Pose Alignment Of An Eye-In-Hand System Using Image Morphing

Rahul Singh Richard M. Voyles David Littau Nikolaos P. Papanikolopoulos

Artificial Intelligence, Robotics, and Vision Laboratory
Department of Computer Science and Engineering
University of Minnesota, Minneapolis, MN 55455
E-mail: {singh, voyles, littau, npapas}@cs.umn.edu

Abstract

Positioning an eye-in-hand robotic system with respect to a static target is a challenging research problem since it involves recognition of the object and the desired pose at which alignment is to occur, planning a trajectory for the robot to attain this pose, and careful calibration of the system and the environment. In this paper we introduce a unified framework based on image morphing, to address the above problems and apply it to the task of translational and rotational alignment of an eye-in-hand system to planar objects or planar projections of 3D objects. In our method the desired manipulator pose for each object is defined and stored as a view of the object taken from this pose. The identity of an unknown object in the workspace is established by morphing its image to the views in the database, and using a quantification of the morph as a dissimilarity measure. The synthetic images generated during the morph are used guide an eye-in-hand system to the desired pose. The framework can accommodate partially occluded or deformable targets and smooth trajectories can be generated since an arbitrary number of intermediate images can be used.

1 Introduction

Aligning a robotic manipulator with a given object, using visual information, is an important problem in active vision and robotics. Applications abound in robotic visual-tracking, vision-based grasping, and a variety of tasks involving manipulation of industrial parts like bolt insertion, point-to-point positioning, and part mating. In a general setting, the required alignment position may vary with each object. Pose alignment therefore, subsumes the problem of object recognition. Another problem which needs to be solved for alignment is that of generating a trajectory for the manipulator from the initial to the desired pose. Finally, vision based control is based upon the calibration, which has to be computed *apriori*, or approximated on-line, of the image plane with the 3-D environment.

While much of the early research in vision-based servoing [2, 3, 5, 14] involved precise off-line calibration, such systems usually suffer from lack of accuracy outside the regions where they are calibrated [15]. In contrast, recent work has focussed on the use of on-line estimation of the calibration [6, 7, 9, 15]. Although these methods are less sensitive to calibration errors, it is worth mentioning that they primarily focus on the vision based control aspects of the problem and either do not consider formulations where the required alignment pose may vary with the object [6, 15] or involve manual selection of features which are used to reduce the image error and thereby control the manipulator [7, 9].

This article introduces a methodology which differs significantly from previous approaches in that, given an object, the issue of determination of the alignment pose (the recognition problem) is addressed in conjunction with the issues involving trajectory generation based on reducing an image error and those related to the on-line estimation of the calibration. Furthermore, the method does not require explicit (manual) selection of features. The formulation considered in this paper uses object contours (2-D shapes or planar profiles of 3-D objects), which are extracted automatically.

We base our method on the emerging area of image morphing. Image morphing techniques rely on image interpolation or pixel reprojection between a set of basis views to obtain new, virtual views (see [1, 8] for an overview). The fundamental idea behind our method is to synthesize a sequence of images, which essentially define a *morph trajectory* between the starting and the desired views of the object. Each virtual (synthetic) image constituting the trajectory is then successively used as goal inputs to visually servo the manipulator to the desired pose. Based on a physics-based model for the object contours [11], each morph is quantified by the work done to bring about the corresponding image transformation. Besides satisfying the properties of a metric, this quantification is invariant to translation and rotation, and can therefore be used as a dissimilarity measure for recognition [12, 13]. A morph between any two object views (real or virtual) is distinguished by an optimal (in a minimal work sense) correspondence between the points on the object contours. In our method, this correspondence is used to obtain a *pose error* in the image plane. Starting from an arbitrary initial calibration, an iterative scheme, based on this *pose error*, is used to estimate the actual calibration between the morph plane and the world coordinates.

The organization of this paper is as follows: Section 2 describes the image morphing technique used for recognition and generation of the virtual images. The mathematical formulation of pose alignment by visual servoing using the morphed images is described in Section 3. Section 4 describes the experimental setup and the experimental results. Finally, in Section 5, the paper is summarized and some future research directions are suggested.

