UCSB COMPUTER SCIENCE TECHNICAL REPORT

3D Hand Pose Reconstruction With ISOSOM

Haiying GUAN and Matthew TURK

Computer Science Department
University of California, Santa Barbara
Santa Barbara, CA 93106

{haiying, mturk}@cs.ucsb.edu

Aug. 24, 2005

Abstract

We present an appearance-based 3D hand posture estimation method that determines a ranked set of possible hand posture candidates from an unmarked hand image, based on an analysis by synthesis method and an image retrieval algorithm. We formulate the posture estimation problem as a nonlinear, many-to-many mapping problem in a high dimension space. A general algorithm called ISOSOM is proposed for nonlinear dimension reduction, applied to 3D hand pose reconstruction to establish the mapping relationships between the hand poses and the image features. In order to interpolate the intermediate posture values given the sparse sampling of ground-truth training data, the geometric map structure of the samples' manifold is generated. The experimental results show that the ISOSOM algorithm performs better than traditional image retrieval algorithms for hand pose estimation.

Keywords:
Computer Vision (CV), Human Computer Interaction (HCI), Gesture Recognition, Hand Pose Estimation, Analysis by Synthesis, Image Retrieval.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

Despite the rapid advances in computing, communication, and display technologies, such as IBM's Blue Gene, wearable devices (PDAs and cell phones), 3D volumetric displays, and large scale displays, the development of Human Computer Interaction (HCI) still lags behind the capability of these new technologies. Currently, traditional input devices such as keyboards and mice are still the most commonly used devices for user interface. These devices, however, restrict the flow of information between users and computer systems. Their processing ability generally falls far behind both the user's ability to exchange the information and the computer's ability to process it. Moreover, they are unsuitable for the new displays we mentioned above, and not nature and intuitive for the user. Consequently, there is an interaction bottleneck between the user and the computer.

Gesture is a good candidate for the next generation input devices. It has the potential ability to relieve this interaction bottleneck. Hands play a very important role in interactions with objects and people. Hands perform countless actions, such as pointing, grabbing, throwing, reaching, and pushing. Hand gestures are a natural way to express ideas, communicate with others, and show people's intentions and feelings. For example, we point, pick, and grasp objects with the pointing, picking and grasping gestures; we wave at familiar faces with the hello gesture; we greet to people with the greeting gestures; we celebrate the achievement with the victory gestures; we use the "call me" gesture (to bend three middle fingers and extend thumb and little finger) for keeping in touch. Gesture is a deeply rooted part of human communication skill and it is capable of providing a direct, natural, and effortless way to express human ideas, commands, and intentions to the computer. Numerous researchers and inventors dedicated to human computer interface design are seeking for the possibility of using gesture for the interaction interface. To realize this possibility, we must explore the techniques which could model, detect, analyze and recognize gestures without intruding the user. Computer vision provides a passive, non-intrusive sensing technique for gesture interpretation. In addition, as high quality cameras with less cost are widely used nowadays, we are coming closer to making vision-based gesture interface practical.

1.2 Related Work

Currently, many studies for vision-based gesture interpretation have been undertaken within the context of particular applications [11] [19], such as, tracking a hand pointer in an indoor environment [8] [10] , recognition of a small set of well-defined hand poses [18], or recognition of a set of well-defined gestures [3] [16] . Although progress in the field is encouraging, further theoretical as well as computational advances are needed before gestures are widely used for real applications such as manipulations of virtual objects in 3D space, translations of the sign language and interactions with the small-size wireless devices. In these applications, the vision systems not only require the interpretation of the location, trajectory and time-spatial information of the gestures, but also the exact hand poses. Ultimately, 3D hand pose estimation is vital for the successful utilization of the gesture as an interactive interface.

Many approaches of 3D hand pose estimation to support gesture recognition may be classified into two categories: model-based approaches with 3D data [6] [12] [17] and appearance-based approaches with 2D data [2] [9] [13] [15]. For model-based approaches, Lee *et al.* [6] proposed a inverse kinematic algorithm for 3D hand pose estimation given 3D locations of seven markers. Lien [7] proposed a scalable model-based hand posture analysis system and addressed the scalability and the occlusion problem in conventional inverse kinematic algorithm. Both of the papers need markers and they are inconvenient in real applications.

In the appearance-based approaches with 2D data, the 3D hand model is generally used to estimate the 3D hand pose. Nolker *et al.* [9] described GREFIT (Gesture Recognition based on FInger Tips) approach. First, the finger tips were detected by a hierarchical system of artificial neural networks. Then, a neural-network-based system established a mapping between the 2D

positions of finger tips and the 3D configurations of the hand angle parameters. The system tested on the front views in a well controlled environment.

Another appearance-based method is based on analysis-by-synthesis approaches. Athitsos *et al.* [2] formulated the problem of hand pose estimation to a problem of database indexing. Chamfer distance and probabilistic line matching method are used to measure the similarity of the real image and the synthetic images. Shimada *et al.* [14] proposed a system with this approach. 125 possible candidate poses with 128 view points were generated with a 3D model of 23 Degree of Freedoms (DOF). The real input hand image was matched to pre-computed models with a transition network, and possible pose candidates were found. The candidates are projected to the image plane and the discrepancy of the silhouette of projected image and input image were evaluated. Extended Kalman filter (EKF) was used to find the best match for fitting the model to the image.

1.3 Problem Description

In this research, we take an image retrieval approach based on analysis by synthesis method. It utilizes a 3D realistic hand model and renders it from different viewpoints to generate synthetic hand images. A set of possible candidates is found by comparing the real hand image with the synthesis images. The ground truth labels of the retrieved matches are used as hand pose candidates. Because hand pose estimation is such a complex and high-dimensional problem, the pose estimate representing the best match may not be correct. Thus, the retrieval is considered successful if at least one of the candidates in top N matches is sufficiently close to the ground-truth (similar to [1]). If N is small enough, with additional distinguishable contextual information, it may be adequate for automatic initialization and re-initialization problems in hand tracking systems or sign language recognition systems, where the correct estimation could be found and the incorrect ones could be eliminated in the later tracking.

