

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Autonomous 3D Shape Modeling and Grasp Planning for Handling Unknown Objects

Yamazaki Kimitoshi (*1), Masahiro Tomono (*2)
and Takashi Tsubouchi (*3)

**1 The University of Tokyo*

**2 Chiba Institute University*

**3 University of Tsukuba*

1. Introduction

To handle a hand-size object is one of fundamental abilities for a robot which works on home and office environments. Such abilities have capable of doing various tasks by the robot, for instance, carrying an object from one place to another. Conventionally, researches which coped well with such challenging tasks have taken several approaches. The one is that detail object models were defined in advance (Miura et al., 2003) , (Nagatani & Yuta, 1997) and (Okada et al., 2006). 3D geometrical models or photometric models were utilized to recognize target objects by vision sensors, and their robots grasped its target objects based on the handling point given by manual. Other researchers took an approach to give information to their target objects by means of ID tags (Chong & Tanie, 2003} or QR codes (Katsuki et al., 2003). In these challenges, what kind of information of the object should be defined was mainly focused on.

These researches had an essential problem that a new target object cannot be added without a heavy programming or a special tools. Because there are plenty of objects in real world, robots should have abilities to extract the information for picking up the objects autonomously. We are motivated above way of thinking so that this chapter describes different approach from conventional researches. Our approach has two special policies for autonomous working. The one is to create dense 3D shape model from image streams (Yamazaki et. al., 2004). Another is to plan various grasp poses from the dense shape of the target object (Yamazaki et. al., 2006). By combining the two approaches, it is expected that the robot will be capable of handling in daily environment even if it targets an unknown object.

In order to put all the characteristics, following conditions are allowed in our framework:

- The position of a target object is given
- No additional information on the object and environment is given
- No information about the shape of the object is given
- No information how to grasp it is given

According to our framework, robots will be able to add its handling target without giving shape and additional marks by manual, except one constraint that the object has some texture on its surface for object modeling.

The major purpose of this article is to present whole framework of autonomous modeling and grasp planning. Moreover, we try to illustrate our approach by implementing a robot system which can handle small objects in office environment. In experiments, we show that the robot could find various ways of grasp autonomously and could select the best grasping way on the spot. Object models and its grasping ways had enough feasibility to be easily reused after they acquired at once.

2. Issues and approach

2.1 Issues on combination with modeling and grasp planning

Our challenge can roughly be divided two phases, (1)the robot creates an object model autonomously, and (2)the robot detects a grasp pose autonomously. An important thing is that these two processes should be connected by a proper data representation. In order to achieve it, we apply a model representation named "oriented points". An object model is represented as 3D dense points that each point has normal information against object surface. Because this representation is pretty simple, it has an advantage to autonomous modeling.

In addition, the oriented points representation has another advantage can in grasp planning because the normal information enables to plan grasp poses effectively. One of the issues in the planning is to prepare sufficient countermeasures against the shape error of the object model which is obtained from a series of images. We take an approach to search good contacts area which is sufficient to cancel the difference.

The object modeling method is described in section 3, and the grasp planning method is described in section 4.

2.2 Approach

In order to generate whole 3D shape of an object, sensors have to be able to observe the object from various viewpoint. So we take an approach to mount a camera on a robotic arm. That is, multiple viewpoint sensing can be achieved by moving the arm around the object. From the viewpoint of shape reconstruction, there is a worry that a reconstruction process tends to unstable comparing with a stereo camera or a laser range finder. However, a single camera is suitable to mount a robotic arm because of its simple hardware and light weight.

A hand we utilize for object grasping is a parallel jaw gripper. Because one of the purposes of the authors is to develop a mobile robot which can pick up an object in real world, such compact hand has an advantage. In grasp planning, we think grasping stability is more important than dexterous manipulation which takes rigorous contact between fingers and an object into account. So we assume that fingers of the robot equip soft cover which has a role of conforming to irregular surfaces to the object. The important challenge is to find stable grasping pose from a model which includes shape error. Effective grasp searching is also important because the model has relatively large data.

3. Object Modeling

3.1 Approach to modeling

When a robot arranges an object information for grasping it, main information is 3D shape. Conventionally, many researchers focused on grasping strategy to pick up objects, the representation of object model has been assumed to be formed simple predefined shape primitives such as box, cylinder and so on. One of the issues of these approaches is that such model is difficult to acquire by the robot autonomously.

In constrast, we take an approach to reconstruct an object shape on the spot. This means that the robot can grasp any object if an object model is able to be acquired by using sensors mounted on the robot. Our method only needs image streams which are captured by a movable single camera. 3D model is reconstructed based on SFM (structure from motion) which provides an object sparse model from image streams. In addition, by using motion stereo and 3D triangle patch based reconstruction, the sparse shape improved into 3D dense points. Because this representation consists of simple data structure, the model can be autonomously acquired by the robot relatively easily. Moreover, unlike primitive shape approach, it can represent the various shapes of the objects .

One of the issues is that the object model can have shape errors accumulated through the SFM process. In order to reduce the influence to grasp planning, each 3D point on reconstructed dense shape is given a normal vector standing on the object surface. Oriented points is similar to the ``needle diagram" proposed by Ikeuchi (Ikeuchi et al., 1986). This representation is used as data registration or detection of object orientation.

Another issue is data redundancy. Because SFM based reconstruction uses multiple images, the reconstructed result can have plenty of points that are too much to plan grasp poses. In order to cope with this redundancy, we apply voxelization and its hierarchy representation to reduce the data. The method described in chapter 5 improves planning time significantly.

