

# USING FAST STATISTICAL DYNAMIC CONTOURS FOR GRASPING OCCLUDING CONTOURS

DOUG PERRIN<sup>1</sup>, OSAMA MASOUD<sup>1</sup>, CHRISTOPHER E. SMITH<sup>2</sup>,  
NIKOLLAOS P. PAPANIKOLOPOULOS<sup>1</sup>

<sup>1</sup>Dept. of Computer Science and Engineering, University of Minnesota,  
4-192 EE/CS Building, 200 Union St, Minneapolis, MN 55455

<sup>2</sup>Dept. of Computer Science and Engineering, University of Colorado at Denver  
2605 North Classroom Building, 1200 Larimer Street Denver, CO 80217-3364

**Abstract.** Object grasping is one of the basic functions required for many manipulator tasks. In particular, the grasping of unknown objects is often a desired functionality in manipulator system applications ranging from space exploration to factory automation. Due to the amount of object and environment data typically required to execute an unknown object grasp, computer vision is the sensor modality of choice. This paper presents a method for the automatic determination of plausible grasp axes on unknown objects using an eye-in-hand robotic system and a novel deformable contour model. The system finds potential grasp point pairs, ranks all the possible pairs/axes based upon measurements taken from the contour, and executes a vision-guided grasp of the object using the highest ranked grasp point pair to determine the gripper alignment constraints.

**Key Words.** Robotic Grasping, Statistical Dynamic Contours, Eye-in-Hand Robotic Systems.

## 1. Introduction

Grasping an unknown or a partially known object is a big challenge for robotic devices. Robotic grasping systems require extensive calibration and *a priori* knowledge about the grasping targets. However, it is often desirable to have a method that can analyze and plan grasps of objects that are novel to the robotic planning system.

Many techniques have been proposed for planning grasps of known or partially known objects. In general, these techniques possess other system characteristics that are undesirable, such as high computational overhead or need for extensive calibration. In addition, methods that attempt to register the environment and then execute an entirely blind grasp are susceptible to problems in calibration and changing environmental conditions.

What is desired is a method for planning grasps that performs well on completely unknown objects, that incorporates object measures that are also useful for

visual guidance of the manipulator, and that reduces (or is free of) calibration restrictions.

In particular, we propose using statistical pressure contour models to guide grasp axis selection and robot motion. Results from this work are preliminary, yet still demonstrate great promise for the use of a knowledge-based, weak-model grasping method. Our approach is based upon finding grasp point pairs and executing vision-guided grasps given no previous information about the object to be grasped.

In this paper, we extend recent innovations in statistical dynamic contours [6] to extract grasp axes and to track the contour of our target. Potential grasp axes are filtered by heuristic methods that measure the quality of each potential grasp axis. We use an eye-in-hand system to experimentally execute the grasps found by our method, showing that our system can grasp various shapes (e.g., a block, a tape dispenser, and a pair of pliers). The grasps are also computed under a variety of backgrounds, ranging from a uniform gray background to a wood textured background. Multiple objects in the scene do not affect the method provided that there are no overlapping objects of similar image statistics.

## 2. Previous Work

Early work on grasping contours by Faverjon and Ponce [5] demonstrated how to compute good grasps using piecewise polynomial contours. However, their work did not include the use of a grasp planner in order to control a robot. Bendiksen and Hager [2] showed promising experimental results on vision-based robotic grasping. Taylor et al. [10] made the initial steps toward combining visual servoing with a contour-based grasp planner. Their work describes a system that uses snakes to grasp prolate spheroids. Their proposed algorithm addresses the problem of having the grasp points of different depths on the contour. In part, our work extends this by looking at the entire contour rather than using sets of parallel snakes that match the gripper width to search for graspable opposing surfaces. Using the entire contour allows us to look for grasps on more complicated objects. We also use a tracking strategy based upon the entire contour. The utility of tracking the entire contour was demonstrated by Yoshimi and Allen in [11]. By tracking the entire contour, the fingers/gripper contact with the object can be detected. Finally, Allen et al. [1], and Smith and Papanikolopoulos [8] presented successful robotic grasping using eye-in-hand vision systems.

## 3. Active Deformable Models

Our method uses active deformable models (i.e., snakes) to extract contour information from image measures. The snakes deform based upon statistics derived from the image data, resulting in a contour model for objects in the workspace. An advantage of snakes in general is that these models can track partially-occluded objects or semi-rigid targets [4][9]. Since the model itself can be used to visually guide the manipulator, the utilization of model and planning data by the control system is relatively straightforward.

