Proceedings of the 8th IEEE Mediterranean Conference
on Control and Automation (MED 2000)
Rio, Patras, GREECE 17-19 July, 2000

DEPTH EXTRACTION FOR CONTOURS BY MONOCULAR EYE-IN-HAND SYSTEMS

DOUG PERRIN[1], CHRISTOPHER E. SMITH[2], NIKOLAOS P. PAPANIKOLOPOULOS[1]

[1]Dept. of Computer Science and Engineering, University of Minnesota,
4-192 EE/CS Building, 200 Union St, Minneapolis, MN 55455, USA
[2]Dept. of Computer Science and Engineering,University of Colorado at Denver
2605 North Classroom Building, 1200 Larimer Street Denver, CO 80217-3364, USA

Abstract. Many vision-based robotic applications either require, or can be improved by, accurate object depth measures. Several computer vision methods exist for extracting depth of features, including stereo vision, structured-light systems, and active monocular depth recovery. Previous efforts using these methods suffered from a variety of problems related to calibration and computational complexity. This paper presents a novel method for active monocular depth recovery that combines new, highly stable active deformable models (snakes) with a structured camera motion along the optical axis to produce depth estimates for all the snake control points.. In experiments with a variety of objects and depths, this method produced control point correspondences and calculated the depth of a large number of control points in the order of 1 ms. Accuracy is demonstrated by results that exhibit errors near the predicted errors when assuming a single pixel mis-measurement in control point location on the image plane.

Keywords. Robotic Grasping, Statistical Dynamic Contours, Eye-in-Hand Robotic Systems.

1. Introduction

Accurate depth measurements in several robotic applications are required to perform a variety of tasks and functions. Such measures allow systems to estimate grasp poses, identify objects, and plan manipulator motions. A large body of research exists for the recovery of depth, including stereo [13], structured light [3], and depth-from-motion [1][2] [12][14][15][19].

These techniques have some drawbacks in common robotic environments. Stereo techniques require precise camera calibration that is sensitive to changes in the environment and can be affected by vibrations and manipulator motions. Active depth extraction is often too time consuming when the depths of many object points are required for manipulator motion planning and grasp planning. Structured light methods often require precise calibration and typically use laser light stripers that may be undesirable in certain environments. A simple, fast method for the monocular depth derivation that does not require extensive calibration is needed.

In this paper, we present a method to recover accurate object-contour depth from a single camera mounted on the end-effector of a robotic manipulator. Our method requires only a simple Z-axis motion and does not require the calibration of many of the intrinsic and extrinsic camera parameters. The method has a wide range of applications including grasping tasks with unknown objects, inspection tasks with constrained lateral motion and in confined environments where stereo or structured light systems are too large. Examples of application include pipe inspection, collapsed building search and rescue, and endoscopic examination.

This method is based upon active deformable models (snakes) that capture the object's occluding contour. We use our new statistical pressure snake formulation that yields stable control points over time and object/camera motion. These snakes give an accurate representation of the object contour during the camera's translation along the optical axis (Z-axis of the manipulator) and allow us to use

control point correspondences between two images to calculate the depth for each control point on the snake. We then extend this method to produce more accurate results (when possible) by interpolating control point correspondences on the second object contour.

We demonstrate the effectiveness of this method by presenting experimental results from a vision-guided robotic workcell. These results show that our method is very accurate for objects at a variety of depths from the end-effector of the manipulator.

This method supplies depth data that can be used in many robotic applications, including grasp planning and execution, object recognition, inspection, etc. Furthermore, the method works with multiple snakes, potentially providing richer data that includes object surface appoximations and terrain mapping.

2. Previous Work

2.1. Stereo Vision

Stereo vision and the study of moving cameras in static and non-static scenes is an extensive research area. There are several books [6][9][11] which cover several of these issues and techniques.

