The Isometric Self-Organizing Map for 3D Hand Pose Estimation

Haiying Guan, Rogerio S. Feris, and Matthew Turk
Computer Science Department
University of California, Santa Barbara
{haiying,rferis,mturk}@cs.ucsb.edu

Abstract

We propose an Isometric Self-Organizing Map (ISOSOM) method for nonlinear dimensionality reduction, which integrates a Self-Organizing Map model and an ISOMAP dimension reduction algorithm, organizing the high dimension data in a low dimension lattice structure. We apply the proposed method to the problem of appearance-based 3D hand posture estimation. As a learning stage, we use a realistic 3D hand model to generate data encoding the mapping between the hand pose space and the image feature space. The intrinsic dimension of such nonlinear mapping is learned by ISOSOM, which clusters the data into a lattice map. We perform 3D hand posture estimation on this map, showing that the ISOSOM algorithm performs better than traditional image retrieval algorithms for pose estimation. We also show that a 2.5D feature representation based on depth edges is superior to intensity edge features commonly used in previous methods.

1. Introduction

Non-intrusive hand gesture interpretation plays a crucial role in a wide range of applications, such as automatic sign language understanding, entertainment, and human computer interaction (HCI). Because hand gestures are natural, intuitive, and provide rich information to computers without extra cumbersome devices, they offer a great potential for next generation user interfaces, being especially suitable for large scale displays, 3D volumetric displays or wearable devices such as PDAs or cell phones.

In this paper, we address the particular problem of 3D hand posture estimation. Given a test hand image, our task is to estimate the hand pose, which is defined by joint angles and the viewpoint. Although substantial progress has been made on this topic over the past decades, recognizing 3D pose configurations remains a challenge, due to the following reasons:

(1) High Degrees of Freedoms (DOF). Tracking or estimating hand postures in a high dimensional space is a very challenging task.

(2) For many applications, the temporal continuity can not be exploited due to fast hand motions [1].

(3) Reliable Features. Skin color is a very important feature for extracting the hand boundary, but it is a great challenge to obtain such features reliably if the hand gesture is performed in complex backgrounds, which may contain similar skin color.

(4) Hand appearance changes dramatically with different viewpoints.

(5) The self-occlusion problem. Many hand configurations involve finger occlusions which can not be detected with conventional segmentation algorithms due to low intensity variation across skin-color regions.

Previous approaches to 3D hand pose estimation are generally classified into two categories: model-based and appearance-based approaches. Model-based methods rely on estimating the parameters of a 3D hand model to fit a given hand image [2]. Although they provide more precise hand pose estimation than appearance-based approaches, the high degrees of freedom of hand configurations impose a search in a high dimensional space, which requires good initialization and often leads to inaccurate local minima solutions. Correlation among joint angles has been exploited to reduce the complexity of the problem [2], [3], but producing good results with complex backgrounds remains an open problem.

Appearance-based, or exemplar-based approaches estimate pose by matching a hand image with a discrete set of labeled hand pose images. Depending on the application, the problem may be simplified by considering a small subset of convenient hand configurations in a particular viewpoint [4]. However, for estimating joint angles and viewpoint, a large number of exemplars need to be considered. In this case, the pose estimation problem can be formulated as a database indexing or image retrieval problem [5] [6]. A realistic 3D hand model is typically used to produce such a large set of exemplars. Due to the database size and high dimensional image data, current methods are still limited in

recognition accuracy and efficiency.

In this paper, we formulate hand pose estimation as a non-linear mapping problem between the hand pose space and the image feature space. Similar to appearance-based approaches, we use a 3D realistic hand model to generate synthetic images with associated pose ground truth, encoding a smooth, non-linear mapping in a high dimensional space. We then learn an organized structure of this mapping in a low dimensional manifold, by using a novel non-linear dimensionality technique which we call Isometric Self-Organizing Mapping (ISOSOM). Our algorithm combines Kohonen's self-organizing maps [7] with ISOMAP dimension reduction algorithm, so that samples are clustered according to their geometric distance on the manifold. Hand pose retrieval is performed on the low dimensional space learned by ISOSOM. In contrast to current appearance-based approaches, which represent each exemplar as an isolated item in the database, ISOSOM reduces the information redundancy by clustering similar vectors generated by similar poses together. This avoids the exhaustive search in the whole database, leading to better results.