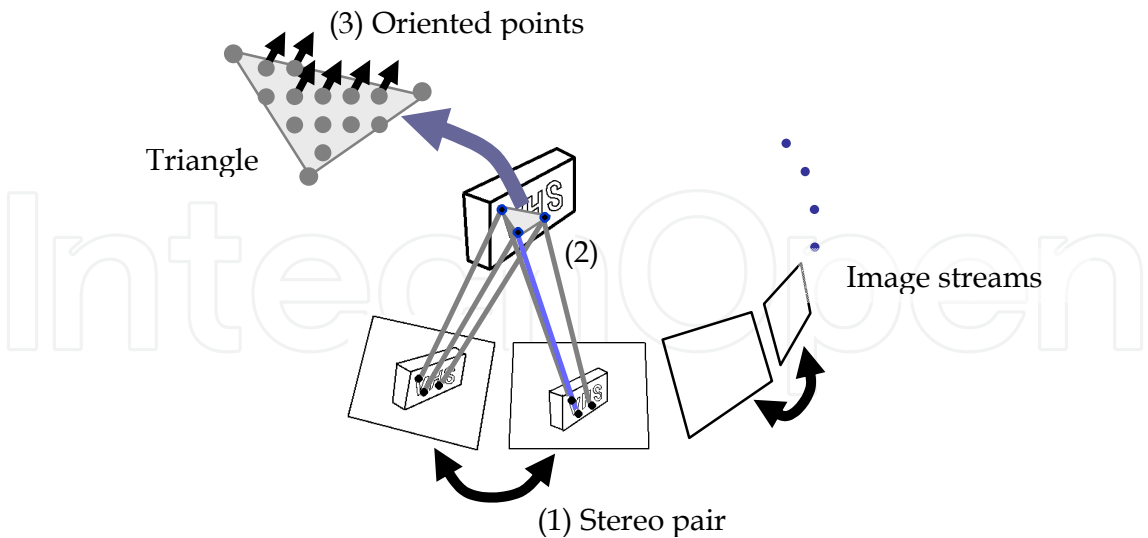


Fig. 1. Surface model reconstruction

3.2 Modeling Outline

Fig.1 shows modeling outline. An object model is acquired according to following procedure: first, image feature points are extracted and tracked from a small area which has

strong intensity by using KLT-tracker (Lucas & Kanade, 2000). From these points, object sparse shape and camera poses are reconstructed by means of SFM (we call this process “sparse model reconstruction” in the rest of this paper). Next, dense shape is acquired from a close pair of images (“dense shape reconstruction” in the rest of this paper). As a result, quite a number of points are reconstructed in online. Details of these two phases are described in next subsection.

3.3 Sparse Shape Reconstruction

In our assumption, because there are almost no given information about an object when the robot tries to grasp it, what the robot has firstly to do is to acquire its shape by using sensors mounted on. We especially focus on SFM by means of a single camera because of its small and light system. This means that the robot can have an ability to acquire whole shape of an object with observing from various viewpoints by moving its manipulator. In this approach, it is hoped that we should also consider a viewpoint planning which decide manipulator motion on the spot, so that sequential reconstruction should be applied.

Factorization method (Tomasi & Kanade, 1992) is a major approach to SFM. 3D shape can be acquired only from image feature correspondences. However, because it is basically batch process, this property prevents our purpose which demands sequential reconstruction. So we apply the factorization only initial process, and use the result as input to sequential reconstruction process. The process consist of motion stereo and bundle adjustment.

Moreover, there are other issues to utilize the result to object grasping, that is, (1) the reconstruction result includes the error of camera model linearization, (2) the scale of reconstructed object is not considered, (3) the shape is basically sparse. We cope with the item (1) by compensating the result of factorization method by means of bundle adjustment. The item (2) will be solved by using odometry or other sensors such as LRF before reconstruction. The item (3) will be solved by an approach described in next subsection.

3.3.1 Initial Reconstruction

In our assumption, the position of a target object is roughly given in advance. What the robot should firstly do is to specify the position of the object. In this process, the robot finds the target object and measures the distance between itself and the object. Next, image streams which observe the object from various viewpoints are captured, and feature points are extracted from the first image and tracked to other images. By using feature correspondences in several images which are captured from the beginning, a matrix \mathbf{W} is generated. A factorization method is suitable in this condition because it is able to calculate camera poses and 3D position of feature points simultaneously. The \mathbf{W} is decomposed as follows :

$$\mathbf{W} = \mathbf{MS}$$

where the matrix \mathbf{M} includes camera poses and the matrix \mathbf{S} is a group of 3D feature points. We use the factorization based on weak perspective camera model (Poalman & Kanade, 1997) whose calculation time is very short but its reconstruction result includes linear approximation error. In order to eliminate the linearization error, bundle adjustment is applied. Basically the adjustment needs the initial state of camera poses and 3D feature points, the result of factorization applies it with good input. After the robot acquired the

distance between itself and a target object, nonlinear minimization is performed obeying the following equation:

$$C = \sum_{i=0}^n \sum_{j=0}^2 \left\{ \left(\frac{\mathbf{r}_x^T \cdot \mathbf{m}_i}{\mathbf{r}_z^T \cdot \mathbf{m}_i} - f_1 \frac{X_i + t_{xj}}{Z_i + t_{zj}} \right)^2 + \left(\frac{\mathbf{r}_y^T \cdot \mathbf{m}_i}{\mathbf{r}_z^T \cdot \mathbf{m}_i} - f_2 \frac{Y_i + t_{yj}}{Z_i + t_{zj}} \right)^2 \right\}$$

where \mathbf{m}_i denotes i th coordinates of a feature point in j th image. P is number of observable feature points. \mathbf{r} is a column vector of a rotation matrix \mathbf{R} , t_x , t_y and t_z are the elements of translation vector from world coordinates to camera coordinates. X , Y and Z indicate 3D position of the feature point.

Through this process, despite the factorization includes linear approximation error, finally obtained result has good values for the next step.

3.3.2 Sequential Reconstruction

The initial reconstruction result provides next process with a part of 3D shape and camera poses, remained object shape is reconstructed sequentially in the next step. One of the issues on this phase is the influence of occlusion, that is, image feature points disappear or arise according to viewpoint changes. In such condition, the reconstruction should be performed whenever a new image is captured.

