Towards Fast, View-Invariant Human Action Recognition

Srikanth Cherla Kaustubh Kulkarni Amit Kale V. Ramasubramanian
Siemens Corporate Technology
SISL - Bangalore, India
{srikanth.cherla, kulkarni.kaustubh, kale.amit, v.ramasubramanian}@siemens.com

Abstract

In this paper, we propose a fast method to recognize human actions which accounts for intra-class variability in the way an action is performed. We propose the use of a low dimensional feature vector which consists of (a) the projections of the width profile of the actor on to an "action basis" and (b) simple spatio-temporal features. The action basis is built using eigenanalysis of walking sequences of different people. Given the limited amount of training data, Dynamic Time Warping (DTW) is used to perform recognition. We propose the use of the average-template with multiple features, first used in speech recognition, to better capture the intra-class variations for each action. We demonstrate the efficacy of this algorithm using our low dimensional feature to robustly recognize human actions. Furthermore, we show that view-invariant recognition can be performed by using a simple data fusion of two orthogonal views. For the actions that are still confusable, a temporal discriminative weighting scheme is used to distinguish between them. The effectiveness of our method is demonstrated by conducting experiments on the multi-view IXMAS dataset of persons performing various actions.

1. Introduction

Human action recognition is an active area of research in computer vision. This has been motivated by the need for fast video indexing applications and robust and reliable automated video surveillance systems. The successful deployment of such video surveillance systems can be of great use not only in identifying abnormal activity on roads, in restricted areas, high-security zones, etc., but also in problems like assisted living. Recognizing human actions from a video is a challenging task due to several reasons. Firstly, there is a problem of identifying the action independent of viewing direction. Secondly, there are differences in the way an action is performed by different people. The system must be capable of incorporating these intra-class variations. Thirdly, certain actions such as waving one's hand, scratching one's head and so on do not usually last for a fixed length of time. The system should be able to account for this temporal uncertainty. Finally all of this has to be accomplished fast so that the system runs in near real-time.

Several researchers have addressed the problem of human action recognition. In recent work, Weinland et. al. [17] address the problem of view invariant action recognition. Their method is based on using a 3D occupancy grid as a feature to learn a set of exemplars and a HMM. For recognition, these 3D exemplars are used to produce 2D image information for matching with the observations. The work of Lv and Nevatia [10] is similar in spirit to this. One of the difficulties in using these methods is that synthesizing the high dimensional 2D images from the 3D exemplars and comparing them with the observed 2D image can incur a high computational cost.

One of the problems in building a robust human action recognition system comes from the fact that the self occluding articulated nature of the human body results in the non-rigid behavior in the 2D projections. It is known from common experience that some views are better suited for recognizing certain activities. For example, hand waving can be better recognized from a frontal view while pointing, kicking etc. are better recognized from a side view. This suggests that accurate recognition of human actions requires a minimum of two orthogonally placed synchronized cameras and using the appropriate camera to recognize a particular activity. Even in a single camera view, several problems remain in the quest of achieving a fast and reliable human action recognition system. An important consideration in performing action recognition is the choice of features. The feature chosen must be capable of capturing the unique aspects of an action performed by different actors. Furthermore its dimensionality also determines how fast recognition can be performed. In interests of brevity, we restrict our discussion here to methods that extract features from video directly as opposed to those relying on joint angles [12, 13]. Veeraraghavan et al.[16, 15] propose the use of shape features and a Procrustes distance as a distance metric. One of the problems of this approach is that the articu-

lated, non-rigid nature of human actions can make the registration of the point sets pretty challenging. Furthermore, the high dimensionality of the feature vector makes the distance computation time consuming. Gorelick et al. [2, 5] proposed a method in which the Poisson equation is used to obtain space-time shape features from the silhouette, the Hessian of which provides information on the shape and orientation of different parts of the human body. This involves computation of the distance transform which can be time consuming. The weighted moments of these local features are used to generate a global 280-dimensional feature vector that is classified as a particular action by comparing it with the trained action using the nearest neighbor approach with Euclidean distance after normalization. In [11], localized spatio-temporal salient regions are used as features. The linear time-warped sequence of these features is used to recognize actions using RVM classifier.

Given a feature vector sequence, the next problem is to decide what temporal matching method is employed. Two predominantly used paradigms for performing temporal matching are Hidden Markov Models (HMMs) and template matching using Dynamic Time Warping (DTW). The use of template matching using DTW has the advantage that it works well even when the training data is limited. One of challenges in using DTW is to accommodate intra-class variations in the activity being performed. Veeraraghavan et al [16] propose the use of constraints on the warping region around the nominal activity trajectory in order to model intra-class variations. Such an enhanced warping region accounts for intra-class temporal and feature variability and allows for an improved matching of two activity patterns belonging to the same class. However, such an enlarged warping region suffers the risk of good matching of the template of a particular class with highly variable patterns of another class, particularly if the two classes are intrinsically confusable. Therefore, the question of dealing with improved class separability or increasing the inter-class discriminability is not addressed by such a scheme.