The traditional deformable model was first proposed by Kass et al. [7]. It is a parametric curve  $S$  of the form

$$S(u) = (x(u), y(u)), u \in [0, 1], \quad (1)$$

where  $x$  and  $y$  are the coordinates of the curve. The curve is placed onto a potential field derived from the following energy equation:

$$E = \frac{\alpha}{2} \oint \left| \frac{\partial S(u)}{\partial u} \right|^2 du + \frac{\beta}{2} \oint \left| \frac{\partial^2 S(u)}{\partial u^2} \right|^2 du + \rho \oint P(I(S(u))) du - \text{Pressure}, \quad (2)$$

where  $\alpha$ ,  $\beta$ , and  $\rho$  are weights. The first term corresponds to the tension force, the second term corresponds to the curvature force,  $P(I(S(u)))$  is the potential induced by the image values (edges, corners, or dark spots on the image) along the curve, and the last term corresponds to the pressure force. The energy along the length of the curve is minimized by allowing the model to change shape and position.

A problem with these formulations is that in the absence of image energy, these models collapse to a point. Pressure snakes (balloons) [3] have been developed to alleviate this problem by adding an internal pressure term to

force the model to expand. Unfortunately, the constant pressure term introduces new problems with the model. For instance, the initial placement of the snake had to be within the target.

Several forms of dynamic pressure models were proposed by Ivins and Porrill [6] to address the issues of constant pressure models. The pressure models are based upon first order statistics and utilize a seed region of the image to identify positive vs. negative pressure regions. That is, image regions that are statistically similar to the seed region yield positive pressure while image regions that are some number of standard deviations away from the seed mean will yield negative pressure. When a portion of the contour is in a positive region, it will expand away from the center of the contour. When the contour portion is in a negative region, it will contract toward the center. It follows that the minimum energy of the contour lies on the pressure boundary between positive and negative.

For our particular method, we use a dynamic statistical pressure snake that does not require an image energy term. We use an energy function with only internal energy and the dynamic statistical pressure model. The pressure model from [6] is given by

$$F(S) = \left( \frac{\partial S}{\partial u} \right)^\perp \left( 1 - \frac{I(S) - \mu}{k\sigma} \right), \quad (3)$$

where  $S$  is the curve,  $\mu$  and  $\sigma$  are the mean and standard deviation of the seed region, and  $k$  is a user specified parameter. There are issues with this model (and snakes in general) related to the automation of seed selection and the automation of the selection of the  $k$  parameter.

A significant problem with dynamic pressure snakes is the coupling between energy terms. Curvature, pressure, and tension can all apply force in a direction perpendicular to the curve. The curvature in Eq. (2) pushes the points toward a line. Tension applies a force along the curve in the direction that reduces overall curve length. Pressure by definition is expanding or contracting the area of the snake and acts perpendicularly to the curve. To de-couple these forces, we first looked at what each force is trying to achieve, then developed forces that achieve the same goal with minimal coupling.

The objective of the curvature term is to smooth the contour. The objective of pressure term is to explore the image and find the region boundaries. The objectives of the tension term are to have the control points uniformly distributed around the curve and to shrink the snake when there are no image forces.

Our first observation is that shrinking of the snake due to tension is redundant with the pressure, therefore our replacement term for tension is only required to evenly space the points. The original tension term that minimizes the length of the curve would pull the control point (B) towards the line connecting its two adjacent control points (A,C) (Fig.2). This perpendicular motion

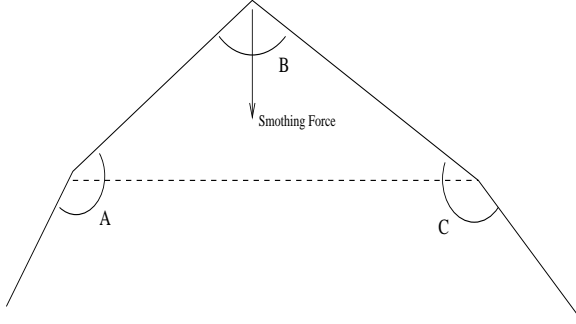


Fig.1. Curvature force.

to the curve strongly interfered with both curvature and pressure forces.