As often as new image is obtained, following processes are applied:

- A. A camera pose is estimated by means of bundle adjustment by using feature points which are well tracked and their 3-D position has already obtained in the former processes.
- B. 3D position of newly extracted feature points are calculated by means of motion stereo.

Feature point extraction will have frequent changes obeying the viewpoint of the camera. In this situation, motion stereo is effective because it can calculate the 3-D position of a point in each. However this method needs a pair of pre-estimated camera poses, the position of a new camera pose is firstly calculated by means of bundle adjustment. Several feature points whose 3D position is known is utilized to this process. The evaluation equation is as follows:

$$C = \sum_{i=0}^p \left\{ \left(\frac{\mathbf{r}_x^T \cdot \mathbf{m}_i}{\mathbf{r}_z^T \cdot \mathbf{m}_i} - f_1 \frac{X_i + t_x}{Z_i + t_z} \right)^2 + \left(\frac{\mathbf{r}_y^T \cdot \mathbf{m}_i}{\mathbf{r}_z^T \cdot \mathbf{m}_i} - f_2 \frac{Y_i + t_y}{Z_i + t_z} \right)^2 \right\}$$

where \mathbf{m}_i denotes i th coordinates of a feature point, P is number of observable points.

By using this equation, back projection error is evaluated and adjusted by means of Newton method. On the other hand, the equation of motion stereo is as follows:

$$C = \|\mathbf{X} - s_1 \tilde{\mathbf{m}}_1\|^2 + \|\mathbf{X} - s_2 \mathbf{R} \tilde{\mathbf{m}}_2 + \mathbf{T}\|^2$$

where $\tilde{\mathbf{m}}_1$ and $\tilde{\mathbf{m}}_2$ denotes extended vectors about corresponded feature point between two images. $\mathbf{X} = (X, Y, Z)$ indicates 3D position of the feature point, \mathbf{R} and \mathbf{T} denotes relative

rotation matrix and relative translation vector between two images, respectively. From this equation, 3D feature position is calculated by means of least squares.

In this step, each process is fast and reconstruction of the target object can be performed sequentially when an image is captured. This enables a robot to plan next camera viewpoint to acquire better shape model from the reconstructed shape in realtime.

3.3.3 Dense Reconstruction

3D dense shape is approximately calculated by using triangle patches (Fig.1, (2)). By using three vertices which are selected from neighboring features in an image, 3D patches are generated by means of motion stereo. In addition, pixels existing inner the triangle are also reconstructed by means of affine transformation based interpolation.

The reconstruction procedure is as follows: first, three feature correspondences in a pair of images are prepared, and a triangle patch is composed. Next, image pixels are densely sampled on the triangle. At this time, normal information of the patch is also added to each point (Fig.1 (3)). These process is applied to mutiple pairs of images, and all the results of 3D points are integrated as a 3D shape of the target object.

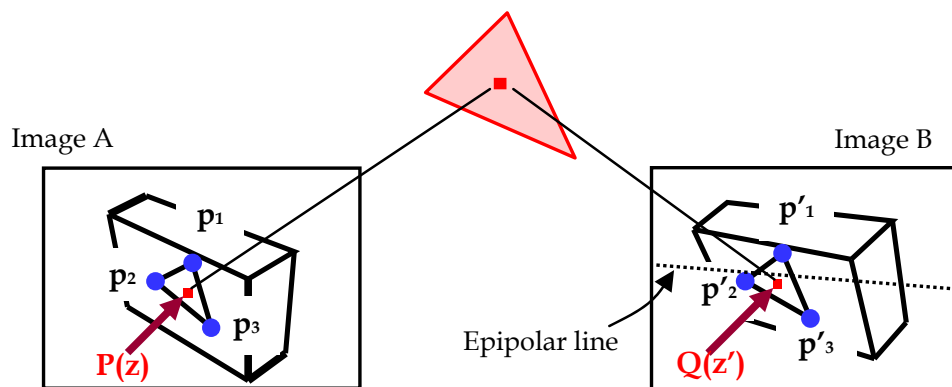


Fig. 2. Feature correspondence by using affine invariance

Fundamentally, dense 3D shape reconstruction is achieved by a correlation base stereo, all the correspondence of pixels in two images must be established and camera poses of them are known. However, making correlation is computational power consuming process and takes long time. So this section describes a smart and faster algorithm for dense 3-D reconstruction, where sparse correspondence of the feature points which is already obtained in the sequential phase is fully utilized. The crucial point of the proposed approach is to make use of affine invariance in finding a presumed pixel Q in Image B in Fig.2 when a pixel P in Image A in Fig.2 is assigned in a triangle that is formed by the neighbor three feature points. The affine invariance parameter α and β is defined as follows:

$$P(\mathbf{z}) = \alpha(\mathbf{p}_2 - \mathbf{p}_1) + \beta(\mathbf{p}_3 - \mathbf{p}_1) + \mathbf{p}_1$$

where \mathbf{z} is a coordinate vector of pixel P , and \mathbf{p}_n ($n = 1, 2, 3$) is a feature point in image A in Fig.2. α and β are invariant parameters which enable to correspond a pixel P in image A with a pixel Q in image B by following equation:

$$Q(\mathbf{z}') = \alpha(\mathbf{p}'_2 - \mathbf{p}'_1) + \beta(\mathbf{p}'_3 - \mathbf{p}'_1) + \mathbf{p}'_1$$

where \mathbf{p}'_n ($n = 1, 2, 3$) in image B in Fig.2 is a corresponding feature point to \mathbf{p}_n ($n = 1, 2, 3$) in image A respectively.

In this approach, we must take notice that the proposed approach employs 2 dimensional affine transformation, and the presumed point Q contains an error in the coordinate vector \mathbf{z}' . Therefore, it is necessary to verify the point \mathbf{z}' with the criteria as follows:

- Distance between presumed pixel \mathbf{z}' of Q to epipolar line in image B in Fig.2 from image A is within a certain threshold.
- A radiance of the pixel Q in image B in Fig.2 is same with the pixel P in image A.

