International Journal of Advanced Computer Technology (IJACT)

ISSN:2319-7900

Categorization of Human Action

M. G. Pallewar, M.E-II [Digital Systems], Prof.M. M. Sardeshmukh, E & TC Department, SAE Kondhwa Pune, India.

Abstract

Human action analysis is a recent topic of interest among the computer vision and video processing community. Research in this area is motivated by its wide range of applications such as surveillance and monitoring systems. We consider the problem of recognizing human actions from videos or images. Our learning objective is designed to directly exploit the pose information for action recognition. Our experimental results demonstrate that by inferring the latent poses, we can improve the final action recognition results. In this project we describe a system for recognition of various human actions from compressed video based on motion history information. 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 spatiotemporal and motion vector 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, and SVM classifiers for recognizing a set of seven human actions.

The main purpose of this project is development of Human Classification System. This system can be utilized for different smart visual surveillance systems. The system can compare the available algorithms and find the optimal solution. If possible go for solution that which algorithm provide better results for particular action.

Keywords- Motion Flow History (MFH), Motion History Image (MHI), SVM, KNN, ANN, Video surveillance.

Introduction

It is of great practical and scientific interests to understand expressed body actions, especially those of the human body. In computer vision, one interesting problem is to represent the different types of human actions with effective models. In this project, we focus on human action classification with available image frames. In the project we have used the term "action", to describe the units, into which human behavior shall be classified. The reason for this terminology is that there is another unresolved issue looming behind our question, namely the definition of what constitutes an action. Obviously, the amount of information which needs to be accumulated, and also the number of relevant classes for a given application, both depend on the complexity of the action (e.g., recognizing a high-jump takes longer than separately recognizing the three components running, jumping, and falling on the back).

We assume that a relatively small set of basic actions, such as walking or waving, form the set of possible action labels,

and that the labels are relatively unambiguous (the most subtle difference we take into account is between running and jogging).

Based on the recent works in human motion categorization [2, 8, 12, 14], we make two key observations that will in turn influence the design of our model. The first observation is based on the usage of different feature descriptors to represent human body and/or human motion. The second observation deals with the choice of the category model that uses such features for corresponding classification. Using good features to describe pose and motion has been widely researched in the past few years.

The primary goal of this work is to classify actions from images. In still images, the information about the action label of an image mainly comes from the pose, i.e. the configuration of body parts, of the person in the image. However, not all body parts are equally important for differentiating various actions. The configurations of torso, head and legs are quite similar for both walking and playing golf. The main difference for these two actions in terms of the pose is the configuration of the arms. A standard pose estimator tries to find the correct locations of all the body parts. The novelty of our work is that we do not need to correctly infer complete pose configuration in order to do action recognition. In the example of "walking" versus "playing golf", as long as we get correct locations of the arms, we can correctly recognize the action, even if the locations of other body parts are incorrect.

Human action classification has been receiving increasing attentions from researchers in computer vision community. The aim of human activity categorization is to recognize human actions from images so that the system could understand the scene so as to make further classification or semantic description of the actions becomes feasible. The results can be applied to many applications such as visual surveillance, human-computer interfaces, content based video retrieval etc. Human action classification is a challenging research area because the dynamic human body actions have unlimited underlying representations.

Model representation and learning are critical for the ultimate success of any recognition framework. In human mo-

ISSN:2319-7900

tion recognition, most models are divided into either discriminative models or generative models. For example, based on the spatial-temporal cuboids, Dollar et al. [6] applied an SVM classifier to learn the differences among videos containing different human motions. Ramanan et al. [13] recently proposed a Conditional Random Field model to estimate human poses. While discriminative frameworks are often very successful in the classification results, they suffer either the laborious training problem or a lack of true understanding of the videos or images. In the CRF framework, one needs to train the model by labeling by hand each part of the human body. And in the SVM framework, the model is not able to "describe" the actual motion of the person.

Review Of Literature

In the literature, the term "Human action" in video sequences can be seen as silhouettes of a moving torso and protruding limbs undergoing articulated motion. Human action recognition is the process of labeling image sequences with action labels.

Classifying human actions from sequence of image data enables applications such as understanding action, semantic retrieval. Depending on the application, a classification system may be constructed in different ways.

