The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Toward a vision based hand gesture interface for robotic grasping

Raghuraman Gopalan and Behzad Dariush

Abstract— **The challenging problem of planning manipulation tasks for dexterous robotic hands can be significantly simplified if the robot system has the ability to learn manipulation skills by observing a human demonstrator. Toward this goal, we present a novel computer vision based hand posture recognition system to serve as an intelligent interface for skill transfer in robotic manipulation. We use the Inner Distance Shape Context (IDSC) as a hand shape descriptor to capture variations in the hand state (open or closed) under large in-plane rotations and considerable out-of-plane rotations. The proposed technique is further examined in applications involving grasp recognition and gesture based communications. The experiments show that the proposed approach can be generalized to recognizing a selected taxonomy of grasp types. At present, skin color is used to segment the hand region from the scene, but this method has its own limitations. We show preliminary results suggesting that the IDSC can be used to segment parts of the articulated object, including segmenting the hand from the human body silhouette without using skin color information.**

I. Introduction

Future robots promise to become an integral part of our everyday lives, serving as caretakers for the elderly and disabled, providing assistance in homes and offices, and assisting in surgery and physical therapy. For this to happen, programming must become simpler, and movements more natural and human-like. In response to this challenge, there has been a growing interest in using captured human motion data as examples to simplify the process of programming or learning complex robot motions [1], [2]. Transferring gross whole-body human motion to humanoids has been well studied and off-line algorithms have been developed using prerecorded motion capture data. Methods which use prerecorded motions are not applicable for interactive, gesture driven applications where a robot is tele-operated by human gestures. To address this issue, Dariush et al. demonstrated an online, vision based system to retarget whole body motion to the humanoid robot ASIMO [3].

The aforementioned whole body retargeting systems do not consider hand articulations in their motion transfer protocols. However, skill transfer focusing on only manipulation tasks based on human hand observations has been examined by other researchers [4], [5]. Such studies have suggested that the time required to add grasping behaviors may be significantly reduced if the robot is capable of transferring manipulation skills from a human demonstrator. Typically, the human demonstrator is instrumented with a data glove while performing the example grasp [6]. Sensors attached to the data glove may directly measure the articulation angles or the Cartesian positions of selected feature points on the glove. Although direct measurement of the glove configuration simplifies the sensing required to detect important grasp features, the glove often obstructs the demonstrators contact with the object and may prevent a natural grasp. Moreover, calibration and adjustments for proper fit for different size hands is required to ensure accurate measurements.

In liu of using a data glove, marker based motion capture has been used to record hand articulations, particulary in computer animation applications [7]. Reconstructing the complete hand posture can still be a challenge with marker based methods due to partial marker occlusions. To minimize the effects of marker occlusions, Chang et. al [8] proposed an algorithm to determine the minimal set of hand markers needed to represent the type of grasp performed by the demonstrator. The reduced marker protocol simplified the capture procedure and described the hand configuration in a low-dimensional space. Although marker based methods are less obtrusive than a data glove, the data collection process is time consuming and requires considerable calibration in an instrumented and controlled environment.

This paper explores a vision based hand gesture interface, whereby hand states (open/close) and a class of hand postures in a taxonomy of grass types can be detected and recognized with a single passive camera. Inferring the full articulations of the fingers form a single camera is a challenging problem due to the complexity of the hand articulations, the occlusions of the fingers, and complications in segmentation of the hand from the background image. In the past, several researchers have developed hand pose estimation methods for vision-based gesture interfaces. Wu and Huang [9] provide a survey of vision based approaches for hand posture recognition. Specifically, there are algorithms that deals with view-invariance [10], recognition under complex backgrounds [11], adaptive learning using SIFT features [12] among many others [13], [14], [15], [16].

In this paper, we focus on recognizing general hand states, invariant to viewpoint using an Inner Distance Shape Context descriptor (IDSC) [17]. The IDSC is invariant to translation, scale and small affine distortions. The IDSC descriptor is particularly compelling for use in describing articulated shapes, such as the hands, because of its ability to capture part structures. This property is important since hand shapes exhibit variations in the organization of its part structures. To our knowledge, this work is the first to use the IDSC for hand posture recognition.

This work was performed while the first author was at Honda Research Institute.

R. Gopalan is in the Department of Electrical and Computer Engineering, University of Maryland, College Park MD, USA raghuram@umiacs.umd.edu

B. Dariush is with Honda Research Institute, Mountain View CA, USA, dariush@hra.com

This paper is organized as follows. We present the proposed algorithm in Section II, followed by the experimental results of hand posture recognition, including grasp taxonomy and sign language classification, in Section III. We then, in Section IV, address a vital pre-processing problem of extracting the hand region from the human body (in a very generic setting), before it can be represented using the IDSC descriptor. Specifically, we introduce an interesting viewpoint of using the Inner distance shape context to detect keypoints/ segments of the human body, by analyzing the patterns of their IDSC shape description across different poses. We then discuss some relevant issues and challenges in Section V and conclude the paper in Section VI.