In this paper, we address several of the above mentioned issues. Given the binarized silhouette of a person, we compute the width of the outer contour of the silhouette. We then project this width vector onto a five-dimensional *action basis* which can be learned easily from the training data. In addition we add simple spatio-temporal features such as variance and centroid displacement of the silhouette, yielding a nine-dimensional feature. We show how this feature vector serves to capture the unique characteristics of each action, while retaining robustness to noise. When using DTW for matching, an important issue is to get the template representation for each action in the training set. The computational complexity of the recognition process is highly dependent on these representations. Intuitively, the average-template which is computed using non-linear warping of all training templates for a specific action, provides the least computationally expensive representation of the action. This amounts to an assumption of unimodality in action space. However, we show that such an assumption does not necessarily hold for a large class of actions. In this paper, we explicitly model the multimodality in the action space occurring due to significant intra-class variance using a non-parametric approach first proposed in [6]. In the presence of multiple views available in the training data, we propose a simple data fusion method to improve view-independent recognition. We present experimental results on the IXMAS dataset which is publicly available on INRIA's Perception Laboratory webpage[1]

2. Features

Given the video of an action, background subtraction [14, 4] can be applied to obtain the binarized silhouettes of the person. The largest blob in the binary image is assumed to be the subject. The binarized silhouette of the person provides a reasonable starting point for performing action recognition and has been used in [3, 7, 9]. We propose to use only the outer contour of the binarized silhouette, specifically the width of the outer contour, as we believe that it contains adequate information for recognizing actions. As we shall demonstrate, this feature captures both structural and dynamic information for an action. Width features for gait representation have been used by [7, 9].

2.1. Width Features

2.1.1 Width Extraction

The first step in generating the width vector corresponding to the binary silhouette is to place a bounding box around the silhouette. The size of the bounding box is set dynamically based on the size of the silhouette. However, there is little uniformity in the size of the silhouette as different people have different heights and also, its size varies with the person's distance from the camera. It is necessary that our width vector is of uniform size for later calculations. In order to normalize the size of the width vector, we uniformly scale the size of the bounding box so that it has a fixed height of hundred pixels, keeping the aspect ratio constant. Morphological close and open operations are applied to the frame in order to deal with noise due to background subtraction. The width along a given row is simply the difference in the locations of the right-most and left-most silhouette boundary pixels in that row. The advantage in using the width profile of a person as a feature is that it encompasses structural and dynamic information peculiar to each action well. Also, use of the width feature provides uniformity to feature representation across different individuals.

[1] https://charibdis.inrialpes.fr/html/sequences.php

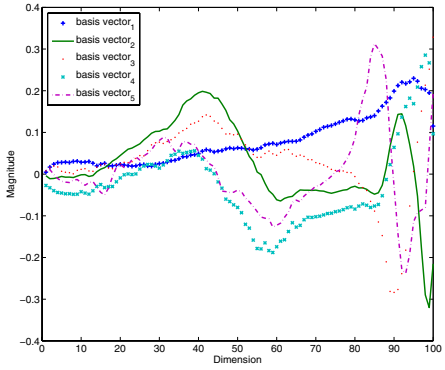

Figure 1. A plot of the five eigenvectors corresponding to the five most significant eigenvalues of the PCA of the concatenated width matrix W. These five vectors form the action basis used to obtain the five dimensional projections of the hundred dimensional width vectors.

We denote the hundred dimensional width vector of a given action at time t as $v(t)$.

2.1.2 Basis Generation & Dimensionality Reduction

One of the important considerations for performing fast recognition is that dimensionality of the feature be small. One way to reduce the dimensionality of the width feature is to project it to a one dimensional basis and use these projections as a lower dimensional representation. Examples of such one dimensional bases include DCT or polynomial bases such as Legendre polynomials. It was noted in [7], that eigen analysis of the width vector sequences for the gait of a particular individual reveal that the top 2 or 3 eigen vectors suffice to capture most of the width variation for the particular individual. It was also shown that, intuitively, the first and second eigen-vectors reflect information contained in the leg and hand motions respectively. Since most human activities involve some combinations of hand and leg motion, and the walk activity encompasses both of these, we propose to build an activity basis using the walk sequences of a large number of individuals. We collect width features from different individuals and perform PCA on the concatenated width matrix W. Observing that around top five eigenvectors account for some 95% of the variance, we choose these eigenvectors to form our action basis E (Fig 1) for dimensionality reduction. One of the limitations of this basis is of course it holds only for the side view. However in our experiments we found that projections on this basis provide discrimination even for frontal views of activities such as waving. In future work, we shall examine other bases.