The hand is modeled as a 3D articulated object with 21 DOF of the joint angles [6] and 6 DOF of global rotation and translations [1]. A hand configuration is defined by these 21 local parameters and a hand pose is defined by a hand configuration augmented by the global rotation parameters.

The main problem of analysis by synthesis is the complexity in such a high dimension space. The size of the samples in the synthesis database could be roughly estimated as follows: if each degree of freedom has the range $0^\circ \sim 360^\circ$ and it is discretized by 36°, the number of poses is approximately 10^{24}! This is obviously an intractable number for both database processing and image retrieval. Even though the articulation of the hand is highly constrained, the complexity is still intractable for both database processing and image retrieval. Wu *et al.* [20] and Zhou *et al.* [23], for example, reduced the dimensionality to 6 or 7 based on data collected with the data glove. Furthermore, 36° is very sparse interval in the sampling procedure. If we reduce the dimensionality even further, the hand reconstruction system's performance would decrease because of the lack of accuracy.

In this report, we formulate hand pose reconstruction as a nonlinear mapping problem between the angle vectors (hand configurations) and the images. Generally, such mapping is a many-to-many mapping in high dimension space (Section 2). To simplify the problem, the hand configuration vector is augmented with the 3 global rotation parameters. The mapping from the augmented hand configurations space (the hand pose space) to the images feature space becomes a many-to-one mapping problem. This report focuses on the many-to-one mapping problem in high intrinsic dimension space.

Instead of representing each synthesis image by an isolated item in the database, the idea of this research is to cluster the similar vectors generated by similar poses together and use the ground-truth samples to generate an organized structure in low dimension space. With such structure, we can interpolate the intermediate vector and reduce the complexity. According to the geodesic distance of the points in the manifold of the training sample, we propose an ISOmetric Self-Organizing Mapping algorithm (ISOSOM). Similar as Kohonen's self-organizing mapping (SOM) [5], ISOSOM organizes the samples in a low dimension manifold by the geometric distance on the

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

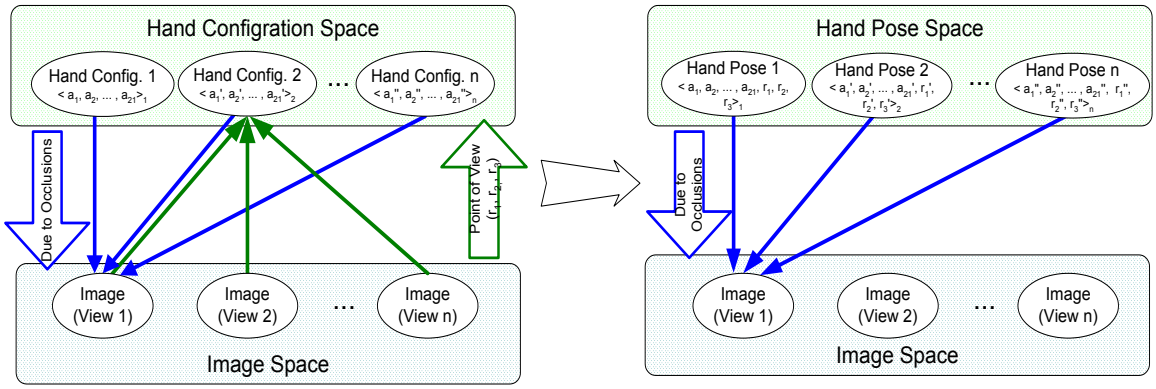


Figure 1: Nonlinear, many to many, continuous mapping

manifold instead of Euclidian distance in the 2D grids. Invariant moments [4] and Fourier descriptor [22] are used to extract the invariant features to represent the hand images. The experimental results shows that our algorithm is better than traditional image retrieval algorithms.

The report is organized as follows. The problem is formulated in Section 2. ISOSOM algorithm is proposed in Section 3. Section 4 describes the hand pose reconstruction method with ISOSOM. The experimental results are shown in Section 5. Finally, the conclusions are given in Section 6.

2 Problem Formulation

Given an hand image, the task for hand pose reconstruction algorithm is to output the corresponding hand pose (that is, the hand configurations with the 3 global rotation parameters). Thus, the input space for hand pose reconstruction algorithm is the hand image space, the dimension of this space is determined by the image size. The output space is the hand pose space which could be treated as 24 dimensions (21 hand configuration components plus 3 rotation components). Due to occlusions, different hand poses could be rendered to the same image. In such cases, many joint angle vectors could generate one hand image, the mapping along this direction is many-to-one (Figure 1). On the other hand, the same pose can be rendered from the different view points and thus generates many images. The mapping along this direction is one-to-many. Intrinsically, the mapping between the hand pose vectors and the image vectors is a many-to-many mapping.

Because the dimension of image space is very high, feature extraction algorithm, which could be considered as a nonlinear dimension reduction process from the image space to the feature space, are generally used to extract the feature vectors to represent the image. On the other hand, hand is a highly constrained object and the joint angles are correlated. The dimension of hand configurations space thus could be reduced. Figure 2 shows the relationship between image space, feature spaces, hand configuration space, hand pose space and low dimension reduction space of hand configuration space.

To simplify the problem, we eliminate the second, one-to-many case by augmenting the hand configuration vector with the 3 global rotation parameters to construct the augmented vector as hand pose vector. The hand pose vector determines the hand image and the feature vector which represents the image. The mapping between the hand pose space and image space (or feature space) is thus many-to-one mapping. Finally, we focus on the many-to-one, nonlinear, continuous mapping problem between the hand pose space and the feature space.