II. Algorithm

An overview of the proposed approach is described as follows. The hand (or palm) region is segmented from the images (or video) using skin color segmentation. The segmented region is then described using Inner Distance Shape Context (IDSC) signatures, which is a histogram of the contour points in the log-polar space that describes how each point is related to all other contour points in terms of distance and angle. With this feature representation as the input, we first address the problem of hand state classification. This is essentially a two class pattern matching problem - open or closed, and the classification is done using the Support Vector Machines (SVM) [18] classifier.

Before classification, we introduce a pre-processing step to group the training and test images based on their primary orientation direction, which is estimated by computing the scatter direction of the images through principal component analysis (PCA) [19]. The training images are then grouped into different bins (spread uniformly into 10 intervals in the 0 - 180 degree range). We then train the SVM offline (for each orientation bin) using the IDSC features of the different hand state instances of the corresponding orientation. Finally, the test image is projected onto the appropriate SVM for classification. We then extend this framework to classify a more generic set of grasp taxonomy shapes. The following sub-sections describe the details of the proposed algorithm.

A. Skin color based segmentation

Proper hand segmentation is a critical step in this process. We use skin color as the cue to segment the hand. Color based segmentation is a very well known method in computer vision [e.g., [20]]. In line with previous techniques, we build a Gaussian models of skin regions corresponding to the hand and non-skin regions, and measure how the pixels in the test image correlate with the models. Normalized color space was used in this process. We worked on the RGB color space, and the Gaussian mixture models were created based on the normalized R and G components of the training image pixels. For instance, each pixel was represented by a vector,

$$Y = \begin{bmatrix} R/(R+G+B) \\ G/(R+G+B) \end{bmatrix}, \quad (1)$$

where R,G, and B are the red, green and blue components of that pixel. Pixels Y_i corresponding to similar regions (skin/ non-skin) are grouped together from the training images as

$$X(i) = [Y_1 \quad Y_2 \quad \cdots \quad Y_N], \quad (2)$$

where $i = \{1, 2\}$ (skin/non-skin), and N represents the number of pixels. The mean value and covariance of the N pixels are computed to build the Gaussian models

$$\begin{aligned} \mathrm{N}(\mu_1, \Sigma_1) &\rightarrow skin \\ \mathrm{N}(\mu_2, \Sigma_2) &\rightarrow non-skin. \end{aligned} \quad (3)$$

The test pixels are then classified as belonging to the skin class (or otherwise), depending on their strength of affinity to the two Gaussian models. We cast this classification problem into a MAP (maximum-a-posteriori) framework. This fits the problem into a well known Bayesian paradigm of expressing the posterior probability as a function of the likelihood function and prior probability. Formally,

$$p(\theta|X) = \frac{p(X|\theta)\ p(\theta)}{p(X)} \quad (4)$$

where $p(\theta|X)$ is the posterior distribution (i.e. probability that a given test pixel will belong to the class θ (here, skin or non-skin), $p(X|\theta)$ is the likelihood function (measure of affinity of a pixel for a particular class) and $p(\theta)$ is the prior probability (normal occurrence rate of a particular class).

In our framework, equal priors were assumed. So, for a two class problem, a pixel (X) is said to belong to class1 if,

$$p(\theta_1|X) > p(\theta_2|X) \quad (5)$$

$$p(X|\theta_1)\ p(\theta_1) > p(X|\theta_2)\ p(\theta_2) \quad (6)$$

$$p(X|\theta_1) > p(X|\theta_2) \quad (7)$$

The likelihood function which is used for decision making, is computed as follows,

$$P(X|\theta_{skin}) = \frac{1}{|\Sigma_1|^{1/2}(2\pi)^{n/2}} e^{-\frac{1}{2}(X-\mu_1)^T\Sigma_1^{-1}(X-\mu_1)}$$

$$P(X|\theta_{non-skin}) = \frac{1}{|\Sigma_2|^{1/2}(2\pi)^{n/2}} e^{-\frac{1}{2}(X-\mu_2)^T\Sigma_2^{-1}(X-\mu_2)}$$

Thus, if $p(X|\theta_{skin}) > p(X|\theta_{non-skin})$,the test pixel is classified as skin region, or otherwise. This process is done for every pixel in the test image to obtain the skin segmentation output as shown in Figure 1. The segmented

Fig. 1. Left-to-Right: Input image, Skin segmentation map, cropped hand region.

result is then subjected to morphological operations such as dilation to fill-in the pixels that could possibly be mislabeled.

Dilation is similar to low pass filtering that smooths the segmented results to maintain regional homogeneity. The dilation process makes sense intuitively because, the hand regions are mostly continuous and well separated from the background. But, care must be taken so that the low pass filtering shouldn't connect two separated fingers. So we used a 3×3 low-pass filter to achieve this objective. The resultant skin/ non-skin map is then automatically cropped to yield the hand (or palm) region.

B. Scatter Direction Estimation

We then estimate the primary scatter direction of the segmented hand image, to aid the classifcation process. This idea stems from the observation that the hands have a primary orientation direction, be it open or closed. This information can be used to group the hand images into different blocks and then classified separately. A simple way to estimate this primary orientation direction is through principal component analysis (PCA). This well known technique [19] in computer vision projects the data along the direction of maximum scatter.

