2013 IEEE International Conference on Robotics and Automation (ICRA)
Karlsruhe, Germany, May 6-10, 2013

Grasp Mapping Using Locality Preserving Projections and kNN Regression

Yun Lin, Yu Sun

Abstract— **In this paper, we propose a novel mapping approach to map a human grasp to a robotic grasp based on human grasp motion trajectories rather than grasp poses, since the grasp trajectories of a human grasp provide more information to disambiguate between different grasp types than grasp poses. Human grasp motions usually contain complex and nonlinear patterns in a high-dimensional space. In this paper, we reduced the high-dimensionality of motion trajectories by using locality preserving projections (LPP). Then, a Hausdorff distance was performed to find the k-nearest neighbor trajectories in the reduced low-dimensional subspace, and k-nearest neighbor (kNN) regression was used to map a demonstrated grasp motion by a human hand to a robotic hand. Several experiments were designed and carried out to compare the robotic grasping trajectory generated with and without the trajectory-based mapping approach. The regression errors of the mapping results show that our approach generates more robust grasps than using only grasp poses. In addition, our approach has the ability to successfully map a grasp motion of a new grasp demonstration that has not been trained before to a robotic hand.**

I. Introduction

Learning from Demonstration (LfD) has been a powerful mechanism for a robot to learn new tasks by observing people's demonstrations without any reprogramming. For most applications using LfD, a number of human movements are recorded during a task, then robot can mimic human motions by reproducing similar movements, as in the demonstration [1]. In LfD, mapping between human and robot has been a crucial problem to resolve the correspondence issue.

The most common way of mapping from a human to a robotic hand is direct mapping, which can be further divided into mapping in finger joints space and in fingertip workspace. Mapping in joint space finds the correspondence of joint angles in a robot's configuration space, which usually creates a hand pose for the robotic hand similar to the human hand pose. This is suitable for power grasps but not for precision grasps, because it may lead to low accuracy in fingertip positions. Mapping in fingertip space, on the other hand, is more suitable for precision grasps, because it computes the correspondence of fingertip positions in workspace. In [2], neural network techniques were used to calibrate the relationship between the Cartesian position of the fingertips and data glove reading. Mapping between a human hand and a robot hand is determined by a linear translation between the two workspaces. [3] used a point-to-point algorithm to map a human hand to a three-finger gripper in a teleoperation. However, in general, since most robotic hand designs simplify the mechanical structures to make control easier, some fingertip positions demonstrated by humans are not reachable by all robotic hands.

Since direct mapping remains challenging, classification approaches have been developed to avoid the correspondence problem; human grasps are classified into different grasp types, and human demonstration is recognized as one of the grasp types. Many of the grasp classifications are based on Cutkosky's grasp taxonomy [4]. Cutkosky's taxonomy classifies common user-performed grasps into 16 classes based on task requirements and dexterities. To recognize the demonstrated grasp as one type in Cutkosky's taxonomy, pattern recognition techniques can be applied. [10] used Hidden Markov models to recognize grasp type from the taxonomy based on an entire grasp sequence. A neural network was used to map in a configuration space. The recognition rate was 97% for a single user, which exists in both the training and test dataset when there are 10 grasp types. The recognition rate dropped to 65% for unseen users that were not in the training dataset. There is no information on performance if unseen objects were tested. Aleotti and Caselli [6] performed grasp recognition using static grasp poses in virtual reality for six grasp types, with a recognition rate of 94% for two expert users without using real objects. [7] compared the different classification methods for recognizing six grasp types. The best recognition rate of all the methods was 83.82% for seen users and 71.67% for both unseen users and objects.

One motivation of this paper is to explore the trajectories of hand joints, which provide richer information than static poses. It is necessary to disambiguate between grasp types that share similar static poses but differ in grasp trajectories, because some similar poses belonging to different classes in human configuration space may be far apart from each other in robot configuration space. For example, the lateral pinch and small wrap have a similar user-performed grasp pose, whereas, due to much less dexterity in some robotic hands, e.g. the simple three-fingered Barrett hand, the lateral pinch has to be performed in a way distinct from the small wrap (Figure 1).