Generally speaking, there are three popular types of features: static features based on edges and limb shapes [5, 9, 13]; dynamic features based on optical flows [5, 7, 16], and spatial temporal features that characterizes a space-time volume of the data [2, 4, 6, 11]. Some researchers, therefore, have proposed several algorithms based on probabilistic graphical model frameworks in action categorization/recognition. Song et al. [18] and Fanti et al. [8] represent the human action model as a triangulated graph. Boiman and Irani [3] recently propose to extract ensemble of local video patches to localize irregular action behavior in videos. Dense sampling of the patches is necessary in their approach and therefore the algorithm is very time-consuming. It is not suitable for action recognition purpose due to the large amount of video data commonly presented in these settings. For structured objects such as human bodies, it is important to model the mutual geometric relationship among different parts. Constellation models offer such a solution [8, 19]. Unfortunately due to the computational complexity of the model, previous works can only use a very small number of features (typically 4 to 6) or approximate the connections by triangulation [8, 18]. Another approach is to lose all the geometric information and consider "bag of words" models. They have proven to be highly efficient and effective in classifying objects [10, 17] and human motion [6, 12]. We propose here a method to exploit both the geometric power of the constellation model as well as the richness of the "bag of words" model. We recognize the computational limit of having a very small number of fully connected parts in the constellation model. But instead of applying it directly onto the image level features, we attach a "bag of words" model to each part of the constellation model. Our model is partly inspired by a hierarchical model proposed by Bouchard and Triggs in [4]. In their framework, they also use the idea of attaching large number of feature at the image level to a handful of intermediate level parts.

Block Diagram for proposed system:

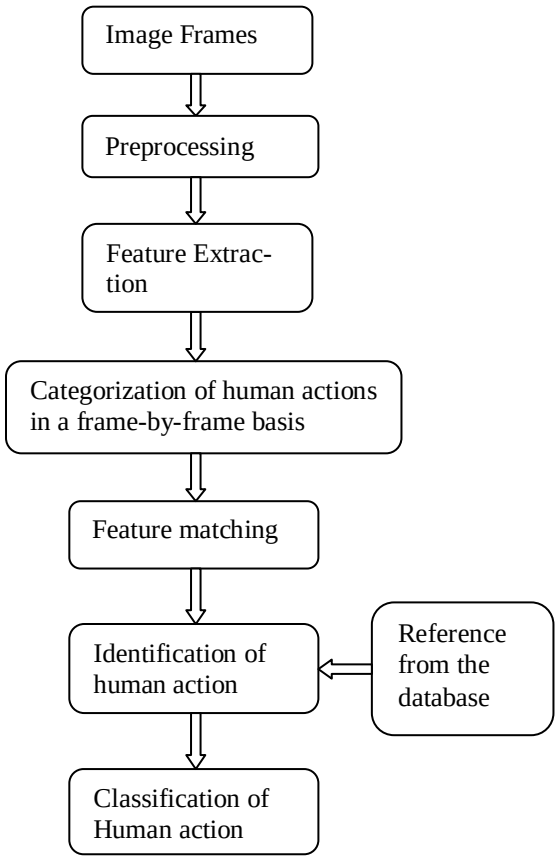


Figure 1: Block diagram of proposed system

Methodology

A. Data Base Preparation:

We will prepare a database of 100 images for different actions. The data base includes 10 distinct actions, for each action we are planning to include at least 10 different images with different scenarios. The images included can be for the actions such as walking, running, waving, jumping, bending etc.The following actions considered for recognition from the database: walk, run, jump, bend up, bend down, wave. For collecting the database, each subject was asked to perform each action many times in front of the fixed camera

ISSN:2319-7900

inside the laboratory. The actions were captured at the angle at which the corresponding to different types of actions and scenarios. 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.

B. Preprocessing:

The preprocessing techniques are utilized to remove noise from the image for the better results.
In the preprocessing steps, we extract foreground, eliminate shadow, and then apply filtering. We then define the action boundary from the foreground image sequence.

Background modeling
We use background subtraction to extract the foreground since the background is relatively static for all image sequences. We adopt a simple background modeling technique such as multiple Gaussian background modeling, for foreground extraction.For each subséquent frame, pt = [pR(t), pG(t), pB(t)], we assume independence among different color channels. Several background images are accumulated and we extract the mean, standard deviation, and variance of the background images.
Let μR, μG, μB be the mean values, and σR, σG, and σB be the standard deviation of the background images which are computed over N frames, then, we extract the foreground according to

$$p(xt) = \begin{cases} 1 & \text{if } |pR(t) - \mu R| \geq 2\sigma R \text{ or } |pG(t) - \mu G| \geq 2\sigma G \text{ or } |pB(t) - \mu B| \geq 2\sigma B \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Shadow elimination
After background subtraction, there still exits some noises in the foreground, such as motion shadow. Therefore, the shadow elimination method should be adopted. In this method, for a given pixel, the expected background value Et = [μR, μG, μB] is computed from N training frames representing the static background.

