Distribution-based dimensionality reduction applied to articulated motion recognition

Sunita Nayak[1], Sudeep Sarkar[1], Barbara Loeding[2]
[1]Department of Computer Science & Engineering, [2]Department of Special Education
University of South Florida, Tampa, FL 33620, USA
{snayak,sarkar }@csee.usf.edu , bloeding@lakeland.usf.edu

***Abstract*— Some articulated motion representations rely on frame-wise abstractions of the statistical distribution of low-level features such as orientation, color, or relational distributions. As configuration among parts change with articulated motion, the distribution changes, tracing a trajectory in the latent space of distributions, which we call the configuration space. These trajectories can then be used for recognition using standard techniques such as dynamic time warping. The core theory in this paper concerns embedding the frame-wise distributions, which can be looked upon as probability functions, into a low-dimensional space so that we can estimate various meaningful probabilistic distances such as the Chernoff, Bhattacharya, Matusita, Kullback Leibler (KL) or symmetric-KL distances based on dot products between points in this space. Apart from computational advantages, this representation also affords speed-normalized matching of motion signatures. Speed normalized representations can be formed by interpolating the configuration trajectories along their arc lengths, without using any knowledge of the temporal scale variations between the sequences. We experiment with five different probabilistic distance measures and show the usefulness of the representation in three different contexts - sign recognition (with large number of possible classes), gesture recognition (with person variations), and classification of human-human interaction sequences (with segmentation problems). We find the importance of using the right distance measure for each situation. The low-dimensional embedding makes matching two to three times faster, while achieving recognition accuracies that are close to those obtained without using the low-dimensional embedding. We also empirically establish the robustness of the representation with respect to low-level parameters, embedding parameters, and temporal scale parameters.**

***Index Terms*— Human motion classification, Embedding probability density functions, Gesture recognition, Sign language recognition**

I. Introduction

Articulated motion analysis is fundamental to gesture recognition. Reviews of work in articulated human motion appear in [1], [2], [3], [4]. While some approaches are based on magnetic or optical markers [5], [6], [7], [8], we are interested in pure image-based input approaches [9], [10], [11], [12]. Also, instead of parametric models [13], [14], [15], [16], [17], we consider non-model-based approaches [18]. Approaches to articulated human motion recognition include those that rely on body part labeling and tracking [13], [19], [20], [17] and those that do not [21], [22], [23], [24]. This latter class of approaches does not require part tracking or correspondence between frames, and does not rely on geometric object models like skeletons. One approach is Bobick and Davis' representation [25] of human motion in short video sequences using motion history images. On conceptually similar grounds is Masoud and Papanikolopoulos' [26] study on feature images used to represent motion in an image sequence without using limb tracking. Whole body contours have also been used for gesture recognition [27]. Zahedi et al. [28] used appearance-based models for sign recognition. Spatio-temporal shapes have been used recently [23], [18], [29] for action recognition.

Our current work is concerned with those tracking-free, image-based, parametric model-free approaches that abstract each frame as a distribution of low-level features such as edge orientation, flow distribution or relationships between features. For instance, the orientation histogram has been used in gesture recognition [30] to represent the hand in images of a sequence. Articulated motion involves change in the relative configuration of the parts, which manifests itself as evolution of these distributions. In sign language recognition, there is some experimental evidence that suggests signs can be recognized from point light displays [31], [32], emphasizing that for a broad class of signs just global configuration of hands and face is sufficient for recognition. While model-based approaches can provide detailed information like hand orientation and finger positions and can handle large view-point variations, these details might not be essential for articulated motion recognition in some contexts, such as sign language recognition, and classification of some kinds of gestures and human activities. On the other hand, by using such a global representation of the motion, one cannot make distinctions between small movements, such as movements of fingers when the whole upper body of a person has been captured in the image sequence. Alternative methods, such as using high resolution cameras focused around the hand region or adopting model-based methods, might be warranted in such cases.

The advantage of using distribution-based representations is that they do not need part-level tracking nor do they require that the segmentation be very precise; they are tolerant of boundary fragmentation and some level of spurious features. Such distribution-based ideas have also been recently proposed for the object detection problem in images. For instance, spatial-color joint distribution has been used by Crandall and Luo [33] and joint statistics were used for human detection by Mikolajczyk et al. [34]. Color distributions or histograms are routinely used in image-database indexing. In this work, we demonstrate our ideas using a global statistical representation, called the relational distribution, to represent the global part configuration structure in each image frame in a Gestaltic manner. It uses the distribution of pairwise relationships between the low-level primitives, such as edge points, and has been shown to be useful for human motion recognition [21] and sign language recognition [35].

These distributions, in particular the relational distributions,

have high dimensionality, with the number of dimensions being equal to the total number of bins in a histogram. Apart from the discretization parameter, the number of bins in a histogram is also proportional to the number of attributes one uses to describe the relationship between two low-level primitives. Although we have used just the displacement between two edge points as the attributes resulting in two dimensions, this is not the only possible choice. One could think of also using the local gradient information such as gradient magnitude or direction as two more attributes. Another possibility includes the use of optic flow vector information at the edge point, if available. With the addition of new attributes, the relational distribution gets richer, however, the size of the histogram representation increases exponentially. Thus, there is a need for a low-dimensional embedding of these relational distributions. We will refer to this low-dimensional embedding space of the relational distributions as the *configuration space*.

Note that our representation is essentially state-space based, where instead of the state being specified by the features extracted from the images [10], [21], [27], the model parameters [15], [16], or by pixel locations (motion trajectories) [36], it is specified by the configuration space coordinates. In this space, one could consider discrete state-based methods such as Hidden Markov Models (HMMs)) [37], [38] or continuous state-space methods, such as autoregressive models [15], [22] and statistical continuous curve representations [39]. The use of a low-dimensional embedding allows us to use continuous curve representations of motion through interpolation. This is particularly important when matching two sequences at different speeds.

Matching of motion state space trajectories is typically accomplished using dynamic programming [37] or trajectory correlation [21], or more recently, by using salient points on the trajectories [40]. State-based approaches like Hidden Markov Models [41], Conditional Random Fields [42], Maximum Entropy Markov Models [43] and Hidden State Conditional Random Fields [11] can also be used with relational distributions, but they require more training data. There has also been work recently in aligning motion sequences [29]. In this work, we will demonstrate our ideas using dynamic time warping.

Figure 1 depicts an overview of our approach. Each image frame is represented by a relational distribution that captures the probability that a randomly selected pair of image primitives (e.g., edge primitives) exhibit a certain relationship (e.g., horizontal and vertical displacement). This distribution is estimated by the normalized histogram of the relationships between sampled pairs of edge pixels. We have found that such gross overall representation works well for a wide range of articulated motion scenarios. We embed the relational distributions into a low-dimensional space where meaningful probabilistic distance measures can be computed. This is the core theoretical contribution.

We start with expressing the probabilistic distances in terms of dot products and define two simple transformations of the histograms depending on the nature of the distance measure. The transformed histograms are then embedded into the low-dimensional space by preserving pairwise dot products between them. We call this reduced dimensional space the configuration space. The probabilistic distance measures are then calculated by computations of dot products in the configuration space. This results in computational savings. We show that the same framework works for five different probabilistic distance measures, the Bhattacharya [44], Matusita [45], KL, Symmetric KL [46], and Chernoff [47] distance measures. We apply the embedding for articulated motion recognition, but the embedding process is a general one that can be used to embed probability functions for other vision applications such as in image databases, where histogram representations are common.

To enable the matching of sequences at different speeds, we will explain how to perform interpolation of the embedded trajectories. We will demonstrate recognition results in three different domains - sign recognition using 147 signs from American Sign Language (ASL), gesture recognition using a two-handed gesture dataset (from IDIAP, Switzerland) with seven different hand motions from seven subjects and activity classification using a human interaction dataset (from CMU) that involves two persons interacting with each other in eight different ways against a complex background.

We organized the paper as follows. Section II reviews the concept of relational distributions. It describes how the relative configuration of regions of interest in an image can be expressed as a probability density function and the dynamics involved in a motion sequence can be represented as changes in these probability density functions. Section III describes how we embed the relational distributions, which are probability density functions, into a low-dimensional space. It also explains how a new relational distribution corresponding to an image in a test sequence can be projected onto the space. Next, in Section IV-B, we describe how motion can be represented as a smooth curve in the space using interpolation. We then explain our experiments and show their results in Section V. We conclude the paper with Section VI.

II. Configuration of Low-Level Primitives

We contend that it is possible to discriminate among many articulated motions using just the change in the global configuration or structure of the object. For instance, in sign language recognition, where the frontal view is primarily used, it should be possible to discriminate between a number of signs based on just the relative movement of the hands, without the need for detailed information such as shape of the individual fingers. Indeed there is some experimental evidence that sign recognition is possible with point light displays [31], [32]. Similarly, it should be possible to distinguish between aggressive and benign interactions between two individuals based on their global postures, without tracking limbs. Of course, such a solution is not a complete solution to the problem; there will be situations that will need detailed analysis. However, as can be seen in the results section, we found that global structure is sufficient for a broad range of articulated motion recognition and classification.

How do we capture the global configuration of the object? We start with low-level primitives that are most likely to come from the articulated object. The exact nature of the low-level primitives can vary. Some common choices include edges, salient points, and Gabor filter outputs; we use edges in this work. We start from some level of segmentation of the object from the scene. The exact steps to achieve this are not new and can vary from domain to domain. They consist of background subtraction, skin color detection or color-based blob extraction. The uses of these processes are fairly standard and have been used widely in gesture and sign recognition. We then consider the boundaries of the postulated object regions along with the edges inside them or just