2.2. Active Deformable Models

The traditional deformable model was first proposed by Kass et al. [10]. 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:

$$E = \frac{\alpha}{2}\oint\left|\frac{\partial S(u)}{\partial u}\right|^2 du + \quad (2)$$

$$\frac{\beta}{2}\oint\left|\frac{\partial^2 S(u)}{\partial u^2}\right|^2 du + \rho\oint P((I(S(u)))du)$$

where α, β, and ρ are weights. The first term corresponds to the tension force, the second term corresponds to the curvature force, and $P(I(S(u)))$ is the potential induced by the image values (edges, corners, or dark spots on the image) along the curve. 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) [4] have been developed to alleviate this problem by adding an internal pressure term to force the model to expand. Unfortunately, the constant pressure term solved few of the problems with the model. More successful have been dynamic pressure models.

Several forms of dynamic pressure models were proposed by Ivins and Porrill [8] to address the issues of constant pressure models. These pressure models are based upon first order statistics and utilize a seed region of the image to identify positive vs. negative pressure regions. 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.

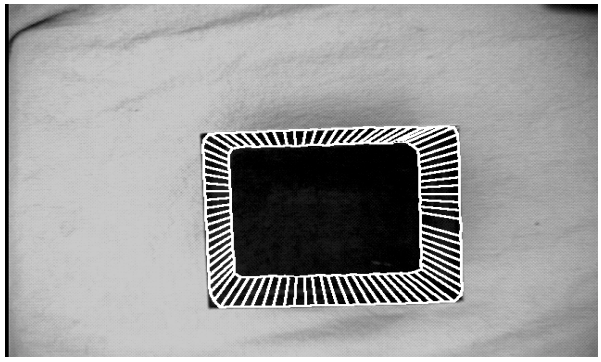

Fig. 1. Two different snakes after a Z translation of the eye-in-hand system. Snake point correspondences are also shown.

A problem with dynamic pressure snakes (addressed in [14]) is the coupling between energy terms. Curvature, pressure, and tension can all apply forces in a direction perpendicular to the curve. The curvature in equation (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. We have developed energy terms that uncouple these forces. These snakes (Fig. 1) are extremely stable and exhibit little motion of the control points due to conflicting internal terms. This stability is very important to this work because the snake control points are the features used to extract the contour depth.

3. Depth from Z-Translation

3.1. Basic Method

Our formulation uses a method that calculates depth estimates for points on the object contour by using corresponding control points from two active deformable models. The models are derived from two images that are taken from camera locations that differ only in their respective distances from the object of interest. This depth difference is produced by a known translation Δ along the Z-axis of the eye-in-hand system. The camera's optical axis has been aligned with the Z-axis of the end-effector of the

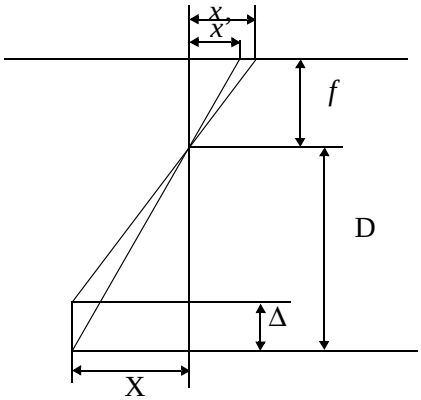

Fig. 2. Base case for finding depth.

robot, but calibration of other intrinsic and extrinsic camera parameters is not required. In fact, our depth formulation is independent of focal length, pixel size, and orientation, assuming they are constant from image to image. The geometry from the two images is shown in Fig. 2. The pinhole camera model produces the following equations:

$$\frac{X}{D} = \frac{x}{f} \tag{3}$$

$$\frac{X}{D-\Delta} = \frac{x'}{f} \tag{4}$$

A simple derivation and substitution (illustrated in (5) through (8)) gives us the final formulation for depth in (9):