2 Contour Metamorphosis

Let $\mathbf{S}^I = [S_0^I, \ldots, S_n^I]$ and $\mathbf{S}^T = [S_0^T, \ldots, S_n^T]$ be the point sets representing the input and the target contours, respectively. Contour metamorphosis of $\mathbf{S}^I$ to $\mathbf{S}^T$ is defined by a sequence of intermediate shapes:

$$\begin{aligned} \mathbf{S}(t) &= u\mathbf{S}^I + t\mathbf{S}^T \\ &= [uS_0^I + tS_0^T, uS_1^I + tS_1^T, \ldots, uS_n^I + tS_n^T] \\ &= [S_0(t), S_1(t), \ldots, S_n(t)] \qquad (1) \end{aligned}$$

where $u = 1 - t$. $S_i(t)$ is the ith contour point in the intermediate shape, formed at time t. The time parameter t is normalized to the interval $[0, 1]$. In practice, the contours of the object are approximated by using a segmentation algorithm. In this work, we use the segmentation algorithm [10]. This algorithm is parameterless and has a linear complexity. The placement of segmentation points in this algorithm is based on minimizing the sum of absolute errors along each piecewise linear approximation of the input curve. It may be pointed out that if the input objects undergo substantial deformations, it may be desirable to use a segmentation algorithm that is not based on reducing an error norm. Segmentation strategies based on the detection of perceptually important point could be used in such cases [13].

Since the contours $\mathbf{S}^I$ and $\mathbf{S}^T$, in general, will have a different number of segmentation points, for the process of metamorphosis (as described by Eq. (1)) to occur, a point correspondence between the segmentation points in the input and the target is needed, wherein every segmentation point on the input contour corresponds to at least one segmentation point on the target contour and vice versa. To find an optimal correspondence (and thereby an optimal morph), we define the cost of a point correspondence as the sum of stretching and bending energies required to bring about the correspondence. The stretching energy is computed for every segment (pair of points) and is defined as:

$$E_s = k_s \frac{(L_T - L_O)^2 - (L_I - L_O)^2}{(1 - c_s)L_{min} + c_s L_{max}} \qquad (2)$$

where

$$L_{min} = min(L_O, \ldots, L_I, L_T)$$

and

$$L_{max} = max(L_O, \ldots, L_I, L_T)$$

In Eq. (2) E_s denotes the stretching energy spent in the current deformation, L_O, L_I, and L_T denote the segment lengths at the beginning, before the current deformation, and after the current deformation, respectively. The term c_s corresponds to the penalty for segments collapsing to points and k_s is the stretching stiffness parameter. The bending energy E_b is computed for point triplets and denotes the cost of angular deformation.

$$E_b = \mid k_b[(\phi_T - \phi_O)^2 - (\phi_I - \phi_O)^2] \mid \qquad (3)$$

where k_b indicates bending stiffness, ϕ_O represents the original angle, and ϕ_I and ϕ_T denote the angle before the current deformation and the angle after the current deformation, respectively.

The optimal metamorphosis, between two contours is defined by the correspondence requiring the least stretching and bending energy. By constraining the deformations at the segmentation points, the following optimal substructure property may be observed: The optimum cost of the point correspondence (S_i^I, S_j^T) equals the optimum cost of the previous point correspondence (S_{i-1}^I, S_j^T) or (S_{i-1}^I, S_{j-1}^T) or (S_i^I, S_{j-1}^T) and the cost of establishing the correspondence (S_i^I, S_j^T). Based on the above, an efficient $(O(mn))$ dynamic programming scheme can be constructed for morphing a contour C_A with m points to another C_B having n