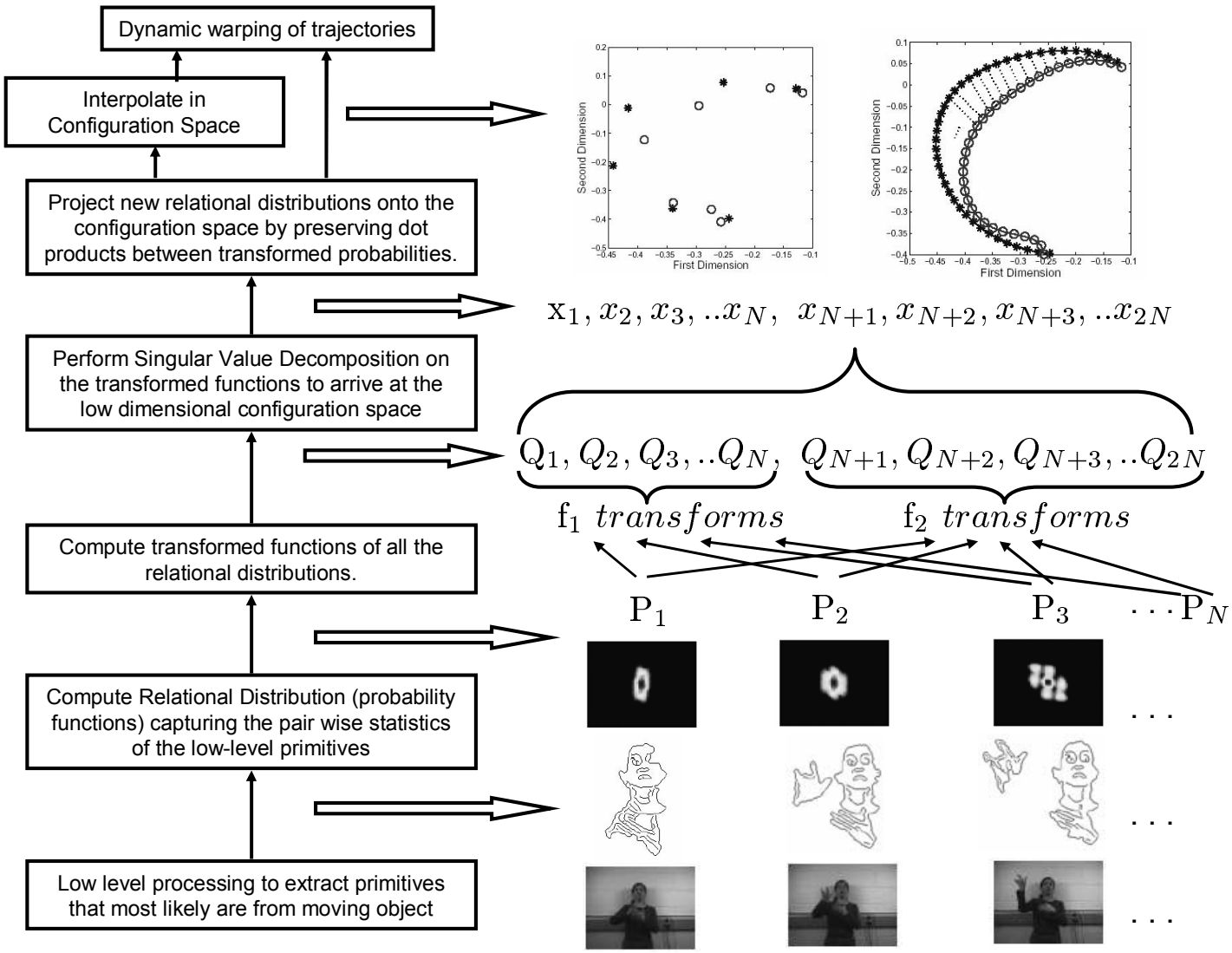


Fig. 1. Overview of the approach along with samples of the intermediate representations. N denotes the number of relational distributions in the training dataset $P_1, P_2, \cdots, P_N$

the region blob boundaries as our low-level image primitives. We capture the global configurations by considering the relationships between these primitives.

Let there be n low-level primitives in an image that are represented as a set $F = \{f_1, \cdots, f_n\}$ and the relationship among k-tuple primitives is represented by a set of M attributes $\mathbf{a} = \{\mathbf{a}_1, \cdots, \mathbf{a}_M\}$. In our experiments, we use the distance between two contour or edge pixels in the vertical and horizontal directions (dx, dy) as the relational attributes, i.e. $k = 2$ and $\mathbf{a} = \{dx, dy\}$. The joint probability function $P(\mathbf{a})$ describes the distribution of the primitives within an image and captures the shape of the pattern in the image. This probability is called a *relational distribution* [21]. It captures the global configuration of the low-level primitives. In rest of the paper, we use the notation $P(\mathbf{a})$ and P interchangeably. Both of them denote a relational distribution.

The rigid or non-rigid motion of the objects in an image sequence will result in changes in the distribution of primitives in the images. Figure 2 shows some example variations in the relational distributions with motion. It shows the top view of the distributions: the region near the center represents the points closer to each other, e.g., the edge points within the face or within the hand; the region further from the center represents the far away points, e.g., the relationship between the edge points of a hand and the face. Notice the change in the relational distributions as the signer moves one of her hands. To be able to discriminate symmetrically opposite motion, we maintain the signs (or directions) of the horizontal and vertical distances between the edge pixels in each ordered pair. This leads to the representation of the probability distribution in a four quadrant system. Given that these relational distributions exhibit complicated shapes that are difficult to be modeled readily using a combination of simple shaped distributions such as Gaussian mixtures, we adopt a non-parametric histogram-based representation. For better discrimination of the probabilities, we do not add counts to the center of the histogram which represents the distance of the edge pixels from itself or very close adjacent pixels. Each bin then counts the pairs of edge pixels between which the horizontal and vertical distances each lie in some fixed range that depends on the location of the bin in the histogram. In our experiments, for simplicity, we use a fixed size histogram for all the sequences. The above range is then defined using linear mapping between the image size and

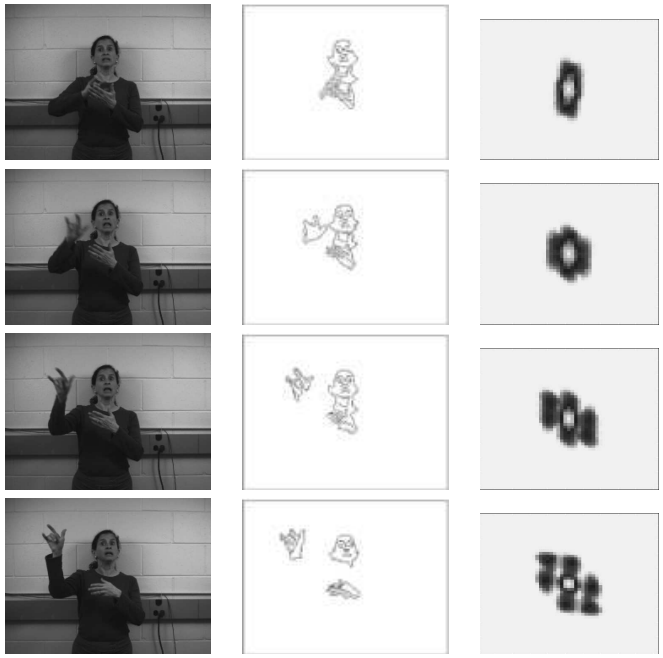


Fig. 2. Variations in the relational distributions with motion. The first column shows a motion sequence, the second column shows the edge pixels in skin color blobs, and the third column shows the relational distributions constructed from the low level features (edge pixels) of the images in the motion sequence. The horizontal axis of the relational distribution represents the horizontal distance between the edge pixels and its vertical axis represents the vertical distance between the edge pixels.

the histogram size, e.g., image size along the horizontal direction corresponds to half the histogram size in the horizontal direction. One could use histogram bin size optimization techniques for optimizing the histograms for each dataset, but we do not address them in this paper. We found the fixed size histograms to be sufficient for our experiments.

III. Embedding Probability Functions in Low-dimensional Configuration Spaces

We have to be careful while constructing the low-dimensional embedding. It should be possible to compute meaningful distance measures from this space. Table I lists some of the commonly used distance measures between two probability distributions. A common method for dimensionality reduction is principal component analysis (PCA), as was done in [21]. However, this type of embedding is appropriate only for computing certain types of distance measures between probability functions such as the Euclidean and Matusita distance measure. Other commonly used dimensionality reduction techniques include kernelPCA [48], multi-dimensional scaling [49], isomap [50], local-linear embedding [51] and Laplacian eigenmaps [52]. However, unlike these approaches, we do not seek to directly match the distances in the embedding space with the probabilistic distances; this will not be possible for many non-linear probabilistic distance measures. Rather we propose a strategy that permits us to construct spaces that allow the indirect estimation of the distances in Table I

We observed that the distance measures, such as the ones listed in Table I, can be expressed as functions of the inner product of transformed versions of the given probability functions. For instance, the Symmetric Kullback-Liebler (Symmetric KL) distance measure between two probabilities $P_1(\mathbf{a})$ and $P_2(\mathbf{a})$, given by $\sum_{\mathbf{a}}(P_1(\mathbf{a})-P_2(\mathbf{a}))\log\frac{P_1(\mathbf{a})}{P_2(\mathbf{a})}$, can be expressed in terms of inner products as: $\langle f_1(P_1), f_2(P_1)\rangle - \langle f_1(P_1), f_2(P_2)\rangle - \langle f_1(P_2), f_2(P_1)\rangle + \langle f_1(P_2), f_2(P_2)\rangle$, where f_1 and f_2 represent the identity and the log transformations respectively, and $\langle a,b\rangle$ is the inner product (dot product for vectors) of a and b. The third and fourth columns in Table I list the required transformation functions, denoted here by f_1 and f_2, for the five different distance measures.

Let the given training set of N relational distributions (or probability functions) that are used to construct the configuration space be denoted by $\{P_j(\mathbf{a})|j=1,\cdots,N\}$. Let $\{Q_1,\cdots,Q_{2N}\}$ represent the set of $2N$ transformed probability functions, where the first N transformed functions are given by $\{Q_j(\mathbf{a})=f_1(P_j(\mathbf{a}))|j=1,\cdots,N\}$ and the next half are given by $\{Q_{j+N}(\mathbf{a})=f_2(P_j(\mathbf{a}))|j=1,\cdots,N\}$. Note that we can now express the distance measures in terms of the Q_j functions. For example, the Symmetric KL distance between two relational distributions P_1 and P_2, can now be written as $\langle Q_1,Q_{N+1}\rangle - \langle Q_1,Q_{N+2}\rangle - \langle Q_2,Q_{N+1}\rangle + \langle Q_2,Q_{N+2}\rangle$. The last two columns in the table describe how each distance can be computed using inner products of transformations of the probability functions. So, if we had a space that would allow us to compute these inner products of the transformed versions of the probabilities (i.e., inner product between the Q_js), then we could construct the corresponding distance measure from them.

Let the entries of the $2N\times 2N$ inner product matrix, S, of these functions be given by

$$s_{ij} = \langle Q_i(\mathbf{a})Q_j(\mathbf{a})\rangle = \sum_{\mathbf{a}} Q_i(\mathbf{a})Q_j(\mathbf{a}) \tag{1}$$

where $i,j=1,\cdots,2N$. We seek to find coordinates, $\mathbf{x}_j$ for each of the Q_j's, such that the inner product computed based on the coordinates matches the inner product computed from the Q_j functions, in other words, $\mathbf{x}_i^T\mathbf{x}_j = s_{ij}$. Expressing this in matrix notation, we need to find a matrix X, whose columns are the coordinates, such that

$$S = X^T X \tag{2}$$

This we find by performing a Singular Value Decomposition(SVD) of the square matrix S and choosing X to be the eigenvectors, U, scaled by the square root of the eigenvalues, Λ.

$$S = U\Lambda U^T = X^T X \tag{3}$$

The coordinates are given by

$$X = \sqrt{\Lambda}U^T \tag{4}$$

The dimension of embedding space can be determined by considering the eigenvalues associated with each dimension. We can retain the number of dimensions, p, in order to keep a certain percentage of the energy, typically 99.5%. The solution given by Equations 3 and 4 is least squares in nature. Hence retaining a higher number of dimensions results in a better approximation of the new coordinates in the embedding space.

It may be noted that the dot product matrix, S, is a symmetric matrix, and as we show here, it is also a positive definite matrix. Let $S=R^TR$ where $R=[Q_1,\cdots,Q_{2N}]$ with each of Q_j's being one column. For any nonzero vector $\mathbf{y}$, $(R\mathbf{y})^T(R\mathbf{y})>0$, i.e. $\mathbf{y}^TS\mathbf{y}>0$, and hence S is a positive definite matrix [53]. So, the eigenvalues are always greater than zero. As an illustration to the distance-based embedding, Figure 3(a) shows the embedding of three $1D$ probability functions for the symmetric KL distance measure. Figure 3(b) shows the embedding of a new probability

TABLE I

PROBABILISTIC DISTANCE MEASURES BETWEEN TWO PROBABILITY DISTRIBUTIONS ALONG WITH THEIR EXPRESSIONS IN TERMS OF INNER PRODUCTS (LAST COLUMN) BETWEEN TRANSFORMED VERSIONS OF THE PROBABILITY FUNCTIONS. THE TRANSFORMATIONS ARE REPRESENTED BY $Q_j = f_1(P_j)$ AND $Q_{N+j} = f_2(P_j)$ WHERE N REPRESENTS THE NUMBER OF TRAINING RELATIONAL DISTRIBUTIONS USED TO CONSTRUCT THE CONFIGURATION SPACE.