Filtering
After the above shadow elimination step, there may exist some small regions and noise.
For further preprocessing, several morphological operations such as erosion, dilation, and connected component analysis should be adopted. Finally, the resulting foreground image is obtained by median filtering. The neighboring window size of the median filtering is 5×5.

Categorization of Human Action
We define the action boundary as the action region in the image sequence where the movements of the person occur or the person exists. The action boundary depends on 1) anthropometry of human body, 2) distance between the video sensor and person performing action, and (3) type of action.

C. Feature Extraction:

The feature extraction is a subjective procedure, as numbers of actions are included in the data base for which we require respective technique for the feature extraction.
We will be referring two algorithms for the feature extraction.

Figure 2: Action database: examples of some actions.

In this technique we can describe a system for recognition of various human actions from compressed video based on motion history information. 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. 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.
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

ISSN:2319-7900

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. 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.

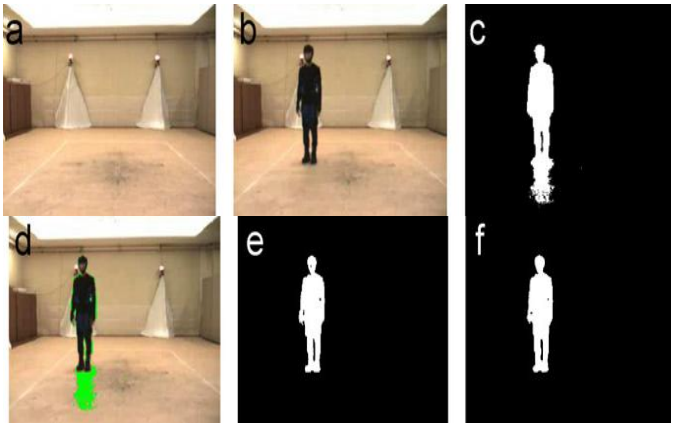


Figure 3: Foreground extraction procedures. (a) Background image. (b) Current image. (c) Extracted foreground image with shadow. (d) Detected shadow pixels (green color). (e) Foreground image after shadow removal. (f) Foreground image after morphological and filter operations.

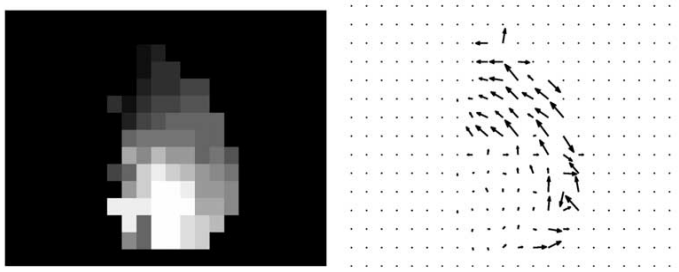


Figure 4: (a) Key-frames of bend-down sequence and corresponding coarse (b) MHI (c) MFH

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 objects 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).

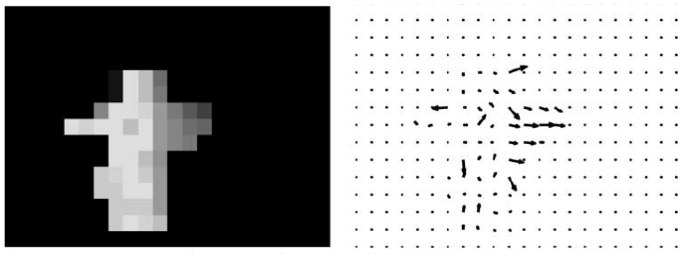


Figure 5: (a) Key-frames of twist left sequence and corresponding coarse (b) MHI (c) MFH

The MFH is constructed from non-zero P-frame motion vectors according to the following:

$$MFH_d(k,l) = \begin{cases} m^{kl}(\tau) & \text{if } E(m^{kl}(\tau)) < T_r \end{cases} \quad (2)$$

