# The Function Space of an Activity*

Ashok Veeraraghavan, Rama Chellappa
University of Maryland
College Park, Md - 20742
{vashok/rama}@umiacs.umd.edu

Amit K. Roy-Chowdhury
University of California
Riverside, CA - 92521
amitrc@ee.ucr.edu

## Abstract

*An activity consists of an actor performing a series of actions in a pre-defined temporal order. An action is an individual atomic unit of an activity. Different instances of the same activity may consist of varying relative speeds at which the various actions are executed, in addition to other intra- and inter- person variabilities. Most existing algorithms for activity recognition are not very robust to intra- and inter-personal changes of the same activity, and are extremely sensitive to warping of the temporal axis due to variations in speed profile. In this paper, we provide a systematic approach to learn the nature of such time warps while simultaneously allowing for the variations in descriptors for actions. For each activity we learn an 'average' sequence that we denote as the nominal activity trajectory. We also learn a function space of time warpings for each activity separately. The model can be used to learn individual-specific warping patterns so that it may also be used for activity based person identification. The proposed model leads us to algorithms for learning a model for each activity, clustering activity sequences and activity recognition that are robust to temporal, intra- and inter-person variations. We provide experimental results using two datasets.*

## 1. Introduction

Activity recognition has attracted tremendous interest in recent years because of its potential in applications such as surveillance, security, and human body animation. An activity is a series of small atomic actions performed by an actor. Walking, running, jogging, climbing, swimming etc., are some examples of activities. Each activity is composed of atomic motion units we call actions. The temporal order of these actions is pre-defined for each activity. For example the activity 'sit down' consists of the following actions - bend knee, lower body, settle on chair and rest back on backrest, in that order. While the order of these actions is pre-defined, the temporal rate at which these actions are executed may vary. Results on gait based person identification shown in [1] indicate that it is very important to study the temporal variations in the execution of an activity.

**Relation to Prior Work:** Activity recognition has been an active research area since the 90's. The reader can refer to the survey on activity recognition [4] for a detailed review. Recently, [15] has explicitly identified three sources that contribute to variability in human activities as a) Viewpoint change, b) Anthropometry of actors and c) Execution rate. Related research efforts have typically concentrated on accounting for the variability in viewpoint [12][10] either by deriving view invariant features or by proposing algorithms that are view-invariant. Some recent research efforts have also looked at the variability due to anthropometry [3]. But very little has been done to account for the variability in the execution rate of the actors. Previous research [12][10][3] indicates that one can subsume the variability due to viewpoint and anthropometry by clever choice of features. In this paper, we explicitly model and learn the variability due to execution rate, while also accounting for other sources of variations. Our model is independent of the choice of feature. Therefore as more sophisticated features become available our model will be able to exploit the characteristics of those features while retaining the ability to deal with variations in execution rate.

**Motivation:** Consider $f(t)$ a function of time, composed of two ramps as shown in Figure 1. Let $g(t)$ be a temporally warped version of $f(t)$, i.e., $g(t) = f(w(t))$, where $w(t)$ is the warping function. Though the structure of these two functions are very similar, simple measures of similarity like correlation that do not account for temporal warping perform very poorly. For the problem of activity modeling, ignoring this temporal warping might lead to structural inconsistencies apart from providing poor recognition performance. The sequence of images shown in the first two rows of Figure 2 correspond to two different instances of the same individual performing the same activity. There is an obvious temporal warping between the two sequences. If this temporal warping is ignored, the distance between these two sequences will be large, leading to incorrect matching.

*This work was supported by the NSF-ITR Grant 0325119



Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)
0-7695-2597-0/06 $20.00 © 2006 IEEE

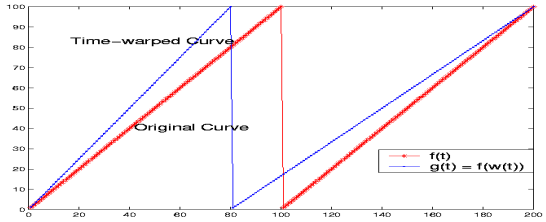


Figure 1. Original curve composed of two ramps and its time-warped version.

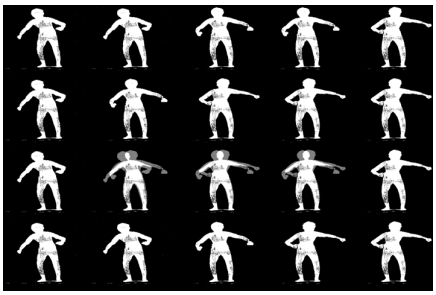

Figure 2. Row 1, Row 2: Two instances of the same activity. Row 3: A simple average sequence. Row 4:Average Sequence after accounting for time warps.