	$D(P_1(\mathbf{a}), P_2(\mathbf{a}))$	$f_1(P_j)$	$f_2(P_j)$	$D(P_1, P_2)$	$D(P_1, P_2)$
Chernoff distance [47]	$-\log\sum_{\mathbf{a}} P_1^{\alpha_1}(\mathbf{a})P_2^{\alpha_2}(\mathbf{a})$	$(P_j)^{\alpha_1}$	$(P_j)^{\alpha_2}$	$-\log\langle f_1(P_1), f_2(P_2)\rangle$	$-\log\langle Q_1, Q_{N+2}\rangle$
Bhattacharya distance [44]	$-\log\sum_{\mathbf{a}} P_1^{1/2}(\mathbf{a})P_2^{1/2}(\mathbf{a})$	$(P_j)^{1/2}$	$(P_j)^{1/2}$	$-\log\langle f_1(P_1), f_2(P_2)\rangle$	$-\log\langle Q_1, Q_{N+2}\rangle$
Matusita distance [45]	$\sum_{\mathbf{a}}(P_1^{1/2}(\mathbf{a}) - P_2^{1/2}(\mathbf{a}))^2$	$(P_j)^{1/2}$	$(P_j)^{1/2}$	$\langle f_1(P_1), f_1(P_1)\rangle -$ $2\langle f_1(P_1), f_2(P_2)\rangle +$ $\langle f_2(P_2), f_2(P_2)\rangle$	$\langle Q_1, Q_1\rangle -$ $2\langle Q_1, Q_{N+2}\rangle +$ $\langle Q_{N+2}, Q_{N+2}\rangle$
KL divergence [46]	$\sum_{\mathbf{a}} P_1(\mathbf{a})\log\frac{P_1(\mathbf{a})}{P_2(\mathbf{a})}$	(P_j)	$\log(P_j)$	$\langle f_1(P_1), f_2(P_1)\rangle -$ $\langle f_1(P_1), f_2(P_2)\rangle$	$\langle Q_1, Q_{N+1}\rangle -$ $\langle Q_1, Q_{N+2}\rangle$
Symmetric KL [46]	$\sum_{\mathbf{a}}(P_1(\mathbf{a}) - P_2(\mathbf{a}))\log\frac{P_1(\mathbf{a})}{P_2(\mathbf{a})}$	(P_j)	$\log(P_j)$	$\langle f_1(P_1), f_2(P_1)\rangle -$ $\langle f_1(P_1), f_2(P_2)\rangle -$ $\langle f_1(P_2), f_2(P_1)\rangle +$ $\langle f_1(P_2), f_2(P_2)\rangle$	$\langle Q_1, Q_{N+1}\rangle -$ $\langle Q_1, Q_{N+2}\rangle -$ $\langle Q_2, Q_{N+1}\rangle +$ $\langle Q_2, Q_{N+2}\rangle$

function onto the low-dimensional space, which we discuss next in section III-A.

For a particular distance measure, if greater number of dimensions are retained (i.e. p is increased), then the final recognition accuracy increases. But different distance measures have different values of p for the same percentage of energy retained. It depends on the nature of the distance measure. For example, with Symmetric KL and KL, f_1 and f_2 transformations result in two very far off sets of points in R^n, and hence the same percentage of energy is retained in a fewer number of dimensions than that with Bhattacharya, Matusita or Chernoff distances where the two sets are either the same or closer to each other. Thus in order to achieve similar discrimination power, we need to keep a higher percentage of energy for the Symmetric KL and KL measures than the other three measures. We later present results for various distance measures by varying the percentage of energy retained from 99.1% to 99.9% (Figure 9).

A. Embedding New Samples

Typically, the configuration space is constructed only once for the model (training) set of sequences, during the "training" phase. At runtime, relational distributions of the new sequences are embedded in this configuration space to compute distances to models. However, it may be noted that unlike other dimensionality reduction mechanisms such as PCA, the dimensions of the configuration space do not have explicit representations. This means that we cannot embed new points into this space by projecting them onto the axes. In order to find the coordinates of a new point, we have to rely on distances of the new point from the previously embedded points. This is not a new problem. Such embedding of new points is known as "out of sample" embedding. However, our particular instance of the problem is novel. For every new relational distribution, $P_z(\mathbf{a})$, we have to embed two points, $f_1(P_z(\mathbf{a}))$ and $f_2(P_z(\mathbf{a}))$, such that the dot product distances of these two points from the currently embedded points are preserved along with the constraint of the dot product distance between them. This implies that we cannot treat these two embeddings independently. We outline a coupled, iterative solution for this process.

Let $\mathbf{x_i}$ denote the coordinate of the i-th embedded point, $Q_i(\mathbf{a})$. Recall that each $Q_i(\mathbf{a})$ is either $f_1(P_j(\mathbf{a}))$ or $f_2(P_j(\mathbf{a}))$; the f_1 and f_2 transformed versions of some relational distribution $P_j(\mathbf{a})$. For any new relational distribution, we would like to find the coordinates, $\mathbf{z_1}$ and $\mathbf{z_2}$ for $f_1(P_z(\mathbf{a}))$ and $f_2(P_z(\mathbf{a}))$ respectively, such that the inner products computed based on the embedded coordinates match the inner products computed from the raw functions. In other words,

$$\begin{bmatrix} \mathbf{x_1}(1) & \cdots & \mathbf{x_1}(p) \\ \vdots & \vdots & \vdots \\ \mathbf{x_{N_a}}(1) & \cdots & \mathbf{x_{N_a}}(p) \end{bmatrix} \begin{bmatrix} \mathbf{z_1}(1) \\ \vdots \\ \mathbf{z_1}(p) \end{bmatrix} = \begin{bmatrix} \langle f_1(P_z(\mathbf{a})), Q_1(\mathbf{a})\rangle \\ \vdots \\ \langle f_1(P_z(\mathbf{a})), Q_{N_a}(\mathbf{a})\rangle \end{bmatrix} \tag{5}$$

and

$$\begin{bmatrix} \mathbf{x_1}(1) & \cdots & \mathbf{x_1}(p) \\ \vdots & \vdots & \vdots \\ \mathbf{x_{N_a}}(1) & \cdots & \mathbf{x_{N_a}}(p) \end{bmatrix} \begin{bmatrix} \mathbf{z_2}(1) \\ \vdots \\ \mathbf{z_2}(p) \end{bmatrix} = \begin{bmatrix} \langle f_2(P_z(\mathbf{a})), Q_1(\mathbf{a})\rangle \\ \vdots \\ \langle f_2(P_z(\mathbf{a})), Q_{N_a}(\mathbf{a})\rangle \end{bmatrix} \tag{6}$$

and

$$\mathbf{z_1}^T\mathbf{z_2} = \langle f_1(P_z(\mathbf{a})), f_2(P_z(\mathbf{a}))\rangle \tag{7}$$

Equations 5 and 6 represent the distance constraint to N_a of the already embedded points. We will refer these N_a points as the *reference points*. Equation 7 is the constraint related to two newly embedded points. These equations can be expressed in matrix form as:

$$\begin{aligned} \mathbf{A}\mathbf{z_1} &= \mathbf{c_1} \\ \mathbf{A}\mathbf{z_2} &= \mathbf{c_2} \\ \mathbf{z_2^T}\mathbf{z_1} &= c_{12} \end{aligned} \tag{8}$$

where $\mathbf{A}$ represents the left most matrix in Eq. 5 or Eq. 6, $\mathbf{c_1}$ and $\mathbf{c_2}$ represent the righthand side matrices in Eqs. 5 and 6 respectively, and c_{12} represents the righthand side value in Eq. 7.

We will choose the embedding points to optimize the following criteria:

$$\{\mathbf{z_1}, \mathbf{z_2}\}_{opt} \quad = \quad \arg\min_{\mathbf{z_1},\mathbf{z_2}} E(\mathbf{z_1}, \mathbf{z_2}) \tag{9}$$

where

$$E(\mathbf{z_1}, \mathbf{z_2}) \quad = \quad ||\mathbf{A}\mathbf{z_1} - \mathbf{c_1}||^2 + ||\mathbf{A}\mathbf{z_2} - \mathbf{c_2}||^2 + ||\mathbf{z_2^T}\mathbf{z_1} - c_{12}||^2$$

This minimization has to be solved using an iterative alternating method [54]. Using the fact that

$$\min_{\mathbf{z_1},\mathbf{z_2}} E(\mathbf{z_1}, \mathbf{z_2}) = \min_{\mathbf{z_2}} \min_{\mathbf{z_1}} E(\mathbf{z_1}, \mathbf{z_2}) \tag{10}$$

we can construct an iterative scheme for solving the above minimization, where at each step we first minimize with respect

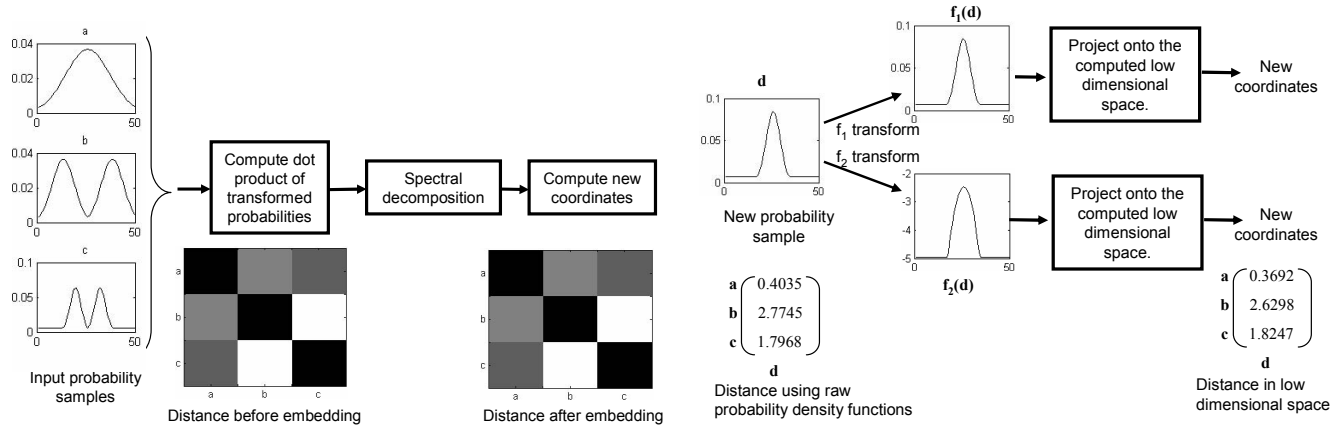


Fig. 3. Embedding probability density functions. (a) Three $1D$ probability density functions, **a**, **b** and **c**, each being 50 dimensional, are embedded into a 6-dimensional space. The Symmetric KL distance between them, both before and after the dimensionality reduction are shown as images. It should be noted that all the pairwise distances are preserved in the embedding. (b) shows a new probability density function, **d**. Two transformed functions of d, $f_1(d)$ and $f_2(d)$, are computed, which are embedded onto the 6-dimensional space. It should be noted that the distance of d from the existing functions **a**, **b**, **c** in the low-dimensional space are very close to those in the original space before embedding.