The hand joint motion of a human demonstration can be treated as a high-dimensional time-series. The noise and variance lying in the motions bring difficulty to the problem. Therefore, it is beneficial if features that discriminate motions can be preserved but unwanted variance can be removed. Dimensionality reduction approaches can be used for this purpose, since the high-dimensional hand motions may

Yun Lin and Yu Sun are with the Department of Computer Science and Engineering, University of South Florida. (yunlin@mail.usf.edu, yusun@cse.usf.edu)

978-1-4673-5642-8/13/$31.00 ©2013 IEEE

have a nonlinear manifold in a low-dimensional subspace. We used locality preserving projections (LPP), presented by He and Niyogi [8], to find the low-dimensional manifold of the training motion data, because LPP has the strength to project the out-of-sample data points to low-dimensional space easily. Mapping is performed between a human and a robotic hand in the low-dimensional subspaces.

Instead of grasp recognition that classifies a new grasp motion as a predefined type, we use k-nearest neighbor (kNN) regression to get the interpolation between classes, given a new demonstration of an unknown object. The benefit is that when the user performs an irregular grasp of an unknown object, the regression will result in an intermediate grasp pose between different grasps. The distance for kNN regression is defined by Hausdorff distance [9], which measures the similarity between trajectories.

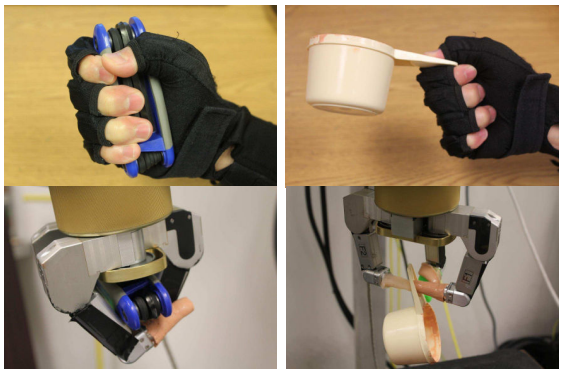

Fig. 1: Corresponding small wrap and lateral pinch of robotic hand to human hand. They look similar to a human grasp but are different for a robotic grasp. Left: Small wrap grasps for a human hand (top) and a robotic hand (bottom). Right: Lateral pinch grasps for a human hand (top) and a robotic hand (bottom).

II. Grasp Mapping Using Trajectories

Given a training dataset of grasp motion sequences, the high-dimensional hand motions in the dataset usually have undesirable properties that bias the learning results. Dimensionality reduction is a typical approach for finding a lower intrinsic dimensionality of data while removing the undesirable noise and variance and leaving a minimum number of needed properties of the data. Typical methods for dimensionality reduction include linear methods, such as principal component analysis (PCA) [14] and linear discriminant analysis (LDA) [15], which both find a linear transformation of high-dimensional data to their low-dimensional counterpart. Nonlinear methods, such as isometric feature mapping (Isomap) [11], local linear embedding (LLE) [12], and Laplacian eigenmap (LE) [13], can model the manifold of the data in the high-dimensional space. These nonlinear methods do not provide a nonlinear transformation that project new data points to the latent low-dimensional space.

We used locality preserving projections (LPP) to perform the dimensionality reduction for the training motion data. LPP is a linear technique that combines the benefit of both linear and nonlinear methods [8]. It finds a linear mapping function that minimizes the cost function of Laplacian eigenmaps; thus, new demonstrated data can be easily projected to the low-dimensional space by a linear transformation computed by LPP.

In the lower-dimensional space, the k-nearest-neighbored trajectories to the demonstrated motion can be found by computing the similarity between trajectories. One approach to measure the similarity between trajectories is Hausdorff distance. The new robot grasp pose can be obtained by kNN regression. The weight of how much each neighbor trajectory contributes to the regression is computed by the Hausdorff distance.

