EXTRAPOLATING LEARNED MANIFOLDS FOR HUMAN ACTIVITY RECOGNITION

Tat-Jun Chin[*], *Liang Wang*[†], *Konrad Schindler*[‡] *and David Suter*

Institute for Vision Systems Engineering
Monash University, Victoria, Australia.

ABSTRACT

The problem of human activity recognition via visual stimuli can be approached using manifold learning, since the silhouette (binary) images of a person undergoing a smooth motion can be represented as a manifold in the image space. While manifold learning methods allow the characterization of the activity manifolds, performing activity recognition requires distinguishing between manifolds. This invariably involves the extrapolation of learned activity manifolds to new silhouettes— a task that is not fully addressed in the literature. This paper investigates and compares methods for the extrapolation of learned manifolds within the context of activity recognition. Also, the problem of obtaining dense samples for learning human silhouette manifolds is addressed.

Index Terms— Human activity recognition, manifold learning, manifold extrapolation methods, dense sampling.

1. INTRODUCTION

Visual analysis of human motions aims to understand human activities or to recognize people. This strong interest is driven by a wide spectrum of promising applications such as virtual reality, video surveillance and perceptual interface. This paper concerns recognizing the *types of human activities* from image sequences. The fundamental problem of such a task is how to effectively account for spatial (e.g. body shapes, clothing) and temporal (e.g. walking speed and style) variations in the images within similar activity classes.

This paper adopts the paradigm of analyzing human motion sequences using silhouettes. Since the silhouettes (binary images) of a person undergoing a smooth motion can be represented as a manifold in the image space, the problem of human activity recognition can be solved using manifold learning methods. A few researchers have attempted to analyze human motions using manifold learning e.g. for pose estimation [1], and have obtained promising results. Nonetheless, to our best knowledge, there seems to have been no reported works on using manifold learning for *activity recognition*. While manifold learning allows the *modeling* of "activity manifolds", identifying the activity in a sequence requires *comparing* learned manifolds. This involves the extrapolation of learned manifolds to novel silhouettes (as will be described in §3.2)— a task not fully addressed in the literature.

One of our contributions lies in comparing several methods for manifold extrapolation. As mentioned before, although previous works have proven the feasibility of manifold learning for *general* human motion analysis, it is unclear which extrapolation technique performs the best for silhouettes for the objective of human activity recognition. Given an extrapolation technique, we also show how to compare activity manifolds for the purpose of classification. In addition, since many human motion databases contain only short sequences of an activity, this paper introduces a method to *interpolate* silhouettes of a human body undergoing a particular motion to produce longer and smoother sequences. This is to examine the effect of different sampling densities on performance.

2. BACKGROUND AND MOTIVATION

Traditional activity recognition methods are based on tracking, obtaining intensity or gradient based features, finding local descriptors on interest points in images etc. For a survey, refer to [2]. Human motion can also be analyzed as temporal variations of human silhouettes. It can be more advantageous to use silhouettes since silhouette extraction is relatively easy and very feasible with current techniques. In contrast, model or feature tracking is complex due to the large variability in the shape and articulation of the human body and imaging conditions. The same difficulties affect methods that use image measurements (e.g. optical flow, spatiotemporal gradients or other intensity-based features). See [2] for more details.

A recent trend is to analyze human motion using manifold learning methods, since images (i.e. not only silhouettes) of a human body undergoing a *specific* activity (e.g. walking, bending) occupy a smooth (mostly likely non-linear) manifold in the image space. Usually the objective is for tracking. For the task of *activity recognition*, however, one needs to compare activity manifolds, and this will invariably involve manifold extrapolation. A manifold extrapolation technique was proposed in [1] to find the positions of novel silhouettes

[*] Currently at the Institute for Infocomm Research, A*STAR, Singapore.
[†] Currently at the Dept of CSSE, the University of Melbourne.
[‡] Currently at the Computer Vision Laboratory, ETH Zurich.

on a previously learned manifold, but their aim was to infer 3D body poses using silhouette images, and it is unclear how their method would perform for activity recognition. For our purpose here, we are interested not only in learning and extrapolating manifolds, but also in comparing learned manifolds to distinguish between human activities.