3 Learning Mapping with ISOSOM Algorithm

In order to learn the nonlinear and high dimension mapping, based on SOM and ISOMAP algorithm, we proposed an ISOmetric Self-Organizing Mapping algorithm (ISOSOM). It utilizes the topological graph and geometric distance of the samples' manifold to define the metric relationships

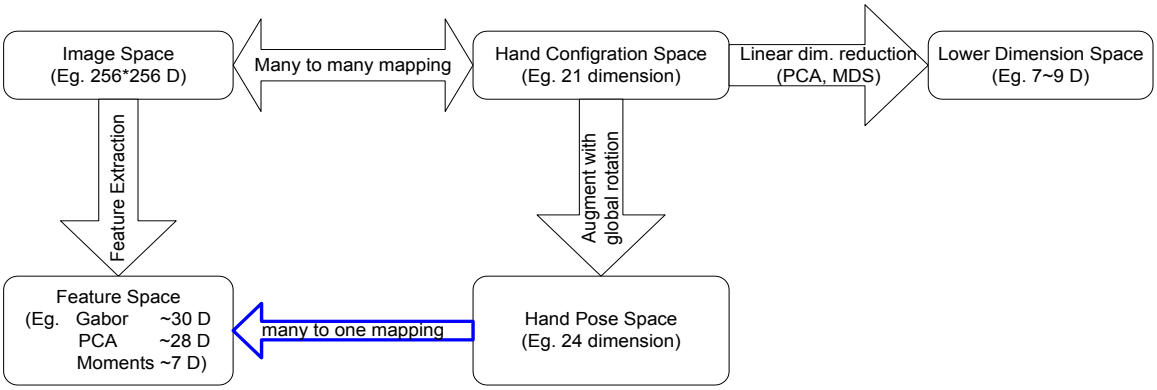


Figure 2: The relationship between five spaces

between samples. By such geometric distance, it enables the SOM to follow better the topology of the underlying data set and preserves the spacial relationships of the samples in low dimension space. Meanwhile, it self-organizes the samples in low dimension space by the information compression.

3.1 SOM

Kohonen's [5] Self-Organizing Map (SOM) 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 2-D visualization space by preserve the topological order of the data, at the same time, it clusters the similar data into clusters. Generally, it is an unsupervised clustering algorithm for dimension reduction.

The SOM structure in lower dimension usually consists of a two-dimensional regular grid of neurons with four or six neighbors (hexagons). Each neuron is associated with a vector. In the training process, the input vector is compared with the vectors of each neuron, fires the best match neuron with the minimum distance (normally it is defined in Euclidean space). The associated vector of the best match neuron and its neighbors are updated iteratively. The associated vectors of the neighbors therefore are similar to each other and the similar vectors are clustered together. Although the input dimension of SOM could be very high, the SOM is efficient for the data samples with the low intrinsic dimension. In our problem, both the intrinsic dimensions of hand pose space (around 10D after dimension reduction) and the feature space are very high, the classical SOM cannot organize the samples efficiently.

3.2 ISOMAP

Tenenbaum's ISOMAP 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 graph in which the edges are generated if two nodes share a common neighbor. The cost of the edge is measured by the Euclidean distance of two nodes. The distance of two nodes is defined by the cost of shortest path between them. The classic multidimensional scaling algorithm is adopted to construct a low dimensional representation of that data. The algorithm is based on estimating and preserving global geometry. This benefit could be used to avoid the feature vectors' mixed-ups in low dimension space due to dimension reduction.

3.3 ISOSOM

Based on ISOMAP's idea, we utilize the topological graph and geometric distance of the manifold in high dimension space to organize the vector in high dimension space. Similar to the first step of ISOMAP algorithm, we define a topological graph of the manifold G over all data points by

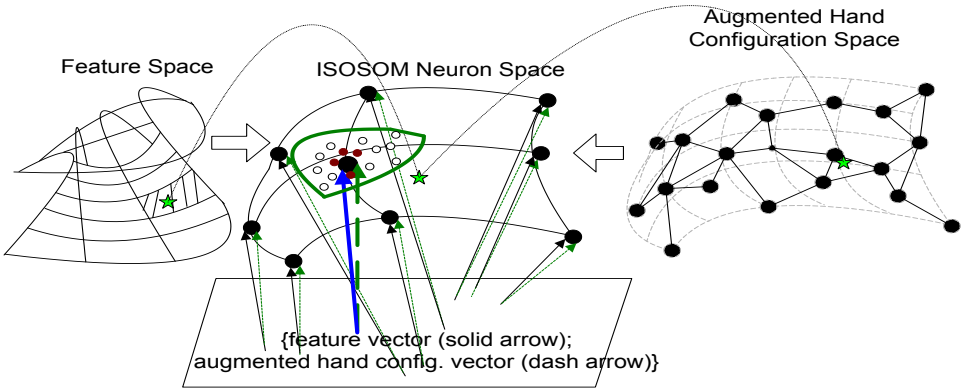


Figure 3: ISOSOM algorithm

connecting node i and j if i is one of the k nearest neighbors of j. We set the edge lengths equal to the Euclidean distance of i and j. On this graph, the distance of any two nodes is defined by the cost of shortest path between them. This distance is approximately the geometric distance on the manifold. Such distance preserves the high-dimension relationship of the samples in low dimension space. Combined this distance with the traditional SOM algorithm, the ISOSOM algorithm compresses information and automatically clusters the training samples in a low dimension space efficiently.

3.3.1 Initialization

Although the ISOSOM algorithm is robust with respect to the initialization, the appropriate initialization allows the algorithm to converge faster to the solution. Before the training, initial vectors associated with neurons on the ISOSOM map are linearly interpolated by the sample nodes of the manifold's topological graph.