$$Xf = Dx \tag{5}$$

$$Xf = (D-\Delta)x' \tag{6}$$

$$xD = (D-\Delta)x' \tag{7}$$

$$\frac{x}{x'} = \frac{D}{D} - \frac{\Delta}{D} \tag{8}$$

$$\frac{\Delta}{\left(1-\frac{x}{x'}\right)} = D \tag{9}$$

Depth is found using only the projections of X and the change in depth Δ. Several factors in equation (9) should also be observed. First, this formulation is independent of focal length. This occurs when equations (5) and (6) are combined in equation (7). We have used two different cameras with different focal lengths (3.5mm and 7mm) to recover contour depths with no software or parameter changes. Second, the formulation does not require the camera scaling factors (pixel size in X and Y) to be known since the ratio x/x' is dimensionless (Note: using a larger focal length lens effectively increases the accuracy of this discrete ratio for a given Z-axis translation, resulting in a more accurate depth measure). Third, this formulation is numerically unstable for control points on the optical axis. The ratio x/x' will equal 1 for such points. For any possible object contour, at most only one control point can lie on the optical axis. This situation is simple to identify and we avoid attempting to solve for depth in these rare cases.

3.2. Calibration Constraints

In a typical vision-based robotic system several intrinsic and extrinsic camera parameters must be calibrated. In this research as in our prior work we have attempted to eliminate as many of these calibration constraints as possible. The basic method requires very few parameters to be calibrated in the eye-in-hand system. Table 1 shows the typical calibration parameters and notes whether the parameter is calibrated or uncalibrated in our method. The parameters that are marked with an asterisk "Uncalibrated" are parameters that we did not calibrate for depth recovery, but are potential parameters to calibrate for increased accuracy or better task performance.

Table 1. Calibration constraints for depth recovery.

Intrinsic Parameters		**Extrinsic Parameters**	
Focal length	Uncalibrated	Camera location in X	Uncalibrated
Pixel size in X	Uncalibrated	Camera location in Y	Uncalibrated

Table 1. Calibration constraints for depth recovery.

Intrinsic Parameters		**Extrinsic Parameters**	
Pixel size in Y	Uncalibrated	Camera location in Z	Uncalibrated*
Image center in X	Uncalibrated*	Camera orientation, pitch	Calibrated
Image center in Y	Uncalibrated*	Camera orientation, yaw	Calibrated
Radial lens distortion	Uncalibrated	Camera orientation, roll	Uncalibrated
Tangential lens distortion	Uncalibrated		

3.3. Control Point Interpolation for De-Noising

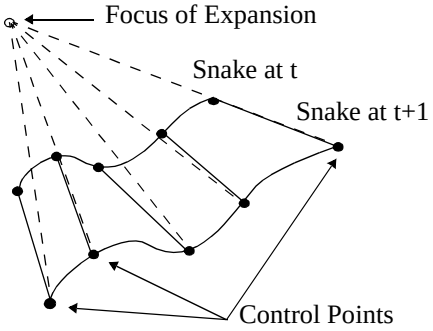

Fig. 3. Interpolation vs. correspondences.

Our first trials for recovering depth from the active deformable models used the labeled control points as correspondences for x and x'. This method yielded acceptable results, since our new snakes have extremely stable control points with respect to contour features over time and object/camera motion. However, there were cases where certain control points would "flip" around corners on the object contour and give degraded depth estimates. We addressed this problem by using a line from the Focus Of Expansion (FOE) to a control point on the initial contour and calculating this line's intersection with the second contour. This provided us with slightly more accurate results and eliminated problems due to "flipping" control points. Like the initial method, this method is numerically unstable when the optical axis (FOE) intersects the contour.

However, a problem arises when using these interpolated control points. If a contour segment is collinear with a radial line from the FOE, then the intersection of the FOE/control-point line on the second snake is a line segment rather than a point. Depth for contour segments that expand radially from the FOE cannot be found due to this ambiguity. Fortunately, this case is easy to test for and the control point correspondences can be used to provide depth data for these control points.