In a nutshell, the PCA algorithm solves the generalized eigen-value problem and computes the eigenvectors from the covariance matrix of the input images. The eigenvectors (corresponding to large eigenvalues) represents the directions of maximum scatter of the data. So, given a segmented hand image, we can estimate its scatter direction based on the co-ordinates of the eigenvector that has the maximum eigenvalue. This can be summarized by the following representative equations,

Consider a set of N sample points of the hand region $\{X_1, X_2, \cdots, X_N\}$, whose values are their corresponding 2D locations. We use PCA to estimate the direction of maximum scatter by computing a linear transformation W^T. We do so by computing the total scatter matrix and solving for the generalized eigenvalue problem. The total scatter matrix is given by

$$S_T = \sum_{k=1}^{N} (X_k - \mu)\ (X_k - \mu)^t \tag{8}$$

where N represents the number of sample points, and μ is the mean location of all the samples. The projection matrix W_{opt} is chosen such as to maximize the determinant of the total scatter matrix of the projected samples, (i.e),

$$W_{opt} = argmax|W^T S_T W| = [W_1 \quad W_2] \tag{9}$$

where, W_1 and W_2 are the set of 2 dimensional eigenvectors. In our case, the eigenvector (say,W_{eig}) corresponding to the maximum eigenvalue gives the direction of maximum scatter. The estimate of the scatter direction is then computed by

$$tan^{-1} \frac{W_{eig}(Y)}{W_{eig}(X)} \tag{10}$$

At this point we have the segmented hand image, and a rough estimate of the hand scatter direction (Figure 2). We then proceed to describe the hand shape using the Inner Distance Shape Context descriptor.

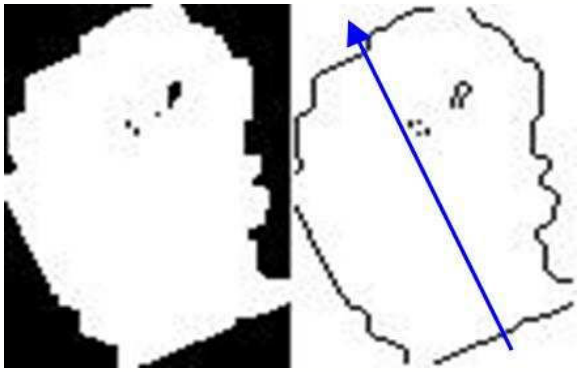

Fig. 2. Segmented hand image, and its primary scatter direction estimate obtained through PCA.

C. IDSC descriptor

The shape context is a descriptor used to measure similarity and point correspondences between shapes [21]. It describes each point along the object's contour with respect to all other points in the contour. Suppose there are n points on the contour of a shape. For the point p_i, the coarse histogram h_i of the relative coordinates of the remaining $n-1$ points is defined to be the shape context of p_i.

$$h_i(k) = \#\{q \neq p_i : (q - p_i) \in bin(k)\} \tag{11}$$

The histogram is computed based on both distance and angle for each point on the contour, with respect to all other points on the contour. The bins are normally taken to be uniform in log-polar space. The key feature of shape context is that it captures the distribution of each point relative to all other points in the contour, thereby resulting in a robust, compact, and highly discriminative description. This shape descriptor has seen widespread use in shape matching applications in computer vision.

The inner distance shape context descriptor (IDSC) is an intuitive extension of the original shape context descriptor, and it is very useful for articulated objects [17]. The IDSC, proposed by Ling et al, primarily differs from shape context in the way the distance and angle between the contour points are computed (as shown in the figure 3 and adopted from [17]). The shape context uses a normal L2 distance measure, whereas the IDSC computes the distance between the points along a path that travels within the object's contour. The angular relation in IDSC was also measured interior to the object's contour, termed as the inner angle. The inner angle was computed by taking the direction between the contour tangent at the start point and the direction of the inner distance originating from it. This is much useful for articulated objects like hand, human body etc. Much of the hand grasp shapes are different by the way in which the fingers are spaced. So IDSC based classifier has tremendous promise in this regard as illustrated in Figure 4.

The IDSC descriptor was derived for the hand images by the following mechanism. For each segmented image, the points were sampled along the contour, and the histogram was computed based on the inner distance and the inner

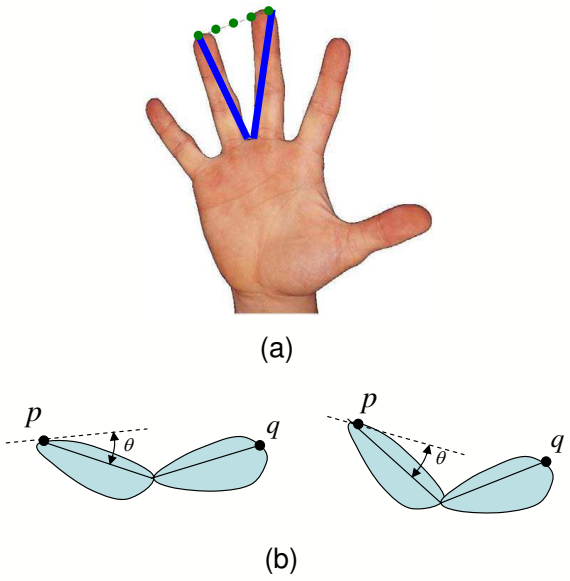

Fig. 3. (a) Inner distance denoted by solid-blue line. Normal shape context distance denoted by dotted green lines. b) The inner angle between the points p and q remain invariant under shape deformation.