3.3.2 Training

Before training, we generate the topological graph of the manifold described above and calculate the global geometric distance between each samples on the graph.

The neurons of the ISOSOM map are connected with their neighbors on the low dimension manifold. In each training step, we randomly choose a sample vector from the training samples and find its Best-Matching Unit (BMU) in the ISOSOM map. Instead of using Euclidean distance, we measure the similarities of the input sample vector with each neuron vector in the ISOSOM map by the geometric distance between the two nodes on the manifold of the training samples. In order to measure this similarity, the nearest nodes of input vector and the neuron vector on the topological graph are retrieved by the Euclidean measurements. The shortest path between the two retrieved nodes on the manifold graph is approximated as the distance of the input vector and the neuron vector of the ISOSOM map. The BMU is the neuron on the ISOSOM map with the smallest geometric distance to the input training vector.

After obtaining the BMU, its associated vectors and its topological neighbors on the ISOSOM map are updated and moved closer to the input vector in the input space as in the classical SOM algorithm.

3.3.3 Retrieval

Given a vector with full components or partial components, the similar neurons are found and sorted by the similarity measurement described above. The mask is used for the vector with partial components and the similarity measurements are also modified to handle the mask.

4 ISOSOM for Hand Pose Reconstruction

With the synthesis method, a realistic hand model are built and the hand poses and the corresponding hand images are generated as our training samples. Our task is to learn the mapping between the feature space and the hand pose space. As we said before, such mapping is many-to-one mapping. Because of the projection and the feature extraction, feature samples of the different poses highly overlap in the feature space. In order to separate the mixed-up features, we form large vectors consisting of both feature vectors and their corresponding hand pose vectors. After augmented feature vector with pose vector, there is no mixed-up in hand pose plus feature vector space. Although the dimension is increased, the distance on the geometrical manifold of the hand pose reflects the true relationship of the sample vectors. If we use such distance in ISOSOM training, we can separate the similar feature vectors generated by different pose. In addition, ISOSOM algorithm is not sensitive to the size of input dimension. This is also the reason why we use the hand pose space directly instead of using its lower linear dimension reduction space. In this way, the original supervised learning problem is converted to an unsupervised algorithm.

Now, using these large vectors as training samples, each neuron vector of ISOSOM map is composed by two vectors: the feature vector and the corresponding pose vector. Figure 3 gives an intuitive depiction of the ISOSOM map. The initial vectors of ISOMAP's neurons are sampled and interpolated from the topological graph of the hand pose and feature vector manifold. We use the sample nodes as the neuron in our ISOMAP algorithm. In the retrieval step, for a given input hand image, we calculate its feature vector. Using a mask to handle the missing hand pose components, we compare the similarity of this feature vector with all feature vectors associated with the ISOSOM neurons. The possible candidates of the hand pose are retrieved by the top n best matches. 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. This is desirable because it reflects the intrinsic nature of the mapping. The confidence of each candidate is also measured by the error measurement of ISOSOM.

5 Experimental Results

5.1 Synthesis database

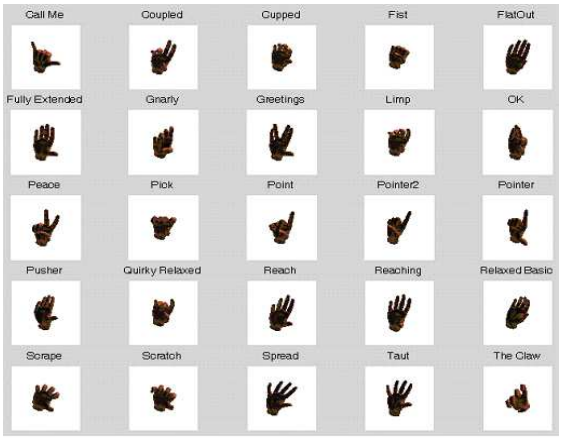
We generate a synthesis database containing 25 commonly used poses. (Figure 4 (a)). For each hand configuration, 48 joint angle parameters are saved as the joint angle vector, which are 3 rotation parameters for hand, 9 parameters (3 rotation parameters for 3 joints respectively) for each finger and thumb. In addition to the 3 global rotation parameters of the camera, the hand pose vector is composed of these 51 parameters.

A sparse sampling database is generated as the training database. In this database, three camera parameters (roll: $0^\circ \sim 360^\circ$, pitch: $-90^\circ \sim 90^\circ$, and yaw: $0^\circ \sim 360^\circ$, interval: 36°) control the global rotation of the camera viewpoints. For each pose, 726 images are rendered in different view points and there are totally 18150 synthesis images in the database. Some randomly picked images in the database are shown in Figure 4 (b).

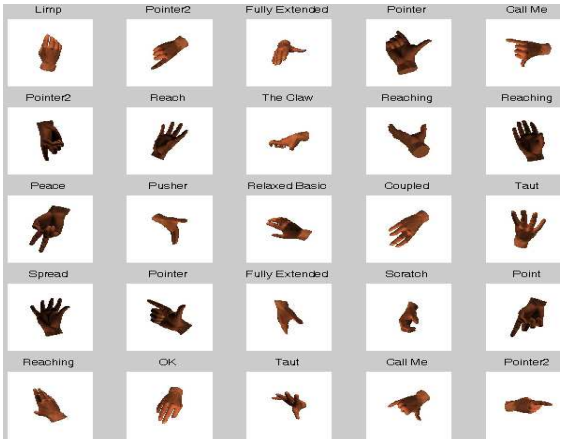
Another dense sampling database is generated as the testing database. Because our feature is invariant to translation, rotation and scale. We don't need to take care of roll parameter of the camera. Thus, two camera parameters pitch ($-90^\circ \sim 90^\circ$) and yaw ($0^\circ \sim 360^\circ$) are concerned. In the dense sampling, the interval is 9° instead of 36°. For each pose, 861 images are rendered in different view points and there are totally 21525 synthesis images in the dense sampling database.