The choice of features is also extremely important for the performance of hand pose estimation. Previous methods tend to rely on intensity edges, which are limited to capture important shape boundaries (due to low-contrast variation across skin-color regions). In addition, many unwanted edges due to background clutter and texture (e.g., wrinkles and nails) are detected. To handle this problem, we adopt a 2.5D hand representation based on depth edges, which are captured with a multi-flash camera [8][9].

This paper is organized as follows: in Section 2, we propose the ISOSOM algorithm for clustering and non-linear dimensionality reduction. Section 3 shows how our algorithm can be used to learn a low-dimensional mapping between hand poses and image features, in order to achieve improved hand pose estimation. Section 4 describes our 2.5D feature representation based on depth edges and a shape context descriptor. Experimental results are shown in Section 5. Finally, conclusions are given in Section 6.

2. Learning with ISOSOM

Tenenbaum's ISOMAP [10] algorithm extracts meaningful dimensions by measuring the distance between data points in the geometric shapes formed by items in a nonlinear data set. It builds a distance graph, which describes the approximate geometric distance on the samples' manifold. The algorithm estimates and preserves the global geometry to avoid the feature vectors' mixed-ups in low dimension space by classic MDS algorithm.

Kohonen's [7] Self-Organizing Map (SOM) is an unsupervised clustering algorithm for dimensionality reduction, which is an effective tool for the visualization of high dimensional data in a low dimensional (normally 2D) display. It is used to build a mapping from high dimension space to 2D visualization space by preserving the topological order of the data, while at the same time clustering similar samples. Although the input dimension of SOM could be very high, the SOM is more efficient for data samples with low intrinsic dimension.

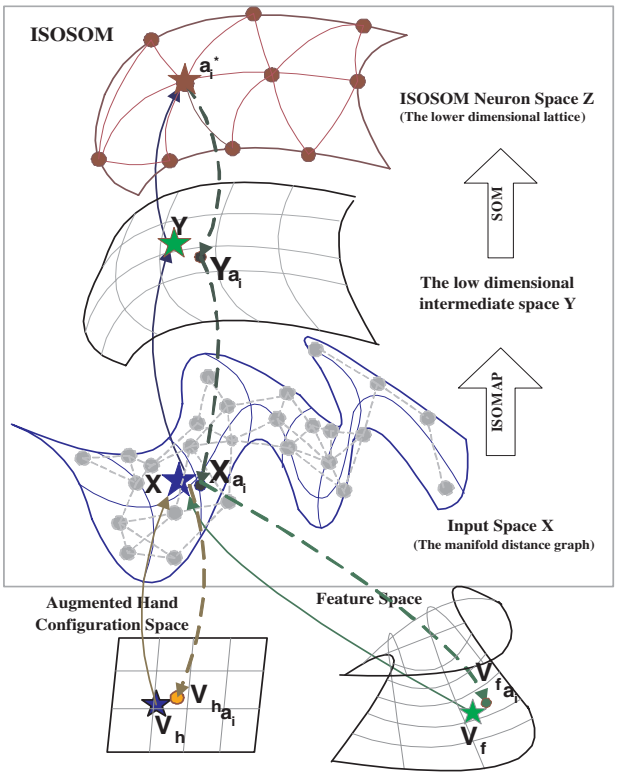


Figure 1. The Isometric Self-Organizing Map

Based on the ISOMAP and SOM algorithms, we proposed an ISOmetric Self-Organizing Mapping algorithm (ISOSOM). Figure 1 illustrates an intuitive depiction of the ISOSOM; a more detailed description of the algorithm follows below. Firstly, the data samples in the high dimensional input space X are mapped into to the low dimensional intermediate space Y by the ISOMAP algorithm (see Alg. 1, Part (I)). Then the data samples in the intermediate space Y are used as the training samples of the SOM algorithm to learn the organized structure, the ISOSOM neuron map, in the same or even lower dimension space Z (Alg. 1, Part (II)). Similar to the map structure of SOM, the ISOSOM map is formed by a set of processing units, called neurons, organized in the space Z with a lattice A. The neurons are connected with their neighbors on the lattice. Each neuron a_i is labeled by an index $i \in \{1 \dots size(A)\}$, and has reference vectors Y_{a_i} attached, which is projecting back into the input space X and associated with another vector X_{a_i} by inverse mapping from Y space to X space (Alg. 1, Part (III)). In the retrieval stage, the best matching unit (BMU) is the closest neuron on the ISOSOM map to