Once the action basis E is generated, the width vector $v_i(t)$ of an action sequence of a person i computed henceforth is projected onto this basis as

$$[w_1(t)\ w_2(t)\ w_3(t)\ w_4(t)\ w_5(t)]^T = E \times (v_i(t) - \bar{v}_i(t)) \quad (1)$$

to obtain the five dimensional eigenprojections corresponding to that frame, where $\bar{v}_i(t)$ is the mean of $v_i(t)$. This procedure reduces the dimensionality of our features while retaining most of the necessary information.

2.2. Spatio-temporal Features

As our approach uses the binary silhouette of a person for action recognition, it is also important to keep a track of how the structure of the silhouette as a whole varies with time. Our spatio-temporal features include the displacements of the centroid c_x and c_y of the silhouette and the standard deviations σ_x and σ_y in both the X and Y directions respectively. These give us four more features (two each for centroid displacement and standard deviation) in addition to the five eigenfeatures. These features give us information about aspects such as pose and motion of the human silhouette. For certain activities, these features hold the key to correct recognition. For example, significant centroid motion in one direction can suggest that the person is walking. Or, a significant change in the value of standard deviation could indicate whether a person is sitting or standing. The centroid displacement is computed as the difference across five frames since over two consecutive frames there is not much displacement of the centroid.

We augment our initial five dimensional eigenfeatures by adding the four spatio-temporal features to obtain a nine dimensional feature vector

$$x = [w_1\ w_2\ w_3\ w_4\ w_5\ c_x\ c_y\ \sigma_x\ \sigma_y]^T \quad (2)$$

Robustness of the feature vector: In order to test the robustness of our feature vector we added different amounts of salt and pepper noise to the data from the IXMAS dataset and compared the results of using our algorithm. The raw and the corrupted images used for testing robustness are shown in Figure 9. The details of the experiment are presented in Section 4. The results are summarized in Table 1. As we can see, our chosen feature is very robust to noise.

3. Average-templates with multiple features

Given the features, the next important step is to get a template representation for each activity to perform a DTW recognition. The computational complexity of the recognition process is highly dependent on these representations.

3.1. Computing average-templates

An obvious way to reduce the complexity of recognition is to use an average or nominal template [18] for that action. In order to achieve this optimally, the non-linear

warping (or DTW) of a new instance of an activity must be carried out with already existing instances of the same activity. Therefore, for each action in the training set we compute an average pattern or average-template R by mapping available training instances $T = \{T_1, T_2, \ldots, T_n, \ldots\}$ by DTW. Here, the training patterns T_n are composed of certain number of frames for a given activity. Each of these frames correspond to the feature vector as given by (2). The accumulated distance $D(i, j)$for the DTW is defined as:

$$\begin{aligned}\min[&D(i-2, j-1) + 3d(i, j),\\ &D(i-1, j-1) + 2d(i, j),\\ &D(i-1, j-2) + 3d(i, j)]\end{aligned} \quad (3)$$

where i is the frame index of the average reference pattern R and j is the frame index of the train pattern T. In order to deal with the disparate components of the feature vector we use weighted Euclidean distance as the local distance $d(i, j)$ between the frame i of R and frame j of T. The weights are computed as the inverse of standard deviation of each component over the entire training set. If I is the length of R and J is the length of T. The path is forced to begin at the point $D(1, 1)$ and end at $D(I, J)$.

Backtracking from the point $D(I, J)$ yields the optimal path $p = [i_k, j_k]$ and the corresponding mapped set of feature vectors $[R(i_k), T(j_k)]$. Here k, is the index of a point on the optimal path p. The average reference pattern R_n for an activity is computed by the successive weighted averaging of n instances as follows:

$$R_n(k) = (1 - \frac{1}{n})R_{n-1}(i_k) + (\frac{1}{n})T_n(j_k); k = 1 \ldots K \quad (4)$$

where K is the number of points on the optimal path p and $R_{n-1}(i_k)$ is the average of the previous $n - 1$ templates. The new time axis for the instance R_n is computed as:

$$p_1(k) = (1 - \frac{1}{n})i_k + (\frac{1}{n})j_k; k = 1 \ldots K \quad (5)$$

We linearly transform this new time axis to a constant length P where P is the average length of all instances of an activity. The transformation is done as follows:

$$p_2(k) = \frac{P}{K}p_1(k) \quad (6)$$

as $p_2(k)$ would have non-integer values we define a time axis $p_3(k')$ where $k' = 1, 2, 3 \ldots P$. The feature values of the average pattern $R_n(k)$ are interpolated to get the new average pattern $R_n(k')$.