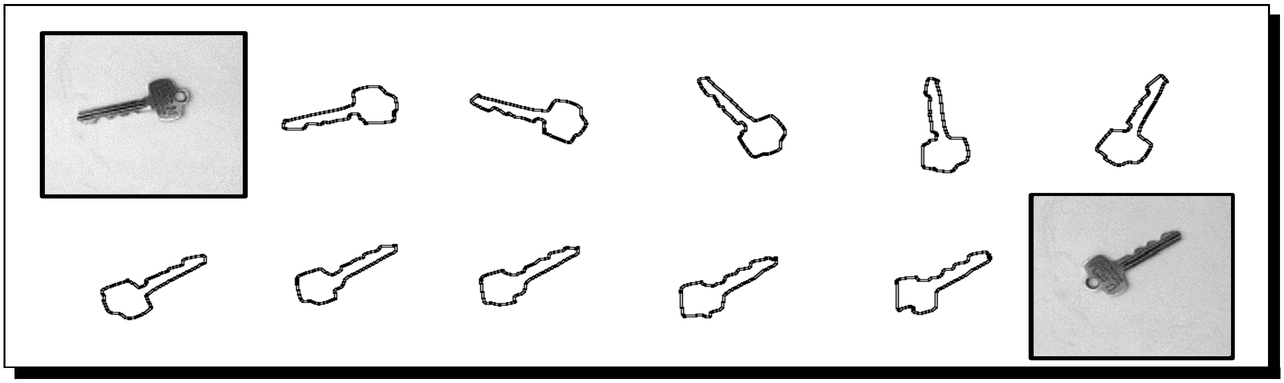

Figure 1: Contour morph sequence showing pose (rotational) rectification and deformation rectification for a key. The initial and final pose of the object are shown by black borders.

points. Since the energy computation described above, requires a starting point correspondence, we define the optimal morphing between two contours C_A and C_B as:

$$D_{morph}(A, B) = \min_{\Omega} E(C_A, C_B) \quad (4)$$

Here, Ω denotes the set of all starting point correspondences between the contours C_A and C_B.

Owing to the fact that the energy measures (Eqs. (2) and (3)) used in the metamorphosis depend on the relative position of the contour points of an object and not their absolute position, they are invariant to translation and rotation. Invariance to scale changes is achieved by mapping the shapes to a unit square (or alternatively, as in the present case, by utilizing the fact that alignment involves movements in the **X-Y** plane of the robot and occurs at a fixed depth **Z** above the object). We note that, since the formulation of metamorphosis used by us (see Eq. (1)) is linear, it does not, by itself, guarantee *physically valid* intermediate shapes in the morph between images of the same object, unless the input and the target contours have rotational alignment. Specifically, the lack of physical validity may be manifested by a cross-over of the object contours during the intermediate morphs. This may be avoided by including in the morph a rectification of the rigid transformations (translation and rotation) between the input and the target shapes, prior to the application of Eq. (1). The rectification of the rigid transformations is based on the observation that during the recognition process, the computation of the optimal morph provides a point correspondence, which is invariant to translation and rotation of the corresponding objects. This correspondence can therefore be used to estimate the elongation vectors for the input and the target shapes. The translational discrepancy between the centers of the input and the target elongation vectors can be used for translational rectification between the images. The rotation between the two shapes can be estimated by computing the angle between the elongation vectors, about their center, after translation rectification. The rectification of the rigid transformations (translation and rotation) involves an update of the coordinates of the input and does not affect the correspondences that were obtained during the computation of the minimal energy. These correspondences along with the updated coordinate values of the input are used to rectify the deformations between it and the target. The modified shape metamorphosis paradigm used by us, thus consists of the following steps:

1. *Shape recognition*: Obtain the shape model, closest to the input, in terms of the stretching and bending energies. The identity of the closest shape is determined by using the optimal substructure property described above.

2. *Rectification of rigid transformations*: Compute the elongation vectors for the input and the target. Recover and rectify the the translational and rotational differences. Update the coordinate values of the input shape.

3. *Rectification of deformations*: Use the updated coordinate values of the input along with the correspondences computed during the recognition stage and rectify the deformations using Eq. (1).

An example metamorphosis sequence is shown in Figure 1. The first and the last frames show the object in its initial and final pose respectively. The intermediate frames show rotational rectification, followed by rectification of the deformations of the input object contour.

3 Pose Alignment By Visual Servoing

Traditionally, the visual servoing formulation is based on relating the feature velocities in the image plane to robot velocities through a Jacobian [4, 9, 15]. The present approach is distinct from traditional visual servoing research in that we use the intermediate (synthetic) images generated by the morph to verify and correct the robot trajectory. Owing to the fact that the intermediate images are inherently position-based, we formulate the problem in terms of homogeneous coordinate transformations rather than a Jacobian.