The spacing term is minimal when all the snake control

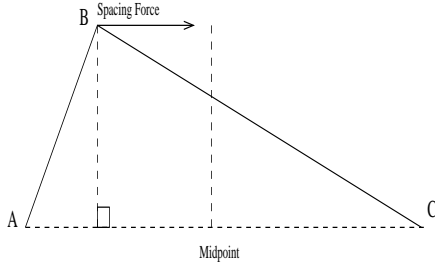


Fig.2. Spacing force.

points have a constant distance between them. This is achieved by a force acting toward the perpendicular bisector of AC (Fig.2). By moving the control point towards the nearest point on the perpendicular bisector, the interference with the curvature force is minimized. While the dynamic addition/subtraction of points can also provide equal spacing, this force is still necessary to maintain spacing within the bounding thresholds.

The curvature term is minimal when all the angles formed at the control points are equal. The lowest energy configuration without a pressure term will be a circle. The original curvature would collapse to a point or flatten out to a line.

There have been several formulations given for curvature [12]. In these formulations, a straight line has the lowest curvature. Since our snakes are closed contours, a better curvature would be one where a circle has lowest value. This is mainly a difference in implementation not in concept since a line can be seen as a segment of an infinitely large circle. In our formulation curvature has been made independent of this circle's radius. We define "Curvature" as the similarity between the control point's angle and the angles of its neighbors. This curvature force moves the control point in such a way that its angle and the neighboring angles are equal (see Fig.1) where the control point is pushed to equalize angles A, B, and C. This formulation is straightforward and easy to implement.

The assurance of evenly spaced points around the contour is necessary because we use the snake contour to approximate the object contour and assume an even dis-

tribution of points in order to find the perceived center of mass

#### 4. Grasp Axis Determination

Once the snake has settled on the object contour, a heuristic is used to extract potential grasp axes. This is done by considering grasp axes formed by each pair of control points. Axes that do not satisfy a filtering condition are eliminated. In Fig.3, points  $i$  and  $j$  are the two control points under consideration. We use the pressure snake to estimate curvature and gripper contact area at the grasp points and thus, the directions of vectors  $\mathbf{p}$  and  $\mathbf{q}$  are assumed to estimate the local object contour. The grasp axis, represented by  $\mathbf{r}$ , must satisfy the following conditions:

$$|\theta| \leq e_1, \quad (4)$$

$$|\alpha - 90| \leq e_2, \quad (5)$$

$$\text{Min}(\text{GripSpan}) < |\mathbf{r}| < \text{Max}(\text{GripSpan}), \text{ and} \quad (6)$$

$$\mathbf{z}' \left( \begin{bmatrix} \mathbf{p} \\ 0 \end{bmatrix} \times \begin{bmatrix} \mathbf{r} \\ 0 \end{bmatrix} \right) < 0 \text{ and } \mathbf{z}' \left( \begin{bmatrix} \mathbf{r} \\ 0 \end{bmatrix} \times \begin{bmatrix} \mathbf{q} \\ 0 \end{bmatrix} \right) < 0, \quad (7)$$

where  $e_1$  and  $e_2$  are preset parameters, and  $\mathbf{z} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}'$ . The final condition (7) guarantees that the grasp axis is inside the object. The width of the opening of the gripper is estimated by taking into account the operating depth of the manipulator. Axes that pass are rank ordered according to their distance from the snake's center of mass. The grasp axis used is the one with the smallest distance to the perceived center of mass. Fig.4 shows an image of an object that was used in our experiments along with the extracted grasp axes.

The promise of this method lies in its ability to produce plausible grasp points for completely unknown objects. This allows the robot to cope with novel situations.

#### 5. Tracking and Grasp Execution

Once an axis is chosen, the axis' offset from the perceived center of mass is found, along with its orientation

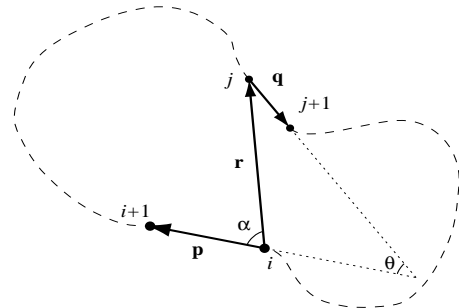


Fig.3. Grasp axis determination.

relative to the elongation axis of the object. The center of mass and the elongation axis are computed from the snake control points by considering the full region enclosed by the snake. This makes these measurements robust despite local variations on the snake contour. Furthermore, the elongation axis and the perceived center of mass are invariant with respect to X- and Y-axis translations and Z-axis rotations. Due to this characteristic, the snake can then be used to track the contour in real-time, extracting the perceived center of mass as well as the orientation given by the elongation axis. The efficiency of our statistical deformable model implementation allows frame-rate visual guidance while the manipulator is in motion.