Moreover, if we are looking for some statistical description of the activity like an average sequence, ignoring the temporal warping could lead to structural inconsistencies like the presence of four arms and two heads in the average sequence, shown in the third row of Figure 2. If we do account for the temporal warping then such inconsistencies are avoided and the distance between the two sequences is rightly small. The fourth row shows a typical average sequence obtained by our method after accounting for the time warping. Most current algorithms for activity recognition do not explicitly model this temporal warping and therefore suffer from the above-mentioned limitations. The only algorithms that we are aware of that explicitly accounts for this temporal warping for activity recognition are [5][9] based on dynamic time warping [11]. While [9] computes an average shape sequence, neither of these [5][9] learn the nature of the warping functions.

### 1.1. Contributions of the paper

• We propose a systematic generative model for activities that accounts for variations in speed profile of an activity. We provide an approach to learn the space of time warps for each activity while simultaneously allowing for other intra- and inter-person variations.

• The model is composed of a *nominal activity trajectory* and a *function space* capturing the permissible activity-specific warping transformations. Given a sequence that is not used in training, the function space of time warpings allows us to 'interpolate' between the training sequences and determine whether the sequence belongs to the activity.

• We also provide algorithms for learning the model and using the model for activity recognition, activity clustering and organizing a database of activities.

Section 2 describes the model we call 'the function space of an activity'. Section 3 addresses issues in feature selection. Sections 4,5 and 6 discuss algorithms for recognition, clustering and organization of a database respectively.

## 2. Time-warping model for activities

Let $a(t)$ for $0 \leq t \leq T_a$ be a vector valued function of time. Let $b(t)$ for $0 \leq t \leq T_b$ be a time-warped version of $a(t)$, with the warping function given by $w(t)$, i.e., $b(t) = a(w(t))$, $w(t) : [0\ \ T_b] \longrightarrow [0\ \ T_a]$. Now $w(t)$ can be decomposed as $w(t) = T_a f(t/T_b)$ where $f : [0\ \ 1] \longrightarrow [0\ \ 1]$, i.e., a global linear dilation (or contraction) and a non-linear warping $f$. Without loss of generality we will use the word time-warping transformation to synonymously denote the non-linear time warping function given by $f$. We will also later indicate how the proposed method can be used to deal with global linear time warping.

Our model for each activity consists of a nominal activity trajectory given by $a(t)$ and $W_s$, the set containing all the time warping transformations permissible for that activity. Each realization of an activity is given by a trajectory $r(t) = a(f(t))$ where $f \in W_s$. To completely specify the model, we need to know a probability distribution function over $W_s$ so that we may sample candidates $f$ for each realization. In the rest of this paper we assume that given a set $W_s$ each member $f_i \in W_s$ is equally likely.

**Physical Significance of the Model:** *The nominal activity trajectory*, $a(t)$, and $W_s$, the *activity specific function space*, together capture all the possible realizations of the activity and provide the description of the activity under different variabilities. In general, the nominal activity trajectories of two different activities will be vastly different. The nominal activity trajectory for 'walking' would consist of key postures like heel-strike, toe-off, mid-stance etc., while that of 'sit down' would consist of the following actions - bend knee, lower body, settle on chair and rest back on backrest. The activity-specific function space of temporal warpings, $W_s$, represents the space of all the permissible time-warping transformations for each activity. By learning this space, we are able to 'interpolate' appropriately between training sequences. Suppose there is a test sequence that is within this space, but was not a part of the training sequences. Most template sequence based recognition techniques tend to misclassify such test sequences. Learning the convex function space of an activity provides our algorithm with the generalization power necessary to correctly classify such test sequences. Moreover, by learning this warping space formally, in a class specific manner, we also obtain better discriminative power than other heuristic techniques for handling time-

warping. The model M=$\{a(t), W_s\}$ represents a *function space* of activities whose elements are composed of functions $a(f(t)) \ \forall \ f \in W_s$.

### 2.1. Properties of the Function Space of Activities

Let $A$ be the function space consisting of all the allowed time-warping transformations. We will show some properties of the functions in this space by imposing certain physical constraints on the activities we model.

• The activity starts at time 0 and ends at time 1. Therefore, if $f : [0 \ \ 1] \longrightarrow [0 \ \ 1] \in A$ be a time warping transformation, then $f(0) = 0$ and $f(1) = 1$.

• The order of action units for each activity is pre-defined and cannot change. Therefore if $f \in A$, then $f'(t) > 0 \, \forall \, t \in (0, 1)$, i.e., every time warping transformation $f$ is a strictly monotone increasing function.

•None of the action units may be skipped, i.e., every time warping transformation $f$ is a continuous function of time. In fact there exists a finite maximum speed for the execution of these action units, i.e there exists some finite constant $c$ such that $f'(t) \leq c \, \forall \, t \in (0, 1)$.

•As a result of the previous properties we note that for every $f \in A$ there exists an inverse $f_{inv} \in A$ such that $f(f_{inv}(t)) = f_{inv}(f(t)) = t, \forall \, t \in (0, 1)$.

•We also note that $A$ is convex, i.e., $\forall \, f_1, f_2 \in A$ and $\alpha \in (0, 1)$, $f = \alpha f_1 + (1 - \alpha) f_2 \in A$.