Let $\{W\}$ denote the world frame, $\{R\}$ the robot frame, and $\{C\}$ the camera tool frame. Let also the morph plane be denoted by $\{I\}$ and the object frame by $\{O\}$. Denoting the transformation between arbitrary frames $\{j\}$ and $\{k\}$ as ${}^{j}_{k}T$, the following two basic relationships can be established between the object frame and the world frame and between the object frame and the morph plane respectively.

$$ {}^{W}_{O}T = {}^{W}_{R}T \, {}^{R}_{C}T \, {}^{C}_{O}T \tag{5} $$

$$ {}^{I}_{O}T = {}^{I}_{C}T \, {}^{C}_{O}T \tag{6} $$

The above two relationships, we note, are coupled by ${}^{C}_{O}T$, which is the transformation describing the object in the camera frame. The calibration problem consists of the following two parts:

- Determining the camera tool transformation ${}^{R}_{C}T$
- Determining the image projection transformation ${}^{I}_{C}T$

Assuming calibration, if we denote the initial transformation between the object and the image frames as ${}^{I}_{O}T_0$ and the goal transformation between them as ${}^{I}_{O}T_{goal}$, then it follows from Eqs. (5) and (6)

$$ {}^{I}_{O}T_0 = {}^{I}_{C}T \, {}^{C}_{O}T_0 \tag{7} $$

$$ {}^{I}_{O}T_{goal} = {}^{I}_{M}T^{-1}_{goal} \, {}^{I}_{O}T_0 = {}^{I}_{C}T \, {}^{C}_{O}T_i \tag{8} $$

where, ${}^{I}_{M}T_{goal}$ is the transformation from the current image to the goal image in the morph plane and ${}^{C}_{O}T_i$ is the intermediate transformation between the camera and the object. Replacing in Eq. 8, the value of ${}^{I}_{O}T_0$ from Eq. 7, we get

$$ {}^{I}_{M}T^{-1}_{goal} \, {}^{I}_{C}T \, {}^{C}_{O}T_0 = {}^{I}_{C}T \, {}^{C}_{O}T_i \tag{9} $$

Taking into account Eqs. (6) and (5), and denoting the desired robot motion by ${}^{R}T_{i_d}$ we have

$$ {}^{C}_{O}T_i = {}^{R}_{C}T^{-1} \, {}^{W}_{R}T^{-1}_{i} \, {}^{W}_{O}T = {}^{R}_{C}T^{-1} \, {}^{R}T_{i_d} \, {}^{W}_{R}T^{-1}_{0} \, {}^{W}_{O}T \tag{10} $$

Finally, simplifying for the desired motion of the robot, we have

$$ {}^{R}T_{i_d} = {}^{R}_{C}T \, {}^{I}_{C}T^{-1} \, {}^{I}_{M}T_{goal} \, {}^{I}_{C}T \, {}^{R}_{C}T^{-1} \tag{11} $$

We note that in the above formulation, the knowledge of the image projection transformation is assumed. If we have an only an estimate of the tool transformation (i.e. we do not assume complete calibration information), the commanded motion to the manipulator may not produce the desired motion in the image plane. An error transformation may then be computed by considering the deviation between the morph trajectory and the resultant view. This transformation can then be used to update (with possible averaging), the estimation of the transformation ${}^{R}_{C}T$. A morph of the new view to the current virtual image being used as a sub-goal, then provides the new move for the manipulator.

4 Experimental Results

Experiments were conducted on a set of six objects. A single image of each object, at a desired pose with respect to the manipulator, was stored as a template in the image database. The contours of an input object were extracted and morphed to the stored templates in the database. The morph consuming the least energy was used to recognize the input. The images generated by the morph sequence were successively used to guide the manipulator. After each move of the manipulator, the real image of the object, obtained from the current position of the manipulator, was morphed to the virtual image being used, as the current subgoal, and a pose difference (translational and/or rotational) between the two was computed (from the morphs). This difference was used to update the estimated transformation matrix and move the manipulator to a new pose with respect to the object. When the pose difference between the actual image and the virtual images became less than a threshold (empirically defined to be 0.10 radians for rotation and within five pixels in **X** and **Y** for translation), the next virtual image from the morph sequence was used. The process terminated when all the images generated in the morph had been utilized.