After making the pixel to presumed pixel correspondence in the two images, conventional motion stereo method yields dense 3-D object shape reconstruction. Avoiding conventional correlation matching of the pixels in the two images provides computation time merit in the reconstruction process.

In the next step, 3-D points which are obtained by above stereo reconstruction are voted and integrated into a voxel space. Because the reconstruction method by affine invariance includes 2-D affine approximation, reconstruction error will become larger at a scene which has long depth or a target object which has rough feature points. There will be phantom particles in shape from the reconstruction by two images. Therefore, voting is effective method to scrape redundant or phantom particles off and to extract a real shape. Fig.3 shows the voxelization outline. The generated model (oriented points) becomes a group of voxels with giving normal information in each voxel.

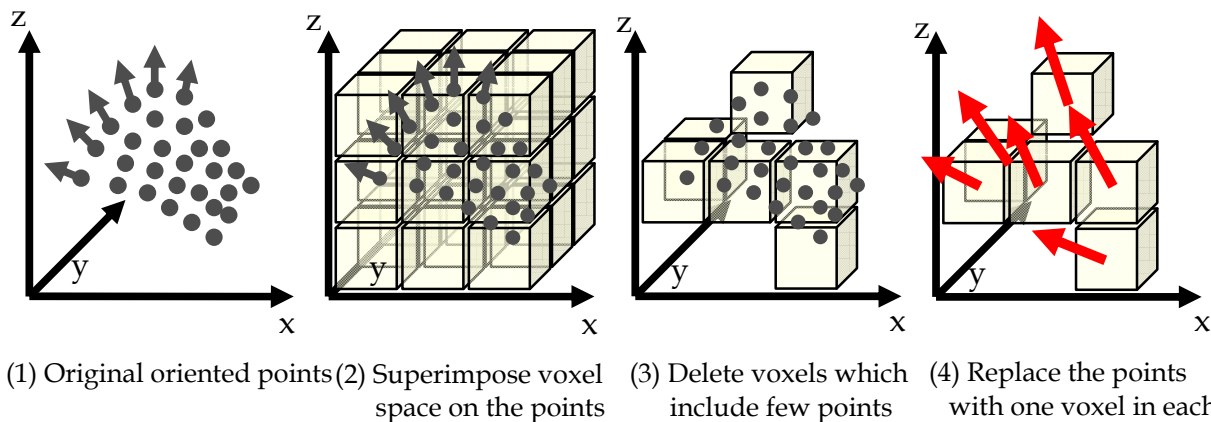


Fig. 3. Model voxelization

In addition to above voxelization process to cope with 3-D error originated from Affine transformation, not only the voxel just on the surface of the reconstructed 3-D shape but also the adjacent voxels are also voted into the voxel space. After finishing the vote from all the reconstructed shapes originated from the image stream around the target object, voxels that has the large voted number exceeding the threshold are saved and other voxels are discarded. The result of reconstruction is presented by a group of voxels which has thickness in its shape.

We also propose hierarchy data representation for effective grasp planning. It is described in section 5 in detail.

4. Grasp Planning

The purpose of our grasp planning is to find reasonable grasp pose based on automatically created model.

4.1 Approach to Grasp Planning

Grasp planning in this research has two major issues:

- how to plan a grasp pose efficiently from the 3D dense points,
- how to ensure a grasp stability under the condition that the model may have shape error.

It is assumed that fingers will touch the object by contacting with some area not at a point. Because the object model obtained from a series of images in this paper is not perfectly accurate, the area contact will save the planning algorithm from the difference of the model shape and real shape of the object.

In order to decide the best grasp pose to pick up the object, planned poses are evaluated by three criteria. First criterion is the size of contact area between the hand and the object model, second criterion is a gravity balance depending on grasp position on the object, and third criterion is manipulability when a mobile robot reaches it hand and grasps the object.

4.2 Evaluation method

The input of our grasp planning is 3D object model which is built autonomously. The method should allow the model data redundancy and the shape error. The authors propose to judge grasp stability by the lowest sum total of three functions as follows:

$$F = w_1 F_1(\mathbf{P}_1, \mathbf{x}_h^o) + w_2 F_2(\mathbf{P}_1, \mathbf{x}_h^o) + w_3 F_3(\mathbf{P}_1, \mathbf{x}_h^o, \eta)$$

where \mathbf{P}_1 is a center point of finger plane on the hand. This point is a point to contact with object. \mathbf{x}_h^o is a hand pose (6-DOF), η is a position of a robot. w_i is a weight.

$F_1(\cdot)$ represents the function of contact area between the hand and the object. The evaluation value becomes smaller if the hand pose has more contact area. $F_2(\cdot)$ represents the function of a gravity balance. The evaluation value becomes small if a moment of the object is small. $F_3(\cdot)$ represents the function of the grasping pose. The evaluation value becomes small if the amount of robot motion to reach to the object is small. The policy of grasp planning is to find \mathbf{P}_1 , \mathbf{x}_h^o and η which minimize the function of F .

As it is necessary to yield the moment of inertia of the object, the model must be volumetric. For this purpose, voxelized model are extended to everywhere dense model through following procedure: a voxel space including all the part of the model is defined, then the voxels of outside of the object are pruned away. Finally, the reminder voxels is a volumetric model.