4. Experimental Design

4.1. Hardware

Our current experimental setup consists of two mini-cameras mounted on the gripper of a Puma 560 manipulator. The camera outputs are 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 algo-

rithms, producing cartesian coordinate changes for the manipulator. These changes are transmitted to the manipulator control subsystem via a serial connection. The serial interface connects to a VME-based Sun Sparc Station that serves as the host for Chimera, a real-time operating system [16]. Data is read from the serial interface into a Ironics 68030 VME Single-Board Computer (SBC) via a Bit-3 VME-to-VME bus adapter. The SBC calculates the inverse kinematics for the Puma every three milliseconds and, through a Trident Robotics VME-to-Puma interface, sends signals to the joint amplifiers.

4.2. Experimental Results

The depth and intersection calculations both require a computational time in the order of 1 ms. For each pair of control points, the total computational time is constant.

Fig. 4 shows a flat paper target 33 cm away from the cam-

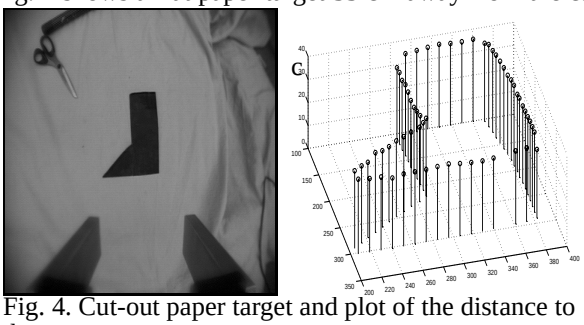

Fig. 4. Cut-out paper target and plot of the distance to the camera.

era and the resulting depths for the snake control points. Fig. 5 shows the same plot as Fig. 4 from two different

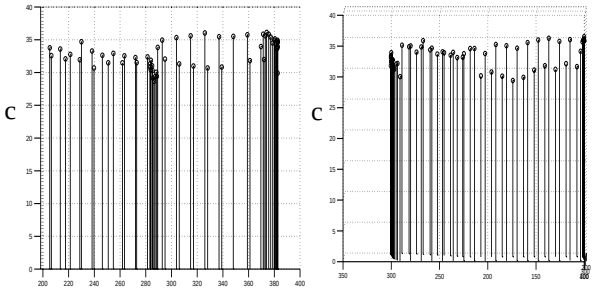

Fig. 5. Plot of the distance to the camera for the target in Fig. 4 viewed from the X and Y axis.

views.What is apparent from these views is the noise in the depth measured. The minimum distance is 29 cm while the maximum is 36 cm. The mean is excellent at 32.9 cm with a standard deviation of 2. Computing a theoretical, predicted pixel error term is somewhat complicated since the pixel error will vary with the control point distance from the optical axis. We computed the predicted error in depth estimate for a shift of one pixel for each control points This gave errors between 1 and 3.5 cm, depending upon how far the control point was from the optical axis. The pixel error for a given point can be found using the known depth, the distance to the optical axis, and a known move along the Z-axis. First, a predicted x is found for the given x' according to equation (10). The erroneous depth is then computed using equation (11) and the pixel error is given by equation (12). This analysis demonstrates that our method produces depth estimates that are accurate to within a one pixel error in location measure for virtually all the control points.

$$x_{predicted} = \left(1 - \frac{\Delta}{D_{actual}}\right)x' \tag{10}$$

$$D_{error} = \frac{\Delta}{1 - \frac{(x_{predicted} + 1)}{x'}} \tag{11}$$

$$PixelError = D_{actual} - D_{error} \tag{12}$$

The pixel error for the paper target in Fig. 4 is shown in Fig. 6. The errors increase as the control point distance to

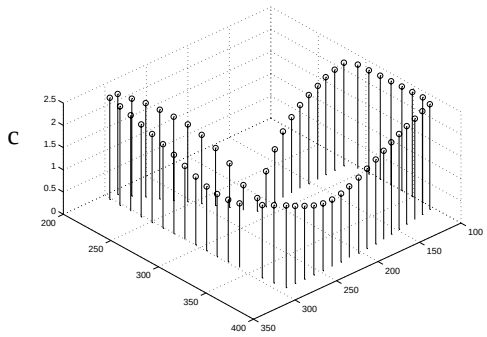

Fig. 6. Computed pixel errors for the paper target.

the optical axis increases (the FOE is near the concavity of the object).