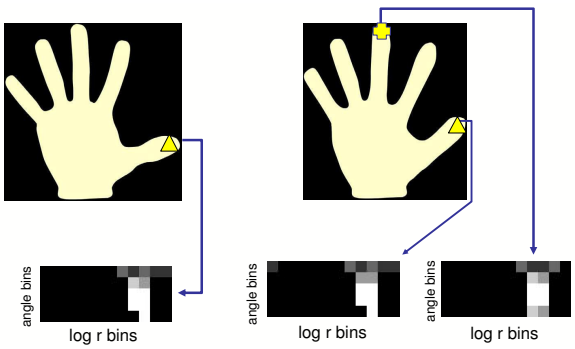

Fig. 4. The inner distance shape contexts of the same thumb point in the two hand images (denoted by a triangle) are very similar even under some shape deformation. The shape context of a point on the middle finger is very different from the shape contexts of the thumb points.

angle, resulting in a vector for each contour point. This resultant matrix will be our description of the input images, and will be given as the feature on which the SVM will be trained and tested.

D. Classification using Support Vector Machines

Support Vector Machines (SVM) is a statistical technique which is well known for classifying two-class problems [18]. The SVM's are trained using labeled data (IDSC features belonging to both open hand state and closed hand state). SVM's then attempt to find a linear separating hyperplane that separates the data (as shown in Figure 5 and adopted from [18]).

If x_i are the training instances, and y_i are their corresponding labels, the SVM tries to find an optimal separating hyperplane that satisfies the following equation:

$$y_i(x_i \cdot w + b) \geq 0 \qquad (12)$$

for all i, where w is the normal to the hyperplane and $|b|/||w||$ is the perpendicular distance of the hyperplane from the origin

But in practice, the data may not be linearly separable. But the hope is, such data that are linearly non-separable in their original dimension, can become well separated in a higher dimensional space. So, the SVM projects the data into a higher dimensional space using kernels, to find the best linear separating hyperplane that classifies the data with very few errors. In this process, the algorithm identifies the training samples that are crucial in separating the two classes as the 'support vectors' and bases the further classification on these vectors.

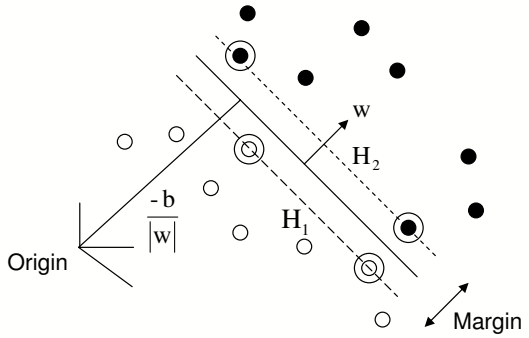

Fig. 5. Linear separating hyperplane for the two class problem (white and black dots). The rounded dots show the support vectors identified by the SVM. The margin is desired to be as large as possible.

III. Experiments

A. Hand State Recognition

We conducted several experiments to test and validate the use of IDSC descriptor to recognize the hand state. We trained the SVM using the iDSC descriptions of open/ closed hand shapes, about 50 examples per state. The RBF kernel was then used to map the data to the higher dimension, and the LIBSVM [22],outine was used to perform this classification task. Once the SVM is trained, we tested the algorithm with eight different videos with the individuals performing different routines with open/ closed hands. The video was sampled on the frame rate, and the resulting images were segmented using skin color to obtain the hand region. The IDSC descriptor was then obtained for the segmented hand region and projected onto the SVM (corresponding to its primary orientation direction category) for classification. Some sample images used in our experiments are given in Fig 6.

We obtained between 85 to 95 % recognition on the eight datasets. The experiment setting also contained very high in-plane rotations (upto +/-180 degree) and substantial out-of-plane rotation (upto +/- 45 degree). Given below (in Fig 8) is the confusion matrix for the hand state recognition experiment.

Fig. 6. Top row: training exemplars for closed hand state; Middle row: training exemplars for open hand state; Bottom row: sample test images

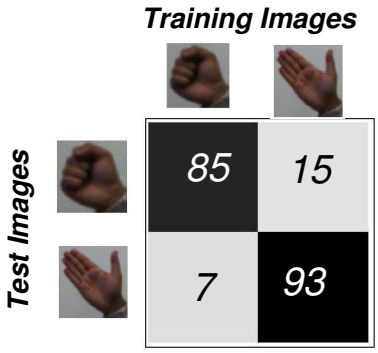

Fig. 7. Confusion matrix: Rows: testing (1:open, 2:close) Columns: training(1:open, 2:close). Key: darker blocks signify more number of correct matches.