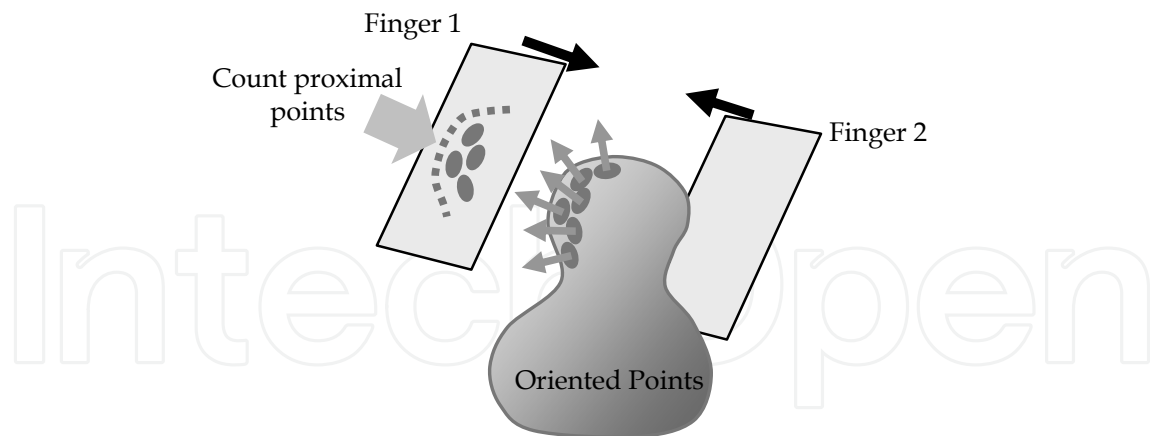


Fig.4 Grasp evaluation based on contact area

4.2.1 Grasp Evaluation based on Contact Area

In order to calculate the function $F_1(\mathbf{P}_1, \mathbf{x}_h^o)$, contact area between the hand and the object is considered. Detail equation can be represented as follows :

$$F_1(\mathbf{P}_1, \eta) = \begin{cases} c & (\text{if } S(\mathbf{P}_1, \eta) \geq S_0) \\ \exp(S_0 / S(\mathbf{P}_1, \eta)) & (\text{if } S(\mathbf{P}_1, \eta) < S_0) \\ \infty & (\text{if } S(\mathbf{P}_1, \eta) = 0) \end{cases}$$

where S_0 is a threshold. $S(\mathbf{P}_1, \mathbf{x}_h^o)$ is the size of contact area. c is a positive constant.

The size of contact area is approximately estimated by counting the voxels in the vicinity of the fingers. The advantage of this approach is that the estimation can merely be accomplished in spite of complexity of the object shape. As shown in Fig.4, the steps to evaluate the contact area are as follows: (i) assume that the hand is maximally opened, (ii) choose one contact point \mathbf{P}_1 which is a voxel on the surface of the model, (iii) consider the condition that the center of the one finger touches at \mathbf{P}_1 and the contact direction is perpendicular to the normal at \mathbf{P}_1 , (iv) calculate contact area as the number of voxels which are adjacent \mathbf{P}_1 with the finger tips. (v) Assume that the other finger is touched with the counter side of the object and count the number of voxels which are touched with the finger plane.

The grasping does not possible if any of following contact conditions applies.

- contact area is too small for either one or both of fingers,
- the width between the finger exceeds the limit,
- the normal with the contacting voxel is not perpendicular to the finger plane.

Change the posture \mathbf{P}_1 by rotating the hand around the normal with certain step angles, above evaluation (i) to (v) is repeated.

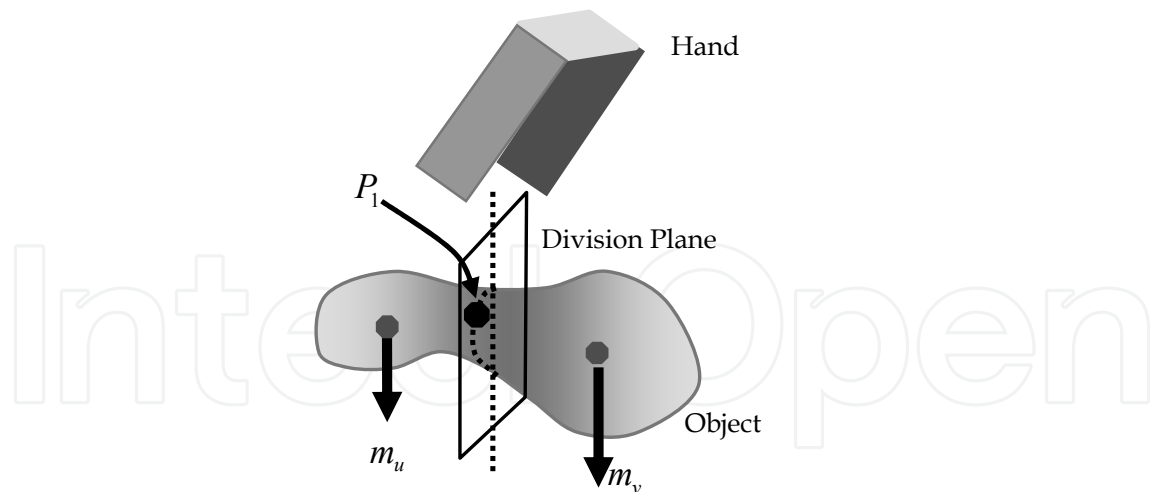


Fig. 5. Grasping evaluation based on gravity balance

4.2.2 Grasp Evaluation of Gravity Balance at Gradient

In order to calculate the function $F_2(P_1, x_h^o)$, a moment caused by a gravity is considered. The moment is easily calculated by investigating voxels which occupies in the volume of the object model. As shown in Fig.5, the model is divided into two volumes by a plane which is parallel to the direction of gravitation. If the two volumes give equivalent moment, good evaluation is obtained:

$$F_2(P_1, \xi) = \begin{cases} c & (\text{if } M < M_0) \\ \exp(M) & (\text{if } M \geq M_0) \\ \infty & (\text{if } m_u = 0 \text{ or } m_v = 0) \end{cases},$$

where

$$M = K \frac{|m_u - m_v|}{m_u + m_v}.$$

The m_u is a moment to u derived from gravitation. K is a positive constant. The equation to calculate M has a role of normalization which prevent a difference of the moment at volume u, v relying on the size of the object.