Experimental results for pose alignment with a pair of pliers and a wedge are presented in Figure 2 and

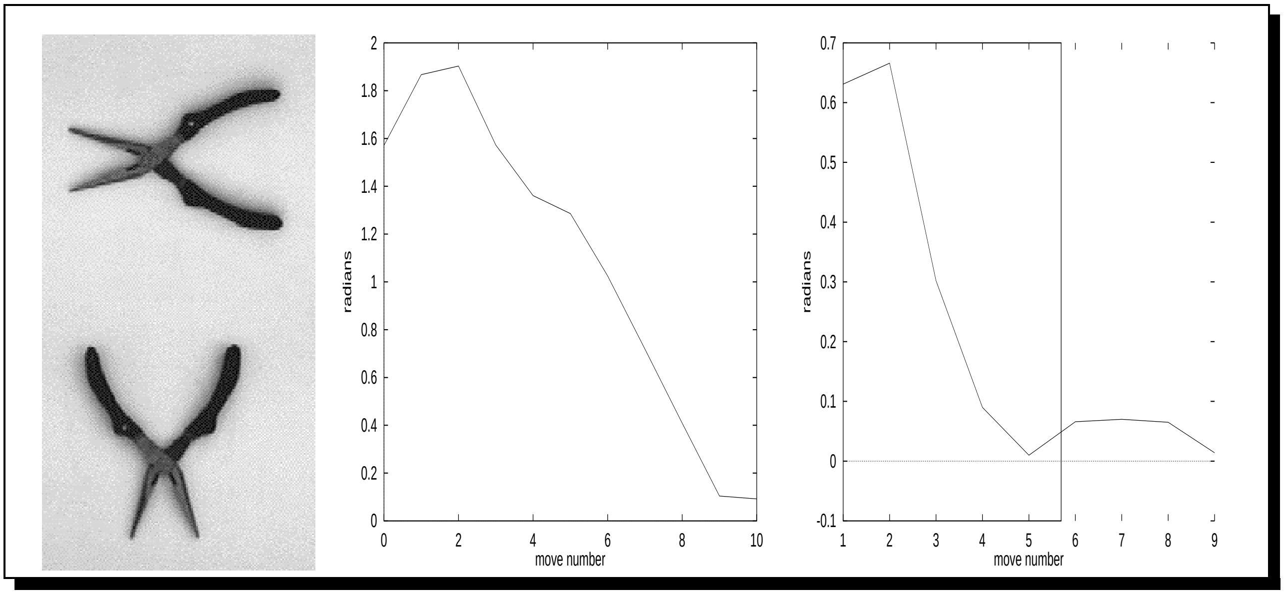

Figure 2: Experimental results for a pair of pliers. The image on the top left indicates the starting pose. The final pose is is shown in in the bottom left image. The first graph shows the angular pose difference (with respect to the final desired pose) after each move of the manipulator. The second graph shows the angular pose error between each virtual image and the corresponding real images, obtained by using the virtual image as an intermediate goal pose.

Figure 3, respectively. Figure 2 shows the results of rotational pose alignment for a pair of pliers. The first graph plots the angular pose error as it decreases through the alignment process. The second graph shows the rotational error between each virtual image constituting the *morph trajectory* and the corresponding real images obtained during the servoing. Graphs of the error in the morph plane, the error in the world frame and the manipulator trajectory for translational pose alignment with a wedge are shown in Figure 3. Figure 4 shows a morph sequence for the pair of pliers. In Figure 5 the images seen during the pose alignment of the manipulator with the object are shown interleaved with the virtual images generated during the morph. Each virtual image (shown as object contours) denotes the desired pose which the manipulator needs to attain with respect to the object. The arrows indicate manipulator poses which are within the designated threshold with respect to the desired pose as obtained from the morph. The starting pose and the desired pose are denoted in black borders. The experimental setup consisted of an IRIX Indigo workstation for generating the morphs. The results from morphing were communicated to a PUMA 560 manipulator, used as the eye-in-hand system.