B. Grasp taxonomy / Sign language pattern matching

The results of the previous section suggest that the IDSC shape descriptor is very effective for hand state recognition. We conducted more experiments to assess the generalizability of the algorithm in classifying more complex hand shape patterns. In particular, we applied our algorithm for recognizing hand postures used in grasping as well as hand sign language. Since such applications involve solving the N-class pattern matching problem (where N is the total number of classes), N SVM's were used in one-against-all configuration. Otherwise, similar training and testing procedures were followed. For grasp recognition, we examined a subset of the taxonomy of grasps proposed by Cutkosky et. al. [23]. In particular, we considered four grasp signatures illustrated in Figure 8. We recorded videos of

Fig. 8. Grasp patterns (L-R): Small diameter grasp, four-finger thumb grasp, Precision Disc grasp, Platform grasp.

three different people demonstrating each of the four grasp categories in different viewing poses from the camera. The poses contained substantial in-plane rotations.

The methods of the previous section were used to extract the IDSC descriptors. For classification, the leave one out strategy was used. So we have three such settings, and it resulted in 84% recognition rates on an average. The confusion matrix given in Figure 9. For sign language pattern

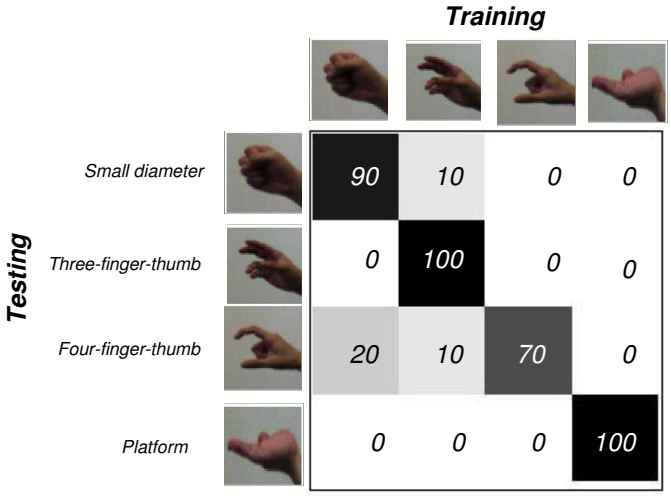

Fig. 9. Rows: testing (1:large diameter, 2:small diameter, 3: four-fingered thumb, 4: platform) – Columns: training(1:large diameter, 2:small diameter, 3: four-fingered thumb, 4: platform

matching, 8 different sign languages (adopted from [24] were matched against one another. Similar experimental setup like that of grasp taxonomy was used, resulting in 80% classification accuracy. A representative set of sign language patterns is given in Figure 10 and its corresponding confusion matrix is provided in Figure 11.

Fig. 10. Sign language patterns (L-R): One, two, three, four, five, follow me, call, thumbs up.

In these experiments, the training and test data are not always similar as shown in Figure 12. The subjects were free to rotate their hands during the collection of both training and testing data. This is in sharp contrast with most of the existing hand gesture recognition algorithms wherein all the subjects are required to perform identical gestures for classification. In this sense, the results reported in our study are much more generalizable to the real world settings, and it is not person dependent. So these results are very encouraging and can readily be applied to tele-robotic grasping applications. It is also interesting to see the impact of a larger training set on these algorithms.

IV. Hand Region Segmentation

As discussed previously, segmentation of the hand is a critical first step in developing a practical system for recognizing hand gestures. Our interest is in developing a system to transfer whole body motions including the posture

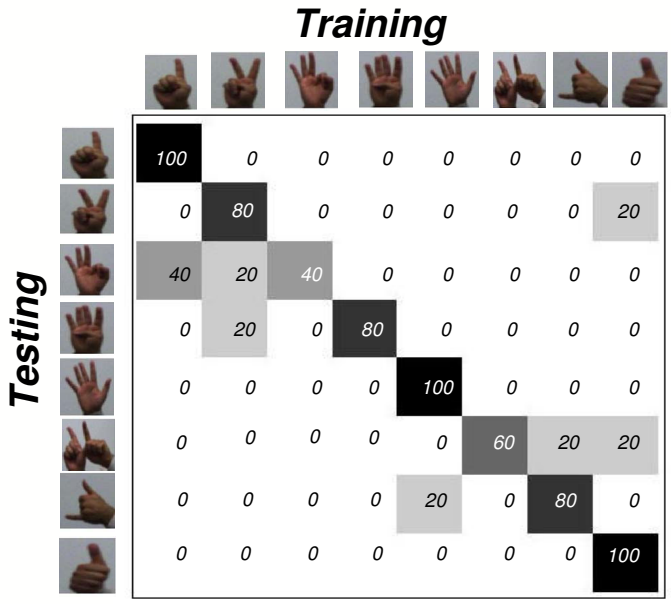

Fig. 11. Rows: testing (1:one, 2:two, 3: three, 4: four, 5: five, 6: follow me, 7: call, 8: thumbs up) – Columns: training(1:one, 2:two, 3: three, 4: four, 5: five, 6: follow me, 7: call, 8: thumbs up)

