions. These MHI and MFH represent 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 MHI and MFH for representing the actions.

VIII. REFERENCES

[1] J.K. Aggarwal, Q. CAI, Human motion analysis: a review, Computer Vision and Image Understanding 73 (3) (1999) 428 – 440.

[2] D. Ayers, M. Shah, Monitoring human behavior from video taken in an office environment, Image and Vision Computing 19 (12-1) (2001) 833 – 846.

[3] R. Venkatesh Babu, B. Anantharaman, K.R. Ramakrishnan, S.H. Srinivasan, Compressed domain action classification using HMM, Pattern Recognition Letters 23 (10) (2002) 1203 – 1213.

[4] R. Venkatesh Babu, K.R. Ramakrishnan, Compressed Domain Human Motion Recognition Using Motion History Information, IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), April, vol. 3, 2003, pp. 41 – 44.

[5] A.F. Bobick, J.W. Davis, The recognition of human movement using temporal templates, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (3) (2001) 257 – 267.

[6] A.F. Bobick, A.D. Wilson, A state-based approach to the representation and recognition of gesture, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (12) (1997) 1325 – 1337.

[7] K. Boehm, W. Broll, M. Sokolewicz, Dynamic gesture recognition using neural networks; a fundament for advanced interaction construction, Proceedings of the SPIE—The International Society for Optical Engineering 2177 (1994) 336 – 346.

[8] C. Bregler, Learning and recognizing human dynamics in video sequences, Proceedings of the IEEE CVPR (1997) 568 – 574.

[9] T.J. Darrell, A.P. Pentland, Space – time gestures, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (1993) 335 – 340.

[10] J. Davis, Hierarchical motion history images for recognizing human motion, IEEE Workshop on Detection and Recognition of Events in Video (2001) 39 – 46.

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

[12] D.J. Fleet, M.J. Black, Y. Yacoob, A. Jepson, Design and use of linear models for image motion analysis, International Journal of Computer Vision 36 (3) (2000) 171 – 193.

[13] D.M. Gavrila, The visual analysis of human movement: a survey, Computer Vision and Image Understanding 73 (1) (1999) 82 – 98.

[14] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall, 1998.

[15] B.K.P. Horn, B.G. Schunck, Determining optical flow, Artificial Intelligence 17 (1981) 185 – 203.

[16] M.K. Hu, Visual pattern recognition by moment invariants, IRE Transactions on Information Theory 8 (2) (1962) 179 – 187.

[17] Y.A. Ivanov, A.F. Bobick, Recognition of visual activities and interactions by stochastic parsing, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 852 – 872.

[18] T. Kohonen, The self-organizing map, Proceedings of the IEEE 78 (9) (1990) 1464 – 1480.

[19] M. Su, H. Huang, C. Lin, C. Huang, C. Lin, Application of neural networks in spatio temporal hand gesture recognition, Proceedings of the IEEE World Congress on Computational Intelligence, 1998.

[20] A. Madabhushi, J. K. Aggarwal, A Bayesian Approach to Human Activity Recognition, Second IEEE Workshop on Visual Surveil-lance, 25 – 30, 1999.

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

[22] C.W. Ng, S. Ranganath, Real-time gesture recognition system and application, Image and Vision Computing 20 (13 – 14) (2002) 993 – 1007.

[23] N.M. Oliver, B. Rosario, A. Pentland, A bayesian computer vision system for modeling human interactions, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 831 – 843. Table 8 Performance of feature sets (%) Input features Classifier results (%) Name Numbers KNN ðk¼3Þ Bayes NNet SVM (RBF) Affine (6) 1 – 6 78.4 92.2 94.1 86.3, Proj. profile (2) 7 – 8 64.7 58.8 70.6 11.8, Centroid (2) 9 – 10 68.6 68.6 72.6 66.7, 1D proj. (10) 11 – 20 92.2 84.3 86.3 90.2, 2D polar (12) 21 – 32 86.3 86.3 90.2 86.3, Overall (32) 1 – 32 98.0 92.1 98.0 98.0, R. Venkatesh Babu, K.R. Ramakrishnan / Image and Vision Computing 22 (2004) 597–607 606.

[24] R. Polana, R. Nelson, Low level recognition of human motion, Workshop on Non-Rigid Motion (1994) 77 – 82.

[25] A. Psarrou, S. Gong, M. Walter, Recognition of human gestures and behaviour based on motion trajectories, Image and Vision Computing 20 (5 – 6) (2002) 349 – 358.

[26] J. Rittscher, A. Blake, S.J. Roberts, Towards the automatic analysis of complex human body motions, Image and Vision Computing 20 (12) (2002) 905 – 916.

[27] R. Rosales, Recognition of human action based on moment based features, Technical Report BU 98-020, Boston University, Computer Science, 1998.

[28] M. Shah, R. Jain, Motion Based Recognition, Kluwer, 1997.

[29] Standard MPEG1: ISO/IEC 11172, Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s, 1996.

[30] T. Starner, A. Pentland, Real-time american sign language recognition from video using hidden Markov models, MIT Media Lab, TR-375, 1995.

[31] T. Starner, A.P. Pentland, Real-time american sign language recognition using desk and wearable computer based video, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (12) (1998) 1371 – 1375.

[32] X. Sun, B.S. Manjunath, Panoramic capturing and recognition of human activity, IEEE International Conference on Image Processing 2 (2002) 813 – 816.

[33] M. Umeda, Recognition of multi-font printed chinese character, Proceedings of sixth Computer Vision and Pattern Recognition (1982) 793 – 796.

[34] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 1995.

[35] J. Yamato, J. Ohya, K. Ishii, Recognizing human action in time-sequential images using hidden Markov model, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (1992) 379 – 385.