The paper is organized as follow: §3 discusses manifold learning, extrapolation and comparison in the context of human motion analysis. §4 describes how silhouettes can be interpolated. §5 presents experimental results to support the proposed methods, and a conclusion is drawn in §6.

3. LEARNING ACTIVITY MANIFOLDS

Let the set of input vectors sampled from an underlying manifold of the input space $\mathbb{R}^m$ be given by $\mathcal{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$. Manifold learning computes the corresponding outputs (the embedding) $\mathcal{E} = \{\mathbf{e}_1, \cdots, \mathbf{e}_n\} \in \mathbb{R}^r$ in such way that the manifold structure innately present in $\mathcal{X}$ is preserved. This entails that nearby inputs should be mapped to nearby outputs, while faraway inputs should be mapped to faraway outputs [3]. Typically, $r \ll m$, and r reflects the intrinsic dimensionality of the underlying manifold. Popular methods include LLE, ISOMAP and Laplacian Eigenmaps [3].

Fig. 1(a) shows an example of applying LLE on a bending motion sequence (down then up) using the silhouette images (64×49 pixels). LLE uncovered a 2D embedding with the points forming a smooth 1D curve, indicating that the manifold structure is preserved in the embedding space.

3.1. Manifold extrapolation methods

Given a learned manifold (i.e. the embedding coordinates of the training points such as in Fig. 1(a)), we wish to find the position of a new point (not necessarily from the same manifold) in the embedding space. Several promising extrapolation techniques are surveyed here and tested for human activity recognition using silhouettes in §5.

3.1.1. Neural networks

A solution is to train multi-layer feed-forward neural networks to produce a mapping from the input space $\mathbb{R}^m$ to the embedding space $\mathbb{R}^r$ [4]. Formally, the aim is to produce a function

$$\mathbf{e} = f_{NN}(\mathbf{x}; \mathbf{W}) \ , \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^m$, $\mathbf{e} \in \mathbb{R}^r$ and $\mathbf{W}$ represents a set of weights and biases which defines the architecture and characteristics of the neural network. Given input points $\mathcal{X}$ and embedding coordinates $\mathcal{E}$ from manifold learning, *training* a neural network involves performing the following task:

$$\mathbf{W}^* = \arg\min_{\mathbf{W}} \sum_{i=1}^{n} \|\mathbf{e}_i - f_{NN}(\mathbf{x}_i; \mathbf{W})\|^2 \ . \tag{2}$$

The error-backpropagation algorithm can be used to update the weights. Also, instead of minimizing the error in (2) only, one can include the criterion of minimizing the error of a validation set to improve the generalization capability of f_{NN}. After $\mathbf{W}^*$ is obtained, f_{NN} can be freely applied to any new input point to find its embedding coordinates.

3.1.2. Generalized Radial Basis Functions (GRBF)

Another solution is to construct GRBF interpolation functions [1]. This involves estimating an interpolation function from $\mathbb{R}^r$ to every dimension (pixel) in $\mathbb{R}^m$ (a total of m functions are created). All the functions can be combined to form an *embedding* space $\mathbb{R}^r$ to *input* space $\mathbb{R}^m$ function

$$\mathbf{x} = f_{GRBF}(\, \psi(\mathbf{e}) \,;\, \mathbf{B} \,) = \mathbf{B}\psi(\mathbf{e}) \ , \tag{3}$$

where $\psi(\mathbf{e})$ involves evaluating the chosen basic functions on $\mathbf{e}$ and each cluster center, while matrix $\mathbf{B}$ defines each individual interpolant. A manifold extrapolation function can be derived by inverting f_{GRBF}, and it is shown in [1] how this can be achieved using the pseudo-inverse of matrix $\mathbf{B}$, i.e. given a novel point $\mathbf{x}^*$, the following is evaluated

$$\psi(\mathbf{e}^*) = f^{-1}_{GRBF}(\mathbf{x}^*; \mathbf{B}) = \mathbf{B}^\dagger \mathbf{x}^* \tag{4}$$

and the embedding coordinate $\mathbf{e}^*$ of $\mathbf{x}^*$ can be recovered from $\psi(\mathbf{e}^*)$ in closed form. As argued in [1], creating an extrapolation function by inverting f_{GRBF} is more practical since constructing GRBF interpolants directly from $\mathbb{R}^m$ to $\mathbb{R}^r$ would require a prohibitively large amount of samples.