5 Conclusions and Discussion

This article proposes the use of image morphing for vision-based robotic tasks involving hand-eye coordination. In particular, we have successfully achieved pose alignment of an eye-in-hand system with various objects using the proposed framework. In our formulation, image morphing is used to generate a set of virtual images, which guide the manipulator to the desired pose. Of particular importance is the fact that no manual feature selection is needed for the operation of the method.

We believe that the idea presented in this paper holds promise in many challenging tasks in active vision and robotics including grasping of 3-D objects using their planar projections as well as grasping of 3-D objects in a full 6-DOF formulation. The latter formulation is especially interesting, since for 3-D objects self-occlusions may render feature based techniques unsuitable. Of particular interest is grasping in the context of programming by human demonstration. Using this paradigm, a camera can be attached to the human demonstrator for acquisition of the goal image, thus obviating the need for absolute position sensing.

An important open problem is the possibility of a transition, based on the proposed approach, from a geometry based task specification to a specification

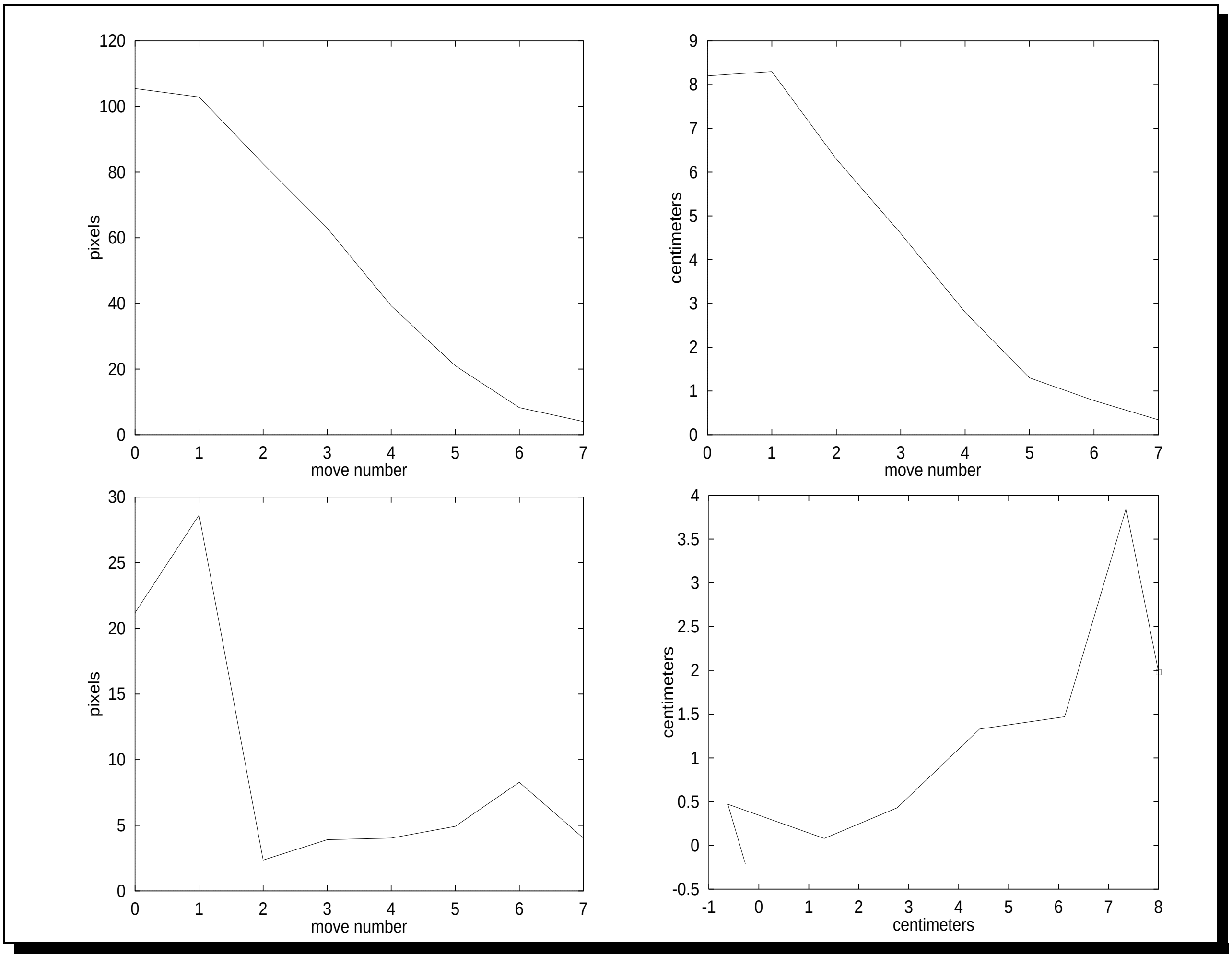

Figure 3: Translational pose alignment with a wedge. The error from the desired pose in the morph plane and the world coordinates is plotted in the left and the right graphs of the top row. The pose error between each virtual image used as a sub-goal, and the corresponding real image is shown in the first graph in the bottom row. In the second graph the trajectory of the manipulator, during the pose alignment is shown.