3.2. Combining average templates with multiple features

The average or nominal activity template is the clearly the best in terms of computational complexity. This representation posits an underlying Gaussianity in the activity space. Such an assumption for the intra-class variation can be inaccurate, however. A simple solution to this is to include multiple templates to represent each class. The downside to this is of course, a manifold increase in the computation time. Clearly, a method that combines the benefits of both the above approaches would be desirable.

We propose to address this problem by using the average-template with multiple features representation first proposed for speech recognition in [6]. In this method the templates in each action class from which the average is computed here are aligned using DTW to the average template for that action class. The optimal path contains information about which frame in the training template corresponds to which frame in the average-template. Using this information we bin together all frames corresponding to each frame of the average-template. Now, the local DTW distance between a frame of test data and a frame of the average-template is computed as the minimum of the distance between the respective test frame and the multiple feature vectors in the bin corresponding to that frame of the average. The sequence of warping is dictated by the average-template. In this way all the variations seen in a particular action class can be combined. Unlike the multiple template representation, the average-template with multiple features is like using a single template representation. If the number of training templates for each action class is large these bins corresponding to each frame of the average can be vector quantized (VQ). Since the database we are using is not that large, we did not use the VQ clustering.

To summarize the above steps, every k' in an average pattern for a category $R_n(k')$, where $k' = 1, 2, 3 \ldots P$, is associated with a bin of frames of size M. The local weighted euclidean DTW distance is computed as:

$$d(k', q) = min(d(k', m)) \quad (7)$$

where, q is the frame of the test sequence of length Q and m varies from $m = 1 \ldots M$ which is nothing but the m^{th} frame associated with k' frame the average template $R_n(k')$ for a given action class. The local continuity constraint is given by (3).

Discriminative training: There are several activities such as cross arms/check watch and pick up object/sit down that can still be confusable. A closer examination of one of the component trajectories for check watch and cross arms (Figure 3) reveals that for the first half, the temporal evolution is pretty similar while the second half is very different. A simple way to disambiguate such activities, then is to emphasize the first and last halves differently viz. a greater weightage is assigned to the latter halves when comparing the actions. These weights are computed under the framework of Fisher linear discriminative analysis [1].

To do the discriminative training, we first determine the

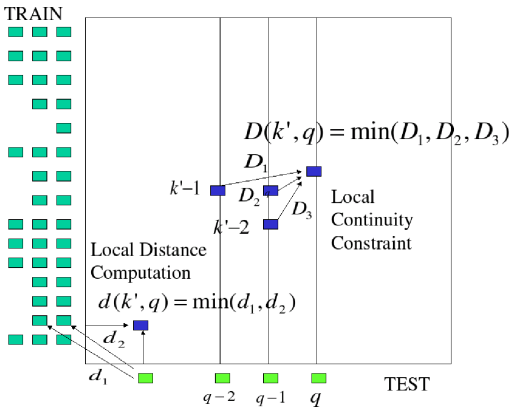

Figure 2. The Local Continuity constraint and the local distance filling for DTW

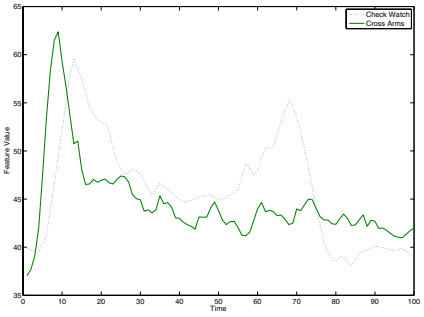

Figure 3. Plots of the second eigenfeature corresponding to actions: Check Watch (CW)(dotted) and Cross Arms (CA)(solid). As we can see the activities are similar in the beginning while towards the end they differ. Discriminative training provides a higher temporal weighting for the latter part of the feature when performing recognition, when an action gets classified as either CW or CA.

equivalence classes manually, i.e. we group the similar actions together on the basis of observation, such as check watch and cross arms. This is a two pass recognition process in the first pass we determine whether the test template belongs to a particular equivalence class or not. If they do belong to that equivalence class a second pass is done by weighting the optimal path with variable weights learned from the training set. The improvements we achieve in the recognition performance are presented in Section 4.