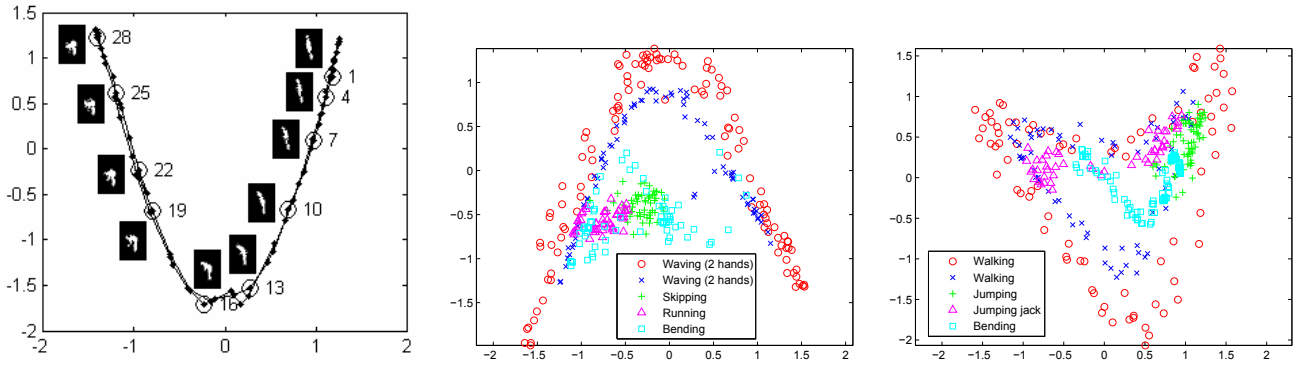
3.1.3. The Nyström estimator

Spectral embedding methods (e.g. LLE, ISOMAP, Laplacian Eigenmap) eventually arrive at a symmetric matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$ of which the eigenvectors provide the desired embedding coordinates [3]. How $\mathbf{K}$ is obtained depends on the specific algorithms. From this perspective, extrapolating a learned manifold for a new point can be achieved by extending the eigenvectors of $\mathbf{K}$. This can be achieved with the Nyström formula in conjunction with a data dependent kernel [5]. A data dependent kernel $k_n(\cdot, \cdot)$ is defined as

$$\mathbf{K}_{ij} = k_n(\mathbf{x}_i, \mathbf{x}_j) \quad \forall \ \ \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X} \ , \tag{5}$$

and evaluating $k_n(\mathbf{x}_i, \mathbf{x}_j)$ depends not only on $\mathbf{x}_i$ and $\mathbf{x}_j$ but also on the training data $\mathcal{X}$. In the limit of $n \to \infty$, the Nyström estimator for the eigenvectors approach the true underlying eigenfunction [5]. To extrapolate a new point $\mathbf{x}$, the Nyström estimator with n samples for the k-th eigenvector is

$$f_{k,n}(\mathbf{x}) = \frac{\sqrt{n}}{\lambda_k} \sum_{i=1}^{n} \nu_{ik} k_n(\mathbf{x}, \mathbf{x}_i) \ , \tag{6}$$



(a) Learned manifold for bending motion from a sequence with 65 silhouette images. Connecting lines between successive points are drawn for better visual presentation of the manifold structure.

(b) Extrapolation of learned two-handed waving manifold (red circles) to other testing sequences. All activities were done by different individuals. The same activities produced similar trajectories.

(c) Extrapolation of learned walking manifold (red circles) to other testing sequences. All activities were done by different individuals. The same activities produced similar trajectories.

Fig. 1. Learning and extrapolating activity manifolds of human silhouettes.

where λ_k is the k-th eigenvalue of $\mathbf{K}$ and ν_{ik} is the i-th element of the k-th eigenvector of $\mathbf{K}$. The actual algebraic form of $k_n(\cdot,\cdot)$ depends on the specific spectral embedding method, and it is shown in [5] how $k_n(\cdot,\cdot)$ can be defined for LLE, ISOMAP and Laplacian Eigenmap.