the query vector (Alg. 1, Part (IV)). In our hand estimation case, the input vector $X_a = [V_h, V_f]$ is formed by the hand configuration vector V_h and the feature vector V_f.

Algorithm 1 The Isometric Self-Organizing Map

(I) ISOMAP Construct a distance graph G of the manifold over all data points in input space X by adding an edge between two nodes i and j, if i is one of the k nearest neighbors of j. The cost of the edge is measured by the Euclidean distance of i and j. On the graph, the distance of any two nodes is defined by the cost of shortest path between them. The low dimensional embedding is constructed by classic MDS algorithm through the mapping from the high dimension input space X to the low dimension intermediate space Y [10].

(II) SOM The data samples in the intermediate space Y is used for the SOM training. The initial vector values associated with neurons on the ISOSOM map could be linearly interpolated along the subspace spanned by the principal eigenvectors of the data samples. In each training step, we randomly choose a sample vector y from the training samples, and the response of a neuron to the vector y is determined by the comparisons between y and the reference vector Y_{a_i} of each neuron with the geometric distance defined by the distance graph. The Best-Matching Unit (BMU) is defined as the winner neuron a_i which has its reference vector Y_{a_i} closest to the given input y. After obtaining the BMU, its prototype vectors Y_{a_i} of the ISOSOM and its topological neighbors are updated and moved closer to the input vector y [7].

(III) Inverse Mapping from Y space to X space
Because the ISOMAP algorithm is a nonlinear mapping algorithm, given a vector y in the intermediate space Y, it is hard to find its exact inverse mapping in input space X. Thus, we find the close points of y in the sample set, $y_1, ...y_k$, and approximate its inverse mapping x with those points' inverse mapping $x_1, ..., x_k$ by the distance preserving method such as the nearest neighbor or the similar method as Locally Linear Embedding(LLE). Thus, every neuron $a_i$in the ISOSOM has a vector X_{a_i} associated with it.

(IV) ISOSOM Retrieval
Given a query vector with full components or partial components with the mask w, the similar neurons are found and sorted by the similarity measurement described below:

$$Top\_N\_BMU = \underset{\forall a \in A}{arg\ top\ N\ min} Distance(w(x_a), w(x)) \qquad (1)$$

where $w(x_a)$ is the mask function defined by W representing the existing components. For example, for the input $x_a = (x_{a1}, *, x_{a3})'$, $W = (1, 0, 1)'$ (where "0" is indicated by the missing components " $*$ "), and $w(x_a) = W. * x_a = (x_{a1}, *, x_{a3})'$ (we don't care the second component).

Intrinsically, ISOSOM utilizes the geometric distance to perform the nonlinear dimension reduction. This geometric distance is defined by the metric relationship between samples and preserves the relationship of the samples in high dimension space. In the SOM learning process, this relationship is also preserved in the ISOSOM map's organized neuron structure, where similar neurons are closer to each other in the grid than dissimilar ones. Thus ISOSOM follows better the topology of the underlying data set and preserves the spatial relationships in high dimensional input space X to the low dimension ISOSOM lattice map.

3. ISOSOM for Hand Pose Estimation

We model the hand as a 3D articulated object with 21 DOF of the joint angles [2] and six DOF of global rotation and translations [1]. A hand configuration is defined by these 21 joint angle parameters. A hand pose is defined by a hand configuration augmented by the three global rotation parameters. Given a hand image, our task is to output the corresponding hand pose, that is, the hand configurations with the three global rotation parameters. The retrieval is considered successful if at least one of the candidates in the top N matches is sufficiently close to the ground truth (similar to [11]). If N is small, with additional distinguishable contextual information, it may be adequate for automatic initialization and re-initialization problems in hand tracking systems, where the correct estimation could be found and the incorrect ones could be eliminated in later tracking.