A. Dimensionality Reduction Using LPP

The problem of linear dimensionality reduction is briefly described as follows based on [8]. Given a set of data points $x_1, x_2, ..., x_n$ in high-dimensional space R^D, find a transformation matrix A that maps these n points to a set of points $y_1, y_2, ..., y_n$ in low-dimensional space R_d ($d << D$), such that $y_i = A^T x_i$, where i is among 1 to n.

Step 1: Construct the adjacency graph. A weighted graph $G = (V, E)$ with n nodes can be constructed from the dataset. An edge is put between the nodes i and j if x_i and x_j are neighbors, which can be defined by either ϵ-neighborhoods ($||x_i - x_j||^2 < \epsilon$) or k-nearest neighbors. We do not choose ϵ-neighborhoods, because choosing an optimal ϵ relies on a good understanding of the data.

Step 2: Choose the weights. W is a sparse symmetric $n \times n$ matrix with w_{ij} having the weight of the edge joining vertices i and j. To better separate the classes, we set $w_{ij} = 0$ if the node x_j is not in the k-nearest neighbor of the node x_i, or they are not in the same class; otherwise, w_{ij} is defined by heat kernel, $w_{ij} = exp(-\left\|x_i - x_j\right\|^2 / t)$, justified by Belkin and Niyogi [16]. Parameter $t \in R$ is the heat kernel factor.

Step 3: Compute eigenmaps. Solve the generalized eigenvector problem:

$$XLX^T \overrightarrow{a} = \lambda XCX^T \overrightarrow{a} \tag{1}$$

where C is a diagonal matrix whose entries are column (or row) sums of the symmetric W, i.e. $C_{ii} = \sum_j w_{ji}$; $L = C - W$ is the Laplacian matrix; the ith column of matrix X is the ith data point x_i. The solution $\overrightarrow{a}$ is the column vector of the transformation matrix A.

Figure 2 shows the three-dimensional representation of the 14 dimensional hand-joint trajectories, captured by a data glove for 12 different grasp types in Cutkosky's taxonomy (Figure 3). Each grasp type has five trials. This demonstrates the ability of LPP to preserve the locality of the nonlinear structure. Although there is partial overlap between two classes (such as the beginning of the motion sequence because the hand is initially open for all grasp types), there is a distinguishable variance among different classes of grasp sequences but little in-class variance.

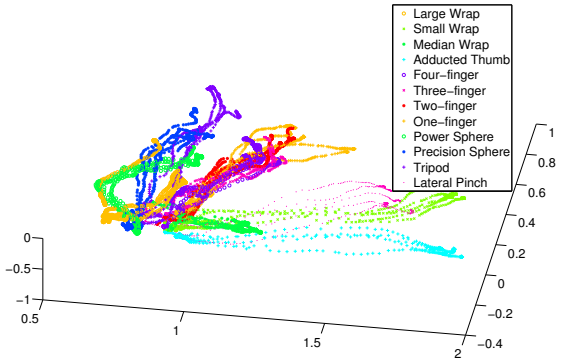

Fig. 2: 3D representation of high-dimensional grasp motion data using LPP.

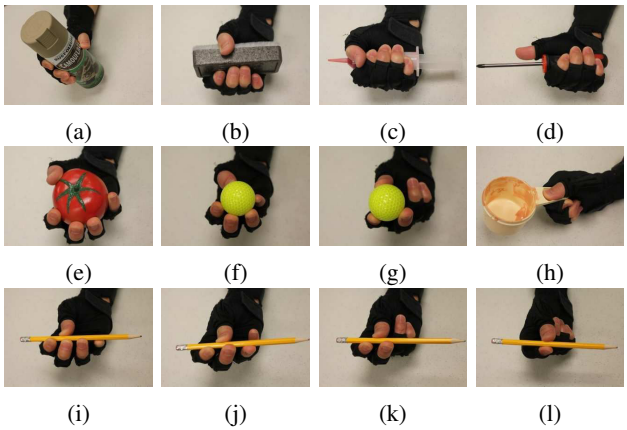

Fig. 3: Twelve human grasp types used for training: (a) large wrap; (b) medium wrap; (c) small wrap; (d) adducted thumb; (e) power sphere; (f) precision sphere; (g) tripod; (h) lateral pinch; (i) four-finger-thumb precision; (j) three-finger- thumb precision; (k) two-finger-thumb precision; (l) one-finger-thumb precision.