The system orients the gripper using closed-loop control with feedback from the real-time computation of the snake's center of mass and orientation. Once the gripper is aligned, a blind Z-axis translation is done and the grasp is completed. Our choice of using visual guidance to perform the alignment significantly reduces the required calibration from a complete calibration down to a few parameters (three parameters if we assume that the gripper points down vertically). In this version of the system, we assume that the object lies on the same plane as the manipulator's mounting surface, leading to a known depth measure that is used to execute the blind Z-axis motion.

One of the areas for future work in this area is to investigate the potential of using either the snakes or some other visual measurement to visually guide the grasping through all manipulator motions and to determine when to close the gripper [8].

## 6. Experimental Design

### 6.1. Hardware

Our current experimental setup consists of a mini-camera mounted on the gripper of a Puma 560 manipulator. The camera's output is sent to a Matrox Genesis vision board that occupies a PCI slot in a dual processor Pentium Pro PC. The Matrox board and the system processors of the PC are used to implement the vision and control algorithms, producing cartesian coordinate changes for the manipulator. These changes are transmitted to the manipulator control subsystem via a serial connection.

### 6.2. System Initialization

The system implements a statistical snake model. The mean and standard deviation of the pixel values for the object are calculated from a seed region. In this preliminary system, the user supplies the seed region by using a

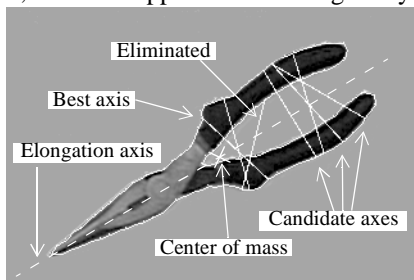


Fig.4. Extracted grasp axes.

mouse or other pointing device. For the snake, the  $\alpha$  and  $\beta$  are set to one, and  $k$  is set to 3.

The grasp axis is determined using the heuristic described above. The parameters  $e_1$  and  $e_2$  were set to 3 degrees. In the preliminary version of the system, the computed grasp axis is presented to the user for verification. While not required, this user intervention is utilized while the system is under development.

### 6.3. Experimental Results

Our first experiments with the system used a somewhat traditional rectangular prism that has many grasp axes along the major axis of the prism. This object was picked to determine the effectiveness of our heuristic method that selects the "best" axis to use for alignment and grasping. Since there are many candidate grasp axes, a flaw in the method would be readily apparent in the rank ordering of the potential grasp axes. In addition, the block used for these experiments narrowly fits into the gripper opening, allowing us to evaluate the quality of our on-line calibration and gripper alignment methods. The results of several experimental runs are presented in Fig.6, Fig.7 and Fig.8. In all the graphs in the figures, measurements are relative to the initial gripper position. The first part of the grasping process, the gripper alignment, involves simultaneous translation in the X and Y directions and rotation around the Z-axis. Once the alignment is completed, a translation along the Z-axis is performed with a constant speed. For clarity, the plots for the X- and Y-position and Z-rotation are shown only during the alignment stage after which, they remain constant.

The first set of graphs (Fig.6) show typical grasps of a block on a gray background. The block in Fig.6 was placed 58 mm away from the origin in X, 48 mm in Y, and rotated by .4 radians relative to X-axis. All the plots show that as the arm moves, it overshoots and has to correct itself. Corrections are about 3-5 mm. These corrections show an advantage of using a vision-based controller, where visual feedback eliminates the need for highly calibrated systems which are susceptible to inaccuracies and to problems in environmental changes. In Fig.7 and Fig.8, the grasp was performed on a tape dispenser sitting on a piece of plywood. Fig.8 shows the largest displacement of all the grasps (50 mm in X and 190 mm in Y). A closer look at the Z rotation graph shows that around time 4.5 sec, the gripper suddenly starts rotating after it appeared to have settled. This is due to the fact that during the X and Y fast translation, the snake lagged behind a little and underwent some dis-

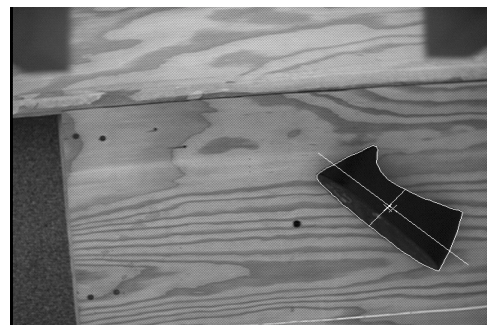


Fig.5. It shows the initial position of the tape dispenser target as viewed from the gripper camera. This target was used for the results in Fig.7.