ISSN:2319-7900

$$= M(m_d^{kl}(\tau)) \quad \text{otherwise}$$

where,

$$E(m_d^{kl}(\tau)) = \| m_d^{kl}(\tau) - med(m_d^{kl}(\tau) \ldots m_d^{kl}(\tau - \alpha)) \|_2$$

and

$$M(m_d^{kl}(\tau) = med(m_d^{kl}(\tau) \ldots m_d^{kl}(\tau - \alpha))$$

Here med refers to median filter, $m^{kl}_d(\tau)$ can be horizontal (m_x) component or vertical (m_y) component of motion vector located at *k*th row and *l*th 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 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 α value beyond 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. The purpose of this threshold T_r 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 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 T_r is set at 4 for generating MFH. In other words, this threshold T_r 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.

$$MHI(k, l) = \tau \quad \text{if } (|m_x^{kl}(\tau)| + |m_y^{kl}(\tau)|) \neq 0 \qquad (3)$$
$$= 0 \quad \text{otherwise}$$

Figs. 7 and 8 show the key frames of the bend-down and twist left actions and the corresponding coarse MHI and MFH. The coarse MHI and MFH of other actions are shown in Fig. 9. 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 loss of a part of the motion information. However, it might be representative enough for the considered human actions.

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.

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

$$Fpp = \frac{\sum_{i=1}^{h_c} P_h[i]}{\sum_{i=h_c+1}^{M} P_h[i]} \quad \frac{\sum_{i=1}^{v_c} P_v[i]}{\sum_{i=v_c+1}^{N} P_v[i]} \qquad (4)$$

where h_c and v_c are the horizontal and vertical centroids of MEI. The above feature (Fpp) 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

$$Fc = [MHIxc - MEIxc \quad MHIyc - MEIyc]$$

The centroid of MHI differs from the centroid of MEI because it is computed using the gray-level time stamp values as weights in the summation. The above vector indicates the approximate direction of the movement of centroid for the corresponding action.

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 in Fig. 7. The affine parameters are estimated by standard linear regression techniques.

The regression is applied separately on each motion vector component since the x affine parameter depends only on ho-

ISSN:2319-7900

rizontal component of motion vector and y parameter depends only on the vertical component of motion vector.
Let c ¼ ½c1…c6_ be the 6D affine parameter vector. Then the linear least squares estimate of c is given by:

$$cT = [\sum \pi(p)T\ \pi(p)]\text{-}1\ .\ \sum \pi T(p)v(p) \qquad (5)$$

where

$$\pi(p) = \begin{matrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{matrix}$$

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 center of low and medium speeds are set apart by 5 pels 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 dimensions. Table 1 summarizes the features used in our experiment.

D. Feature Matching:

In case of feature matching we will be referring two algorithms.
The combination of algorithm for feature extraction and feature matching will be done to get the result of classification. We can use different types of classifiers for recognizing the action, namely

1. Normalized KNN,
2. ANN
3. Support Vector Machines (SVM)

Support Vector Machines (SVM)

Support Vector Machines (SVMs) are state-of-the-art large margin classifiers which have recently gained popularity within visual pattern recognition ([13, 14] and many others). In this section we provide a brief review of the theory behind this type of algorithm; for more details we refer the reader to [5, 12]. Consider the problem of separating the set of training data (x1, y1), (x2, y2) . . . (xm, ym) into two classes, where xi Є RN is a feature vector and yi Є {−1, +1} its class label. If we assume that the two classes can be separated by a hyper plane w · x + b = 0 in some space H, and that we have no prior knowledge about the data distribution, then the optimal hyper plane is the one which maximizes the margin [12]. The optimal values for w and b can be found by solving a constrained minimization problem, using Lagrange multipliers αi (i = 1 . . . m).

$$F(x) = \operatorname{sgn} \sum_{I=1}^{m} \alpha i\ yi\ K\ (xi, x) + b) \qquad (6)$$

where αi and b are found by using an SVC learning algorithm[12].Those xi with nonzero αi are the “support vectors”. For K(x, y) = x· y, this corresponds to constructing an optimal separating hyper plane in the input space _N.