Fig. 7 shows a box on a rectangle of paper and the resulting

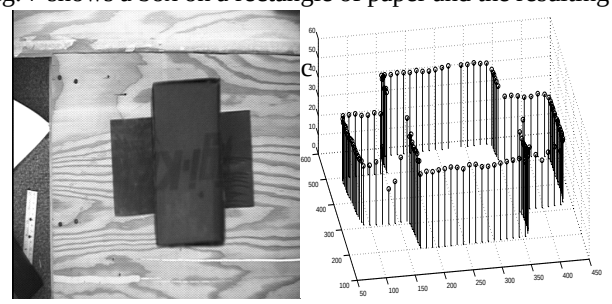

Fig. 7. A black box on black paper and a plot of the distance to the camera.

plot of the depth from the camera. The plot clearly shows the two different depths of the contour, the high center and the lower lobes on the sides of the object.

Fig. 8 shows a box with a ramp attached to one side. The

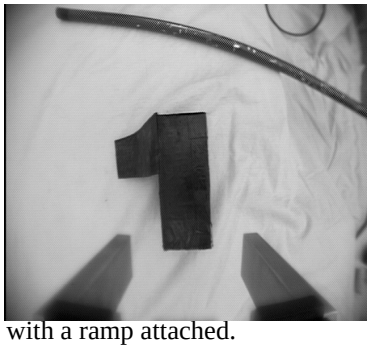

Fig. 8. A box with a ramp attached.

depth plot in Fig. 9 shows the smooth change in the depth of the ramp. The ramp is rather flexible and is steeper closer to the top and flattens out toward the bottom. This curvature can be seen clearly in the plots in Fig. 9.

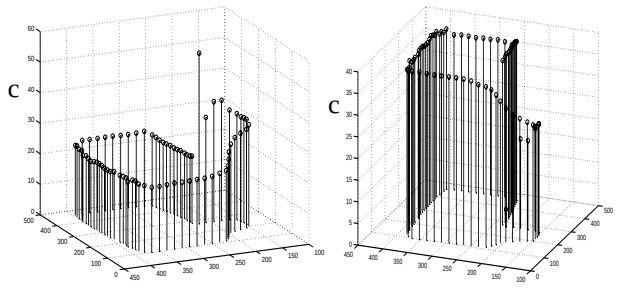

Fig. 9. Plot of the distance to the camera for the target in Fig. 8. The right plot shows the height with respect to the support base.

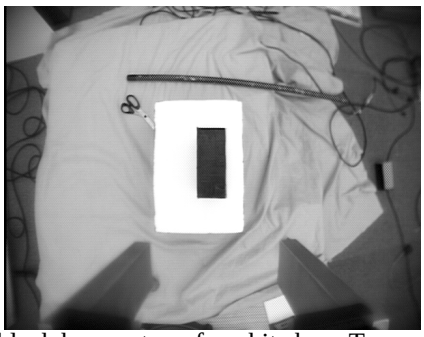

Fig. 10. A black box on top of a white box. Two snakes were used in order to capture each box's contour.

Fig. 11 shows two boxes stacked one on top of the other. Two snakes were used simultaneously to find the depths of the boxes. One snake was used to capture the white box's contour and the other was used to capture the black box's contour. The depth plot and the height plot (with respect to the support base) are shown in Fig. 11. The mean depth for

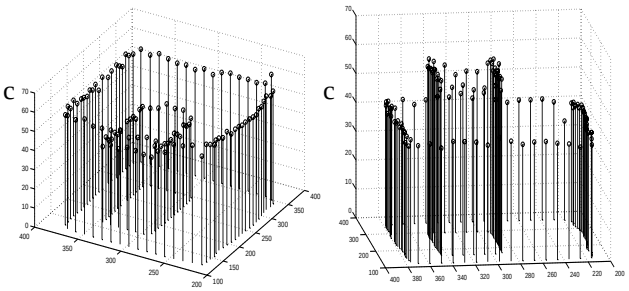

Fig. 11. Plot of the distance from the camera for the target in Fig. 10. The right plot shows the height with respect to the support base.

the white box is 58 cm with a standard deviation of 1.7. The mean depth of the black box is 41 cm with a standard deviation of 1.9.