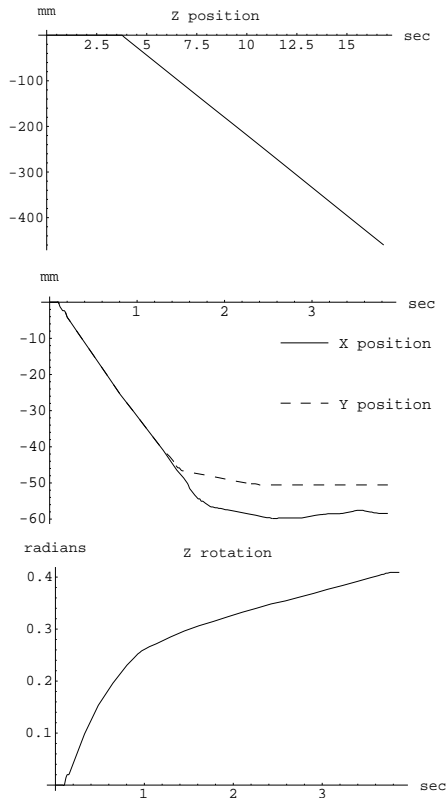


Fig.6. Plots of arm motion during the grasp of a rectangular block.

tortion. Although it still tracked the target, and thus estimated the center of mass, it did not accurately estimate the target orientation. As the translation speed slowed down, the snake recaptured the target contour and provided an accurate estimate of the orientation which is  $\sim 0.02$  radians different from the skewed direction. We do not consider this to be a serious problem because no major attempt has been made to optimize any of the code. Optimizing the code would help reduce or eliminate the lag in tracking. This would allow for an even faster translation and rotation during the alignment stages.

For all practical purposes, the translation along the Z-axis can be made as fast as physically possible. We used a conservative approach of setting the speed to about 35 mm/sec during our experimentation.

## 7. Conclusion

In this paper, we have presented a novel method for the determination of plausible grasps for unknown objects. This method is based upon statistical active deformable models (statistical snakes). The method is experimentally verified using both simple and complex, unknown, grasping targets. Because we allow the dynamic addition and removal of control points for the snake, we obtain a dense and evenly distributed set of control points along the contour of the object. The points are pairwise analyzed to determine those points that define possible grasp axes. We can apply any of several heuristic methods to these possible grasp axes to rank order the axes from best to worst. Once ranked, the best (most plausible) axis is used to determine the alignment constraints for the grasp. We extract the perceived center of

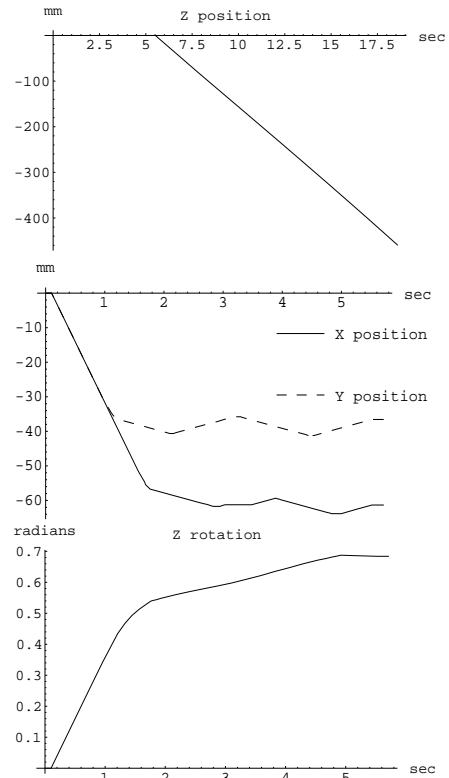


Fig.7. Plots of arm motion during the grasp of the tape dispenser.

mass and the elongation axis of the object from our snake and use these measures to visually guide gripper alignment. Once aligned, the manipulator executes a blind Z-axis motion to grasp the object. We present the results of several experiments to verify the validity of this method and to demonstrate its effectiveness in finding quality grasp axes.