B. Similarity among Motion Trajectories

Related approaches include Hidden Markov Model (HMM), Dynamic Time Warping (DTW), and Hausdorff distance. HMM measures the probability that a trajectory belongs to a class of trajectories, demonstrated to be an effective tool to temporal sequence recognition. DTW is an alternative way to measure similarities by aligning sequences that may vary in time; it has been used in matching time series such as speech recognition and economics. Hausdorff distance is used to establish the similarity between two sets or trajectories. In this paper, we chose Hausdorff distance to measure the similarity between trajectories, because Hausdorff distance can be applied in kNN regression for our mapping purpose.

Hausdorff distance is described as follows. Let X and Y be two motion trajectories, the Hausdorff distance from X to Y is represented as:

$$d_h(X,Y) = \max_{x\in X}(\min_{y\in Y}(\|x-y\|)) \tag{2}$$

where x and y are data points in trajectories X and Y respectively. The distance from Y to X is represented as:

$$d_h(Y,X) = \max_{y\in Y}(\min_{x\in X}(\|x-y\|)) \tag{3}$$

The distance between the two trajectories X and Y is defined by:

$$D_H(X,Y) = max(d_h(X,Y), d_h(Y,X)) \tag{4}$$

The Hausdorff distance handles three cases of similarity between grasp motion sequences, illustrated in Figure 4. The two trajectories start from approximately the same position because they share the same initial pose.

Figure 4a demonstrates Case 1, where trajectory Y is roughly a part of trajectory X. This usually happens for the same grasp types but slightly different object sizes. The inter-trajectory distance, therefore, becomes the distance between the end poses of X and Y.

In Case 2 (Figure 4b), trajectory X and Y share the same start and end points but differ in intermediate paths. This usually happens when the two grasp types are different but share a similar end pose, like a lateral pinch and a small wrap, which actually span a larger Euclidean volume in robotic hand configuration space. In this situation, Hausdorff distance is beneficial for distinguishing between two grasp types that share ambiguous grasp poses.

Case 3 (Figure 4c) is the general case, in which trajectory X and Y differ in intermediate paths as well as end points.

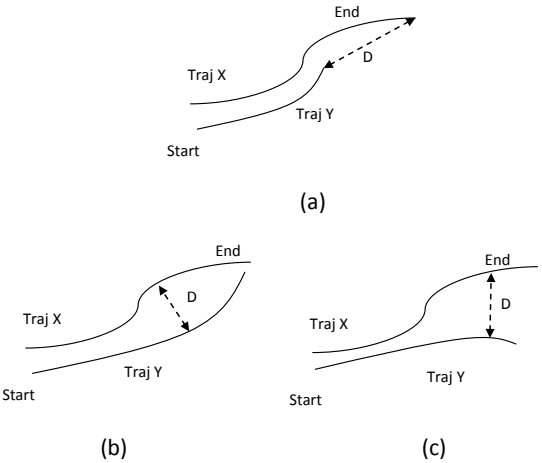

Fig. 4: Three cases of Hausdorff distance between two grasp types. (a): Case 1, trajectory Y is a part of trajectory X; (b): Case 2, trajectory X and Y meet at the end but differ on the way; (c): Case 3, general case, where trajectories X and Y go further away until the end.

Hausdorff distance can also be modified to other metric, such as mean pairwise distance, depending on the applications.