4. Experimental Results

We use the IXMAS dataset, which is freely available on INRIA's Perception Laboratory website, to test our algorithm. The multi-view data in it was obtained with the help of five synchronized cameras placed at different positions around the region where several actors performed the set of actions to be identified. We observed that, though the cameras might have been static, not always did they have the same view of all the persons performing the actions. That is, if a camera, say camera 1, viewed a person from a certain angle, it is not always the case that the same camera had the same view of another person. In order for our algorithm to perform recognition, view consistency was necessary while training. We re-organized the camera views in the dataset into 6 categories, depending on the direction in which the person performing the actions faces as follows:

- In view 1, person faces westward (left)
- In view 2, person faces south-westward (bottom-left)
- In view 3, person faces southward (bottom)
- In view 4, person faces south-eastward (bottom-right)
- In view 5, person faces eastward (right)
- In view 6, person faces north-eastward (top-right)

Views corresponding to cases where the person was facing northward (top) and north-westward (top-left) were either unavailable or were not in sufficient number to train our classifier. The results of using view 1 as gallery and view 5 as probe are shown in Figure 7 and Figure 6. All numbers in the tables are percentages. The results are as follows:

- For a single average template representing an action the recognition result is 75.10%.
- For an average template with multiple features the recognition result is 80.05%.

As we can see, using the average template with multiple features gives a near 5% improvement in recognition performance. With the use of average with multiple templates the greatest improvement is seen in the walk activity (a relative improvement of 50%).This can be reasoned as follows: For the walk activity, it is clear that all the nine components of the feature vector will be quite energetic as it involves a considerable amount of both hand and leg movements. This increases the likelihood of larger intra-class variability in the templates for walk sequences. This, in turn, would make it harder to register each instance with an average template. More importantly, this also suggests that "walk" provides a very good activity for performing activity specific human identification. In general, whenever an average template with multiple features works better than the average template, that activity will be very good for discriminating between people performing it. This is illustrated in Figure 8.

	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave	punch	kick	point	pick up	throw
check watch	34.21	10.53	18.42	0	0	0	0	36.84	0	0	0	0	0
cross arms	0	78.38	10.81	0	0	0	0	10.81	0	0	0	0	0
scratch head	6.06	0	27.27	0	0	0	0	63.64	0	0	3.03	0	0
sit down	0	0	0	100	0	0	0	0	0	0	0	0	0
get up	0	0	0	0	100	0	0	0	0	0	0	0	0
turn around	2.5	2.5	2.5	0	0	72.5	0	15	2.5	0	2.5	0	0
walk	0	0	2.56	0	0	15.38	79.49	0	0	0	0	0	2.56
wave	10.53	7.9	0	0	0	0	0	68.42	0	0	13.16	0	0
punch	2.44	2.44	2.44	0	0	0	0	26.83	48.78	0	4.88	4.88	7.32
kick	0	5.13	0	0	0	20.51	0	0	10.26	61.54	2.56	0	0
point	0	2.86	0	0	0	2.86	0	40	14.29	0	34.29	0	5.71
pick up	0	0	0	0	0	0	0	2.5	0	0	0	97.5	0
throw	0	15.15	0	0	0	0	0	27.27	6.06	0	6.06	0	45.45

Figure 4. Confusion matrix for experiment with view1 as gallery and views 2, 4 and 6 as probes using average template with multiple features.

	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave	punch	kick	point	pick up	throw
check watch	55.26	7.9	7.9	0	0	0	0	23.68	0	0	5.26	0	0
cross arms	0	89.19	2.7	0	0	0	0	8.11	0	0	0	0	0
scratch head	9.09	3.03	45.45	0	0	0	0	36.36	0	0	6.06	0	0
sit down	0	0	0	100	0	0	0	0	0	0	0	0	0
get up	0	0	0	0	100	0	0	0	0	0	0	0	0
turn around	2.5	2.5	0	0	0	87.5	0	2.5	2.5	0	2.5	0	0
walk	0	0	0	0	0	2.56	97.44	0	0	0	0	0	0
wave	7.9	7.9	0	0	0	0	0	65.79	0	0	18.42	0	0
punch	2.44	2.44	0	0	0	0	0	12.2	65.85	0	4.88	4.88	7.32
kick	0	2.56	0	0	0	5.13	0	0	7.69	84.61	0	0	0
point	0	5.71	14.29	0	0	0	0	28.57	14.29	2.86	28.57	0	5.71
pick up	0	0	0	0	0	0	0	0	0	0	0	100	0
throw	0	9.09	0	0	0	0	0	18.18	3.03	0	9.09	0	60.61

Figure 5. Confusion matrix for experiment with views 1 and 3 as gallery and views 2, 4 and 6 as probe using average template with multiple features.