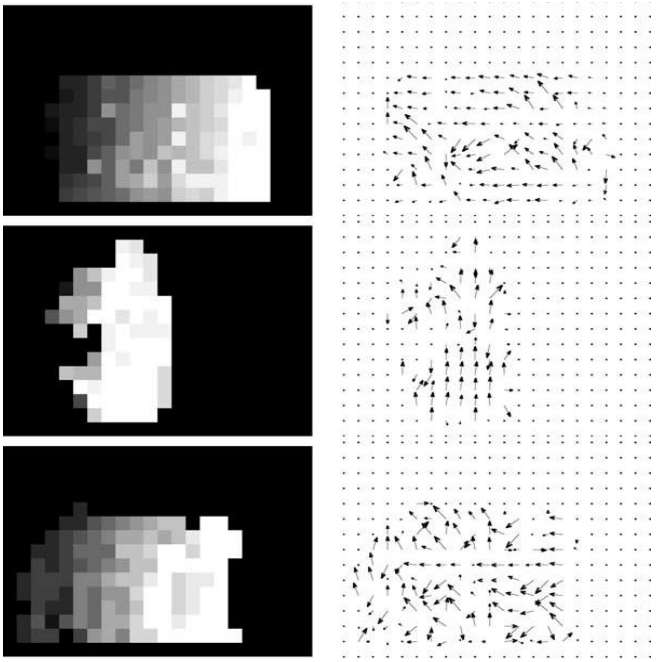
Experiments

SVM classification combined with motion descriptors in terms of local features (LF) and feature histograms (HistLF) define two novel methods for motion recognition. In this section we evaluate both methods on the problem of recognizing human actions and compare the performance to other approaches using alternative techniques for representation and/or classification.

Classification results and discussion

The following six actions were considered for recognition: walk1, walk2, walk3, run1, run2, run3, jump1, jump2, jump3. 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 network:

ISSN:2319-7900



(a) (b)

Fig. 6. (a) The coarse MHI and the corresponding (b) MFH of walk, jump, run action.

Multi-Layer feed forward Perceptron (MLP) and Support Vector Machines (SVM). 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 three 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.

1. K-nearest neighbors classifier

The KNN algorithm simply selects 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 higher dimensional spaces these distributions may be often erroneous. Even in situations where second order 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 1 shows the classification results of KNN classifier with all aforementioned features.

2. 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 2 shows the classification results obtained with an MPL trained with two hidden layers with 15 neurons in each layer using all the features.

3. SVM classifier

SVM [34] are powerful tools for data classification. SVM is based on the idea of hyperplane 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 1: Confusion Matrix for KNN

----------KNN----------

	bend	pjmp	jump1	jump2	jump3	run1	run2	run3	walk1	walk2	walk3	wave1	wave2
bend	25	0	1	0	0	0	0	0	0	0	0	0	1
pjump	1	23	0	0	0	0	0	0	0	0	1	1	1
jump1	0	2	22	0	2	0	0	0	0	1	0	0	0
jump2	0	3	0	14	0	6	0	2	0	0	1	0	1
jump3	0	0	0	0	23	0	0	1	0	0	2	1	0
run1	1	2	0	1	0	19	2	0	1	1	0	0	0
run2	0	0	0	0	1	1	16	0	7	2	0	0	0
run3	1	0	0	1	2	3	0	15	0	3	2	0	0
walk1	0	0	1	0	0	2	3	0	19	0	0	0	2
walk2	0	0	1	0	0	0	0	2	1	23	0	0	0
walk3	0	0	0	0	0	6	1	0	3	1	16	0	0
wave1	0	0	0	0	1	0	0	0	0	0	0	26	0
wave2	0	1	0	0	0	0	0	0	0	0	0	2	24

Table 2: Confusion Matrix for ANN

----------ANN----------

	bend	pjmp	jump1	jump2	jump3	run1	run2	run3	walk1	walk2	walk3	wave1	wave2
bend	27	0	0	0	0	0	0	0	0	0	0	0	0
pjump	0	26	1	0	0	0	0	0	0	0	0	0	0
jump1	0	1	26	0	0	0	0	0	0	0	0	0	0
jump2	0	0	0	27	0	0	0	0	0	0	0	0	0
jump3	0	0	0	4	17	2	1	1	2	0	0	0	0
run1	0	0	0	0	0	23	2	2	0	0	0	0	0
run2	0	0	0	0	0	1	21	5	0	0	0	0	0
run3	0	0	0	0	0	0	0	26	0	1	0	0	0
walk1	0	0	0	0	0	0	0	7	20	0	0	0	0

ISSN:2319-7900

walk2	0	0	1	0	0	0	0	1	2	24	0	0	0
walk3	0	0	0	0	0	0	0	0	2	5	19	1	0
wave1	0	0	0	0	0	0	0	0	0	0	1	24	2
wave2	0	0	0	0	0	0	0	0	0	0	0	3	24