### 2.2. Activity specific time-warping space

Even though $A$ represents the space of all plausible time-warping transformations, every individual activity may only be able to access a subset $W$ of the candidate functions in $A$ because of the physical constraints imposed on the actor and the activity. For example, let us consider the activity of 'jumping'. The actor may in principle speed up certain portions of the activity relative to the others. But, during the actual moments the actor has no contact with the ground, the only external forces on the actor are those from gravitation and therefore, much as he might attempt to, he will not be able to change the speed of his activity during such times. There are thus physical, aesthetic and structural constraints that restrict each activity to pick candidate time warping transformations from $W \subset A$ instead of $A$ itself. The constraints themselves vary with activity and therefore the space $W$ is different for each activity. Below, we discuss and visualize some properties of this activity specific time warping space $W$.

- $W$ is a subset of $A$, i.e., $W \subset A$.
- $f(t) = t$ is a candidate function in $W$, i.e., $f(t) = t \in W$. This represents no time warping.
- It is reasonable to assume that the function space is pointwise convex, i.e., $\forall \, f_1, f_2 \in W$ and $\forall t \in (0, 1)$ and $\alpha \in (0, 1)$, there exists atleast one function $f \in W$ such that $f(t) = \alpha f_1(t) + (1 - \alpha) f_2(t)$. This also means that the class specific time warping space $W$ is a convex set. Moreover, since the derivative is a linear operator, this means that if

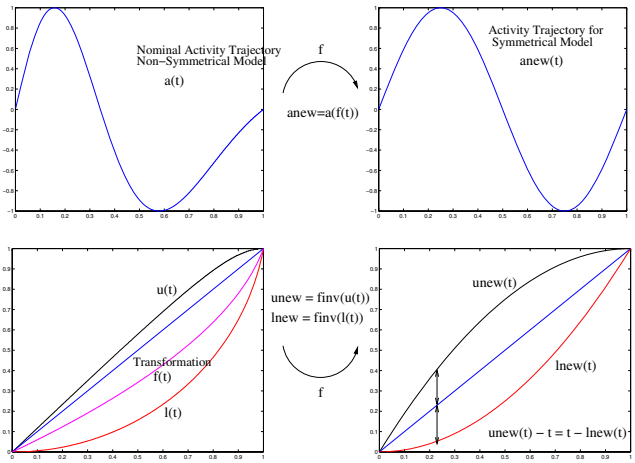


Figure 3. Non symmetrical model {a(t), $W_{ul}$} and the corresponding symmetrical version {$a_{new(t)}$, $W_{unew,lnew}$}. The warping transformation between the two models is given by $f$.

the rate of execution of some action unit can be speeded up by factors $\alpha_1$ and $\alpha_2$ then it can also be speeded up by any factor $\beta$ in between $\alpha_1$ and $\alpha_2$. This is not just reasonable but in fact desirable. These properties imply that $W$ can be bounded above and below by functions $u, l \in W$ such that

$$u(t) \geq t \geq l(t) \quad \forall t \in (0, 1) \tag{1}$$

$$u \geq f \geq l \quad \forall f(t) \in W \tag{2}$$

$$\text{where} \quad f \geq g \Longrightarrow f(t) \geq g(t) \quad \forall t \in (0, 1) \tag{3}$$

So, we can now index any such convex space $W$ by the functions $u$ and $l$ and call it $W_{ul}$.

### 2.3. Symmetric representation of an Activity Model

The representation of the activity model given by $M_1 = \{a(t), W_{ul}\}$ is not unique. Let $u_{new}(t) = f^{-1}(u(t))$ and $l_{new}(t) = f^{-1}(l(t))$ and let $f$ be a member function in $W_{ul}$. Consider the new model $M_2 = \{b(t), W_{u_{new} l_{new}}\} = \{a(f(t)), W_{u_{new} l_{new}}\}$. For every realization of the model $M_1$, i.e., $a(f_1(t))$ there exists a corresponding realization of the model $M_2$ given by $b(f^{-1}(f_1(t)))$. Therefore the two models $M_1$ and $M_2$ are equivalent (refer to Figure 3).

Before comparing two models, we need to account for this non-trivial ambiguity. The ambiguity can be resolved by arbitrarily specifying some notion of synchronizing time that is consistent both within class and across classes. The choice of any such synchronization time would perform equally well. What is important is that having specified a synchronizing time, we map the all the activity models obtained to this synchronizing time. This serves as a sort of 'reference frame' on the temporal axis.

Specifically, let us choose synchronization time, $f(t)$ such that the upper bounding function $u_{new}$ is as much above the identity warping function $i(t) = t$ as the lower bounding function $l_{new}$ is below the identity warping function $i(t) = t$. This synchronization time is appealing since

this ensures that the average of all the warping functions in $W_s$ is the identity warping function. The *symmetric* representation of the model is such that $u_{new}(t)-t = t-l_{new}(t)$. Therefore the activity specific warping space can be represented as $W_s = W_{u_{new}l_{new}}$ where $s(t) = u_{new}(t) - t = t-l_{new}(t)$, represents the extent of possible temporal warpings. This symmetric representation of the model is unique, i.e., if $M_1 = \{a_1(t), W_{s1}\}$ and $M_2 = \{a_2(t), W_{s2}\}$, then $M_1 = M_2 \Longleftrightarrow a_1 = a_2$ and $s_1 = s_2$.