4.1. Recognition across dissimilar views

Based on the organization of data as explained above, we conducted experiments by training our system with a certain view(s) and testing it on different views to demonstrate its view-invariant capabilities. We summarize the results, shown in Figure 4and 5, below.

- In the experiment using view 1 as gallery and views 2, 4 and 6 as probes, for an average template with multiple features, the recognition rate is 66.05%.
- In the experiment using views 1 and 3 as galleries and views 2, 4 and 6 as probes, for an average template with multiple features, the recognition rate increases to 76.28%.

In the former case, since the result of training the algorithm with only one view and testing it with three completely different views is reasonably high, it can be said that the features grasp various activities quite well. However, some confusions still prevail between actions like wave and scratch head, check watch and cross arms, etc.

In the latter case, when performing recognition, we use the minimum of the DTW score computed between the test template and the two orthogonal view training templates. As shown in [8], this is a specialization of the SUM fusion rule to the MAX rule if the DTW scores are transformed to probabilities. Using this strategy, most of the previous confusions can be eliminated. Also, there is no change in the recognition accuracy of actions like 'get up' and 'sit down' as they are well distinguishable from other actions irrespective of the viewing angle.

4.2. Robustness to Noise

Our method uses background subtracted silhouettes as a basis for generating features used in recognition. Often, the extracted human silhouettes are accompanied by noise

	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave	punch	kick	point	pick up	throw
check watch	70	25	0	0	0	0	0	5	0	0	0	0	0
cross arms	0	94.44	0	0	0	0	0	5.56	0	0	0	0	0
scratch head	0	5.88	52.94	0	0	0	0	35.29	0	0	0	5.88	0
sit down	0	0	0	100	0	0	0	0	0	0	0	0	0
get up	0	0	0	0	100	0	0	0	0	0	0	0	0
turn around	5.26	0	0	0	0	94.74	0	0	0	0	0	0	0
walk	0	0	0	0	0	31.58	68.42	0	0	0	0	0	0
wave	10.53	15.79	0	0	0	0	0	68.42	0	0	0	0	5.26
punch	0	0	0	0	0	0	0	5	80	0	0	0	15
kick	0	0	0	0	0	5.26	0	0	5.26	78.95	5.26	0	5.26
point	5.88	0	0	0	0	0	0	5.88	11.76	0	76.47	0	0
pick up	0	0	0	0	0	0	0	0	0	0	0	100	0
throw	0	6.25	0	0	0	0	0	31.25	6.25	0	0	0	56.25

Figure 6. Confusion matrix for experiment with view 1 as gallery and view 5 as probe using average template with multiple features.

	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave	punch	kick	point	pick up	throw
check watch	70	25	0	0	0	0	0	5	0	0	0	0	0
cross arms	0	88.89	11.11	0	0	0	0	0	0	0	0	0	0
scratch head	0	5.88	70.59	0	0	0	0	17.65	0	0	0	5.88	0
sit down	0	0	0	100	0	0	0	0	0	0	0	0	0
get up	0	5.26	0	0	94.74	0	0	0	0	0	0	0	0
turn around	5.26	0	0	0	0	94.74	0	0	0	0	0	0	0
walk	0	0	26.32	0	0	36.84	36.84	0	0	0	0	0	0
wave	15.79	15.79	26.32	0	0	0	0	42.1	0	0	0	0	0
punch	0	5	5	0	0	0	0	0	85	0	0	0	5
kick	0	0	0	0	0	5.26	0	0	10.53	78.95	5.26	0	0
point	5.88	0	11.77	0	0	0	0	0	11.77	0	70.59	0	0
pick up	0	0	0	0	0	0	0	0	0	0	0	100	0
throw	6.25	6.25	6.25	0	0	0	0	37.5	0	0	6.25	0	37.5

Figure 7. Confusion matrix for experiment with view1 as gallery and view5 as probe using average template.

from background subtraction which could be due to change in illumination of the scene, shadows, reflections, etc. In order to test the robustness of our system to noise, we synthetically added salt & pepper noise of different variances to the silhouette data (see Figure 9) and performed the same experiments on them as in the case of noiseless data. Table 1 displays the results of our experiments with noisy data. As illustrated in Table 1, we see that there is no sig-