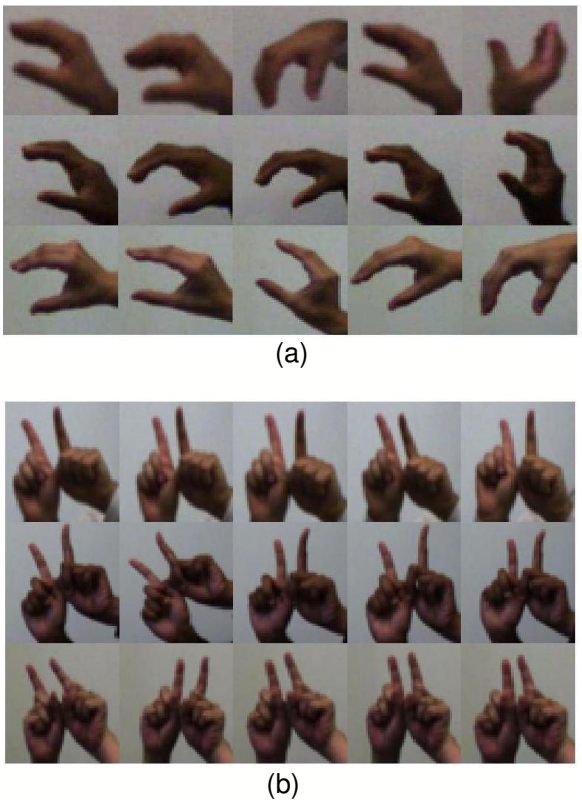

Fig. 12. Experimental data variations - A) grasp pattern, B) sign language; 1st and 2nd rows: training data for a particular pattern, 3rd row: the corresponding test data.

of the upper and lower extremity limbs as well as the posture of the hands.

Extracting the hand region from the rest of the body is very important. In the experimental results presented in previous sections, the subjects were asked to wear full-sleeved shirts so that we could use skin color to segment out the hand (palm) from the rest of the arm region.

A more principled approach would be to try and fit models corresponding to the hand (palm) and the arm region based on the skin based segmentation result. There can be instances wherein the use of skin-color as the segmentation cue, could fail. In such cases, depth based segmentation promises to be a viable alternative. Use of depth images can also help in another setting that we haven't dealt with here: which is, occlusion. Depth images (in combination with skin color) can be used to overcome occlusions resulting in stable segmentation. Given such alternatives, we here propose a novel way of segmenting the parts of an articulated object using the shape description based on the Inner Distance Shape Context.

A. IDSC for segmentation - Motivation

Recall that the IDSC descriptor gives a holistic definition of the contour points relative to all other points in the image, in terms of the inner distance and the inner angle. Given two images of the same object under different articulations, the IDSC of the points belonging to the same 'part' of the object (as shown by the points denoted by $\triangle$ on the thumb in two images in Figure 4) produce nearly identical IDSC. Whereas, the points corresponding to different parts of the object produce dissimilar IDSC signatures (as illustrated by the symbols $\triangle$ and $+$ in Figure 4).

Motivated by this observation, we would like to see if: given the silhouette of the human body, whether the points corresponding to the hand region will have similar IDSC descriptions irrespective of the amount of articulations that the body undergoes. First up this makes sense intuitively because, since the IDSC describes a point relative to all the other points, it effectively captures the information about the points with a good sense of the overall configuration of the object. This then leads us to a broader question: Can inner distance shape context be used for key-point detection, and hence, segmentation of the parts of an articulated object? This is a very challenging problem, and has wide applications in human body part detection, face and hand landmark detection - all of which have a great impact on object-part segmentation.

B. IDSC for segmentation - Algorithm

The segmentation algorithm is formulated as follows. Given a silhouette of a human performing an action, we're interested in reliably detecting certain anatomical points (key-points) which are of interest for segmentation. These key-points can be, for example, the hands, elbows, shoulders, and the head. During training, these seven key-points are manually localized under different body poses and their associated IDSC descriptions are computed and stored as the

test gallery. Then, given the silhouette of a test image in a new body pose, its IDSC signature is analyzed to determine the contour point with an IDSC description that is most similar to that in the gallery.

Specifically, if $IDSC\_Gallery(\mathbf{j})$ (where $\mathbf{j} = \mathbf{1}...\mathbf{7}, \mathbf{j} = [\mathbf{j_1}, \mathbf{j_2}, ..., \mathbf{j_M}]$ with M denoting the number of IDSC feature examples for that particular gallery) represent the IDSC gallery created for each of the seven key-points we are interested in. Then, given a test instance X of the human silhouette, N points $(x_1, x_2, \cdots, x_N)$ are sampled along its contour to compute its IDSC, say IDSC(X) where $X = [x_1, x_2, \cdots, x_N]$. Then for $j = \mathbf{1} \cdots \mathbf{7}$ (corresponding to the key-points), D_j (the point in X that has the IDSC description similar to the $IDSC\_Gallery(\mathbf{j})$) is identified by

$$D_j = \min_{i=1\cdots N}(\min_{k \in S}(\|IDSC(x_i) - IDSC\_Gallery(j_k)\|_1)) \quad (13)$$

where S is the number of IDSC descriptors in Gallery '$\mathbf{j}$'.

C. IDSC for segmentation - Experiments

In our experiments, we worked with two human body motion sequences (with no self occlusion of the human body parts), wherein one sequence is significantly different from the other. Snapshots from each sequence is shown in Figure 13, where the top. Sample keypoints were hand-marked from representative frames of the sequences to create the gallery. The algorithm was then tested in one(sequence)-against-another framework, with the standard correlation distance measure used for classification. We achieved 85% detection rate with a eight-pixel neighborhood support, when compared with the ground truth (indicating, we accept the keypoint detection to be correct if the location of the detected keypoint lies within a 4×4 region centered around its actual location specified by the ground truth). Sample detection result is given in Figure 13.