Given a non-symmetric representation of the model, i.e., $M_1 = \{a(t), W_{ul}\}$, we still need to determine a time-warping function $f$ such that the upper bounding and the lower bounding functions of the new model are symmetric about the diagonal. This is achieved as

$$\begin{aligned} & u_{new}(t) - t && = t - l_{new}(t) \qquad (4)\\ \Rightarrow \quad & f^{-1}(u(t)) - t && = t - f^{-1}(l(t)) \\ \Rightarrow \quad & f^{-1}(t) - u^{-1}(t) && = u^{-1}(t) - f^{-1}(l(u^{-1}(t))) \\ & \text{(applying the } u^{-1} && \text{ operator)} \\ \Rightarrow \quad & f^{-1}(t) && = 2u^{-1}(t) - f^{-1}(l(u^{-1}(t))) \\ \Rightarrow \quad & f(t) && = \{2u^{-1}(t) - f^{-1}(l(u^{-1}(t)))\}^{-1} \end{aligned}$$

The implicit function equation can be solved by fixed point iterations as

$$f_{(i)}(t) = \{2u^{-1}(t) - f^{-1}_{(i-1)}(l(u^{-1}(t)))\}^{-1}, \qquad (5)$$

where $f_{(i)}$ represents the approximation of $f$ in the $i^{th}$ iteration. We initialize the iteration with $f_{(0)}(t) = \frac{u(t)+l(t)}{2}$. We observe that it converges within very few iterations with such an initialization.

Once we have obtained this symmetrizing time warp $f$ then any non-symmetric model parameters $M_1 = \{a(t), W_{ul}\}$ can be transformed to its symmetric (unique) counterpart as $M = \{b(t), W_s\}$, where $b(t) = a(f(t))$ and $s(t) = u_{new}(t) - t = t - l_{new}(t) = f^{-1}(u(t)) - t$. Figure 3 illustrates such a symmetrizing transformation.

### 2.4. Learning Model Parameters

The unique symmetric representation of the model is given by $M = \{a(t), W_s\}$. Learning the model parameters amounts to learning both the nominal activity trajectory $a(t)$ and the symmetric bounding function $s(t)$ for the activity specific warping set $W_s$. Let $b_i(t)$ for $i = 1, 2, ...N$ be $N$ realizations of an activity $\{a(t), W_s\}$. Suppose we knew the time-warping functions for each realization, i.e., we knew that $b_i(t) = a(f_i(t))$ for $i = 1, 2, ..., N$. Now, since we know both the actual realizations and the corresponding time-warping functions, we can invert the effect of time-warping to obtain an estimate of the unwarped nominal activity trajectory $a(t)$ as

$$\hat{a}(t) = E(b_i(f_i^{-1}(t))) = \frac{1}{N}\sum_{i=1}^{i=N} b_i(f_i^{-1}(t)), \qquad (6)$$

where $E(.)$ represents the expectation operator. In fact, in [8], they denote such time synchronised averages as "functional convex average" and address in detail some of the properties of such functional averages. They prove that for any given (convex) family of time-synchronizing mappings the functional convex mean exists and can be estimated as above if we knew the time-synchronisation. In particular, they look at time-synchronising maps like the area under the curve synchronisation. But the results on the convex operations, convex averages and asymptotics they derive are actually valid for any given synchronising map.

In practice though, we would be able to observe only the actual realizations $b_i(t)$ for $i = 1, 2, ...N$ while the time warping transformations $f_i$ would not be explicitly available. If we have a method to synchronize two warped activity trajectories, then as shown in [8] we can estimate the functional convex mean of the set of activity trajectories by averaging the time-synchronised versions of these realizations. The DTW algorithm [11] allows us to do this. It is a method for computing a non-linear time normalization (warping) between a template sequence and a test sequence. The DTW algorithm which is based on dynamic programming computes the best non-linear time normalization of the test sequence in order to match the template sequence by searching over the space of all time-warpings. The advantage of using DTW is that by cleverly using dynamic programming the complexity of the search space is considerably reduced. Since this space of all time-warpings over which the DTW algorithm searches must match a discrete version of the function space of time-warpings ($A$), the temporal consistency constraints used in the DTW algorithm must correspond to the properties of functions in $A$. The temporal consistency constraints used in our work are:

- End Point constraints: The 'start' and the 'end' of the activity trajectories must match exactly.
- Monotonicity: The warping function should be monotonically increasing, i.e., the sequence of action units must be unchanged.
- Continuity:The warping function must be continuous.