Due to occlusions, the different hand configurations could be rendered to the same image. In such cases, many hand configuration vectors could generate one hand image, so that the mapping along this direction is many-to-one (see the top-down arrows in Figure 2). On the other hand, the same hand configuration can be rendered from different viewpoints and thus generates many images. The mapping along this direction is one-to-many (see the bottom-up arrows in Figure 2). Intrinsically, the mapping between the hand configuration vectors and the image vectors is a many-to-many mapping. To simplify the problem, we eliminate the second, one-to-many case by augmenting the hand configuration vector with the three global rotation parameters to construct the augmented vector as hand pose vector. The hand pose vector determines the hand image and the feature vector representing the image. The mapping between the hand pose space and image space (or feature space) is thus a many-to-one mapping.

3.1. Pose Estimation as a Non-Linear Mapping

The objective is to learn the many-to-one, nonlinear, continuous mapping between the feature space (input) and the hand pose space (output). Such high dimensional mapping is encoded by training samples generated by a realistic 3D hand model. Each sample corresponds to a feature vector with the associated pose, obtained from the model with a particular hand configuration, and rendered at a particular viewpoint. It is a typical supervised learning problem.

The challenge is that the feature vector of different poses is highly mixed up in the feature space. Figure 3 illustrates two examples in which the different poses look similar in the second viewpoint. The second row of Figure 3 shows that two different poses have similar appearances in the particular viewpoint. It indicates that even though two feature vectors are similar or two hand images are similar, their hand configuration might be quite different in hand con-

[1]The translation parameters could be estimated by hand segmentation algorithms or neglected if the translation and scale invariant features are adopted.

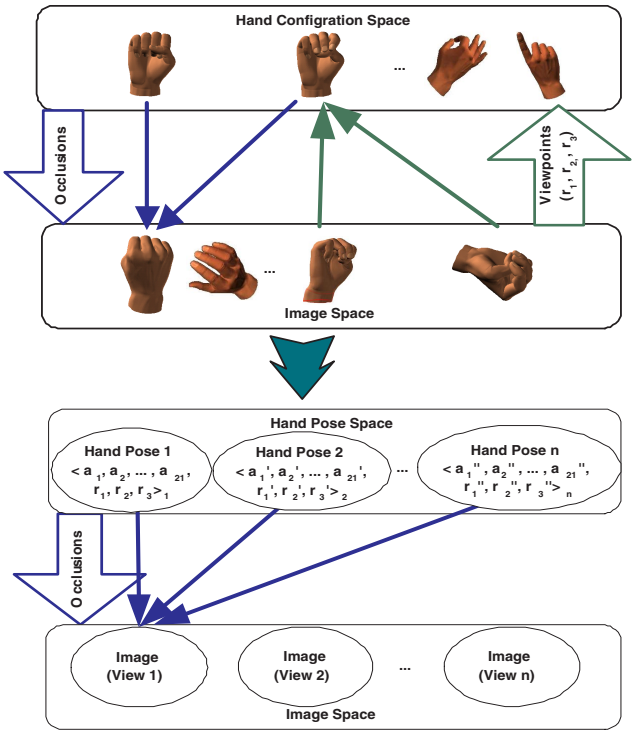
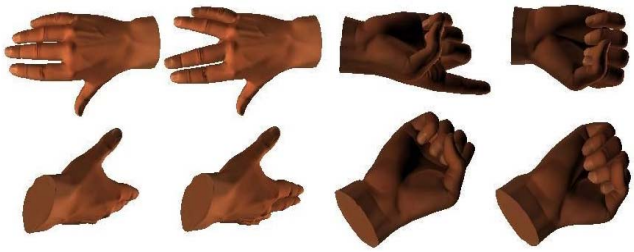


Figure 2. High dimension nonlinear mapping



(a) Open Hand (b) Greetings (c) Pointer (d) Fist

Figure 3. The similar appearances (second row) of two sets of different hand poses

figuration space. If we just cluster the features in the feature space with traditional methods, it is difficult to retrieve the corresponding hand pose vector in the hand pose space. We thus convert the original supervised learning problem to an unsupervised learning problem by constructing the large vector, $x = [V_h, V_f]$, which consists of both feature vectors V_f and their corresponding hand pose vectors V_h. The bottom part of Figure 1 gives a description of such operation. In the ISOSOM input space X, the similar feature vectors with the different poses are separated by hand pose vector and not be mixed-up anymore. Then we train the ISOSOM to encode the nonlinear mapping between feature space and the hand configuration space.