To our knowledge, this is the first attempt in looking at the use of IDSC descriptor for keypoint detection and segmentation. Most of the work in the literature uses shape context based descriptors to do shape matching. But, this new way of looking at the shape context is very exciting and the results obtained are very encouraging, given that the gallery had considerably different poses from that of the test images. This new result has a potentially good impact in using the 'shape description' of the object for keypoint detection. The detected keypoints give a good estimate of different regions of the objects, which in turn gives a good head-start for segmentation. Hence, this kind of approach can readily be used to segment the hand region from the body, which is very important for robotic grasp recognition systems!

V. FUTURE WORK

The IDSC descriptor is found to be very efficient in modeling the hand shapes across variations in size and rotation. It also has shown robust performance in detecting the key-points to aid the segmentation task. But, to achieve the bigger goal of classifying hand grasp patterns in more practical situations that allow ripe probability of self-occlusions between the palm and arm, and significant out-of-plane rotations, the formulation of 3D variations of IDSC looks to be an interesting and challenging problem. The 3D inner distance shape context may be used in tandem with depth-skin color segmentation to detect key-points and describe the grasp patterns. Another area of interest is to use the motion information from the video to predict the hand grasp patterns. It is quite natural for humans to follow some good pattern in any grasping task. This 'predictive' motion information can help resolve ambiguities in vision based shape description. It is very exciting to investigate this feedback mechanism of shape and motion for robotic grasping applications.

VI. CONCLUSION

In summary, an important contribution of this work is to show that IDSC descriptor can be used to capture the variations in hand postures across viewpoints. It is quite encouraging that this descriptor can handle pose variations in a very generic experimental setting. Another finding of this work is the novel way of looking at IDSC descriptions for detecting key-points from articulated shapes. This conceptual way of using a shape descriptor to identify different parts of an articulated object has wide spread usage in different segmentation tasks, such as segmenting the hand from the body. The above experiments promise very good potential for our approach in the interesting area of hand grasp recognition.

REFERENCES

[1] A. Nakazawa, S. Nakaoka, K. Ikeuchi, and K. Yokoi. Imitating human dance motions through motion structure analysis. In *Intl. Conference on Intelligent Robots and Systems (IROS)*, pages 2539–2544, Lausanne, Switzerland, 2002.

[2] S. Schaal. Learning from demonstration. In M.C. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, chapter 9, pages 1040–1046. MIT Press, 1997.

[3] B. Dariush, M. Gienger, A. Arumbakkam, C. Goerick, Y. Zhu, and K. Fujimura. Online and markerless motion retargeting with kinematic constraints. In *Int. Conf. Intelligent Robots and Systems(IROS)*, pages 191–198, Nice, France, 2008.

[4] K. Ikeuchi and T. Suehiro. Toward an assembly plan from observation. *IEEE Trans. Robot. Automat.*, 10(3):368–385, 1994.

[5] S.B. Kang and K. Ikeuchi. Toward automatic robot instruction from perception-mapping human grasps to manipulator grasps. *IEEE Trans. Robot. Automat.*, 13(1):81–95, 1997.

[6] S. Ekvall and D. Kragic. Grasp recognition for programming by demonstration. In *Int. Conf. Robotics and Automation (ICRA)*, pages 748– 753, 2005.

[7] N. Pollard and V. B. Zordan. Physically based grasping control from example. In *ACM SIGGRAPH/Eurographics Symp.on Computer Animation*, pages 311–318, 2005.

[8] L. Chang, N. Pollard, T. Mitchell, and E. Xing. Feature selection for grasp recognition from optical markers. In *Intelligent Robots and Systems (IROS)*, pages 2944–2950, San Diego, CA, 2007.

[9] Y. Wu and T.S. Huang. Vision-Based Gesture Recognition:A Review. *Lecture Notes in Computer Science*, 1739:103, 1999.

[10] Y. Wu and T.S. Huang. View-Independent Recognition of Hand Postures, 2000.

[11] J. Triesch and C. von der Malsburg. A System for Person-Independent Hand Posture Recognition against Complex Backgrounds. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, pages 1449–1453, 2001.

[12] C. Wang and K. Wang. Hand Posture Recognition Using Adaboost with SIFT for Human Robot Interaction. *LECTURE NOTES IN CONTROL AND INFORMATION SCIENCES*, 370:317, 2008.