Although it is naturally strict to consider another balance requirement such as force-closure, the authors rather take $F_2(\cdot)$ for moment balance criterion according to the following reasons. The one reason is that it is difficult to evaluate the amount of the friction force between the hand and grasped object, because there are no knowledge about the material or mass of the object. The second reason is that a grasping pose which is finally fixed on the basis of this evaluation can be expected to maintain the gravity balance of the object. Our approach assumes that the grasping can be successfully achieved unless the grasp position is shifted in very wrong balance, because a jaw gripper hand is assumed to have enough grasping force. This means that the finally obtained grasp pose by the method proposed here roughly maintains forceclosure grasp.

4.2.3 Grasp pose evaluation based on robot poses

Although evaluation criteria described above are a closed solution between an object and a hand, other criteria should be considered when we aim to develop an object grasping by a real robot. Even if good evaluation is acquired from the functions $F_1(\cdot)$ and $F_2(\cdot)$, it may be worthless that the robot cannot have grasping pose due to kinematic constraint of its manipulator.

In order to judge the reachability to planned poses, we adopt two-stage evaluation. At first, whether or not inverse kinematics can be solved is tried to a given grasp pose. If manipulator pose exist, the $F_3(\cdot)$ is set to 0. In other case, second phase planning is performed. Robot poses including standing position of the wheelbase are also planned. In this phase grasping pose is decided by generating both wheelbase motion and joint angles of the manipulator (Yamazaki et al, 2008).

4.3 Efficient grasp pose searching

In the pose searching process, oriented point which is touched to P_1 is selected from the model in order. Because such monotonous searching is inefficient, it is important to reduce vain contact between finger and the object model. In order to implement fast planner, oriented points which can have good evaluation are firstly selected. This can be achieved to restrict the direction of the contact by utilizing normal information of each point. In addition, another approach to reduce the searching is also proposed in next section.

5. Model Representation for Efficient Implementation

As described in section 3, the model represented by oriented points has redundant data for grasp planning. By transforming these points to voxelized model, redundant data can be reduced. This section describes some issues on the voxelization and its solution.

5.1 Pruning voxels away to generate thin model

From a viewpoint of ensuring grasping success rate, it is expected that the size of voxel is set 2mm to 5mm because of allowable shape error. One of the issues of voxelization under the setting is that the voting based model tends to grow in thickness on its surface. This phenomenon should be eliminated for effective grasp planning.

An algorithm to acquire a "thin" model is as follows: (1) select a certain voxel from voxelized model, (2) define cylindrical region whose center is the voxel and its direction is parallel to the normal of the voxel. (3) Search 26 neighbor voxels and find voxels which are included the cylindrical region. This process is performed recursively. (4) calculate an average position and normal from the listed voxels, and decide a voxel which can be ascribed to object surface.

Through this thinning, number of reconstructed points reduces from several hundred thousands to several hundreds. Moreover, this averaging has effect of diminishing shape error of the model.

As described in section 4.2, volumetric model is also needed. Such model is generated from the model created through above procedure. Because the process consumes few time, this is one of the advantage of voxelized model.

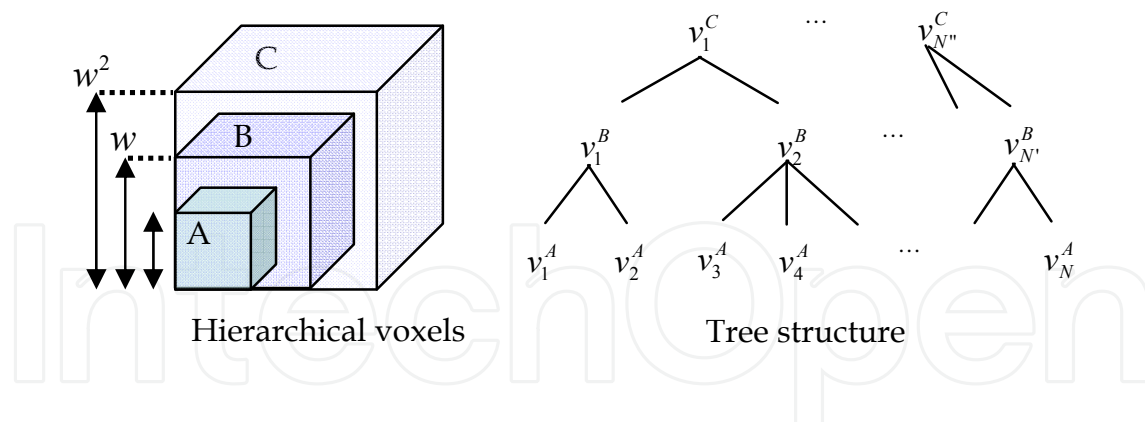


Fig. 6. hierarchical representation

5.2 Hierarchical Data Representation

The method mentioned in 5.1 can reduce the number of pose searching. However, the searching has potential to be still capable of improving. For instance, there are somewhat points which obviously need not to be checked. From this reason, hierarchical data representation is adopted to exclude needless points before judging the quality of grasp pose. Using the new formed model, the searching can be performed at some parts on object model where will have rich contact area with fingers.

The hierarchical representation is similar to octree. Octree is often used for judging collision in the field of computer graphics. The transformation procedure is as follows: at first, initial voxels which construct original voxelized model are set hierarchical A. Next, other voxel space which is constructed w times larger voxels than hierarchical A is superimposed on the voxels of hierarchical A. A new model is represented by the larger voxels which are set hierarchical B. In this processing, only voxels belonging to hierarchical B are adopted when these voxels include much number of voxels which has similar orientation at hierarchical A. The same hierarchy construction is performed from hierarchical B to hierarchical C, too. As a result, one voxel of hierarchical C includes several voxels of hierarchical A. Because these voxels of hierarchical A are grouped and has similar orientation, the area can be expected that it supplies rich contact area with finger.

In the grasp pose searching, voxels of hierarchical C are selected in order. The evaluation is performed about inner voxels which belong to hierarchical A. This approach can achieve efficient searching with selecting only voxels which are guaranteed to provide good evaluation result about contact area.