Let us assume without any loss of generality that $b_1(t)$ is the template sequence. Then DTW would find functions $g_i$ such that $b_i(t) \approx b_1(g_i(t))$. Now, the time synchronized activity trajectories can be obtained as $b_i(g_i^{-1}(t))$ and therefore the nominal activity trajectory can be estimated as

$$\hat{b}(t) = E(b_i(g_i^{-1}(t))) = \frac{1}{N}\sum_{i=1}^{i=N} b_i(g_i^{-1}(t)) \qquad (7)$$

Moreover, the upper and the lower bounding functions for the activity specific time-warping set can also be estimated from $g_i (i = 1, 2, ...N)$ as

$$\hat{u}(t) = \max_{i=1,2,...N} g_i(t) \ \forall t \in (0, 1) \qquad (8)$$

$$\hat{l}(t) = \min_{i=1,2,...N} g_i(t) \ \forall t \in (0, 1) \qquad (9)$$

Since each $g_i$ is constrained to be monotonously increasing and the end points are fixed, it is easy to see that the

estimates $\hat{u}(t)$ and $\hat{l}(t)$ also inherit these properties. Moreover, by construction, the estimates $\hat{u}$ and $\hat{l}$ are such that $\hat{u}(t) \geq t \geq \hat{l}(t)\ \forall t \in (0,1)$. Therefore $\hat{u}(t), \hat{l}(t)$ are valid upper and lower bounding functions for the convex activity specific time warping set and therefore $W_{ul}$ exists and is well defined. Thus the estimated model $\hat{M}$ is given by $\hat{M} = \{\hat{b}(t), W_{ul}\}$. This model parameters correspond to the non-symmetric version of the model and can be easily transformed to the equivalent symmetric version of the model given by $\hat{M}_s = \{a(t), W_s\}$. This is done by finding a time warping function $f$ such that the new upper and lower bounding functions $u_{new}(t), l_{new}(t)$ of the new model are symmetric about the diagonal and then setting $a(t) = b(f(t))$ and $s(t) = u_{new}(t) - t = t - l_{new}(t) = f^{-1}(u(t)) - t$. The procedure for determining $f$ was discussed in Section 2.3.

### 2.5. Global Speed of activity

We have restricted our attention to time-warping functions that do not contract or dilate the duration of the activity, i.e., we have concentrated on variations in speed profile. But this is not restrictive, since any other time-warping transformation can be decomposed into two parts: a linear global scaling of the temporal axis and the non-linear time-warping functions that we have addressed so far. In all our experiments we have first identified the global temporal scaling factor by identifying the start and stop instants of each activity. The identification of the start and stop instants of each activity is also done automatically by template matching. Once the global temporal scaling factor is found, each realization of the activity is temporally dilated or contracted linearly so that the total duration of the activity is a constant for all realizations of the activity.

## 3. Features for describing action units

**Choice of appropriate features:** In principle, the feature chosen to describe the action units must have physical significance and one must be able to directly identify the relationship between the features extracted and the basic human pose. For the problem of activity recognition, 3-D joint angles would be ideal features. Moreover, since the model for learning the function space time-warpings is not explicitly dependent on the choice of features, one could potentially use the same model to learn individual specific function spaces in order to perform activity based person identification. The only difference would be that we would choose a feature that is person-specific (e.g., silhouette). The nominal activity trajectory would be individual specific in this case. The function space of temporal warpings for each individual will now amount to learning the person specific warping functions. Unfortunately, estimating features like 3-D joint angles from images is extremely difficult and unreliable. So researchers have used several other features for describing the action units[12][9][14][6]. We use the shape of the silhouette as a feature[16].

**Kendall's Statistical Shape Feature:** "Shape is all the geometric information that remains when location, scale and rotational effects are filtered out from the object"[2]. We use Kendall's statistical shape as the shape feature. The binarized silhouette denoting the extent of the object in an image is obtained. A shape feature is extracted from this binarized silhouette. This feature vector must be invariant to translation and scaling since the objects identity should not depend on the distance of the object from the camera. This yields the pre-shape of the object in each frame. Let the configuration $(X)$ of a set of k landmark points be given by a k-dimensional complex vector containing the positions of the landmarks. Centered pre-shape is obtained by subtracting the mean from the configuration and then scaling to norm one as,

$$Z_c = \frac{CX}{\parallel CX \parallel}, \quad where \quad C = I_k - \frac{1}{k} 1_k 1_k^T, \qquad (10)$$

where $I_k$ is a identity matrix and $1_k$ a vector of ones.

**Distance between shapes:** The pre-shape vector lies on a spherical manifold. Therefore a concept of distance between two shapes must include the non-Euclidean nature of the shape space. Several distance metrics have been defined in [2] of which we use the partial Procrustes distance. Consider two complex configurations X and Y with corresponding preshapes $\alpha$ and $\beta$. The partial Procrustes distance between configurations X and Y is obtained by matching their respective preshapes $\alpha$ and $\beta$ as closely as possible over rotations, but not scale.

$$d_P(X,Y) = \inf_{\Gamma \epsilon SO(m)} \parallel \beta - \alpha\Gamma \parallel . \qquad (11)$$

The interested reader may refer to [2] for a detailed description of partial Procrustes distance.