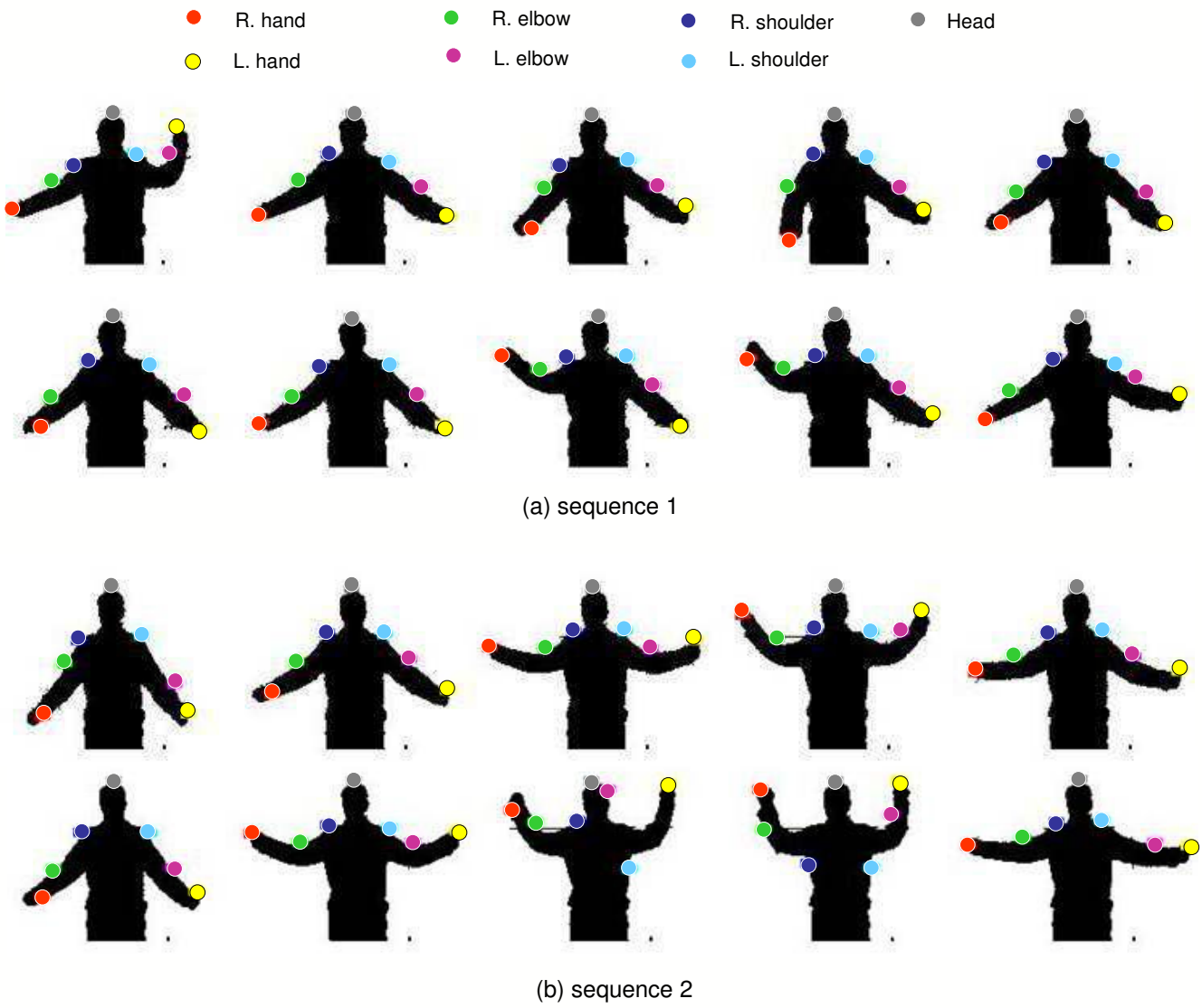

Fig. 13. Human motion sequences and detected keypoints. Top: Sequence 1, Bottom: Sequence 2.

[13] J. Rehg and T. Kanade. Visual tracking of high dof articulated structures: An application to human hand tracking. In *3rd Eur. Conf. Computer Vision (ECCV '94)*, pages 35–46, 1994.

[14] M. Imai E. Ueda, Y. Matsumoto and T. Ogasawara. A handpose estimation for vision-based human interfaces. *IEEE Trans. Ind. Electron.*, 50(4):676684, 2003.

[15] V. Athitsos and S. Sclaroff. Estimating 3d hand pose from a cluttered image. In *Computer Vision and Pattern Recognition (CVPR '03)*, pages 432–439, 2003.

[16] C. Schwarz and N. Lobo. Segment-based hand pose estimation. In *2nd Canadian Conf. Computer and Robot Vision, May 2005*, pages 42–49, 2005.

[17] H. Ling and D.W. Jacobs. Shape Classification Using the Inner-Distance. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, pages 286–299, 2007.

[18] C.J.C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[19] M. Turk and A. Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition (CVPR 91)*, pages 586–591, 1991.

[20] R.L. Hsu, M. Abdel-Mottaleb, and A. Jain. Face detection in color images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(5):696–706, 2002.

[21] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. *IEEE Trans. Pattern Analysis and Machine Intel. (PAMI)*, pages 509–522, 2002.

[22] C.C. Chang and C.J. Lin. LIBSVM: a library for support vector machines. *Software available at http://www. csie. ntu. edu. tw/cjlin/libsvm*, 80:604–611, 2001.

[23] MR Cutkosky. On grasp choice, grasp models, and the design of hands formanufacturing tasks. *Robotics and Automation, IEEE Transactions on*, 5(3):269–279, 1989.

[24] K. Fujimura and X. Liu. Sign recognition using depth image streams. *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, pages 381–386, 2006.