5.2 Feature Extraction

In general, image representation includes color, texture, spatial layout, shape, interest points, and mathematical features (such as PCA). For our tasks, the color, texture and spatial layout features are not very distinctive for different poses. The potential choices could be hand shape. The shape can be represented by global features and local features. The global feature includes



(a) 25 poses in the database



(b) Randomly picked images in the database

Figure 4: The synthesis database

statistics/physical descriptors such as the area, contour length, and the centroid, region-based descriptors such as moments, boundary-based descriptors such as Fourier descriptor, and mathematical descriptors such as PCA features. The local feature includes Gabor features, texture features and so on.

In the experiments, Hu moments [4] which consists of 7 moment parameters and Fourier descriptors are adopted. These two global features are invariant to translation, in-plane rotation and scale. Even though the two representations are related by a simple linear transformation, the combination of them is more robust and makes great improvement in real applications.

In order to obtain Fourier descriptors for hand images, closed contours are generated by Gradient Vector Flow (GVF) algorithm [21]. The benefits of GVF algorithm are its automatical initialization and its capability of convergence to boundary concavities. The convex hull of skin color segmentation is served as the initial contour. The GVF algorithm iteratively refines the contour until a criterion is satisfied.

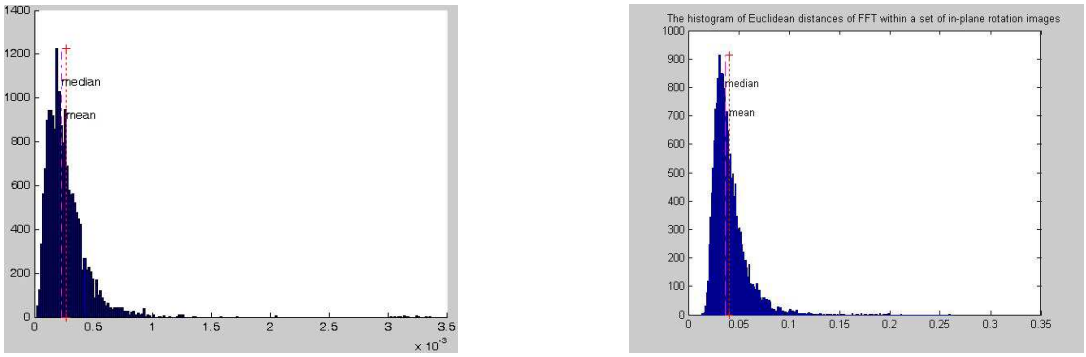
The FFT coefficients of the hand contour are calculated for each image in the database. The first $0 \sim N$ coefficients are called Fourier Descriptors (FD) of the shape. In our experiments, the first 15 FFT coefficients are considered. The rest coefficients are regarded as high frequency noises. The first frequency coefficient FD_0 is the direct current (DC) coefficient. It is related to the position of shape and is discarded. Each coefficient of a has two components: amplitude and phases. The in-plane rotation invariance is achieved by only using the amplitude component. The scale invariance is achieved by dividing all amplitudes of other descriptors by amplitude value of the second frequency coefficient FD_1.

5.3 Feature Performance Analysis

For these two features, we are interested in their performance to represent the hand images. In other word, since the descriptor are translation, in-plane rotation and scale invariant, the same hand shape under different translation, in-plane rotation and scale invariant should be described by the same descriptor. On the other hand, the hand images generated from different pose should be represented by different descriptors.

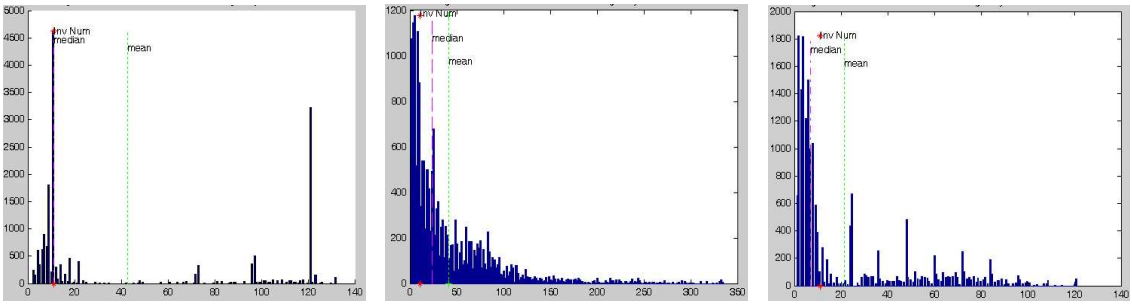
There are 11 images in the database which have the same hand shape except the differences of in-plane rotation (the roll parameter of the camera), scale, translation and lighting conditions. For these 11 similar hand shapes, we calculated Euclidean distances among their descriptor vectors and choose the median distance to measure the precision of the descriptors for such shape. Figure 5 (a) and 5 (b) show the histogram distribution of such differences of Hu moments descriptors and Fourier descriptors based on 18150 images with 1650 shapes. The precision of Hu moments and Fourier Descriptor could be find from the histogram distribution (The median distances are set as

the precision threshold).