## 4. Activity Recognition

Suppose we have $M$ different activity models given by $M_i = \{a_i(t), W_{s_i}\}$ for $i = 1, ..M$. Given a test sequence $h(t)$, the activity recognition problem is one of identifying the model that generated the test sequence $h(t)$. We do this in two steps. Firstly, assuming that the test sequence $h(t)$ is generated from the model $M_i$, we estimate the best warping transformation $\hat{f}_i$ from $W_{s_i}$ that would warp $a_i$ to $h$, i.e.,

$$\hat{f}_i = \min_{f \in W_{s_i}} dist(h(t), a_i(f(t))) \qquad (12)$$

$$\hat{I} = \arg \min_{i=1,...M} dist(h(t), a_i(\hat{f}_i(t))) \qquad (13)$$

Activity recognition is performed by minimizing the warping error between the nominal activity trajectory and the test sequence. Note that the search of warping functions is performed only over the corresponding activity specific warping set. The above-mentioned intuitive idea for

activity recognition can be easily implemented by a simple variation of the DTW. In the DTW algorithm, instead of arbitrarily limiting the warping function to lie within some window (typical choices are uniform window and parallelogram window), we replace the window constraints by the upper and lower bounds for the warping function that we have learnt for each model. Thus, the DTW algorithm with the window width being given by $u(t) = s(t) + t$ and $l(t) = t - s(t)$ computes the distance that is being minimized in Equation (13).

$$\hat{I} = \min_{i=1,...M} DTW(a_i, h, s), \tag{14}$$

where, $DTW(a_i, h, s)$ stands for the implementation of the DTW algorithm with the warping window constraints given by $u(t) = s(t) + t$ and $l(t) = t - s(t)$.

### 4.1. Relationship with other algorithms

It is interesting to note how the recognition algorithm (that arises from the model) is related to other algorithms designed with similar intent. Most methods that attempt to tackle time-warping have not been based on a model where observed trajectories are viewed as a realization of a stochastic process. Instead they were typically based on a template (eg., DTW[11]). A method for computing an average shape for a set of dynamic shapes is provided in [9]. A functional curve synchronisation model to estimate a longitudinal average (referred to as "functional convex average") is presented in [8]. Neither of these methods address the issue of learning the nature of time-warping transformations for each class from the data. Our model can be viewed as a generalization of these methods where we also learn the nature of the time-warping transformations for each class. A method to learn the best class of time-warping transformations for a given classification problem is proposed in [13]. Their algorithm is based on an optimization of recognition performance over a training set. Our recognition algorithm can also be viewed as class-specific, model based generalization of their algorithm. Template based recognition algorithms are very effective when the test sequence is one among those in the gallery. But they usually have very poor generalization power. Our algorithm has sufficient generalization power since we explicitly make the function space of an activity convex.

### 4.2. Common Activities Dataset

We collected a dataset of common activities to perform preliminary experiments to validate our model. The dataset consists of 10 activities and 10 different instances of each activity. These activities were captured using two synchronized cameras that were about 45 degrees apart. We perform a round-robin activity recognition experiment on this database. We partition the dataset into 10 disjoint sets each containing 1 instance of every activity. In order to test the recognition for each set, we first learn the model parameters from the remaining nine sets and then perform recognition for the test sequences. We repeat the process for each of the 10 sets. Thus we ensure that there is no overlap between the training set and the test sequences. Figure 4 shows the 10 X 100 similarity matrix for this experiment. Each column corresponds to a different test sequence while each row corresponds to a different activity. The strongly block diagonal nature of the similarity matrix indicates that the recognition algorithm performs well. In fact, on this database we obtained 100% recognition using our algorithm.

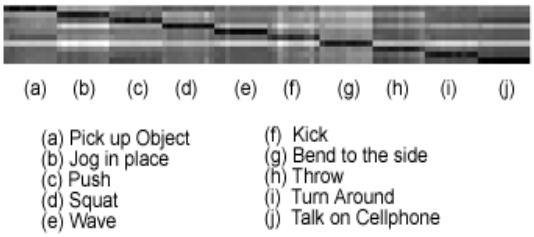


Figure 4. 10 X 100 Similarity matrix of 100 sequences and 10 different activities.

### 4.3. USF Gait Database

Since the model for learning the function space of time-warpings is not explicitly dependent on the choice of features, we use the same model to learn individual specific function spaces in order to perform activity based person identification. The nominal activity trajectory is now individual specific. Various external conditions (like surface, shoe) induce systematic time-warping variations within the gait signatures of each individual. The function space of temporal warpings for each individual amounts to learning the class of person specific warping functions. By learning the function space of these variations we are able to account for the effects of such external conditions.