Table 3: Confusion Matrix for SVM

----------SVM----------

	bend	pjmp	jump1	jump2	jump3	run1	run2	run3	walk1	walk2	walk3	wave1	wave2
bend	27	0	0	0	0	0	0	0	0	0	0	0	0
pjump	0	27	0	0	0	0	0	0	0	0	0	0	0
jump1	0	0	27	0	0	0	0	0	0	0	0	0	0
jump2	0	0	0	27	0	0	0	0	0	0	0	0	0
jump3	0	0	0	0	27	0	0	0	0	0	0	0	0
run1	0	0	0	0	0	27	0	0	0	0	0	0	0
run2	0	0	0	0	0	0	27	0	0	0	0	0	0
run3	0	0	0	0	0	0	0	27	0	0	0	0	0
walk1	0	0	0	0	0	0	0	0	27	0	0	0	0
walk2	0	0	0	0	0	0	0	0	0	27	0	0	0
walk3	0	0	0	0	0	0	0	0	0	0	27	0	0
wave1	0	0	0	0	0	0	0	0	0	0	0	27	0
wave2	0	0	0	0	0	0	0	0	0	0	0	0	27

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.

Performance analysis of features

In this section, we present the performance of each feature set for various classifiers. Fig. 8 shows the recognition performance of each feature with test and training samples using the nearest neighbor criterion. The individual performance of the first 10 or 11 features is good on both, the test as well as the training samples.

Other features perform slightly better with test samples compared to training samples. Here, the considered test subjects are different from the training ones and the subjects were given freedom to perform the action at their own pace at any location in front of the camera. So the features show invariance to translation, scale and speed of action.

Conclusion

In this paper, we have proposed a method for constructing coarse MHI and MFH from compressed MPEG video with minimal decoding. Various useful features are extracted from the above mentioned motion representations for human action recognition. We have shown the recognition results for three classification paradigms. The performance of these features is analyzed and compared. Though the test instances are from entirely different subjects other than those used for training the classifiers, the results show excellent recognition accuracy. The KNN, Neural network (MLP) and SVM (RBF kernel) classifiers give the best classification accuracy of 98% and 1D projected and 2D polar features show consistent performance with all the classifiers. Since the data is handled at macroblock level, the computational cost is extremely less compared to the pixel domain processing.

References

[1] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts", IEEE Trans. Pattern Anal. Mach. Intell., 24(4):509–522, 2002.

[2] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes", In ICCV, 2005.

[3] O. Boiman and M. Irani, "Detecting irregularities in images and in video", In ICCV, 2005.

[4] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance", In ECCV, 2006.

[5] P. Doll´ar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features", In VS-PETS, 2005.

[6] A. A. Efros, A. C. Berg, G. Mori, and J. Malik, "Recognizing action at a distance", In ICCV, 2003.

[7] C. Fanti, L. Zelnik-Manor, and P. Perona, "Hybrid models for human motion recognition", In CVPR, 2005.

[9] X. Feng and P. Perona, "Human action recognition by sequence of movelet codewords", In 3DPVT, pages 717–723, 2002.

[10] K. Grauman and T. Darrell, "Unsupervised learning of categories from sets of partially matching image features", In CVPR, 2006.

[11] I. Laptev and T. Lindeberg, "Velocity adaptation of spacetime interest points", In CVPR, 2004.

[12] J. C. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words", In BMVC, 2006.

[13] D. Ramanan, "Learning to parse images of articulated bodies", In Advances in Neural Information Processing Systems, 2006.

[14] D. Ramanan and D. A. Forsyth, "Automatic annotation of everyday movements", In Advances in Neural Information Processing Systems ,2004.

[15] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach", In ICPR, 2004.

[16] H. Sidenbladh and M. J. Black, "Learning the statistics of people in images and video", International Journal of Computer Vision, 54(1-3):183–209, 2003.

ISSN:2319-7900

[17] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering objects and their localization in images", In ICCV, 2005.

[18] R. V. Babu, K. R. Ramakrishnan, "Recognition of human actions using motion history information extracted from the compressed video", Image and Vision Computing, Vol. 22, No. 8. 597-607 2004.

[19] Mohiuddin Ahmad, Seong-Whan Lee, "Human action recognition using shape and CLG-motion flow from multi-view image sequences", Department of Computer Science and Engineering, Korea University, Anam-dong, Seongbuk-ku, Seoul 136-713, December 2007.

[20] Yang Wang and Greg Mori, "Hidden Part Models for Human Action Recognition: Probabilistic vs. Max-Margin," in IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI), 33(7) pp.1310-1323 2011.