(a) Hu moments (b) Fourier Descriptors

Figure 5: The histograms of the distance of the descriptor vectors representing the same shape



(a) Hu moments (b) Fourier Descriptors (c) The combination

Figure 6: The histograms of the number of the similar images for the given hand shapes

Given the precision threshold, the descriptive properties of the two descriptors are measured. For each query hand images, the similar images within the database whose feature vectors are within the distance threshold are retrieved. Ideally, we wish each hand shape image is represented by an unique feature vector. In our cases, for the same hand shape, there are 11 hand images in the database and they should be represented by an unique feature vector. On the other hand, if the descriptor is less descriptive, many other images could fall in the threshold and the descriptor cannot distinguish difference poses. Using the sparse sampling database, the histogram of the number of the images, whose feature vector are fall into the threshold, are shown in Figure 6 (a) and 6 (b). Ideally, the distributions are pulses at 11. However, some numbers of similar images are larger than 11 and some numbers are less than 11 because we choose the median as the precision threshold instead of maximum. It should be point out that some numbers of similar images are around 120, it reveals that Hu moments have limited distinctive ability in some cases. Figure 6 (b) shows the distribution for Fourier Descriptors. It shows that the median of the similar image numbers is larger than 11. It indicate that the descriptive ability of Fourier Descriptors is decreased compared with ideal cases, which may caused by the performance of contour extraction steps.

The retrieval performance of the combination of two descriptors are also measured by the following method: for the given query hand shape, the number of the similar images falling into both precision thresholds are counted. Figure 6 (c) shows that the number of similarity is greatly reduced in the combination scheme. It indicates that the combination of two features gives better representations of hand images.

5.4 Traditional Hand Image Retrieval

5.4.1 Hand Image Retrieval with the Precision Thresholds

Based on the combination of the two features, the image retrieval scheme (**Scheme A**) is designed as follows: first, all the images in the database are sorted according to their Hu moment distance to the query image. Next, all the images in the database are sorted according to their Fourier Descriptor distance to the query image. The top N images are retrieved based based on the combinational priority of these two orders. The complexity of such retrievals are $O(NlogN)$, where N is the size of database.

In order to reduce the retrieval time, after sorting all the images according to Hu moments distance, **Scheme B** only pick the first 200 images and calculate their FD distance to the query images, then the combination priority is used to retrieve the top N matches.

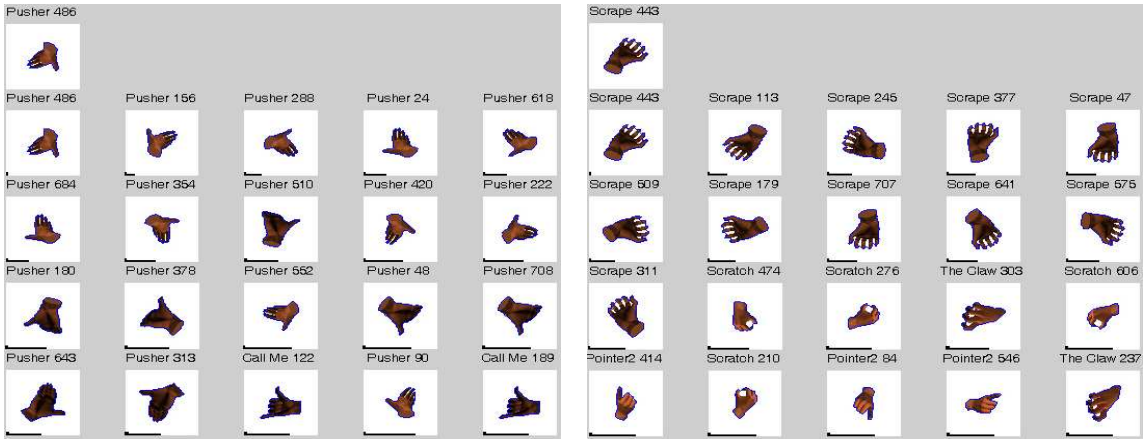


Figure 7: The retrieval results based on Scheme B

Figure 7 illustrates the top 20 retrieval results based on Scheme B. Two small bars in the bottom of each retrieval images represent the distance of the moment features and Fourier Descriptors between the query image. As we known, given one query image, there are 11 images in the database with the same hand shape except in-plane rotation, scale, translation and lighting conditions. Ideally, we hope 11 images should be in top 20 and the rest could be the same hand configuration with different camera pose. But in fact, other hand configurations are also shown in the top 20 retrieved images. It indicates that the feature vectors of different hand configurations are highly mixed up in high dimension space due to projection and feature extraction.

Table 1 shows the comparisons of retrieval performance of two schemes in top 20 matches (the number of images measured: 18150). It indicates that Scheme B doesn't decrease the retrieval performance. The percentage of the hits of the same hand configuration in top 20 are divided by 20. Table 2 shows the comparisons of retrieval performance of two features and the combination scheme in top 40 matches. The average time for one query is 0.1234. The percentage of the hits of the same hand configuration in top 40 are divided by 40. In both tables, the number of images measured is 18150. The percentage of The hits of the same hand pose (except roll) and the percentage of the hits of the wrong hand configuration are both calculated by the hit number divided by 11.

5.4.2 Hand Image Retrieval with the Top N Matches

Now, we measure the performance of the image retrieval algorithms using the dense sampling database, which focuses on pitch and yaw rotation of the camera and use 9° as a interval instead of 36°. The database contains 21525 hand images. It turns out that 92.33% of the testing images in the dense sampling database cannot find the exact match in the sparse sampling database (the training data set). For a given testing image in the dense set, we count a hit if there are one or

Table 1: The comparisons of two schemes

	Average time for one query	The average number of the exact same hand pose (except roll) in top 20		The average number of the exact same hand config. in top 20		The average number of the wrong hand config. in top 11	
Schemes	(sec.)	number	percentage %	number	percentage %	number	percentage %
Scheme A	3.0965	7.2707	66.01	15.2818	76.41	0.6683	6.08
Scheme B	0.1196	7.7597	70.54	15.5338	77.67	0.2574	2.34

Table 2: The comparisons of of two descriptors

	The average number of the exact same hand pose (except roll) in top 40		The average number of the exact same hand config. in top 40		The average number of the wrong hand config. in top 11	
Schemes	number	percentage %	number	percentage %	number	percentage %
Hu moments	10.9967	99.97	23.4718	58.68	0.0864	0.79
FD with the top 200 of Hu results	10.9967	99.67	23.9777	59.94	0.8753	7.96
Scheme B	8.5152	77.41	23.9478	59.87	0.2574	0.0234

more hand pose(s) in the retrieved poses whose global rotations are close to the query image within evaluation threshold 37°. According to this definition, for each hand pose of the query image, it has about 4 similar poses in the training database.