In order to compare the performance of our algorithm with the current state of the art algorithms, we also performed a gait based person identification experiment on the publicly available USF gait database [14]. The USF database consists of 71 people in the Gallery. Various covariates like camera position, shoe type, surface and time were varied in a controlled manner to design a set of challenge experiments [14].The results are evaluated using cumulative match scores(CMS) curves and the identification rate. We performed a round-robin recognition experiment as before. Table 1 shows the identification rate of our algorithm, the baseline algorithm [14], simple DTW on shape features [16] and the image based HMM [6] algorithm on the USF dataset for the 7 probes A-G. Since most of these other algorithms could not account for the systematic variations in time-warping for each class the recognition experiment they performed was not round robin. Therefore, to ensure a fair comparison, we also implemented a round-robin ex-

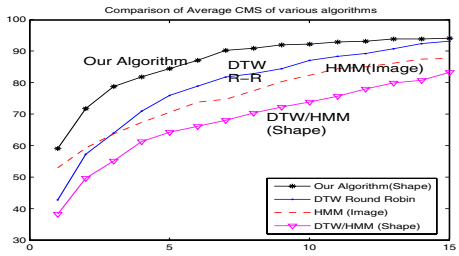


Figure 5. Average CMS of various algorithms on the USF dataset.

periment (DTW R-R) using the normal DTW algorithm and using the average shape sequence as the feature.

Table 1. Comparison of Identification rates on the USF dataset

| Probe | Baseline | DTW Shape | HMM Shape | HMM Image | DTW R-R | Our method |
|---|---|---|---|---|---|---|
| Avg. | 42 | 42 | 41 | 50 | 42 | 59 |
| A | 79 | 81 | 80 | 96 | 52 | 70 |
| B | 66 | 74 | 72 | 86 | 52 | 68 |
| C | 56 | 52 | 56 | 74 | 72 | 81 |
| D | 29 | 29 | 22 | 32 | 33 | 40 |
| E | 24 | 20 | 20 | 28 | 26 | 64 |
| F | 30 | 19 | 20 | 17 | 26 | 37 |
| G | 10 | 19 | 19 | 21 | 36 | 53 |

The average performance of our algorithm is better than all the other algorithms that use the same feature, (DTW/HMM (Shape)[16] and DTW R-R) and is also better than the baseline[14] and HMM[6] algorithms that use the image as a feature. The image based HMM algorithm [6] outperforms our algorithm for a probes A and B. One reason for this is that the image as a feature performs better than shape as a feature for the USF dataset. But, it is a computationally very intensive feature (of the order of number of pixels) and consequently leads to algorithms that are very slow. Therefore, we prefer to use the shape as a feature. Inspite of this obvious handicap, our algorithm outperforms the image based HMM algorithm for most probes. Another noteworthy fact is that the Probes A and B are very similar to the Gallery used for testing the image based HMM algorithm[14]. Therefore, the HMM algorithm is well tuned to the Probes A and B. Since our learning algorithm accounts for the variations in gait by learning the space of an activity using all the available data, it is not specifically tuned to any of the probes and performs 'equally well' for all the probes. Figure 5 shows the average CMS plot of the various algorithms compared.

## 5. Clustering Activity Sequences

**Algorithm for Clustering** The clustering algorithm, based on expectation maximization (EM) is very similar to the Lloyd-Max algorithm and can be used to organize a database of sequences for efficient retrieval. Let us assume that we know the number of clusters, $N$ and the cluster centers $c_1, c_2, ...c_N$. Then, each of the sequences in the database can be associated with one of the $N$ clusters. This can be done using a maximum-likelihood approach as described earlier in Equation (14). This forms the Maximization step of the EM algorithm. The Expectation step of the algorithm involves recomputing the new cluster centers from cluster memberships evaluated during the Maximization step. We iterate these 2 steps until convergence. In all our experiments, we initialized the cluster centers randomly.

**Clustering on Common Activities Dataset** We performed a clustering experiment on the 100 activity sequences collected as a part of the Common Activities dataset. We chose the number of clusters $N$ to be 10 since there were 10 different activities. If clustering were perfect, then the 100 activity sequences would be clustered into 10 different clusters, each cluster containing 10 sequences that correspond to that particular activity. But in reality, clustering would be imperfect and some of the 100 sequences would be misaligned in the wrong cluster. We repeated the clustering experiment several (about 50) times, with a random initialization of cluster centers during each trial. On an average, the algorithm converged in about 10 iterations and about 92% of the sequences were clustered correctly. In order to evaluate the robustness of the algorithm to initial conditions, we initialized the algorithm with the 10 cluster centers being 10 different instances of the same activity. This is one of the most adverse initializations possible for the clustering algorithm. Even during this trial, when the algorithm converged, the 10 clusters represented the 10 different activities. Moreover, 80 out of the hundred sequences were correctly clustered. This shows that the algorithm is fairly robust to initialization.

## 6. Organizing a Large Database of Activities

With the decreasing cost of storage, the size of activity databases is increasing rapidly. For example, the complete USF gait database [14] consists of about 122 classes and a total of more than 1000 sequences. As the size of the database increases, the number of 'distance' computations that must be performed on every query also increases linearly with the size of the database. This poses a significant bottleneck for practical activity recognition systems. We show that organizing the database of sequences using the clustering algorithm described in Section 5 decreases this computational burden significantly. The price paid is a small decrease in recognition performance. We organize the database of activities in the form of a dendrogram as shown in Figure 6. At each level of the dendrogram the number of branches ($B$) was set to 3. The number of levels to which the dendrogram is 'grown' determines the trade-off between computation and accuracy. As the number of levels is increased, the number of 'distance' computations