to $\mathbf{z_1}$ and then with respect to $\mathbf{z_2}$. Thus,

$$\begin{aligned} \mathbf{z_1}^{t+1} &= \arg\min_{\mathbf{z_1}} E(\mathbf{z_1}, \mathbf{z_2}^t) \\ \mathbf{z_2}^{t+1} &= \arg\min_{\mathbf{z_2}} E(\mathbf{z_1}^{t+1}, \mathbf{z_2}) \end{aligned} \tag{11}$$

where we used the superscript t to denote the t-th iteration. Since $E(\mathbf{z_1}^{t+1}, \mathbf{z_2}^{t+1}) \leq E(\mathbf{z_1}^{t+1}, \mathbf{z_2}^t) \leq E(\mathbf{z_1}^t, \mathbf{z_2}^t)$ and $E(\mathbf{z_1}, \mathbf{z_2}) \geq 0$ (i.e. bounded from below), we will produce a monotonically non-increasing set of estimates. Given the local nature of this alternating mechanism, the converging point could either be a minimum or a saddle point. To solve each of the alternating minimizations, $\min_{\mathbf{z_1}} E(\mathbf{z_1}^t, \mathbf{z_2}^t)$ and $\min_{\mathbf{z_2}} E(\mathbf{z_1}^{t+1}, \mathbf{z_2}^t)$, we use the fact that at the minimum point the derivatives, $\frac{\partial}{\partial \mathbf{z_1}} E(\mathbf{z_1}, \mathbf{z_2}^t)$ and $\frac{\partial}{\partial \mathbf{z_2}} E(\mathbf{z_1}^{t+1}, \mathbf{z_2})$, have to be zero. These conditions translate to

$$\begin{aligned} \frac{\partial}{\partial \mathbf{z_1}} E(\mathbf{z_1}, \mathbf{z_2}^t) &= \frac{\partial}{\partial \mathbf{z_1}} \left((\mathbf{A}\mathbf{z_1} - \mathbf{c_1})^T (\mathbf{A}\mathbf{z_1} - \mathbf{c_1}) + (\mathbf{z_2^T}\mathbf{z_1} - c_{12})^2 \right) \\ &= 2\mathbf{z_1^T}\mathbf{A^T}\mathbf{A} - 2\mathbf{c_1^T}\mathbf{A} + 2\mathbf{z_1^T}\mathbf{z_2}\mathbf{z_2^T} - 2\mathbf{z_2^T}c_{12} \\ &= 0 \end{aligned} \tag{12}$$

This condition can be rewritten in a compact form by augmenting $\mathbf{A}$ with $\mathbf{z_2}$ as $\mathbf{B_2}^T = \left[\mathbf{A}^T \mathbf{z_2}\right]$ and using $\mathbf{d_1}^T = \left[\mathbf{c_1}^T c_{12}\right]$.

$$\mathbf{B_2^T}\mathbf{B_2}\mathbf{z_1} = \mathbf{B_2^T}\mathbf{d_1} \tag{13}$$

or

$$\mathbf{z_1} = \left(\mathbf{B_2^T}\mathbf{B_2}\right)^{-1} \mathbf{B_2^T}\mathbf{d_1} = \mathbf{B_2}^{\dagger}\mathbf{d_1} \tag{14}$$

where $\mathbf{B_2}^{\dagger}$ is the pseudo-inverse of $\mathbf{B_2}$. Equivalently, by considering the derivative with respect to $\mathbf{z_2}$ we have

$$\mathbf{z_2} = \left(\mathbf{B_1^T}\mathbf{B_1}\right)^{-1} \mathbf{B_1^T}\mathbf{d_2} = \mathbf{B_1}^{\dagger}\mathbf{d_2} \tag{15}$$

where $\mathbf{B_1}$ is $\mathbf{A}$ augmented with $\mathbf{z_1}$ and $\mathbf{d_2}$ is $\mathbf{c_2}$ augmented with c_{12}. Equations. 14 and 15 form the coupled equations that we solve at each iteration. The iterative equation solving process is stopped when the L_2 norms $||\mathbf{z_1}^{t+1} - \mathbf{z_1}^t||$ and $||\mathbf{z_2}^{t+1} - \mathbf{z_2}^t||$ are less than a given threshold. We used a threshold of 0.001 in all our experiments. We have found that, in practice, this process converges after a few iterations.

B. Choice of Reference Points

The choice of the number of reference points, N_a, is dependent on the dimension of the low-dimensional space p; in general, $N_a > p$. In the results section we will experiment with this number. The choice appears to be dependent on the probabilistic distance measure. We choose the reference points by uniform random sampling from the set of transformed probabilities. For Chernoff, Symmetric KL and KL distances where f_1 and f_2 are the different functions, we sample the N_a points from the set $Q_1, Q_2, \cdots, Q_{2N}$. On the other hand, for Bhattacharya and Matusita distances, where f_1 and f_2 are the same functions and the sets $Q_1, \cdots, Q_N$ and $Q_{N+1}, \cdots, Q_{2N}$ are the same, we sample the N_a reference points from the set $Q_1, Q_2, \cdots, Q_N$.

We have experimented with other strategies for choosing the reference points, such as choosing the N_a subset of points whose inter-point distance is above a chosen threshold, or constructing a subset of N_a independent sets of points in a greedy fashion. However, those methods did not give a consistently stable solution. Random selection seemed to work better. To show the stability of the random selection solution, we will show our results by repeating the experiments multiple times, each with different random set of reference points. As we later see in the results section(Fig 8), the accuracy increases and the standard deviation decreases with an increase in the number of reference points for each of the shown distance measures. Depending on the nature of the probabilistic distance, the increase is seen more for some distance measures than for others.

C. Time and Space Complexities

We analyze the time complexity of finding the distance of a new sample from the existing N samples with and without embedding. Let the dimension of each relational distribution before embedding be M. Then the time complexity of comparing a new relational distribution to all existing N relational distributions is $O(NM)$, without performing embedding. On the other hand, with embedding, the time is $O((M+N)p + cp^3)$, where p is the number of dimensions retained in the low-dimensional space and c is the number of iterations required for the convergence.

It includes the time required for embedding the new point into the low-dimensional space by forming the set of linear equations$(O(Mp))$ and solving them iteratively $(O(cp^3))$, and the time to compute distances from all other points in the p-dimensional space$(O(Np))$. In practice, c is a small integer, typically less than 10. Note that $p << M$ and as N increases the advantage of having a low-dimensional space increases. The space complexity for embedding a new relational distribution into the p-dimensional space is $O(N_a p)$, which is dominated by the step of solving the linear set of equations.

IV. Configuration Space Trajectories

Each motion sequence is represented by two trajectories, corresponding to f_1 and f_2 in the configuration space. For distances like Matusita and Bhattacharya, where f_1 and f_2 are the same, each sequence is represented by a single trajectory. The length of each trajectory is given by the number of frames in the motion sequence. We match a test sequence with a training sequence using dynamic time warping.

A. Matching

Let l_1 and l_2 represent the length of the two sequences, and $d(i,j)$ represent the probabilistic distance between the i^{th} relational distribution from the first sequence and the j^{th} relational distribution from the second sequence. The value of $d(i,j)$ is computed in terms of dot products of the coordinates of the transformed relational distributions embedded in the configuration space. Analogous expressions, as given in the Table I are used for various distance measures. Let D represent the score matrix D of size $(l_1+1)\times(l_2+1)$. The 0^{th} row and column of D are initialized to infinity, except for $D(0,0)$, which is initialized to 0. The rest of the score matrix D is completed using the following recursion:

$$D(i,j) = d(i,j) + \min\{D(i-1,j), D(i-1,j-1), D(i,j-1)\} \tag{16}$$

where $1 \le i \le l_1$ and $1 \le j \le l_2$. The dissimilarity score between the two sequences is then given by $D(l_1,l_2)$. The lower $D(l_1,l_2)$ is, the more similar the two sequences are. Each test sequence is matched to every training sequence and is finally labeled as belonging to the class of the training sequence with lowest dissimilarity score.

B. Interpolation

In many gesture-related applications, it is desirable to be able to match two gesture sequences performed at different speeds or captured at different frame rates. In some cases, the sampling rate might be very sparse, or the frames of two motion sequences performed at the same speed might not be aligned – the sequences might have an offset between them. Dynamic time warping is a commonly used technique in such situations, but its results are more accurate if the two sequences being compared are at nearby temporal scales. In this section, we describe an approach to normalize the gesture sequences with respect to varying speeds by interpolating the series of points in the configuration space. For this, we perform cubic spline interpolation in the configuration space.

Interpolation between two configurations lets us arrive at a continuous representation of the motion in terms of change in the underlying structure. For further matching purposes, the motion is then indexed by arc length along the interpolated curve. The spline interpolation is evaluated at equidistant points along the arc length. In our experiments, we used a fixed sampling distance that is equal to the mean of the L_2-norm distances between all consecutive points in all the motion sequences in the training dataset. Finer sampling can be done by using a fraction of the mean distance as the sampling distance. The resulting equidistant point series representation is the speed-normalized representation for a given motion sequence. Two nearby points along the arc length of the interpolated curve, represent a lesser amount of change in configuration than two points farther away along the curve.

Figure 4 depicts an example of the representation of motion as a smooth curve. Note that the variation in the rate of change in configuration between consecutive frames shows up in non-uniform spacing of the points(Fig 4(a)). This is overcome by interpolation(See Fig 4(b)). Furthermore, the overall shape of the curves remains similar in both the interpolated curves and the number of sampled points is also nearly the same in both of them. As we show later in Section V-E, in some cases, better recognition results can be obtained by using these speed normalized interpolated sequences as inputs to dynamic time warping, than using the raw non-interpolated points of motion trajectories. It should be noted that while interpolating along the arc length of a curve, we do not use any knowledge regarding the speed of the motion sequence. Each sequence(test or train) is first speed normalized separately by interpolating along the arc length of its motion trajectory in the configuration space. A fixed sampling distance is used for all the sequences. The interpolated curves are then matched using dynamic time warping to find a dissimilarity score between a test and a train sequence, and the test sequence is labeled as the class of the train sequence with smallest dissimilarity score. If the temporal scale variation is known, then motion can also be interpolated in the spatio-temporal space along the temporal axis. We compare results of different kinds of interpolations in Section V-E.

An added advantage of a continuous curve representation in terms of cubic splines is the ability to compute derivatives, which in our configuration space would represent either the rate of change in configuration with time, or the amount of change in configuration along the arc. It should be noted that L_2-norm is closest in nature to the Matusita distance measure. Hence, we perform all the interpolation experiments using this distance measure. Interpolation based on Chernoff, Symmetric KL and KL distances would require the adoption of a more involved strategy and adjustment of the sampling intervals to be faithful to original distances; we do not address these issues in the current paper.

V. Experiments and Analyses

We illustrate the versatility of the configuration space ideas to represent and recognize motion patterns using (i) signs from American Sign Language (ASL), (ii) two-handed gestures from human-computer interaction domain, and (iii) classification of human-human interaction sequences. Using these datasets, we thoroughly analyze the different aspects of the representation. Specifically, we considered the following questions:

1) What are the recognition rates? What is the loss in accuracy due to dimensionality reduction? What are the time savings? What are the recognition rates across subjects?