C. Distribution of the Robot Grasping Poses

The corresponding training dataset of robot grasp motions is collected by manually commanding the grasp motions of

the robot hand. The velocity of the motion does not affect the low-dimensional projection because the dimensionality reduction disregards the temporal information (dynamics). No matter what velocity is commanded to the robot, all fingers are closed jointly to form a hand closure. In this paper, we use a Barrett hand to execute the testing results. The Barrett hand has four degrees of freedom – the flexion/extension angle of each of the three fingers and one adduction/abduction angle. The three fingers of the Barrett hand are commanded to the same joint values, so the dimensionality can be reduced to two dimensions. The 2D distribution of grasp types is shown in Figure 5. Since the Barrett hand is much less functional than a human hand, fewer grasp types are defined. For example, four-finger-thumb, three-finger-thumb, and two-finger-thumb (Figure 3) in Cutkosky's taxonomy can be grouped together because the Barrett hand has only three fingers. Overall, the Barrett hand grasps are restricted to six basic types: large wrap, medium wrap, small wrap, power sphere, precision sphere, and two-finger precision. Precision sphere and tripod correspond to the robot grasp type of precision sphere, as depicted in Table I.

TABLE I: Corresponding grasp types between the human and the robotic hand.

Human Grasp Types	Robot Grasp Types
Large Wrap	Large Wrap
Medium Wrap	Medium Wrap
Small Wrap Adducted Thumb	Small Wrap
Power Sphere	Power Sphere
Precision Sphere Tripod	Precision Sphere
Lateral Pinch Four-finger Thumb Three-finger Thumb Two-finger Thumb One-finger Thumb	Precision

From Figure 5, we can observe that the poses of large wrap and precision grasp are close to each other, although they are very different in human configuration space. On the other hand, although small wrap and precision are far away from each other, they are close to each other in human configuration space.

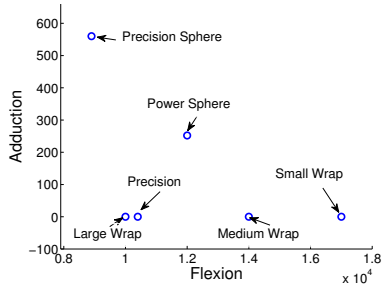

Fig. 5: 2D representation of the Barrett hand grasp types in its configuration space.

D. kNN Regression for Grasp Mapping

Hausdorff distance can be used for kNN classification as well as kNN regression. Here, we choose kNN regression to map a grasp of an unknown object to the robotic hand. Given a testing grasp trajectory q, k-nearest neighbor trajectories $x_1, x_2, ..., x_k$ in the training dataset are found. The weight of each neighbor trajectory contributing to the regression can be defined by:

$$W(q, x_i) = \frac{exp(-D_H(q, x_i))}{\sum_{i=1}^{k} exp(-D_H(q, x_i))} \tag{5}$$

It is clear that the weights defined above will satisfy:

$$\sum_{i=1}^{k} W(q, x_i) = 1 \tag{6}$$

Thus, the corresponding robot grasp pose is defined as:

$$r = \sum_{i=1}^{k} W(q, x_i) r_i \tag{7}$$

where r_i is the robotic hand pose of ith nearest neighbor in the dataset.

III. Experiment

A. Experimental Setup

We measured the sequence of hand motions using a right-handed 5DT data glove 14 Ultra, with 14 fiber-optic-based bend sensors measuring the amount of bending, as shown in Figure 3. The data glove captures proximal interphalangeal (PIP) articulations and metacarpophalangeal (MP) joints for all five fingers and for MP joints between every two neighbored fingers. The flexure resolution is 12 bits for each sensor, and the minimum dynamic range is 8 bits. The sensor does not measure the real joint angles; instead, it measures the proportion of in its full range of motion. The bending values are scaled in between 0 and 1, with 0 being fully open and 1 being fully closed. Hand motions were sampled at a rate of 100 Hz.