3.2. Comparing manifolds

The main assumption for activity recognition is that manifolds of silhouettes of the same activities lie closely in the image space, while manifolds of differing activities are far apart, provided the body shapes do not dramatically differ. Hence, given a learned manifold of a particular activity, extrapolating for a sequence containing the same activity should result in a set of embedding coordinates which give rise to a trajectory that is similar to the learned manifold. The opposite should happen for a sequence with a different activity. Figs. 1(b) and 1(c) support the validity of this assumption.

Based on the above assumption, manifold comparison can be performed by computing distances between trajectories in the embedding space while respecting differences in sequence length and temporal shifts. A variant of the Hausdorff metric, the "mean value of the minimums", can be used:

$$Diff(M_G, M_P) = \frac{1}{t_G}\sum_{i=1}^{t_G} \min_{1\le j\le t_P} \|M_G(i) - M_P(j)\| \ , \tag{7}$$

where M_G and M_P are respectively the gallery and probe sequences (the embedding coordinates), t_G and t_P are respectively the length of M_G and M_P, while $M_G(i)$ indicates the i-th point in the sequence of M_G. To make $Diff(\cdot,\cdot)$ symmetric, the following can be evaluated:

$$D(M_G, M_P) = Diff(M_G, M_P) + Diff(M_P, M_G) \ . \tag{8}$$

$D(M_G, M_P)$ is small when M_G and M_P are similar (and vice versa). It is 0 when M_G and M_P are exactly the same.

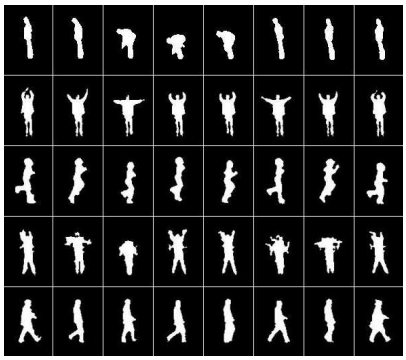
4. INTERPOLATING MOTION SILHOUETTES

Two motivations exist to upsample the training data in time. First, almost all manifold learning methods require a dense sampling of the underlying manifold. Secondly, having more samples can be beneficial for estimating or training the extrapolation function. Many publicly available video databases for activity recognition contain only short sequences.

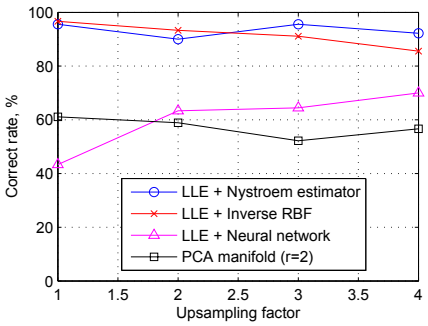
The task hence is to interpolate intermediate silhouettes between two observed ones (in the spatial domain). However, silhouettes change over time in an elastic way (the distance along the boundary from the top of the head to the left foot might be half the boundary length in one frame, but only 40% of the length in the next one). The problem is separated into two parts: first, represent the two observed silhouettes by sequences of equally spaced landmark points on their boundaries, and minimize the *non-linear elastic matching distance* between the two sequences [6], in order to establish correct pointwise correspondence between them. Second, find landmarks for the intermediate frames by linear interpolation between corresponding landmark points, and floodfill the resulting silhouette boundaries to obtain intermediate frames.

5. EXPERIMENTAL RESULTS

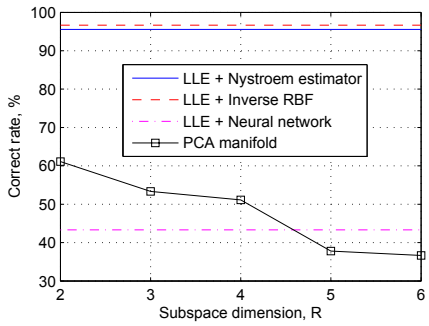
A recent database reported in [7] is used for our experiments. This database is appreciably larger than other publicly available databases, and it consists of 90 low-resolution videos (all less than 100 frames) from 9 different subjects, each performing 10 vastly different activities (e.g. bending, jumping, running, waving). All sequences were employed in our experiments, and we made use of the readily available silhouette masks. See Fig. 2(a). The leave-one (subject)-out procedure is carried out, i.e. the sequences of one subject (all activities) are retained for testing, while manifold learning (LLE) is performed on the other 80 sequences. On each learned mani-



(a) Several examples of the database. Note the variability in body shapes and silhouette quality.