During the retrieval, the feature vector is calculated for a given input hand image. Using a mask to handle the missing hand pose components, the similarities of the feature vector with all feature vectors associated with the ISOSOM neurons are measured. The top N hand pose candidates associated with the neurons are retrieved and the confidences are measured by the error measurement. Because the mapping from the feature space to the hand pose space is a one-to-many mapping, one feature vector could have several possible hand pose candidates, which is desirable since it reflects the intrinsic nature of the mapping.

4. Depth Edges and Shape Context Descriptor

The performance of a pose estimation system depends heavily on the choice of features. Hand shape descriptors (such as shape context or Fourier descriptors) have been commonly used in current systems, rather than color or texture features, which are not very distinctive for different poses and thus have limited discriminability ability power.

A key limitation of current approaches is hand shape acquisition. Most techniques are based on intensity edge detectors (using, for example, the Canny operator), which fail to capture important shape boundaries along the hand due to low intensity variation across skin-color regions. In addition, intensity edges include many unwanted edges due to background clutter and texture (such as wrinkles and nails).

Rather than using intensity edge detectors for hand shape acquisition, we adopt a 2.5D hand representation, detecting edges only at depth discontinuities. A natural way to detect depth edges is to first reconstruct the scene (using stereo algorithms) and then look for discontinuities. However, 3D estimation algorithms are limited to produce accurate results exactly at depth discontinuities, due to occlusions and violation of smoothness constraints.

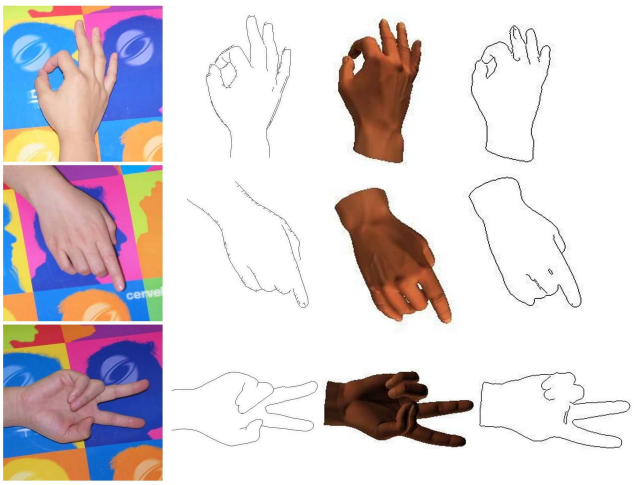
We use the recent method of Raskar et al. [8] to bypass scene reconstruction and detect depth edges directly. We rely on a multi-flash camera with flashes strategically positioned to cast shadows along depth discontinuities. This allow us to reliably acquire hand shape (including internal finger edges), while considerably reducing background clutter.

Feris et al. [9] demonstrated the important use of depth edges on the problem of fingerspelling recognition. They show that that depth discontinuities may be used as a signature to reliably discriminate among complex hand configurations in the American Sign Language alphabet. Figure 4 compares hand shape acquisition based on depth edges and intensity edges. In Figure 5, we show the comparison of our detected hand depth edges and the rendered 3D model edges, which are very similar.

Shape context [12], a translation invariant shape descriptor, gives the one-to-one correspondence for the sample points in two shapes, provides the transformations from one shape to another, and measures the similarity between shapes. We adopt the variant version of shape context,

(a) Original image (b) Canny (Thre. =1) (c) Canny (Thre.=0.3) (e) Depth edges

Figure 4. Canny edge and Depth edge



(a) Original image (b) Depth edges (c) Synthesis model (d) Synthesis edge

Figure 5. Real images and their models

which is described in [9]. It is modified for scale invariance, and uses a 256-dimension feature vector to represent the shape.