based on demonstrations.

Acknowledgements

The authors would like to acknowledge the help of Mike Wade in conducting some of the experiments. This research was partially supported by the National Science Foundation through Grants #IRI-9410003 and #IRI-9502245. This material is partially based upon work supported by the Defense Advanced Research Projects Agency, Electronics Technology Office, *Distributed Robotics*, ARPA order No. G155, program code 8H20, issued by DARPA/CMD under contract # MDA972-98-C-0008.

References

[1] Shai Avidan and Amnon Shashua. “Novel View Synthesis In Tensor Space”. In *Computer Vision and Pattern Recognition 1997*, pages 1034–1040, 1997.

[2] F. Chaumette and P. Rives. “Vision-Based-Control For Robotic Tasks”. In *Proceedings of the IEEE International Workshop on Intelligent Motion Control*, pages 395–400, 1990.

[3] F. Chaumette, P. Rives, and B. Espiau. “Positioning Of A Robot With Respect To An Object, Tracking It And Estimating Its Velocity By Visual Servoing”. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2248–2253, 1991.

[4] J. T. Feddema and C. S. G. Lee. “Vision Guided Servoing with Feature Based Trajectory Generation”. *IEEE Trans. Robotics and Automation.*, 5(5):670–691, 1989.

[5] J. T. Feddema and C. S. G. Lee. “Adaptive Image Feature Prediction and Control For Visual Tracking with a Hand-Eye Coordinated Camera”. *IEEE Trans. on Syst., Man, and Cybern.*, 20(5):1172–1183, 1990.

[6] B. K. Ghosh, D. Xiao, N. Xi, and T. J. Tarn. “Calibration-Free Vision-Based Control of a Robotic Manipulator”. In