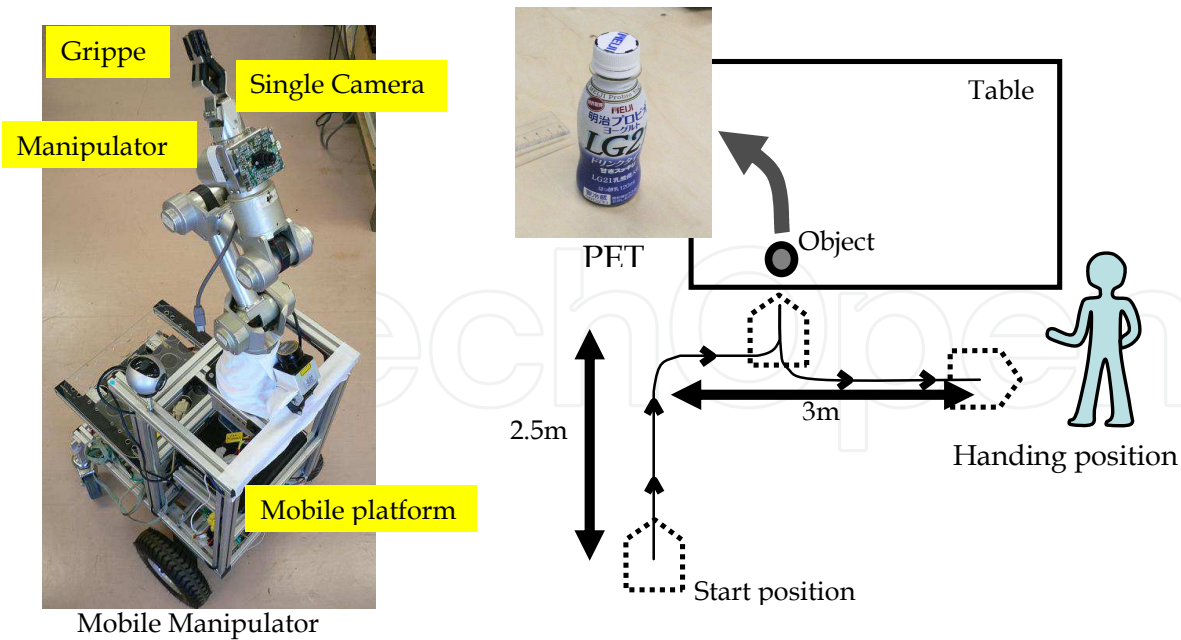


Fig. 7. A robot system and an assumed task

6. Experiments

6.1 System setup

Fig7. shows a robot system in our use. A 5DOF manipulator made by Nuronics Inc. was mounted on the mobile base “Yamabico” which was developed by Intelligent robot laboratory, Univ. of Tsukuba. A camera mounted on the wrist of the manipulator was used to capture image streams with observing a target object while the manipulator moving. A LRF sensor, URG04-LX made by Hokuyo Inc. was mounted on the wheelbase. Two portable computers were also equipped. The One (Celeron 1.1GHz) was to controll the wheelbase and the manipulator from the result of planning. Another (Pentium M 2.0GHz) was to manage reconstruction and planning process.

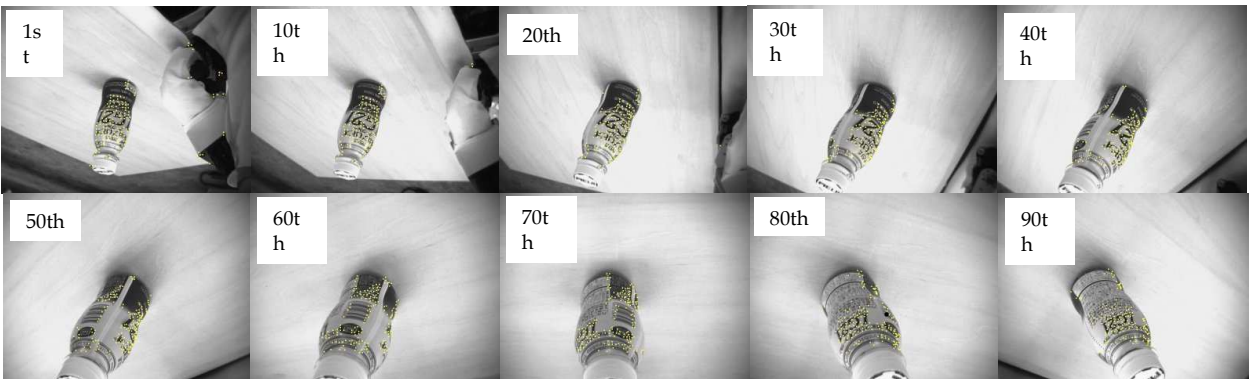


Fig. 8. Image streams in case of a plastic bottle

6.2 Proof experiments of automatic 3D modeling and grasp planning

Firstly, several small objects having commonly texture and shape were selected and they were tried to reconstruct the shape and to plan grasp poses.

The procedure of proof experiments was as follows : (i) the robot moved to the front of instructed position by manual, and (ii) observed to detect a target object position by using LRF. (iii) From the result, camera trajectory (several via points and their interpolation for the end-effector of the manipulator) was calculated to capture image streams related to the target object.

Fig.8 shows an example when a target object was a plastic bottle which had 120mm height. Number of captured images were 134, and 150 feature points were extracted and tracked in each image. These image features were used to reconstruct its 3D shape by means of an algorithm described in section 3.

Factorization method and bundle adjustment took 300 msec for firstly captured 10 images. After that, one time of sequential reconstruction of a camera pose and object shape took 30 msec in each image. Dense shape reconstruction was performed when all images were finished to capture. It used 55 image pairs to make oriented points and the result was integrated into voxel space. The processing time was 2.4 sec.