In the experiment, 18 everyday objects were tested (Figure 6). During training and testing, the grasps were performed by the same user but on different objects. Four subjects participated in the experiment. During training, the users were asked to apply one of the 12 predefined grasp types, with five trials for each type. During testing, each object was performed by the participants for two trials. In the first trial, the subjects were asked to perform a regular predefined grasp type. In the second trial, the subjects were asked to perform an arbitrary grasp of their own choice that did not necessarily exist in the predefined type. For example, a subject was asked to grasp a box using a large wrap grasp for the first trial. Then, in the second trial, the subject applied his own grasp, as shown in Figure 7a, in which he wrapped four fingers around the box (without the little finger, which was fully closed) rather than applying a regular large wrap using all fingers.

Fig. 6: Daily grasping objects used in the experiment.

To evaluate the regression results using grasp trajectories, we compared our results to regression using grasp poses, where the weights for the kNN regression were obtained from the distance between hand poses. The grasp poses were extracted from the grasp trajectories in the training dataset. Similar to the dimensionality reduction we applied on grasp trajectories, the dimensionality of the 14-DOF hand poses were reduced using LDA. LDA [15] is a popular linear dimensionality reduction approach that projects data points into subspaces, maximizing linear class separability and minimizing the within-class variance.

The regression results were applied to a 6-DOF Fanuc L200IC robotic arm and a Barrett hand. During the execution, the positions and orientations of the robot hand were commanded before the regression results were executed. The strain gauge embedded in each finger measured the joint torque, which provided feedback for the force control. The force control is necessary for the robot to reach a force closure, so it can compensate the error between the force closure pose and the estimated hand grasp pose.

B. Execution Results

We did not evaluate the success rate of the grasp because it is highly related to the positions and orientations of the robotic hand relative to the objects. Instead, we computed the mean absolute percentage error between the estimated robotic hand grasp poses and the final force closure pose, as shown in Table II. The error of regular grasps was computed from the first trial of each object specified as one of the predefined grasp types; the error of irregular grasps was obtained from the second trial of each object that was arbitrarily performed by the user. The results in Table II show that regression using trajectories has a lower error than using static poses for both regular grasps and irregular grasps. The significant difference between errors of trajectory and pose is diminished by averaging over all grasp types. The trajectory and pose methods result in similar mean errors of 5.15% and 5.02%, respectively, for six of the regular grasp types: large wrap, medium wrap and 1-, 2-, 3-, and 4-finger-thumb precision; the mean error of the other 6 grasp types – lateral pinch, small wrap, adducted thumb, tripod, power sphere and precision sphere – is highly decreased to 6.27% by the trajectory approach, compared to an 11.56% error from the pose approach.

TABLE II: Mean absolute percentage error between estimated grasp poses and the best grasp poses.

Testing Grasps	Trajectories	Poses
Regular Grasps	5.72%	8.24%
Irregular Grasps	7.38%	10.47%

Figure 7 illustrates three irregular grasps of a box, a pair of pliers, and a gun. The first column is the human demonstration. Columns 2 and 3 compare 3D representation of 14D demonstrated motion trajectories and poses. The trajectory marked by the black stars is the testing grasp. Column 4 shows the execution results from the robot.

In Figure 7a, the participant demonstrated a large-wrap-like grasp of a box, where the little finger closed more than the other fingers. Figure 7b shows that the trajectory of the demonstrated grasp was similar to the trajectories belonging to the large wrap grasp in the training dataset. The five nearest neighbors of this demonstration were all trajectories in large wrap grasp type. Figure 7c shows that the nearest neighbors of the demonstrated poses were the four-finger precision and tripod grasp types. The regression error of trajectory approach is 4.69%, compared to an error of 12.98% from pose approach.

The second row illustrates a grasp of a pair of pliers. The nearest trajectories belonged to small wrap (Figure 7f), and the nearest poses belonged to small wrap and lateral pinch ((Figure 7g). In this example, small wrap is more desirable. The error in robotic execution (Figure 7h) is 6.56% from trajectory approach, compared to an error of 10.86% from pose approach. The robotic hand needs to close more in order to fit the shape of the pliers.