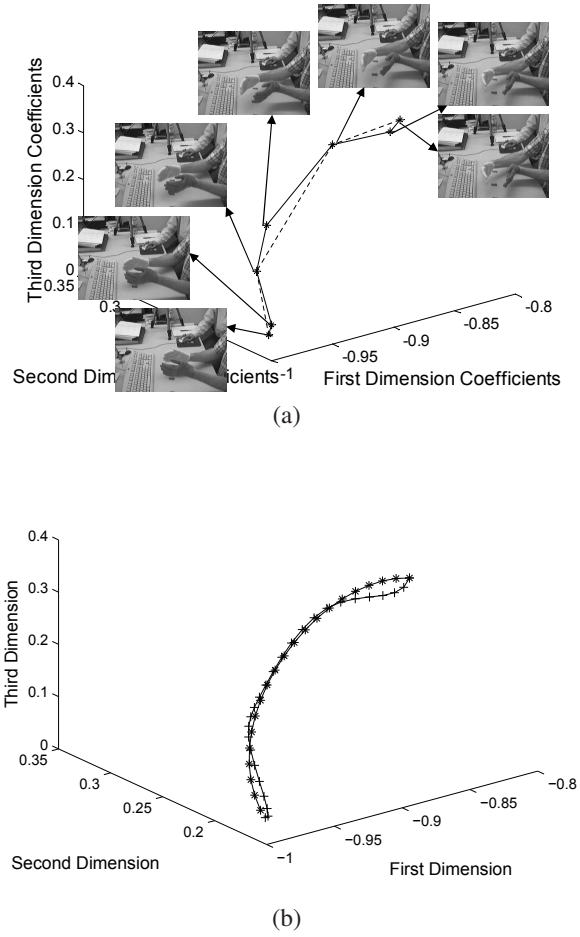


Fig. 4. Example of interpolation in configuration space. (a) First three coordinates corresponding to image frames from two sequences, each representing a part of a motion sequence. Alternate frames in the sequence are dropped in one to simulate differences in speed/frame rate. The original image frames are shown with their corresponding points. (b) Resampled sequences after cubic spline interpolation along arc length of each curve without the knowledge of the existing speed differences.

2) What is the impact of the different probabilistic measures on recognition? Is the use of the Euclidean distance between distributions sufficient?
3) What is the impact of varying the number of reference points used for embedding distributions?
4) What are the trade-offs between the extent of dimensionality reduction and accuracy?
5) How can the representation be used to match sequences at different temporal scales, both with and without the knowledge of the temporal scale parameter?
6) Is it possible to further speed up the recognition process?

For all these studies, as a performance metric we uniformly used the overall recognition rates of the signs, gestures, or actions. We adopted a task-based evaluation philosophy and decided against the use of any measure that evaluates the quality of any intermediate level representation such as the relational distribution. The accuracy reported throughout this section is the average of the accuracies obtained over 25 test runs with the reference points for the embedding process, sampled independently in each run based on uniform random sampling with replacement from the set of $Q_1, \cdots, Q_{2N}$. Along with the average we report the standard deviation. The time reported is also averaged over the 25 runs.

A. Datasets

We start by describing the datasets, the train-test splits for each, and lower level primitives used.

1) Signs from American Sign Language: This dataset is comprised of 147 different ASL signs and is unique in the large number of classes involved. Figure 5 shows one of these signs. The length of the signs varies between 2 to 33 frames, the average being approximately 10 frames, and the resolution of each frame is 490x370 pixels. There are 2 instances of each sign, one is used for training and the other for testing.

We choose Canny edges as our primitives. Skin color blobs were first extracted from the images, by using a Bayesian classifier with a histogram technique [55] for skin detection. Each pixel in a new image was classified as a skin pixel if its probability in the skin histogram was more than that in the background histogram. The skin blobs' contour pixels along with the Canny edges found inside them were selected as the low-level primitives. One might use the contour pixels alone, but in that case, the hand shape would be missed out completely when the hands cross the face.

2) Two-handed Gestures: We used a subset of the two-handed gestures described in [56] that involves variations of the same gesture across persons. It contains 7 different kinds of gestures performed by 7 different subjects. The resolution of the images is 320×240 pixels. Figure 6 shows one of the 7 types of gestures. They are push, rotate-front, rotate-back, rotate-up, rotate-down, rotate-right, and rotate-left.

We will show two kinds of experiments with this dataset. For one, we used one session of each gesture performed by each person for training and two different sessions of each gesture by each person for testing. For the other, which was an across-person recognition experiment, we used 2 sessions of each of the 7 types of gestures performed by 4 subjects for training, and 10 sessions of each of the 7 types of gestures performed by 3 other subjects for testing.

To generate the low-level features, we segmented the hand blobs using color information. Then, we used contour pixels of the two hands as the low level primitives.

3) Human-human Interaction: The third dataset we used is the Human Interaction data from CMU Graphics Lab Motion Capture (MoCap) Database [57]. We did not use the marker data, rather we used the video data of the subjects performing the interactions and moving across a complex background. The 8 different classes of human interactions that we used in our experiments are 'A pulls B, B resists', 'A sits, B pulls A', 'Chicken Dance', 'Walk together', 'Walk away from each other', 'Walk towards each other', 'One gets up after a scramble for last seat', and 'One saves other'; Fig. 7(a) and (c) show two of these interactions. Each class of interaction has two instances. We used one for training and the other for classification. The resolution of the frames is 352×240 pixels. Blobs representing human bodies were extracted based on color information and contour pixels of the blobs were used as primitives. This dataset was particularly challenging, because it was difficult to segment the subjects cleanly from the background as their dark clothing matches the darkness of some of the objects in the background as they move across the scene; therefore segmentation was not perfect. Figure 7(b) and (d) show the blob contours extracted from the frames in (a) and (c) respectively. Note the distortions in the body contours due to the presence of similarly colored objects in the background.

Fig. 5. ASL sign 'AGAIN', an example of the ASL signs dataset(best viewed in color). The complete test dataset consists of 147 different signs. For display purposes, intermediate frames have been skipped and those displayed are reduced in size.

Fig. 6. Two-handed gesture 'Rotate Left', one of the seven different types of two-handed gestures (best viewed in color). For display purposes, intermediate frames have been skipped and those displayed are reduced in size. (Source: IDIAP Research Institute, Switzerland).

(a) Raw Sequence - A Pulls B, B Resists

(b) Contour sequence - A Pulls B, B Resists

(c) Raw sequence - Walk Towards Each Other

(d) Contour sequence - Walk Towards Each Other

Fig. 7. Two of the eight different types of interactions between two humans. (a) and (c) represent the raw sequences (best viewed in color). (b) and (d) represent the contours extracted in the frames shown in (a) and (c) respectively. It should be noted that due to the presence of similarly colored objects in the background, the blobs are not perfectly segmented. For display purposes, intermediate frames have been skipped and those displayed are reduced in size. (Source: Carnegie Mellon University Graphics Lab)

B. Recognition Performance

In the first set of experiments, we studied the effectiveness of the representation in the diverse set of tasks, e.g. sign, gesture, and action recognition. For each of the three datasets, we used the five different distance measures mentioned in Table I. For the Chernoff distance, we used the values $\alpha_1 = \frac{1}{3}$ and $\alpha_2 = \frac{2}{3}$. We used PCA on the relational distributions, followed by a Euclidean distance metric as the baseline.

Depending on the probabilistic measure, each training sequence is represented by either a single sequence of coordinates (for Matusita and Bhattacharya) or two sequences of coordinates (for Chernoff, KL and Symmetric KL) in the low-dimensional space. A test sequence similarly results in either one or two sequences of coordinates, obtained by embedding the transformations of relational distributions corresponding to frames in the test sequence onto the low-dimensional space. This embedding process needs the choice of a set of embedding points to which we compute the distances. To generate recognition performance statistics, mean and standard deviation accuracy, we considered 25 possible random choices of this set for each test sequence. We matched the test and each train embedding using dynamic time warping based on the distances computed in the low-dimensional space, and classified using the nearest neighbor classifier.

Tables II, III and IV list the recognition performances on the three datasets, for various distances measures, as well as with and without the low-dimensional embedding. We can make several observations.

1) The drop in recognition rates using a low-dimensional embedding is small. This is evident when we compare recognition rates with and without any dimensionality reduction.
2) There is about 2 to 3 times speed up in performance with the low-dimensional embedding. The time reported in the tables and the figures other places is the time required to

TABLE II
RECOGNITION PERFORMANCE ON THE ASL DATASET WITH 147 DIFFERENT TEST SIGNS.

Distance Measure	Using dimensionality reduction: Accuracy% (Std. Dev.)	Time (in sec)	# of dim. (p)	Without dimensionality reduction: Accuracy (in %)	Time (in sec)
Matusita	74.6 (0.5)	0.97	50	74.8	2.69
Bhattacharya	80.3 (0.7)	0.61	50	81.0	1.30
Chernoff	75.5 (1.1)	1.05	68	76.9	1.75
Symmetric KL	77.2 (1.4)	2.33	96	78.2	2.60
KL	54.9 (3.9)	1.98	96	61.2	2.25
Euclidean(PCA)	76.2	1.11	283	76.9	2.80

TABLE III
RECOGNITION PERFORMANCE ON THE TWO-HANDED TEST GESTURE SEQUENCES BASED ON 98 TEST SEQUENCES.

Distance Measure	Using dimensionality reduction: Accuracy% (Std. Dev.)	Time (in sec)	# of dim.(p)	Without dimensionality reduction: Accuracy (in %)	Time (in sec)
Matusita	98 (1.03)	3.11	40	99	8.92
Bhattacharya	95 (0.79)	1.78	40	97	4.96
Chernoff	96 (0.66)	2.88	49	96	6.51
Symmetric KL	94 (1.27)	3.29	40	95	8.83
KL	64 (9.34)	3.04	40	86	8.24
Euclidean(PCA)	96	3.61	246	96	10.11

embed all relational distributions of a test sequence into the configuration space and to recognize the test trajectory by matching it to all the training trajectories using dynamic time warping. The time is averaged over all the test sequences. All the results on time in this paper reflect the CPU time on a 3 GHz Xeon workstation with 2 GB of memory.

3) The choice of the probabilistic distance measure matters. For the ASL dataset, the Bhattacharya distance gives the best result, while for the two-handed gestures dataset, the Matusita distance measure gives the best result. At this point, we do not have any theoretical reasons for explaining the difference in performances of different probabilistic distance measures on different datasets. It requires further investigation on the relation between the nature of the datasets and various distance measures. The KL measure results in the poorest performance, which could be due to the inherently asymmetric nature of its distance between the two probabilities. The performance of the PCA+Euclidean approach is not consistently high.

TABLE IV
RECOGNITION PERFORMANCE ON THE HUMAN INTERACTION DATASET FOR 8 TEST SEQUENCES.

Distance Measure	Using dimensionality reduction: # correct	Time (in sec)	# of dim.(p)	Without dimensionality reduction: # correct	Time (in sec)
Matusita	6 (0)	8.42	55	6	17.66
Bhattacharya	6 (0)	5.45	55	6	9.30
Chernoff	7.5 (0.5)	8.55	64	7	13.63
Symmetric KL	6.6 (0.5)	7.67	53	7	20.55
KL	4.8 (0.5)	7.18	53	5	17.04
Euclidean(PCA)	6	8.78	310	6	21.95

The differences among the performances with different distance measures are accentuated when trying to match a low temporal resolution sequence to a high temporal resolution sequence. For the two-handed gesture dataset, we created test sequences that were temporally sub-sampled from the original resolution; we retained one out of every five frames. The training sequences remained at the original resolution. The recognition accuracies were 61%, 69%, 66%, 67% and 59% for the Matusita, Bhattacharya, Chernoff, Symmetric KL, and PCA+Euclidean measures, respectively. This shows the recognition accuracies spread over a wider range and that some of the probabilistic distances outperform the PCA+Euclidean approach by large margins.