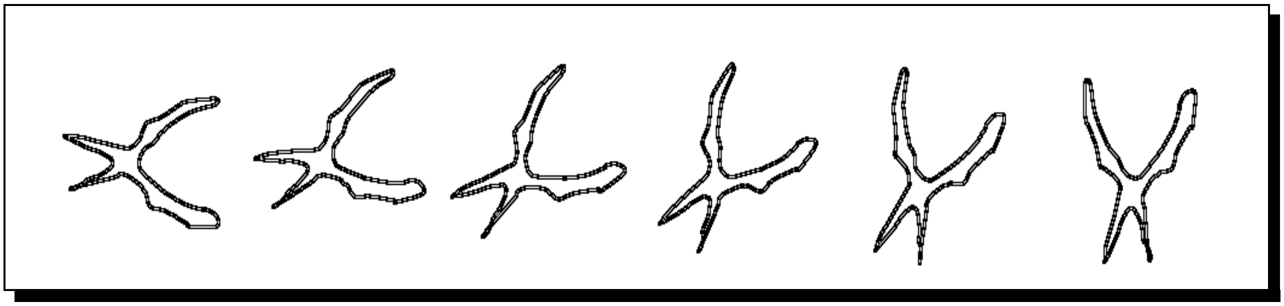

Figure 4: **Contour morph sequence for the pliers.**

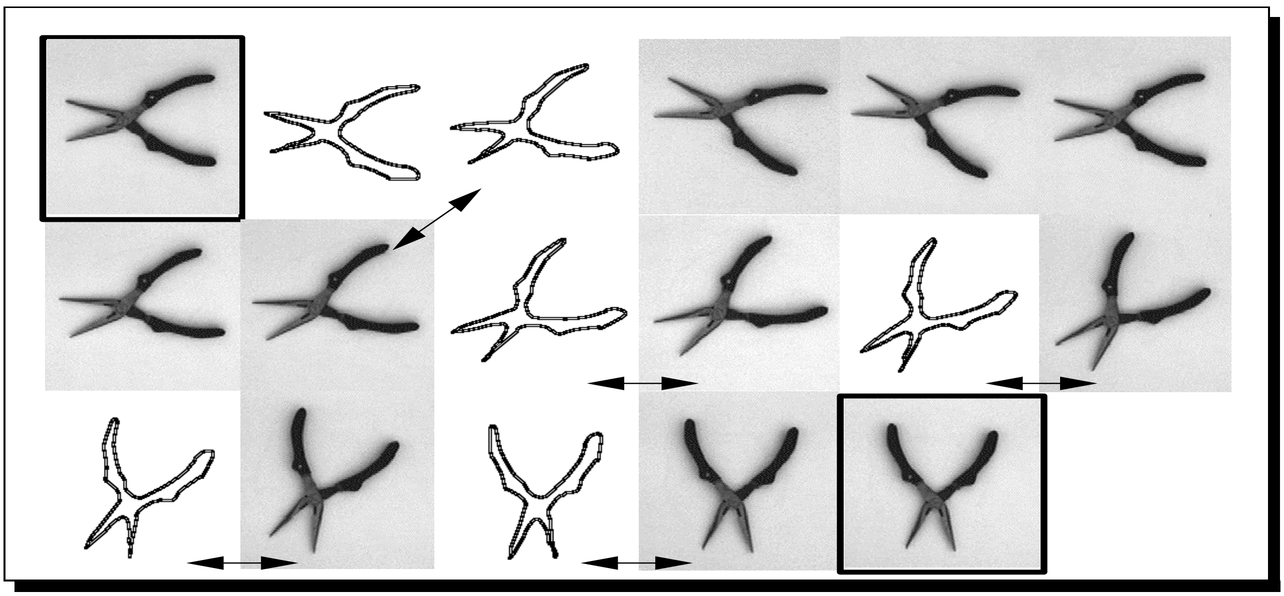

Figure 5: **Pose alignment for a pair of pliers: The views obtained from the eye-in-hand system are interleaved with the virtual images generated by the morph.**

Proceedings of the Thirty-Fourth Annual Allerton Conference on Communication, Control and Computing, pages 593–602, 1996.

[7] G. D. Hager. "Calibration-Free Visual Control Using Projective Invariance". In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1009–1015, 1995.

[8] S. B. Kang. *"A Survey of Image-based Rendering Techniques"*. TR CRL 97/4, Digital Equipment Corporation, Cambridge Research Lab, 1997.

[9] N. P. Papanikolopoulos and P. K. Khosla. "Adaptive Robotic Visual Tracking: Theory and Experiments". *IEEE Trans. on Automatic Control*, 38(3):429–445, 1993.

[10] Bimal Kumar Ray and Kumar S. Ray. "Determination of Optimal Polygon From Digital Curve Using L_1 Norm". *Pattern Recognition*, 26(4):505–509, 1993.

[11] T. W. Sederberg and E. Greenwood. "A Physically Based Approach to 2D Shape Blending". *Computer Graphics*, 26(2):25–34, 1992.

[12] R. Singh, I. Pavlidis, and N. P. Papanikolopoulos. "A Metamorphosis-Based Shape Recognition Method". In *Proceedings of the 1997 International Conference on Digital Signal Processing*, volume 2, pages 679–682, 1997.

[13] R. Singh, I. Pavlidis, and N. P. Papanikolopoulos. "Recognition of 2D Shapes Through Contour Metamorphosis". In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1651–1656, 1997.

[14] L. Weiss, A. Sanderson, and C. Neuman. "Dynamic Sensor-Based Control Of Robots With Visual Feedback". *IEEE Journal Of Robotics And Automation*, 3(5):404–417, 1987.

[15] B. H. Yoshimi and P. K. Allen. "Alignment Using an Uncalibrated Camera". *IEEE Trans. on Robotics and Automation*, 11(4):516–521, 1995.