The third row is the example of grasping a gun. The user employed a medium-wrap-like grasp, but the index finger was put on the trigger. The trajectory of grasping a gun was between 4-finger precision and 3-finger precision (Figure 7j), and the nearest neighbor of the grasp pose is three-finger (Figure 7k). The motion shows a more dexterous motion than wrap grasps, with an error of 6.78% for both methods.

IV. Conclusions

In this paper, we propose a novel approach of mapping based on human grasp motion trajectories rather than grasp poses, since grasp trajectories provide richer information for distinguishing between different grasp types. LPP has been selected to perform dimensionality reduction on the hand grasp data. A mapping was performed between the human and the robotic hand in the low-dimensional subspaces. The Hausdorff distance was used to measure the similarity between trajectories. kNN regression was used to map a new demonstrated grasp into the robotic hand.

Although mean absolute percentage errors, averaged over all grasp types, have less significant difference between regression by trajectory and pose, the trajectory approach greatly improves the ability to distinguish some grasp types

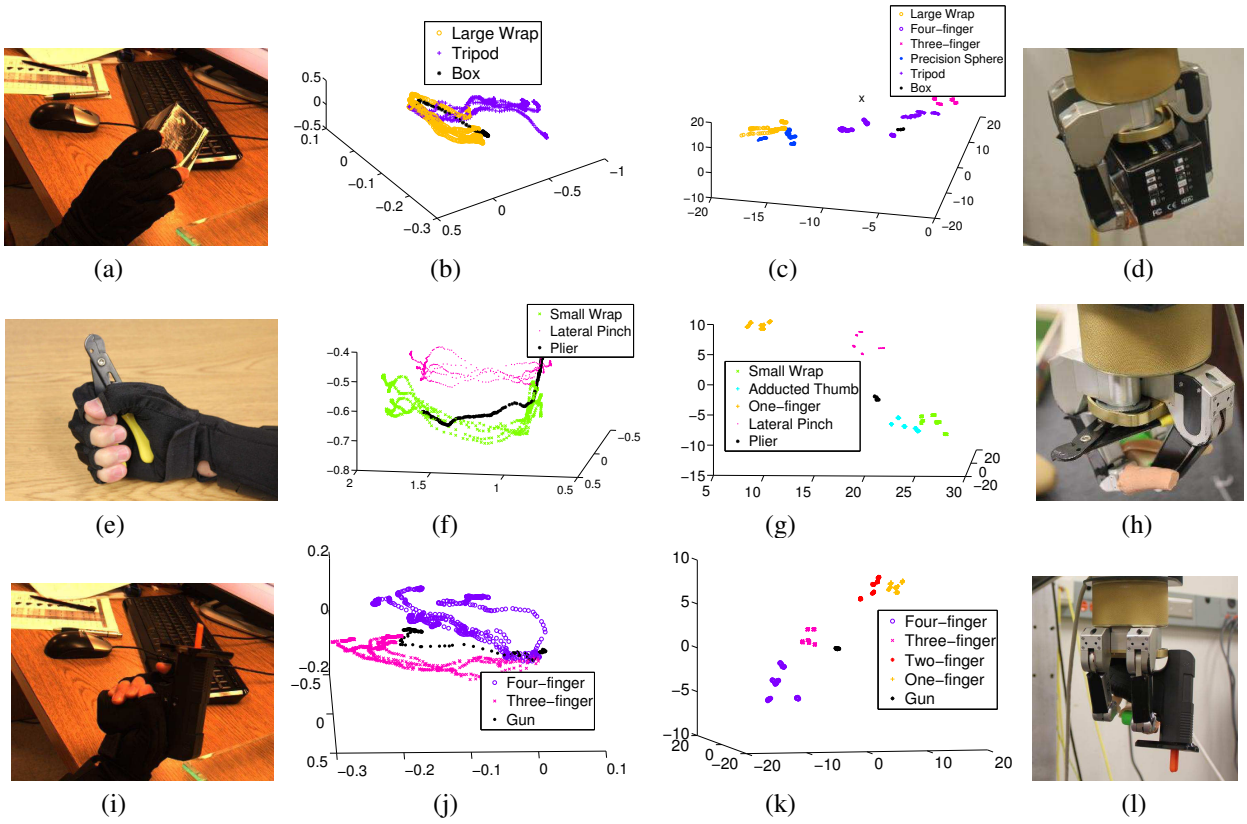

Fig. 7: Three grasp examples: Column 1, human demonstration; Column 2, 3D representation of 14D hand motion trajectories using LPP; Column 3, 3D representation of 14D static hand pose using LDA; Column 4, execution results of regression using hand motion trajectories.

over the pose approach. Experimental results have shown the ability of the proposed method to map the grasp of previously unknown object to the robotic hand. Mapping using the distance between trajectories is more robust than using static poses.

One limitation of the current method is that it is not feasible for new users because of the different geometry between human hands. For every user, a training process is necessary to guarantee a low regression error. To tackle this problem, calibration is needed to map from the data glove sensor values to the joint values.

REFERENCES

[1] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, Robot Programming by Demonstration, *Handbook of robotics*, Springer, 2007.