We also experimented with considering only the odd frames of the training sequences and even frames of the test sequences, using the Matusita distance measure and the gesture dataset. The number of dimensions and the number of reference points used were same as those of the experiment in Table III for the Matusita measure. The average accuracy obtained over 25 test runs was 94%. This shows the robustness of our approach.

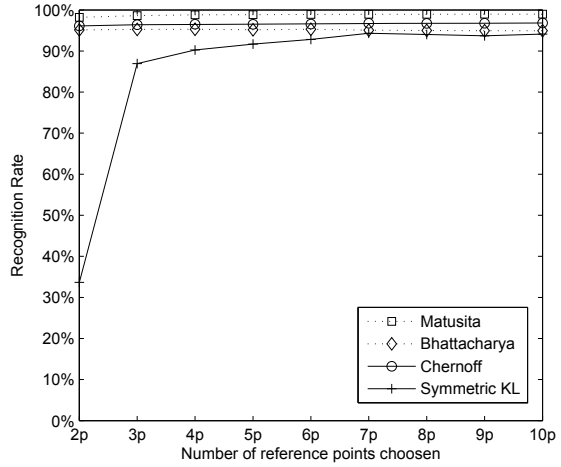
4) The representation also holds promise for recognition across subjects. From the two-handed gesture dataset, we used 2 sessions of each of 7 types of gestures performed by 4 subjects for training, and 10 sessions of each of the 7 types of gestures performed by 3 other subjects for testing. The subjects used for testing were not used for training. Altogether, we used 56 training sequences and 210 test sequences. We obtained an average recognition accuracy of 73% with a standard deviation of 1.1% across the 25 test runs. This is based on the Matusita distance measure, which seems to be the best measure to use for this dataset.

C. Choice of number of reference points

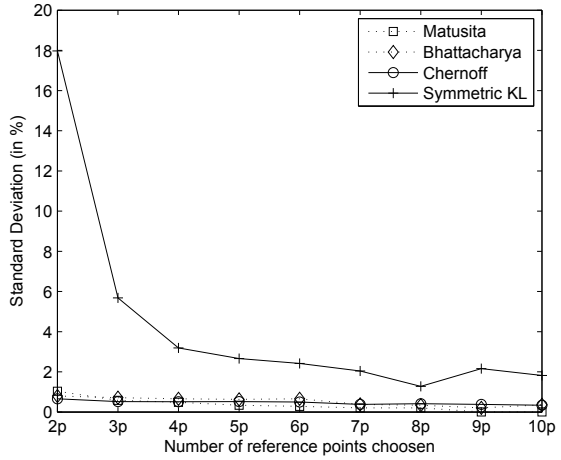
How does the choice of the number of reference points for embedding effect the quality of recognition? Figure 8 shows the results of an experiment designed to address this question. For the two-handed gesture dataset, we repeated the recognition experiments with $2p$, $3p$, $\cdots$, $10p$ number of reference points, where p is the dimension of the low-dimensional embedding. We see from the plots that for Matusita, Bhattacharya, and Chernoff measures the effect on accuracy is quite small. For Symmetric KL, the accuracy increases initially with the increase in the number of reference points and then saturates. An analogous trend is seen with the standard deviation of the accuracy. As expected, the time taken to match the sequences increases with the increase in the number of reference points. However, it should be noted that the time required with even $10p$ reference points is less than the time taken without dimensionality reduction for each of the distance measures (the last column in Table III).

D. Effect of the dimensions of the embedding

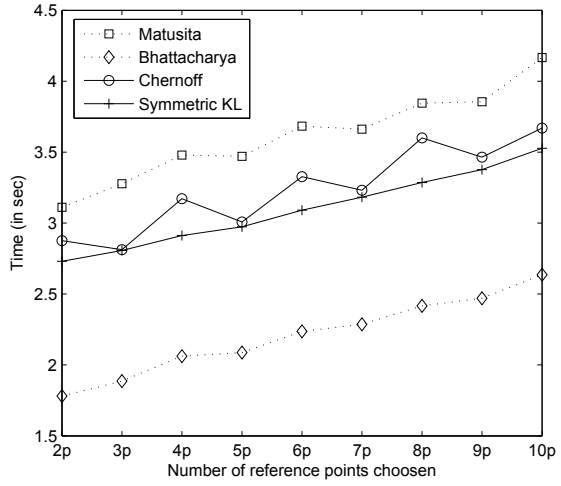
With the increase in percentage of energy retained during the dimensionality reduction, the number of dimensions and accuracy varies differently depending on the nature of the distance measure. Figure 9 shows the variations for different probabilistic measures for the two-handed gesture dataset. As can be seen, a higher percentage of energy is required for the Symmetric KL measure



(a) Accuracy



(b) Standard Deviation



(c) Time

Fig. 8. Effect of the number of reference points used during embedding. (a) shows the average accuracy obtained over 25 test runs, (b) shows the standard deviation of the accuracy, and (c) shows the average time taken to embed and recognize a given series of relational distributions.

than the other three types of measures in order to obtain the same number of dimensions or the same accuracy. The reason for this is the presence of the log transform in the Symmetric KL measure.

E. Interpolation of Motion Sequences

To demonstrate the effectiveness of the interpolation scheme based on the representation scheme, we used temporally sub-sampled test sequences of the two-handed gesture dataset. We kept one out of k frames ($k = 1,2,3,4,5$) in the test sequences and matched them with the training sequence at original scale. We performed the matching based on interpolation along the arc length of the curves in the configuration space without the knowledge of the temporal scale variation, interpolation in the spatio-temporal configuration space knowing the temporal scale variation(k), and with no interpolation. We used the same set of randomly sampled $2p$ reference points.

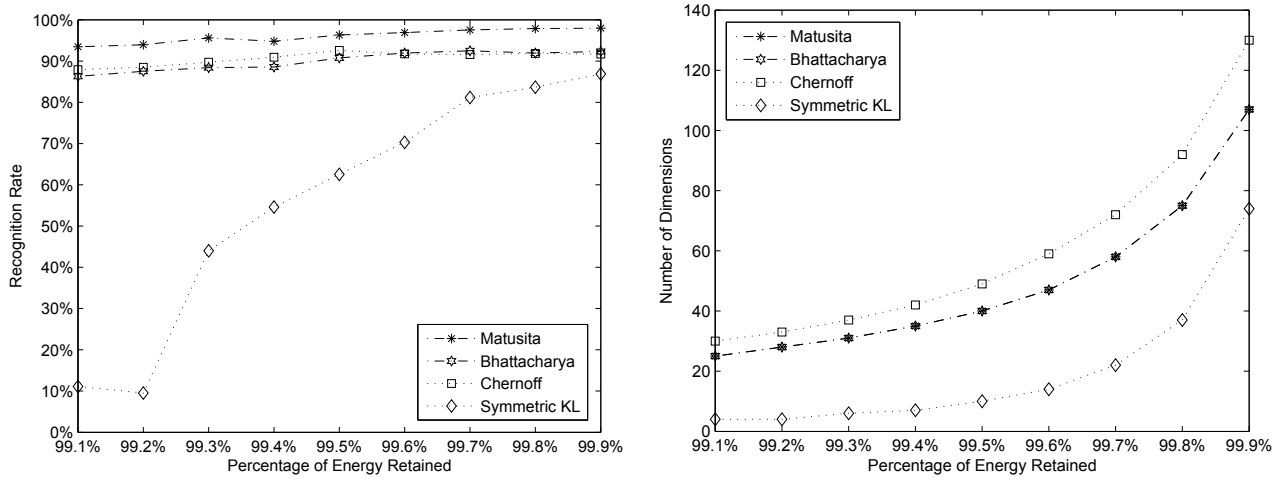
Figures 10(a) and (b) show the overall recognition results and the time required for recognition. As can be observed, interpolation in the configuration space remarkably enhances the accuracy of recognition. It should also be noted that recognition with interpolation in the reduced dimensional space is faster than that in the original space, and there is no significant decrease in recognition accuracy.

F. Time for building the relational distributions

Since the main focus of this work is a novel low-dimensional embedding scheme, which is not tied to the exact nature of the probability distribution used for representing each image frame, the time we have reported so far does not take into account the time to compute the relational distributions. The naive approach to computing relational distributions would involve an exhaustive enumeration of all pairs of edge pixels, which is computationally intensive with a time complexity of $O(n^2)$, where n is the number of edge pixels in an image. For instance, for the ASL dataset, it takes 0.665 seconds to compute the relational distribution for one image. This has been the practice in earlier uses of relational distributions in the representation of periodic human motion [21] and continuous sign language sentences [35], [58].

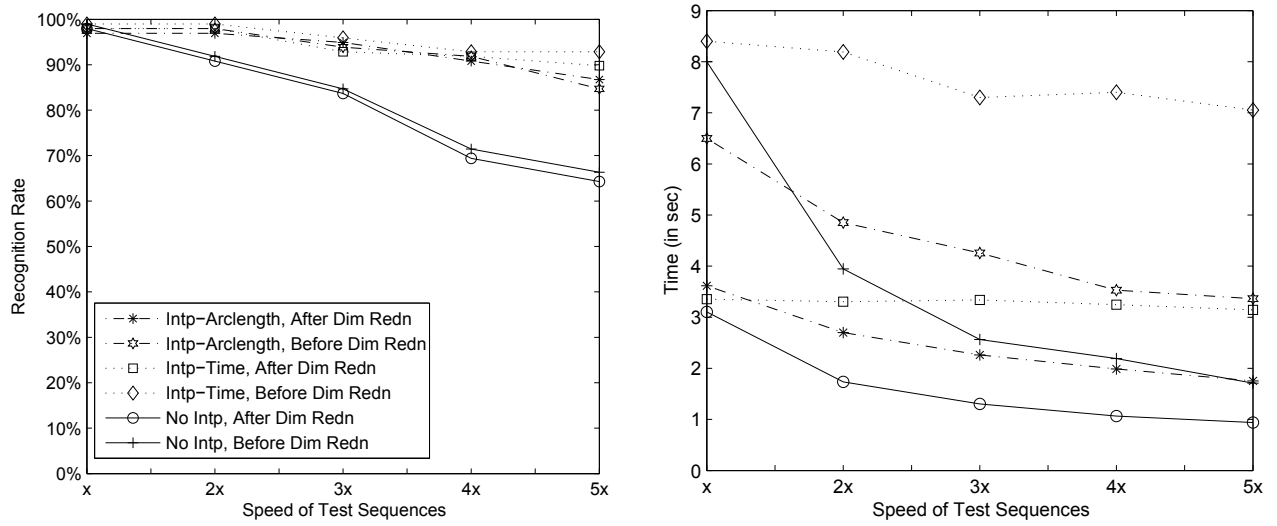
We have found that using a sampling-based method to estimate the relational distribution offers an efficient alternative. At each iteration, we sample m pairs of edge pixels with replacement, where m is directly proportional to the number of histogram bins. We repeat until the change in entropy of the distribution is small. The average complexity of computation of a relational distribution is thus reduced to $O(km)$, where k is the number of iterations for which the relational distribution is updated. From practice we have found that the number of iterations required for a relational distribution to converge, k, and the number of pairs of edge samples sampled at each iteration, m, are smaller than n.