5. Experimental Results

Our synthesis database contains 15 commonly used hand gestures. These 15 gestures are defined at the semantic level. For each hand gesture, 16 hand configurations, which are created by adding a small turbulence to the basic hand configuration, are rended in 81 viewpoints sampled uniformly from the surface of the 3D view sphere, which represents the global rotation of the hand with respect to the camera. Overall, we generate a database containing 19440 synthetic images. For each hand configuration, 48 joint angle parameters are saved as the joint angle vectors, which are 3 rotation parameters for the hand, 9 parameters (3 rotation parameters for 3 joints respectively) for each finger and thumb. In addition to the 3 global rotation parameters of the camera, the hand pose vector is composed of these 51 parameters.

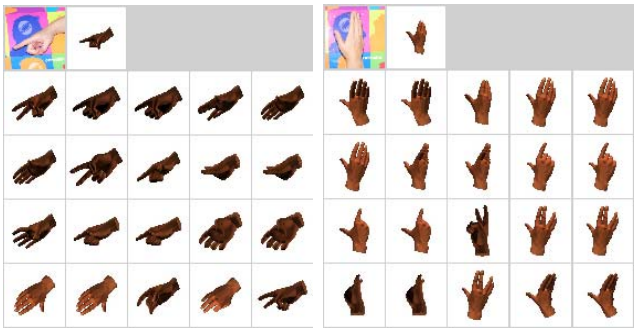
We collect a real hand image dataset with the similar 15 hand gestures in 10 different viewpoints, which are approximately sampled uniformly from the surface of the 3D view sphere. We capture $1 \sim 2$ sets of images for each pose. The real hand image database totally contains 226 cases. We label the pseudo-ground truth for each hand image by manually identifying the similar images in the synthesis database and assigning its hand pose parameters to the real image. The real hand images are captured with several kinds of backgrounds, some of them contains similar skin color and clutter.

Table 1. The correct match rates

Number	IR	SOM	ISOSOM
Top 40	44.25%	62.39%	65.93%
Top 80	55.75%	72.12%	77.43%
Top 120	64.60%	78.76%	85.40%
Top 160	70.80%	80.09%	88.50%
Top 200	76.99%	81.86%	91.59%
Top 240	81.42%	85.84%	92.48%
Top 280	82.30%	87.17%	94.69%

In our experiments, we use depth edges and the variant of shape context descriptor (see [9] for details) to represent the given image. Each hand shape is represented by a 256 component vector. We define the correct match if the hand gesture of the real image is the same as one of the hand gestures in the top N retrieved images (with small changes in the hand configuration), and in addition, the three global hand rotation parameters (roll, pitch, yaw) of the test images are within the 40° range with respect to that retrieved hand image. Using the same criterion, we compare the performances of the hand pose estimation by the traditional image retrieval algorithm (IR), SOM and ISOSOM in Table 1. The IR is implemented by simply comparing the query image with each image in the synthesis dataset and retrieve the top N best match. The result shows that the correct match rate of the ISOSOM increases around 16.5% compared to IR, and increases around 5.6% compared to SOM. N is decided by the application requirements. In order to achieve more than 85% successful rate, IR requires retrieving more than 280 images which provide more than 280 hand pose possibilities. ISOSOM just requires less than half of that, and also has a higher precision rate. The ISOSOM with 1512 neurons needs 0.036 second to retrieve the top 400 images. It is more than 12 times faster than IR, which needs 0.453 seconds. The ISOSOM retrieval results are shown in Figure 6. The first image in the first row is the query image. The second image in the first row is the model generated

by its pseudo-ground truth. The remaining 20 images are the retrieval results from the ISOSOM neurons. Figure 7 shows the comparison of the match rates with respect to the number of the retrieved images. The results indicate that the ISOSOM algorithm is overall better than traditional image retrieval algorithm and SOM [2].