First, we use the Hu moments feature along and the FD feature alone to retrieve the best matches. Then, we design two combination schemes: Scheme 1 sorts all the image according to invariant feature, then pick the top 200 images and sort them according to their FD features, finally combine the two sorting results and find the top matches. Scheme 2 sorts all the images according to the FD feature first, pick the top 200 images and sort them according to Hu features, then combine the invariant features and find the top matches. Table 3 shows the retrieval results of these four schemes with the dense testing set. It shows that Scheme 2 gives the best performance, which achieves 93.21% hit rate within the top 160 matches. The query time is around 0.12 seconds per query.

Table 3: The comparisons of the retrieval performances with the dense testing set by traditional four retrieval schemes with the threshold 37°. (The number of images measured: 21525.)

	The percentages of hits of the similar hand pose (hand configuration and the global rotations)				The percentage of hits of the same hand configuration (without the global rotations)			
No. of images	Hu	FD	Scheme 1	Scheme 2	Hu	FD	Scheme 1	Scheme 2
Top 20	24.97%	55.63%	32.03%	58.67%	35.98%	71.89%	46.69%	73.70%
Top 40	29.41%	62.56%	37.03%	65.66%	45.07%	79.52%	55.17%	80.89%
Top 80	34.95%	69.53%	43.66%	73.13%	55.52%	86.67%	65.58%	87.80%
Top 120	38.10%	73.64%	46.33%	77.12%	60.84%	89.97%	70.11%	91.02%
Top 160	41.49%	76.33%	50.00%	79.81%	66.15%	92.05%	75.01%	93.21%

5.5 Hand Pose Reconstruction with ISOSOM

In the experiments, we only use Hu features to represent given images. It should be pointed out that ISOSOM is a general algorithm for nonlinear mapping and dimension reduction, it does not specify the features. The following experimental shows that even with less descriptive Hu features, our algorithm still can achieve good estimation results.

Using the same dense set as the testing data, based on Hu moments feature, we compare the performance of the traditional image retrieval algorithm (IR), SOM and ISOSOM. We compare

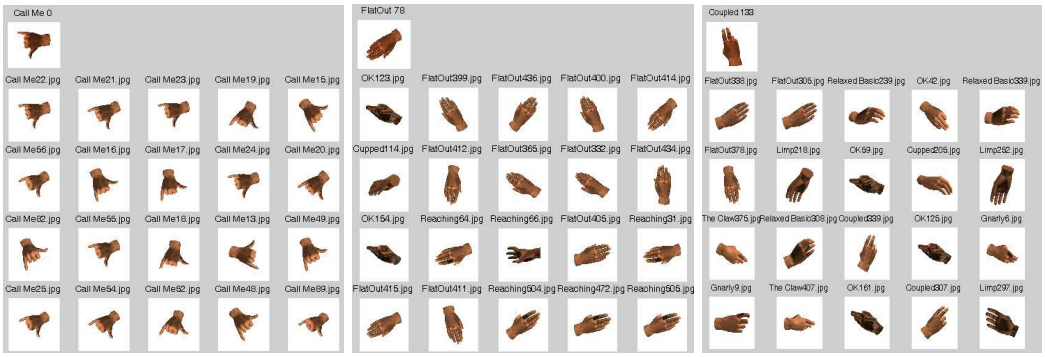


Figure 8: The ISOSOM retrieval results

them when the evaluation threshold for the global rotations is 10°, 20° and 40° respectively. Because the hand configuration vectors of the SOM and the ISOSOM are modified during the training steps, they are not the exact original configurations any more. In order to keep the same criterion for three algorithms, during the hit counting, we define the similar configurations by allowing the parameter components to change within a small range without changing the physical meanings of the postures. Table 4 shows the comparison results. The performance of the ISOSOM is the best among the three algorithms. This results indicates the hit rate increase around $7\% - 27\%$ with ISOSOM compared to the traditional image retrieval algorithm for the top 20 matches. It also indicates that the ISOSOM algorithm not only has the clustering ability, but also has interpolation ability. The experimental results also shows that only with hu moment features, ISOSOM achieves the same performance as Scheme 2 which uses both Hu features and Fourier features.

Table 4: The comparisons of the reconstruction performance by the traditional retrieval algorithm, the SOM and the ISOSOM algorithms. (The evaluation thresholds for the global rotations are 10°, 20° and 40° respectively. The number of images measured: 21525.)

The percentages of hits of the similar hand pose (except roll rotation)									
	Threshold 10			Threshold 20			Threshold 40		
Number	IR	SOM	ISOSOM	IR	SOM	ISOSOM	IR	SOM	ISOSOM
Top 20	19.46%	10.75%	26.69%	29.01%	24.99%	47.40%	38.13%	43.72%	66.65%
Top 40	23.21%	16.13%	36.73%	36.25%	34.27%	59.97%	47.93%	55.64%	78.01%
Top 80	27.13%	23.20%	48.71%	44.48%	44.52%	72.11%	58.36%	67.03%	86.85%
Top 120	29.28%	28.63%	55.77%	49.14%	50.97%	78.82%	63.69%	73.04%	91.64%
Top 160	31.41%	32.39%	61.14%	53.17%	55.37%	83.32%	68.39%	76.28%	94.33%
Top 200	33.35%	35.28%	65.26%	56.98%	58.62%	86.13%	72.78%	78.64%	95.56%
Top 240	34.90%	37.50%	68.47%	59.95%	60.99%	88.12%	75.71%	80.51%	96.55%
Top 280	36.28%	39.61%	71.12%	62.74%	63.53%	89.69%	78.43%	82.98%	97.13%

The ISOSOM retrieval results are shown in figure 8. The first image is the query image. The image name is the name of the hand configuration and the following number is the index number of the particular pose in the testing database. The rest 20 images are the retrieval results from the ISOSOM neurons (the number is the index number of the particular neuron in the ISOSOM map). The experimental results also shows that only with hu moment features, ISOSOM achieves the same performance as Scheme 2 which uses both Hu features and Fourier features.