Obviously, there is some reduction in the fidelity of representation with sampling. Does this loss impact final recognition rates? To better understand this, we studied the impact on the recognition rates on the ASL data set and the time to estimate the relational distribution with number of sampling iterations. We varied m from 1% to 5% of the total number of bins in the histogram ($51 \times 51 = 2601$ in our experiments), and k was varied from 5 to 30 iterations at an interval of 5 iterations. Using exhaustive sampling, the average time taken to compute a relational distribution for each image in the test dataset was 0.665 seconds with an overall accuracy of 79% using the Bhattacharya distance measure. Figure 11 shows the results based on the random sampling approach. As can be seen, the time to compute a single relational distribution is much less with the random



(a) Accuracy (b) Number of dimensions

Fig. 9. Effect of the percentage of energy retained with various distance measures. (a) shows the average accuracy obtained over 25 test runs, (b) shows the number of dimensions for each type of distance measure for a range of the percentage of energy retained.



(a) Accuracy (b) Time

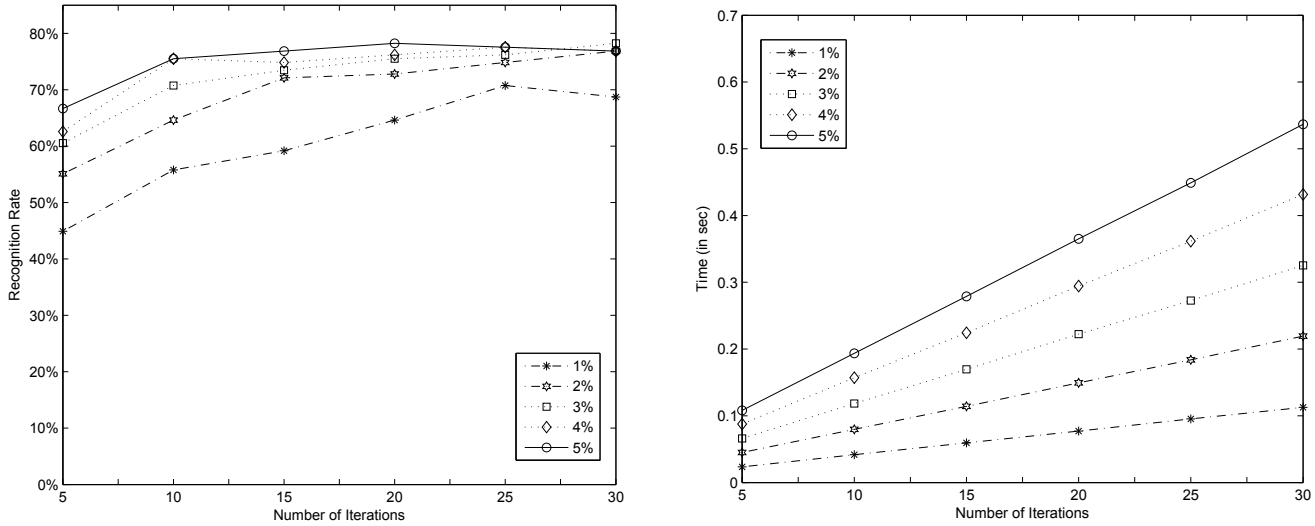
Fig. 10. Recognition results on the two-handed gesture dataset at various temporal scales (x to $5x$). Experiments include spline interpolated data along the arc length without using any temporal scale information, spline interpolated data along temporal axis using temporal scale information, and non-interpolated data. (a) shows the recognition scores (b) shows the average time taken to recognize a sequence of relational distributions. (The legend in (a) applies to (b) as well.)

sampling approach while there is a very little fall in accuracy. For example, using a sampling set size of 2% of the number of bins in a histogram, and improving the histogram over 30 iterations of sampling, takes around one-third of the time taken with exhaustive sampling. The reduction in accuracy, from 79% to 77%, is much less compared to the computational savings achieved. We used the same set of randomly sampled $2p$ reference points for the above comparisons.

Another possible way to sample could have been to sample an edge image across scale space. One could start from a higher scale edge map, and iterate to finer scales. The idea is that gross structure would be captured at high scales and finer levels would be progressively captured until there was no other significant structure to add. However, we have found our simpler sampling idea was sufficient for our experiments. For storage purposes, it should be noted that the relational distributions are sparse matrices and all existing efficient implementations for sparse matrices could be used for them. This would result in further savingsin space and computations.

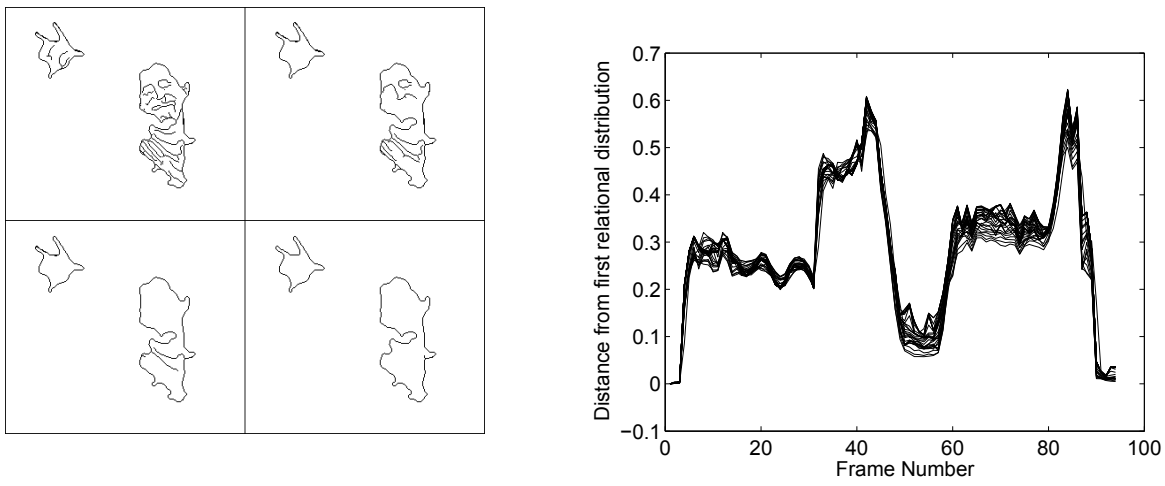
G. Robustness with Imperfect Edges

Segmentation is not the focus of the paper, but we briefly discuss how robust the relational distributions are to the edge detection parameters? In Fig. 12 we show an illustration of the stability of the representation for a motion sequence representing one ASL sentence. We compute the Bhattacharya distance between the relational distribution of each frame from that of the first frame in the sequence. We repeat this plot for each edge detection parameter, in this case it is the Canny edge high threshold, which controls the amount of "clutter" at any given scale. Different plots are obtained by varying the Canny edge high threshold value from $\frac{50}{255}$ to $\frac{200}{255}$ in intervals of $\frac{5}{255}$. If the representation is stable the plots should cluster well and the variation between the curves at each frame should be lower than the variation of each plot over the



(a) Accuracy (b) Time

Fig. 11. Effect of using sampling methods to estimate the relational distributions. For different percentages of total number number of possible samples, (a) shows the overall accuracy obtained with the ASL signs dataset using the Bhattacharya measure and (b) shows the average time taken to estimate one relational distribution.



(a) (b)

Fig. 12. Illustration of robustness of relational distributions with respect to parameters. (a) Edges extracted with different parameters in a frame from a sequence. (b) Bhattacharya distance of the relational distribution of each frame in a sequence from the first frame. Each plot corresponds to a different edge detection parameter.

sequence. The latter represents the variability of the configurations in the ASL sentence, which we expect to be high. We see from Fig. 12 that this is true, at least for this sequence.

We experimented with the impact of varying edge detection parameters on overall recognition rates. We used the Bhattacharya distance measure and ASL signs dataset for this purpose. We used the Canny edge high threshold value of $\frac{50}{255}$ for all the training sequences and varied the threshold values for the test sequences. The recognition accuracies obtained using the same set of $2p$ reference points were 76%, 77%, 80%, 78%, 67%, 64%, and 60% for the threshold values of $\frac{10}{255}$, $\frac{30}{255}$, $\frac{50}{255}$, $\frac{75}{255}$, $\frac{100}{255}$, $\frac{125}{255}$, and $\frac{150}{255}$ repsectively. This shows a gradual degradation in accuracy as the difference between the threshold values used for training and test sequences is increased.

VI. Conclusion & Future Work

We describe a framework for embedding probabilities, representing the configuration of regions of interest in an image, into a low dimensional configuration space, customized to probabilistic distance measures of choice such as the Bhattacharya, Chernoff, Matusita, KL, or Symmetric KL measures. Each new probability function is embedded into the configuration space using a small subset of training probability functions. This results in computational savings, in addition, it facilitates interpolation of motion trajectories. It might appear that our approach is similar to kernel-PCA [48]. However, it is different from kernel-PCA in two respects. First, kernel-PCA seeks to express the distances directly in terms of one inner product, however, we seek approximation in terms of functions of inner products. Thus, in our method, each input data point is mapped into two points in the space,

whereas, in kernel-PCA each data point is mapped to one point. Second, unlike kernel-PCA we use uncentered kernels to find the embedding space.

We presented results from three different contexts - ASL signs, two handed gestures, and a human interaction dataset. We have also compared and shown that recognition in the proposed configuration space gives better results than that in a low-dimensional space obtained by performing conventional PCA on the relational distributions. Also, recognizing a motion sequence that is represented by a sequence of probabilities is faster with our dimensionality reduction approach than recognition in the original space. We have shown that interpolating the trajectories along their arc lengths in the proposed configuration space, with or without any knowledge of temporal scale variations between test and train sequences, remarkably enhances the recognition results.

There are some ways we can further advance the work in this paper. The local nature of the optimization involved while embedding a new relational distribution into the configuration space could be somewhat mitigated by using multiple initializations and then choosing the most frequently occurring solution as the final solution. Next, the relational distributions used are only translation invariant and not rotation or scale invariant. While rotation invariance is often not desirable in gesture recognition, scale invariance is an important issue. We do not address this in the current paper, but one way to accomplish this could be through normalization of the relational data by the standard deviation of the attributes prior to computing the histograms. Also, we used a fixed size of the histograms for all the three datasets. Optimization of the histogram size depending on the dataset might further improve the results. Another important issue that we plan to address in future work is the handling of outliers. In the cases of extremely bad segmentation, e.g., when a relevant body part is missed completely in some frames, then the corresponding relational distributions do not indicate the correct flow of motion; however, it might still be possible to recognize a test sequence if the frames of the training sequences are correctly segmented and segmentation of a large majority of the frames in the test sequence is correct. The outlier frames need to be detected and removed and the motion needs to be interpolated. We plan to explore this possibility further in our future work.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their highly useful comments and reviews. We thank Sameer Agarwal for his insightful discussion on this paper. This work was supported by funding from NSF ITR Grant No. IIS 0312993. The human interaction dataset used in this project was obtained from mocap.cs.cmu.edu that was created with funding from NSF EIA-0196217.

REFERENCES

[1] T. Moeslund, A. Hilton, and V. Krger, "A survey of advances in vision-based human motion capture and analysis," *Computer Vision and Image Understanding*, vol. 103, no. 2-3, pp. 90–126, 2006.

[2] D. Gavrila, "The visual analysis of human movement: A survey," *Computer Vision and Image Understanding*, vol. 73, pp. 82–98, 1999.