that must be performed before finding the class membership of a given test sequence decreases. Therefore, the computational burden of the algorithm also decreases. But this might introduce a decrease in classification performance. When the dendrogram is fully grown (i.e., when each leaf of the dendrogram represents one activity), there will be $log_B N$, levels and therefore $Blog_B N$ 'distance computations'. Let us consider the USF database which consists of 122 subjects and a total of 1870 sequences. A nearest neighbour classifier on this database must perform 1870 distance computations in order to classify a new test sequence. But if we assume that we organize the database in the form of a 'fully grown dendrogram', with each leaf node representing each of the 122 individuals, then one would just have to perform about $Blog_B N = 3 * log_3 122 \approx 14$ 'distance computations'. This is a very significant computational saving.

We performed an experiment to evaluate the efficiency of organizing the database on a subset of the USF database as in Section 4.3. In our experiments, we grow the dendrogram upto 2 levels. We measure efficiency of organization ($\eta$) as a ratio of the recognition rate before and after organization.

$$\eta = 100 * \frac{\text{Identification rate after organization}}{\text{Identification rate before organization}} \quad (15)$$

The efficiency $\eta$ is strongly related to clustering performance and it is reasonable to expect the efficiency $\eta$ to increase with better clustering. Table 2 shows the efficiency of organization for the various probes in the USF dataset. On this data, the dendrogram organization of the database reduced the computational time by a factor of about 30. This means that processing time for large databases will be reduced from the order of days to a matter of hours. For such a significant reduction in processing time the Table 2 shows that the decrease in recognition performance is not drastic. It is also possible to index DTW for efficient retrieval[7]. However, the indexing method in [7] is derived for Euclidean spaces, while our method is derived on spherical manifolds. A more important contribution of our method is that organizing the database using clustering also provides us with a nice graphical visualization of the space of activities, where activities that are similar get separated into different clusters lower down the dendrogram than activities that are dissimilar.

Table 2. Efficiency of Organization on the USF dataset

| Probe | A | B | C | D | E | F | G | Avg |
|---|---|---|---|---|---|---|---|---|
| $\eta$ | 76 | 81 | 84 | 100 | 82 | 100 | 95 | 89 |

## 7. Summary and conclusions

In this paper, we address an important but often neglected problem in modeling activity, that of temporal warping of the activity trajectories. Apart from temporal warping, activities are sometimes not aligned spatially. We are addressing the issue of handling larger spatial misalignments in activities, using mixture models.

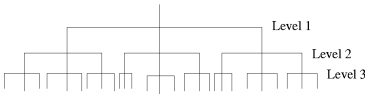


Figure 6. Dendrogram for organizing an activity database

## References


[1] A. Bobick and Tanawongsuwan. Performance analysis of time-distance gait parameters under different speeds. *4th Intl. Conf. on AVBPA*, June 2003.

[2] I. Dryden and K. Mardia. *Statistical shape analysis*. John Wiley and sons, 1998.

[3] A. Gritai, Y. Sheikh, and M. Shah. On the use of anthropometry in the invariant analysis of human actions. *ICPR*, 2004.

[4] W. Hu, T. Tan, L. Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Trans. on Systems, Man, Cybernetics - Part C:Applications and Reviews*, 34(3), 2004.

[5] A. Kale, N. Cuntoor, B. Yegnanarayana, A. Rajagopalan, and R. Chellappa. Gait-based human identification using appearance matching. *Optical and Digital Techniques for Information Security Ed. B Javidi et.al*.

[6] A. Kale, A. Sundaresan, A. Rajagopalan, N. Cuntoor, A. Roy Cowdhury, V. Krueger, and R. Chellappa. Identification of humans using gait. *IEEE Trans. on Image Processing*, Sept. 2004.

[7] E. Keogh. Exact indexing of dynamic time warping. *VLDB*, 2002.

[8] X. Liu and H. Mller. Functional convex averaging and synchronization for time-warped random curves. *J. American Statistical Association*, 99:687–699, 2004.

[9] P. Maurel and G. Sapiro. Dynamic shapes average. *www.ima.umn.edu/preprints/may2003/1924.pdf*.

[10] V. Parameswaran and R. Chellappa. View invariants for human action recognition. *CVPR*, 2003.

[11] L. Rabiner and B. Juang. *Fundamentals of speech recognition.* Prentice Hall, 1993.

[12] C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 2002.

[13] C. Ratanamahatana and E. Keogh. Making time-series classification more accurate using learned constraints. *Proceedings of SIAM International Conference on Data Mining*, pages 11–22, 2004.

[14] S. Sarkar, P. Phillips, Z. Liu, I. Vega, P. Grother, and K. Bowyer. The humanid gait challenge problem: data sets, performance, and analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 162–177, Feb 2005.

[15] Y. Sheikh and M. Shah. Exploring the space of an action for human action recognition. *ICCV*, Oct 2005.

[16] A. Veeraraghavan, A. RoyChowdhury, and R. Chellappa. Role of shape and kinematics in human movement analysis. *CVPR*, 2004.