(a) Pointer2 (View 9) (b) Fully Extended (View 3)

Figure 6. The ISOSOM retrieval results

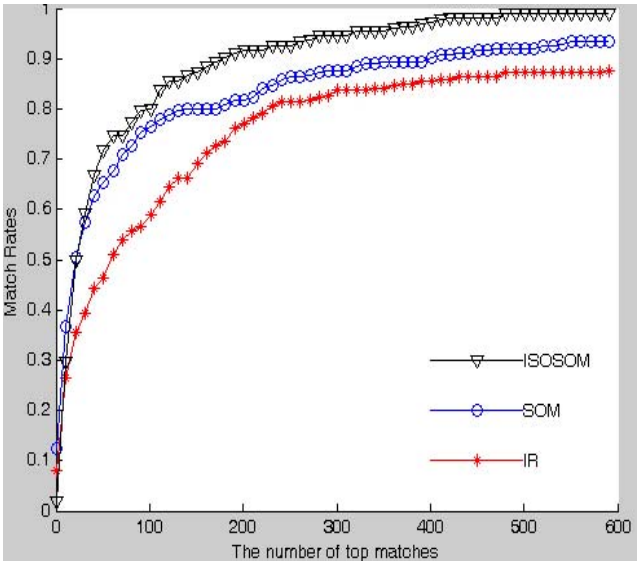


Figure 7. The performance comparisons

6. Conclusion

We have investigated a general nonlinear dimensionality reduction algorithm ISOSOM and applied it to the problem of 3D hand pose estimation from a single 2D image. Instead of representing each synthetic image by an isolated item in the database, the idea of this paper is to build a high dimension mapping and learn an organized structure in a low dimension space. With such generalized structure, we reduce the information redundancy in the database by clustering the similar vectors generated by the similar poses together. The retrieval is done by searching in the low dimensional manifold instead of exhaustedly searching in the whole database.

[2] It shows that the SOM is a little bit better than ISOSOM around $N = 20$, but the retrieval rate of around that point is too low to be considered as a good choice for N in the most applications.

References

[1] C. Tomasi, S. Petrov, and A. Sastry, "3D tracking = classification + interpolation," in *Ninth IEEE International Conference on Computer Vision (ICCV'03)*, vol. 2, pp. 1441–1448, 2003.

[2] J. Lee and T. L. Kunii, "Model-Based analysis of hand posture," *IEEE Computer Graphics and Applications*, vol. 15, no. 5, pp. 77–86, 1995.

[3] J. Lin, Y. Wu, and T. S. Huang, "Modeling the constraints of human hand motion," in *Proceedings of the 5th Annual Federated Laboratory Symposium*, 2001.

[4] J. Triesch and C. von der Malsburg, "A system for person-independent hand posture recognition against complex backgrounds," *IEEE Transactions on Patt. Anal. and Mach. Intell.*, vol. 23, pp. 1449–1453, Dec 2001.

[5] V. Athitsos and S. Sclaroff, "Database indexing methods for 3D hand pose estimation," in *Gesture Workshop*, April 2003.

[6] N. Shimada, K. Kimura, and Y. Shirai, "Real-time 3D hand posture estimation based on 2d appearance retrieval using monocular camera," in *IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pp. 23–30, 2001.

[7] T. Kohonen, *Self-Organizing Maps*. Springer Series in Information Sciences, 2001.

[8] R. Raskar, K. Tan, R. Feris, J. Yu, and M. Turk, "A non-photorealistic camera: Depth edge detection and stylized rendering with multi-flash imaging," in *ACM SIGGRAPH*, 2004.

[9] R. Feris, M. Turk, R. Raskar, K. Tan, and G. Ohashi, "Exploiting depth discontinuities for vision-based fingerspelling recognition," in *IEEE Workshop on Real-time Vision for Human-Computer Interaction (in conjunction with CVPR'04)*, 2004.

[10] J. B. Tenenbaum, V. D. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction.," *Science*, pp. 2319 –2323, Dec 2000.

[11] V. Athitsos and S. Sclaroff, "An appearance-based framework for 3D hand shape classification and camera viewpoint estimation," in *Proceedings of the Fifth IEEE International Conf. on Automatic Face and Gesture Recognition*, 2002.

[12] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, 2002.