[2] M. Fischer; P. van der Smagt; G. Hirzinger; , "Learning techniques in a data glove based telemanipulation system for the DLR hand," Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on , vol.2, no., pp.1603-1608 vol.2, 16-20 May 1998

[3] A. Peer; S. Einenkel; M. Buss; , "Multi-fingered telemanipulation - mapping of a human hand to a three finger gripper," Robot and Human Interactive Communication, 2008. RO-MAN 2008. The 17th IEEE International Symposium on , vol., no., pp.465-470, 1-3 Aug. 2008

[4] M. Cutkosky, "On grasp choice, grasp models and the design of hands for manufacturing trasks," IEEE Transactions on Robotics and Automation, 1989

[5] Guido Heumer, Heni Ben Amor, and Bernhard Jung;, Grasp recognition for uncalibrated data gloves: A machine learning approach. Presence: Teleoper. Virtual Environ. 17, 2, Apr. 2008

[6] J. Aleotti, S. Caselli;, "Grasp Recognition in Virtual Reality for Robot Pregrasp Planning by Demonstration", Proceedings. 2006 IEEE International Conference on , 2006.

[7] G. Heumer; H.B. Amor; M. Weber; B. Jung; , "Grasp Recognition with Uncalibrated Data Gloves - A Comparison of Classification Methods," Virtual Reality Conference, 2007. VR '07. IEEE , vol., no., pp.19-26, 10-14 March 2007

[8] X. He, P. Niyogi, Locality preserving projections, in: Proc. Intl. Conf. Advances in Neural Information Processing Systems, 2003.

[9] R. Tyrrell Rockafellar, Roger J-B Wets, Variational Analysis, Springer-Verlag, 2005, pg.117.

[10] S. Ekvall; D. Kragic; , "Grasp Recognition for Programming by Demonstration," Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on , vol., no., pp. 748-753, 18-22 April 2005

[11] J.B. Tenenbaum. Mapping a manifold of perceptual observations. In Advances in Neural Information Processing Systems, volume 10, pages 682-688, Cambridge, MA, USA, 1998. The MIT Press.

[12] S. Roweis, L. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323-2326.

[13] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, in: Proc. Intl. Conf. Advances in Neural Information Processing Systems, 2001, pp.585-591.

[14] H. Hotelling. Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology, 24:417-441, 1933.

[15] R.A. Fisher. The use of multiple measurements in taxonomic problems. Annals of Eugenics, 7:179-188, 1936.

[16] M. Belkin and P. Niyogi, Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering , Advances in Neural Information Processing Systems 14, Vancouver, British Columbia, Canada, 2002.