6 Discussion and Conclusion

We have investigated a nonlinear mapping approach for 3D hand pose estimation from a single image. Traditional image retrieval algorithms just compare the image features in feature space and retrieve the top matches. Our approach utilizes both the feature vectors and their corresponding

augmented hand configuration vectors to avoids the feature overlapping problem in nature. To deal with the complexity, we reduced the redundancy by clustering the similar feature vectors generated by similar poses together and represented them by neurons in our low dimension ISOSOM map. The experimental results also confirm that the ISOSOM algorithm greatly increase the hit rates in the pose retrievals.

The ISOSOM algorithm proposed by the report is a general algorithm for nonlinear dimension reduction. The ISOSOM algorithm use the geometric distance instead of Euclidean distance, which provides the better relationship measurements of the two points in high dimension space to achieve better clustering performance. It could be considered as a variant of the SOM algorithm which aims at enabling SOM to follow the better topology (ISOMAP's topology instead of 2D grids) of the underlying data set.

References

[1] Vassilis Athitsos and Stan Sclaroff. An appearance-based framework for 3d hand shape classification and camera viewpoint estimation. In *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, page 45, Washington, DC, USA, 2002. IEEE Computer Society.

[2] Vassilis Athitsos and Stan Sclaroff. Database indexing methods for 3d hand pose estimation. In *Gesture Workshop*, April 2003.

[3] A. F. Bobick and A. D. Wilson. A state-based approach to the representation and recognition of gesture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(12):1325–1337, December 1997.

[4] M-K. Hu. Visual pattern recognition by moment invariants. *IRE Trans. on Information Theory*, (8):179 –187, 1962.

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

[6] Jintae Lee and Tosiyasu L. Kunii. Model-Based analysis of hand posture. *IEEE Computer Graphics and Applications*, 15(5):77–86, 1995.

[7] Cheng-Chang Lien. A scalable model-based hand posture analysis system. *Machine Vision and Applications*, 16(3):157–169, 2005.

[8] Shan Lu, D. Metaxas, D. Samaras, and J. Oliensis. Using multiple cues for hand tracking and model refinement. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 443–450, June 2003.

[9] Claudia Nolker and Helge Ritter. Visual recognition of continuous hand postures. *IEEE Transactions on Neural Networks*, 13(4):983–994, July 2002.

[10] K. Oka, Y. Sato, and H. Koike. Real-time fingertip tracking and gesture recognition. *IEEE Computer Graphics and Application*, 22(6):64–71, Nov-Dec 2002.

[11] V. Pavlovic, R. Sharma, and T. S. Huang. Visual Interpretation of Hand Gestures for Human Computer Interaction: A Review. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 19, pages 677–695, July 1997.

[12] James M. Rehg and Daniel D. Morris. Singularities in articulated object tracking with 2-d and 3-d models. Technical Report CRL 97/8, Compaq CRL, October 1997.

[13] Rómer Rosales, Vassilis Athitsos, and Stan Sclaroff. 3d hand pose reconstruction using specialized mappings. In *International Conference on Computer Vision (ICCV)*, pages 378–385, 2001.

[14] Nobutaka Shimada, K. Kimura, and Yoshiaki 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*, pages 23–30, 2001.

[15] Nobutaka Shimada, Yoshiaki Shirai, Yoshinori Kuno, and Jun Miura. Hand gesture estimation and model refinement using monocular camera - ambiguity limitation by inequality constraints. In *Proceedings of the 3rd Conf. on Face and Gesture Recognition*, pages 268–273, April 1998.

[16] Thad E. Starner, Joshua Weaver, and Alex Pentland. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 20(12):1371–1375, December 1998.

[17] B. Stenger, P. R. S. Mendonça, and R. Cipolla. Model-based 3d tracking of an articulated hand. In *Proc. Conf. Computer Vision and Pattern Recognition*, volume 2, pages 310–315, December 2001.

[18] Jochen Triesch and Christoph von der Malsburg. A system for person-independent hand posture recognition against complex backgrounds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12):1449–1453, Dec 2001.

[19] Ying Wu and Thomas S. Huang. Vision-based gesture recognition: A review. In Annelies Braffort, Rachid Gherbi, Sylvie Gibet, James Richardson, and Daniel Teil, editors, *Gesture-Based Communication in Human-Computer Interaction*, volume 1739 of *Lecture Notes in Artificial Intelligence*. Springer Verlag, Berlin Heidelberg, 1999.

[20] Ying Wu, John Y. Lin, and Thomas S. Huang. Capturing natural hand articulation. In *ICCV*, 2001.

[21] C. Xu and J. L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, (3):359–369, March 1998.

[22] C.T. Zahn and R.Z. Roskies. Fourier descriptors for plane close curves. *IEEE Trans. Computers*, (21):269–281, 1972.

[23] Hanning Zhou and Thomas S. Huang. Tracking articulated hand motion with eigen dynamics analysis. In *Ninth IEEE International Conference on Computer Vision*, pages 1102–1109, Oct 2003.