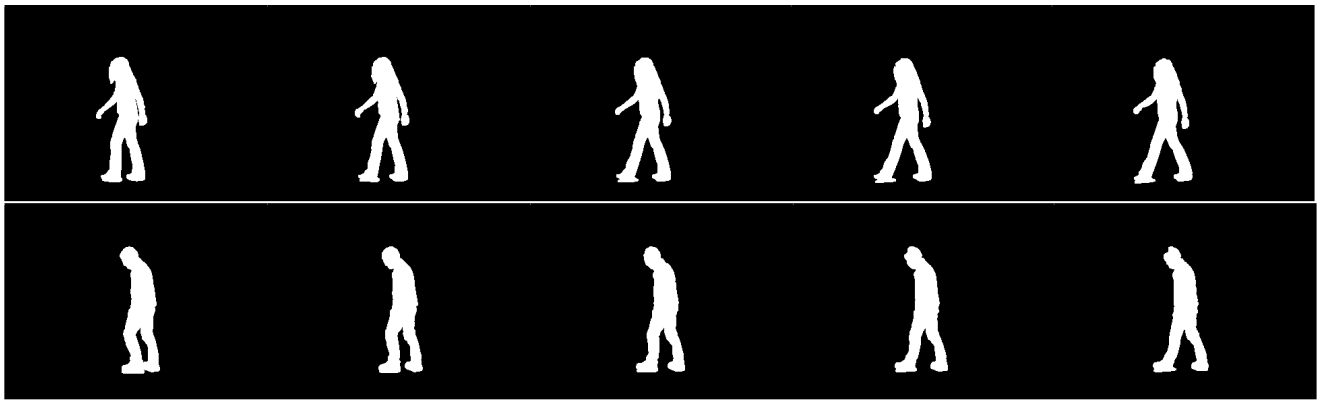

Figure 8. Illustration of different walking styles. As we can see there are differences in walking styles of people, making average template with multiple features work better.

Expt/Noise				
	No Noise	-27.99 dB	-39.95 dB	-48.36 dB
Expt 1	75.10%	75.52%	75.52%	75.93%
Expt 2	80.05%	81.74%	80.91%	81.33%
Expt 3	66.05%	66.67%	66.67%	65.44%
Expt 4	76.28%	76.07%	76.07%	75.87%

Table 1. Table illustrating the effect of noise on recognition results. Experiment 1 corresponds to the one with view 1 as gallery and view 5 as probe using average template, Experiment 2 to view 1 as gallery and view 5 as probe using average template with multiple features, Experiment 3 to view 1 as gallery and views 2, 4 and 6 as probes using average template with multiple features and Experiment 4 to views 1 and 3 as galleries and views 2, 4 and 6 as probes using average template with multiple features.

nificant change in the recognition performance on addition of noise. This change in the results can be considered statistically insignificant. Any level of noise above the ones used in our experiments would be too high even for poor background subtraction and would question the ability of the background subtraction algorithm used. This shows that our features are reasonably robust to noise.

4.3. Comparison with MHIs

We also conducted a comparative study between our method and the method proposed by Bobick et al. [3]. We chose this method for comparison as it is well-known and considered seminal in the area of action recognition. This approach uses Hu Moment Invariants of the Motion History Images (MHIs) of various activities as features for recognition. We performed the experiment with view 1 as gallery and view 5 as probe with our implementation of this technique. For each action in the gallery we computed the mean of the Hu moments across people and used the Mahalanobis distance based vector quantization to classify each instance from the probe. We found that this system achieved a recognition rate of 33.20% as compared to 80.05% rate of our system. One of the problems of using MHIs on the current database is that it can only deal with linear changes in speed. The database, however, includes activities such as check watch, cross arms and scratch head which have temporal segments that can persist for variable duration, while the other segments are of more or less same duration e.g. for cross arms, a person's arms can be crossed for any length of time while the segments when she raises or lowers her arms usually do not vary by much. Such actions are best matched only by using a non-linear time warp such as DTW. Furthermore MHIs also have trouble matching actions such as check watch vs cross arms (viewed from the side), where the motion vectors can be confused. Our chosen features reflect such small differences, and when coupled with DTW for matching, result in good recognition rates.

Discriminative Training Results: The application of the discriminative training to the experiment with view 1 as gallery and view 5 as probe for the single average template experiment helps when we have check watch and cross arms as the equivalence class. The number of confusions between those classes reduce from 5 to 3 which is equivalent to a relative increase of 40%.

5. Conclusion and Future Work

In this paper, we proposed a novel method for fast human action recognition. The method uses a nine-dimensional feature vector which comprises of (a) projections of the width profile on an *action basis* and (b) simple spatio temporal features. To deal with intra-class variability, we used the average template with multiple features representation in the DTW framework. We demonstrated the efficacy of our method in robustly recognizing human actions. We also proposed a simple data fusion technique to fuse information from two orthogonal views to improve monocular