[3] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 428–440, 1999.

[4] T. B. Moeslund and E. Granum, "A survey of computer vision-based human motion capture," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 231–268, 2001.

[5] C. Wang, W. Gao, and S. Shan, "An approach based on phonemes to large vocabulary Chinese Sign Language recognition," *International Conference on Automatic Face and Gesture Recognition*, pp. 393–398, 2002.

[6] J. Hernandez-Rebollar, N. Kyriakopoulos, and R. Lindeman, "A new instrumented approach for translating American Sign Language into sound and text," *International Conference on Automatic Face and Gesture Recognition*, pp. 547–552, 2004.

[7] G. Johansson, "Visual perception of biological motion and a model for its analysis," *Perception and Psychophysics*, vol. 73, no. 2, pp. 201–211, 1973.

[8] Z. Zhang and N. F. Troje, "3d periodic human motion reconstruction from 2d motion sequences," *Neural Comp.*, vol. 19, pp. 1400–1421, 2007.

[9] L. Zelnik-Manor and M. Irani, "Statistical analysis of dynamic actions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1530–1535, 2006.

[10] A. Efros, A. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," *International Conference on Computer Vision*, vol. 2, pp. 726 – 733, 2003.

[11] S. Wang, A. Quattoni, L. P. Morency, D. Demirdjian, and T. Darrell, "Hidden conditional random fields for gesture recognition," *Computer Vision and Pattern Recognition*, vol. 2, pp. 1521 – 1527, 2006.

[12] S. Wong and R. Cipolla, "Continuous gesture recognition using a sparse bayesian classifier," in *ICPR (1)*, pp. 1084–1087, 2006.

[13] Y. Yacoob and M. J. Black, "Parameterized modeling and recognition of activities," *Computer Vision and Image Understanding*, pp. 232–247, 1999.

[14] Y. Sheikh, M. Sheikh, and M. Shah, "Exploring the space of a human action," *International Conference on Computer Vision*, pp. 144–149, 2005.

[15] N. Vaswani, A. Roy-Chowdhury, and R. Chellappa, "Shape activity: A continuous-state HMM for moving/deforming shapes with application to abnormal activity detection," *IEEE Transactions on Image Processing*, vol. 14, pp. 1603 – 1616, Oct 2005.

[16] C. Sminchisescu, A. Kanaujia, L. Zhiguo, and D. Metaxas, "Conditional models for contextual human motion recognition," *International Conference on Computer Vision*, vol. 2, pp. 1808–1815, 2005.

[17] R. Urtasun, D. Fleet, and P. Fua, "Temporal motion models for monocular and multiview 3d human body tracking," *Computer Vision and Image Understanding*, vol. 103, pp. 157–177, 2006.

[18] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *International Conference on Computer Vision*, vol. 2, pp. 1395 – 1402, 2005.

[19] A. O. Balan and M. J. Black, "An adaptive appearance model approach for model-based articulated object tracking," *Computer Vision and Pattern Recognition*, vol. 1, pp. 758–765, 2006.

[20] C. Shan, Y. Wei, T. Tan, and F. Ojardias, "Real time hand tracking by combining particle filtering and mean shift," *International Conference on Automatic Face and Gesture Recognition*, pp. 669–674, 2004.

[21] I. Vega and S. Sarkar, "Statistical motion model based on the change of feature relationships: human gait-based recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 1323–1328, October 2003.

[22] A. Veeraraghavan, A. K. Roy-Chowdhury, and R. Chellappa, "Matching shape sequences in video with applications in human movement analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1896–1909, 2005.

[23] Y. Ke, R. Sukthankar, and M. Hebert, "Event detection in cluttered videos," *International Conference on Computer Vision*, 2007.

[24] H. Cooper and R. Bowden, "Large lexicon detection of sign language," *Lecture Notes in Computer Science*, no. 4796, pp. 88–97, 2007.

[25] A. Bobick and J. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 257–267, 2001.

[26] O. Masoud and N. Papanikolopoulos, "A method for human action recognition," *Image and Vision Computing*, vol. 21, no. 8, pp. 729–743, 2003.

[27] H. Li and M. Greenspan, "Multi-scale gesture recognition from time-varying contours," *International Conference on Computer Vision*, vol. 1, pp. 236 – 243, 2005.

[28] M. Zahedi, D. Keysers, T. Deselaers, and H. Ney, "Combination of tangent distance and image distortion for appearance-based sign language recognition," *Pattern Recognition*, vol. 3663, pp. 401–408, 2005.

[29] Y. Ukrainitz and M. Irani, "Aligning sequences and actions by maximizing space-time correlations," *European Conference on Computer Vision*, vol. 3, pp. 538–550, 2006.

[30] W. Freeman and M. Roth, "Orientation histograms for hand and gesture recognition," *International Workshop on Face and Gesture Recognition*, pp. 296–301, 1995.
[31] H. Poizner, U. Bellugi, and V. Lutes-Driscoll, "Perception of American Sign Language in dynamic point-light displays," *Journal of experimental psychology: Human perception and performance*, vol. 7, no. 2, pp. 430–40, 1981.
[32] V. Tartter and S. Fischer, "Perceiving minimal distinctions in ASL under normal and point-light display conditions," *Perception and psychophysics*, vol. 32, no. 4, pp. 327–34, 1982.
[33] D. Crandall and J. Luo, "Robust color object detection using spatial-color joint probability functions," *Computer Vision and Pattern Recognition*, vol. 1, pp. 379–385, 2004.
[34] K. Mikolajczyk, C. Schmid, and A. Zisserman, "Human detection based on a probabilistic assembly of robust part detectors," *European Conference on Computer Vision*, vol. 1, pp. 69–82, 2004.
[35] S. Nayak, S. Sarkar, and B. Loeding, "Unsupervised modeling of signs embedded in continuous sentences," *CVPR Workshop on Vision for Human-Computer Interaction*, 2005.
[36] M. Yang, N. Ahuja, and M. Tabb, "Extraction of 2d motion trajectories and its application to hand gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1061–1074, 2002.
[37] A. D. Wilson and A. F. Bobick, "Parametric hidden markov models for gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 884–900, September 1999.
[38] H. Lee and J. Kim, "An HMM-based threshold model approach for gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 961–973, Oct 1999.
[39] A. Bobick and A. Wilson, "A state based approach to the representation and recognition of gesture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 1325–1337, December 1997.
[40] C. Rao, A. Yilmaz, and M. Shah, "View-invariant representation and recognition of actions," *International Journal of Computer Vision*, vol. 50, pp. 203 – 226, 2002.
[41] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 2, p. 257286, 1989.
[42] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," *International Conference on Machine Learning*, vol. 18, p. 282289, 2001.
[43] A. McCallum, D. Freitag, and F. Pereira, "Maximum entropy markov models for information extraction and segmentation," *International Conference on Machine Learning*, pp. 591–598, 2000.
[44] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bull. Calcutta Math. Soc.*, vol. 35, pp. 99–109, 1943.
[45] K. Matusita, "Decision rules based on the distance for problems of fit, two samples and estimation," *Annals of Math. Statistics*, vol. 26, pp. 631–640, 1955.
[46] T. M. Cover and J. A. Thomas, "Elements of Information Theory," 1991.
[47] H. Chernoff, "A measure of asymptotic efficiency of tests for a hypothesis based on a sum of observations," *Annals of Math. Statistics*, vol. 23, pp. 493–507, 1952.
[48] B. Scholkopf, A. J. Smola, and K.-R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, p. 12991319, 1998.
[49] T. Cox and M. Cox, *Multidimensional Scaling*. Chapman & Hall/CRC, 2001.
[50] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, 2000.
[51] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2223–2326, 2000.
[52] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," *Advances in Neural Information Processing Systems*, vol. 14, p. 585591, 2002.
[53] R. A. Horn, *Matrix Analysis*, ch. 7. Cambridge University Press, 1985.
[54] R. J. Bezdek, J. C. Hathaway, "Convergence of alternating optimization," *Neural Parallel and Scientific Computations*, pp. 351–368, 2003.
[55] S. Phung, A. Bouzerdoum, and D. Chai, "Skin segmentation using color pixel classification: analysis and comparison," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 148–154, January 2005.
[56] A. Just and S. Marcel, "Two-handed gesture recognition," IDIAP-Research Report 24, IDIAP Research Institute, Switzerland, 2005.
[57] "Human Interaction, CMU Graphics Lab Motion Capture Database," *Carnegie Mellon University*.
[58] S. Nayak, S. Sarkar, and K. Sengupta, "Modeling signs using functional data analysis," *Indian Conference on Computer Vision, Graphics and Image Processing*, 2004.

Sunita Nayak received the B.E. degree in Computer Science & Engineering from National Institute of Technology, Rourkela, India in 2001 and the M.S. degree in Computer Science from University of South Florida in 2005. Her research interests are in image enhancement, object classification, and learning and recognition of human motion sequences. She recently completed her Ph.D. at University of South Florida and is going to join Photometria Inc.

Sudeep Sarkar (Senior Member, IEEE) received the B.Tech degree in Electrical Engineering from the Indian Institute of Technology, Kanpur, in 1988. He received the M.S. and Ph.D. degrees in Electrical Engineering, on a University Presidential Fellowship, from The Ohio State University, Columbus, in 1990 and 1993, respectively. Since 1993, he has been with the Computer Science and Engineering Department at the University of South Florida, Tampa, where he is currently a Professor. His research interests include perceptual organization in single images and multiple image sequences, automated American Sign Language recognition, biometrics, gait recognition, and performance evaluation of vision systems. He is the co-author of the book"Computing Perceptual Organization in Computer Vision," published by World Scientific. He also the co-editor of the book "Perceptual Organization for Artificial Vision Systems" published by Kluwer Publishers. He is the recipient of the National Science Foundation CAREER award in 1994, the USF Teaching Incentive Program Award for undergraduate teaching excellence in 1997, the Outstanding Undergraduate Teaching Award in 1998, and the Theodore and Venette Askounes-Ashford Distinguished Scholar Award in 2004. He served on the editorial boards for the IEEE Transactions on Pattern Analysis and Machine Intelligence (1999-2003) and Pattern Analysis & Applications Journal during (2000-2001). He is currently serving on the editorial board of the Pattern Recognition journal and the IEEE Transactions on Systems, Man, and Cybernetics, Part-B.

Barbara Loeding received a double B.S degree in Communication Disorders and Psychology from the University of MInnesota, in 1975. She received the M.S. in Speech Language Pathology in 1979 (Minnesota State University Mankato) and Ph.D. degree in Special Education, while on a fellowship, from Purdue University, Indiana in 1989. Since 1989, she has been in the Department of Communication Sciences and Disorders (Tampa) and then the Department of Special Education at the University of South Florida, Lakeland where she is currently an Associate Professor. Her research interests include accessibility of technology for people with disabilities, improvement of communication between hearing and Deaf individuals through the use of automated American Sign Language recognition and automated American Sign Language synthesis systems, improvement of human-computer interfaces, usability testing, and functional measures of performance evaluation of vision systems. She is the recipient of a Johns Hopkins Award for her development of an innovative computerized program to assess interpersonal skills of people who use sign language. She has collaborated with DePaul Universitys ASL Synthesizer Project and prior to that, with Hill Abraham on development of a video-based English to Sign translation program. She has served on several editorial boards for the field of Speech Language Pathology including Associate Editor for International Society for Augmentative and Alternative Communication, presented internationally and has published numerous articles on her work.