## 8. Future Work

Our future work in this area will center upon three directions. First, we will investigate the suitability of additional heuristics for ranking the possible grasps. Second, the use of visual guidance throughout all manipulator motions should be researched so that assumptions that we made for this preliminary system can be relaxed. Third, we want to consider the possibility of including an exploratory component that uses visual measurements to reject grasp axes that are identified as plausible when the grasp is in fact not possible. This could also include applications where the gripper has more complex kinematics than a simple parallel jaw gripper.

## 9. Acknowledgments

This work has been supported by the National Science Foundation through Contracts #IRI-9410003 and #IRI-9502245, and the Department of Energy (Sandia National Laboratories) through Contracts #AC-3752D and #AL-3021.

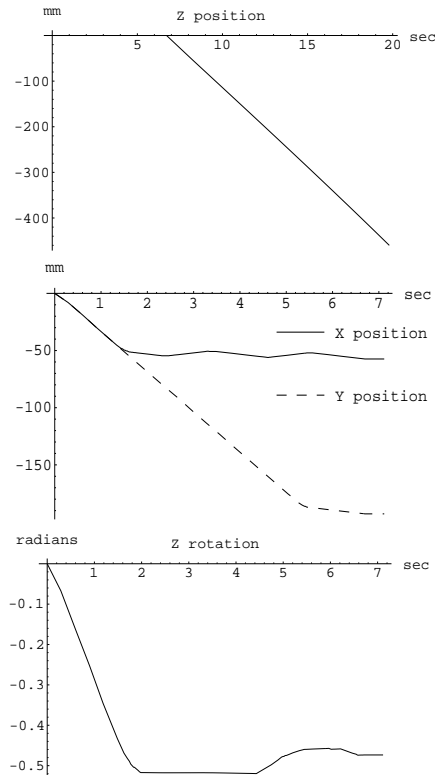


Fig.8. Plots of arm motion during the grasp of the tape dispenser shown in Fig.5.

## 10. References

- [1] Allen, P., Timcenko, A., Yoshimi, B., and Michelman, P., "Automated tracking and grasping of a moving object with a robotic hand-eye system," *IEEE Transactions on Robotics and Automation*, 9(2):152-165, 1993.
- [2] Bendiksen, A. and Hager, G., "A vision-based grasping system for unfamiliar planar objects," *Proceedings of the IEEE International Conference on Robotics and Automation*, 2844-2849, 1994.
- [3] Cohen, L. and Note, D.: "On Active Contour Models and Balloons", *CVGIP (Image Understanding)*, Vol 53, 2, pp211-218, 1991.
- [4] Couvignou, P., Papanikolopoulos, N., Sullivan, M., and Khosla, P., "The use of active deformable models in model-based robotic visual servoing," *Journal of Intelligent and Robotic Systems: Theory and Applications*, 17(2):195-221, 1996.
- [5] Faverjon, B. and Ponce, J., "On computing two-finger force closure grasps of curved 2D objects," *Proceedings of the IEEE International Conference on Robotics and Automation*, 424-429, 1991.
- [6] Ivins, J. and Porrill, J., "Active region models for segmenting medical images," *Proceedings of the IEEE International Conference on Image Processing*, 227-231, 1994.
- [7] Kass, M., Witkin, A., and Terzopoulos, D., "Snakes: active contour models," *Proceedings of the First International Conference on Computer Vision*, 259-268, 1987.
- [8] Smith, C. and Papanikolopoulos, N., "Grasping of static and moving objects using a vision-based control approach," *Journal of Intelligent and Robotic Systems: Theory and Applications*, 19:237-270, 1997.
- [9] Sullivan, M. and Papanikolopoulos, N., "Using active-deformable models to track deformable objects in robotic visual servoing experiments," *Proceedings of the IEEE International Conference on Robotics and Automation*, 2929-2934, 1996.
- [10] Taylor, M., Blake, A. and Cox, A., "Visually guided grasping in 3D," *Proceedings of the IEEE International Conference on Robotics and Automation*, 761-766, 1994.
- [11] Yoshimi, B. and Allen, P., "Visual control of grasping and manipulation tasks," *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 575-582, 1994.
- [12] Williams, D. and Shah, M., "A fast algorithm For Active Contours and Curvature Estimation," *CVGIP (Image Understanding)*, Vol 55, 1, 14-26, 1992.