(b) Classification rates using LLE manifold learning on silhouette images (with silhouette upsampling).



(c) Classification rates of varying PCA manifold dimensions (silhouette upsampling factor = 1).

Fig. 2. Experimental results.

fold, extrapolation (with the techniques described in §3.1) is carried out for each testing sequence and the trajectories are compared using (8). The testing sequence is labeled with the activity with which it has the smallest manifold distance. To examine the effect of different sampling densities on the performance, the silhouettes are upsampled at different factors.

Figs. 2(b) and 2(c) illustrate the results. It can be seen that the Nyström estimator and inverse RBF extrapolation methods gave the best performance (avg. 93% and 92%). Neural networks do not extrapolate satisfactorily hence producing inferior classification rates. It should be noted that a prior tuning of the parameters for inverse RBF and neural networks was carried out. In the experiment in Fig. 2(b), PCA was also used as a baseline method to learn a 2D linear subspace to characterize each activity manifold (one subspace per sequence). A testing sequence is projected onto each 2D subspace and manifold comparisons are evaluated using (8). This produced almost the same performance as neural networks. In addition, it can be seen from Fig. 2(b) that activity classification rates are not affected by the length of the training sequences, despite a more denser sampling of the underlying manifold provided by silhouette interpolation. In Fig. 2(c), the dimension of the PCA subspace for each training manifold was varied and the classification rates were recorded. It can be seen that the performance markedly decreases as more dimensions are used. This is probably because the variance of the sequences are confined in the first few dimensions, and increasing the subspace dimensions involve only learning unwanted noise.

6. CONCLUSIONS

Based on the empirical results, we conclude that the Nyström estimator is the best extrapolation technique (at least for human activity recognition using silhouettes). This is motivated by its good performance and simplicity (no parameter tuning is required!). In contrast, although inverse RBF performs equally well, some effort is required for model selection and parameter tuning such as number of cluster centers, types of basic functions and their parameters. Neural networks suffer from the same difficulties e.g. choosing the number of hidden layers, number of neurons and types of activation function. Finally, although it can provide a denser sampling of the underlying manifold, we find that silhouette interpolation is unnecessary for the particular database. Despite a small number of images per sequence (some as low as 36 frames!), LLE seems to perform well for activity recognition.

7. REFERENCES

[1] A. Elgammal and C.-S. Lee, "Inferring 3D body pose from silhouettes using activity manifold learning," in *CVPR*, 2004, vol. 2, pp. 681–688.

[2] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviours," *TSMC*, vol. 34, no. 3, pp. 334–352, 2004.

[3] L. K. Saul, K. Q. Weinberger, F. Sha, J. Ham, and D. D. Lee, "Spectral methods for dimensionality reduction," in *Semisupervised learning*, O. Chapelle, B. Schölkopf, and A. Zien, Eds. MIT Press: Cambridge, MA, 2006.

[4] H. Gong, C. Pan, Q. Yang, H. Lu, and S. Ma, "Neural network modeling of spectral embedding," in *British Machine Vision Conference*, 2006.

[5] Y. Bengio, O. Delalleau, N. Le Roux, J.-F. Paiement, P. Vincent, and M. Ouimet, "Learning eigenfunctions links spectral embedding and kernel PCA," *Neural Computations*, vol. 16, no. 10, pp. 2197–2219, 2004.

[6] G. Cortelazzo, G. A. Mian, G. Vezzi, and P. Zamperoni, "Trademark shapes description by string-matching techniques," *Pattern Recognition*, vol. 27, no. 8, pp. 1005–1018, 1994.

[7] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Action as space-time shapes," in *ICCV*, 2005, vol. 2, pp. 1395–1402.