5. Conclusion

We have presented a way to extract depth of contours from a monocular eye-in-hand system. Preliminary results are very encouraging. Our approach is based on a modified version of pressure snakes. The method finds depths within one pixel error. This accuracy is satisfactory considering that the image is coming from a camera mounted on the end-effector of a manipulator.

6. Future Work

While our results are encouraging, an error of 3 cm when the griper fingers is only 6 cm deep is still too large for consistent grasping tasks. We intend to improve accuracy by using integration over time and multiple depths.

With only a slight improvement of our accuracy, we will be ready to use depths to do grasp planning and execution on various targets with complex contour depths. We can use these methods to extend existing work of Sullivan [17], Couvignou [5], and Taylor [18] in snake grasping and tracking.

This work could be also be combined with structured motion to extract shape and scale.

7. 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.

8. 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), pp. 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, pp. 2844-2849, 1994.

[3] Boyer, K. L. and Kak, A. C., Color-encoded structured light for rapid active ranging, IEEE Trans. Pattern Analysis and Machine Intelligence, 9(1), pp. 14-28, Jan. 1987.

[4] Cohen, L. and Note, D., On active contour models and balloons, CVGIP (Image Understanding), 53(2), pp. 211-218, 1991.

[5] 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), pp. 195-221, 1996.

[6] Faugeras, O., Three-dimensional Computer Vision, MIT Press, Cambridge, 1993.

[7] 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, pp. 424-429, 1991.

[8] Ivins, J. and Porrill, J., Active region models for segmenting medical images, Proceedings of the IEEE International Conference on Image Processing, pp. 227-231, 1994.

[9] Jain, R., Kasturi, R., and Schunck, B., Machine Vision, McGraw-Hill Inc., 1995.

[10] Kass, M., Witkin, A., and Terzopoulos, D., Snakes: active contour models, Proceedings of the First International Conference on Computer Vision, pp. 259-

268, 1987.
[11] Horn, B.K.P., Robot Vision, MIT Press, Cambridge, 1986.
[12] Matthies, L., Kanade, T., and Szeliski, R., Kalman filter-based algorithms for estimating depth from image sequences. International Journal of Computer Vision, 3(3), pp. 209-238, 1989.
[13] Marr, D., Vision: A computational investigation into the human representation and processing of the visual information., W. H. Freeman and Company, San Francisco, 1994.
[14] Perrin, D., Masoud, O., Smith, C., and Papanikolopoulos, N., Unknown object grasping using statistical pressure models, Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1054-1059, 2000.
[15] Smith, C. and Papanikolopoulos, N., Computation of shape through controlled active exploration, Proceedings of the IEEE International Conference on Robotics and Automation, pp. 2516-2521, 1994.
[16] Stewart, D., Schmitz, D., and Khosla, P., CHIMERA II: A real-time multiprocessing environment for sensor-based robot control, Proceedings of the Fourth International Symposium on Intelligent Control, pp. 265-271, 1989.
[17] 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, pp. 2929-2934, 1996.
[18] Taylor, M., Blake, A. and Cox, A., Visually guided grasping in 3D, Proceedings of the IEEE International Conference on Robotics and Automation, pp. 761-766, 1994.
[19] Tomasi, C., and Kanade T., Shape and motion from image streams under orthography: a factorization method. International Journal of Computer Vision, 9(2), pp. 137-154, Nov. 1992.
[20] Williams, D. and Shah, M., A fast algorithm for active contours and curvature estimation, CVGIP (Image Understanding), 55(1), pp. 14-26, 1992.
[21] 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, pp. 575-582, 1994.