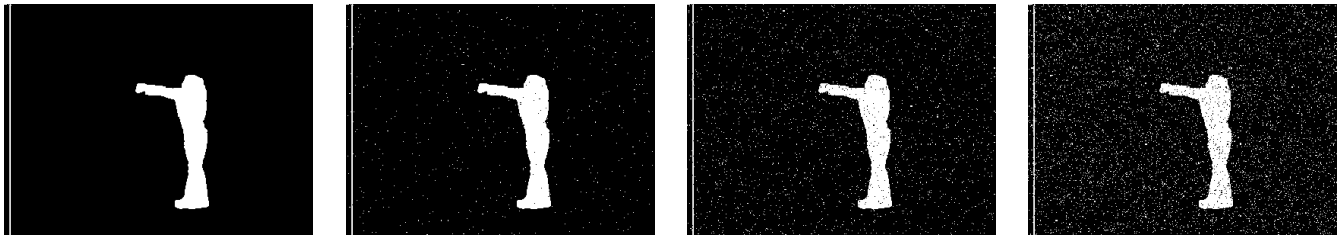

Figure 9. Figure displaying the different levels of noise added to the original data to test the robustness of our features. The first image is one without noise. The second, third and fourth images are corrupted with salt & pepper noise of variances 0.01, 0.04 and 0.1 respectively. Even with these noise levels, our algorithm retains its recognition accuracy.

action recognition performance. Finally, we proposed the use of the temporal discriminative weighting of the optimal DTW path for disambiguation of activities that resemble each other over certain time intervals and show significant differences over certain time intervals. The feature generation and the recognition using the average template with multiple features comfortably runs at 20 frames per second on a 1.8 GHz machine and further optimizations are in progress.

Future work consists of performing continuous action recognition. Furthermore, we also plan to study optimal data fusion strategies when multiple cameras are available. We also plan to study alternative basis functions for reducing feature dimensionality to achieve further improvements in recognition scores. Our method revealed how 'gait' provides a useful action for performing activity specific human identification. It would also be interesting to evaluate the nine-dimensional feature vector along with some of the concepts discussed here for gait recognition.

6. Acknowledgments

The authors would like to thank Padma Madhuri from Siemens Corporate Technology for her help in implementing the MHI action recognition.

References

[1] *Fundamentals of Speech Recognition*. Prentice Hall, New Jersey, USA, 1993.

[2] M. Blank, L. Gorelick, E. Schechtman, M. Irani, and R. Basri. Actions as space-time shapes. *Proc. of IEEE Intl. Conf. on Computer Vision*, Oct. 2005.

[3] A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2001.

[4] A. Elgammal, D. Harwood, and L. Davis. Non-parametric model for background subtraction. *Proc. of European Conf. on Computer Vision*, June 2000.

[5] L. Gorelick, M. Galun, E. Sharon, R. Basri, and A. Brandt. Shape representation and classification using the poisson equation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Dec. 2006.

[6] B. Kaemmerer. Method for producing reference segments describing voice modules and method for modelling voice unit of a spoken test model. United States Patent Application Publication, Dec. 2004. Pub. No. : US 2004/0249639 A1.

[7] A. Kale, N. Cuntoor, B. Yegnanarayana, A. N. Rajagopalan, and R. Chellappa. Gait analysis for human identification. *Proc. of Intl. Conf. on Audio and Video Based Person Authentication*, Jun. 2003.

[8] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Mar. 1998.

[9] Y. Liu, R. T. Collins, and Y. Tsin. Gait sequence analysis using frieze patterns. *Proc. of European Conf. on Computer Vision*, 2002.

[10] F. Lv and R. Nevatia. Single view human action recognition using key pose matching and viterbi path searching. *Proceedings. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Jun. 2007.

[11] A. Oikonomopoulos, I. Patras, and M. Pantic. Spatiotemporal salient points for visual recognition of human actions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, Jun. 2006.

[12] V. Parameswaran and R. Chellappa. View invariants for human action recognition. *Proceedings. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Jun. 2003.

[13] A. B. S. Ali and M. Shah. Chaotic invariants for human action recognition. *Proc. of IEEE Intl. Conf. on Computer Vision*, Oct. 2007.

[14] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 1999.

[15] P. K. Turaga, A. Veeraraghavan, and R. Chellappa. From videos to verbs: Mining videos for events using a cascade of dynamical systems. *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Jun. 2007.

[16] A. Veeraraghavan, R. Chellappa, and A. K. Roy-Chowdhury. The function space of an activity. *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Jun 2006.

[17] D. Weinland, E. Boyer, and R. Ronfard. Action recognition from arbitrary views using 3d exemplars. *Proc. of IEEE Intl. Conf. on Computer Vision*, Oct. 2007.

[18] R. Zelinski and F. Class. A learning procedure for speaker-dependent word recognition systems based on sequential processing of input tokens. *ICASSP*, Apr. 1983.