Types of objects					
Results	Rectangular box	Cup	Toy	Stapler	Ornament
Points	807	986	966	934	1531
Grasp pose candidates	64	37	49	48	58
Processing time (A) [sec]	0.96	1.01	0.94	0.86	1.36
Processing time (B) [sec]	9.3	6.3	6.7	7.2	16.3

Fig. 9. Experimental results of object modeling

Fig.9 shows several results of object modeling and grasp planning. There were five objects which had relatively rich texture on its surface. Numbers in ‘Points’ row shows number of reconstructed points after voxelization, and numbers in ‘Grasp pose candidates’ shows number of grasping poses through the algorithm descibed in section 4. Other 2 rows shows processing times of the planning.

Notice that the planning times were not related to object shape complexity. In these experiments, dozens of grasp poses could be found from each created models about 1 second (Pentium M, 2.0 GHz) as shown in processing times (A). On the other hand, the results in (B) as shown in Fig.9 indicate the processing time without utilizing hierarchy data representation describe in section 5.2. The representation succeeded 7 to 10 times speeding up the planning.

On the other hand, some problems were cleared up through this experiments. For instance, an area where had no texture cannot reconstructed by our modeling method. This means grasp poses which touch to inner of the cup could not be selected in grasp planning.



Fig. 10. Examples of object grasping by using planning result

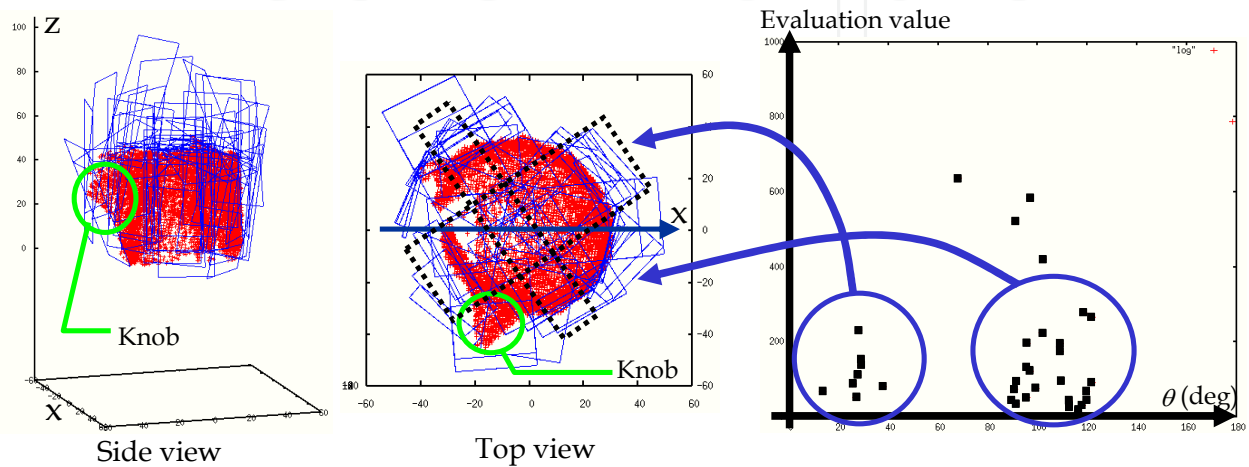


Fig. 11. Planned grasping poses in case of a cup

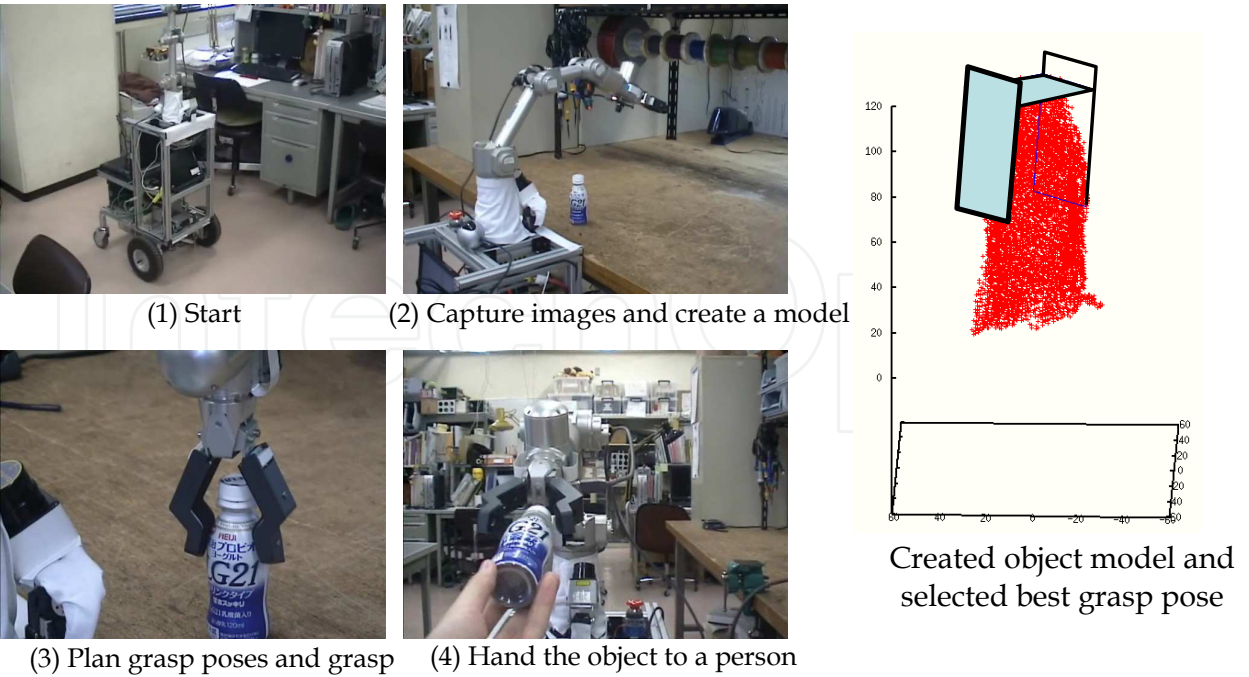


Fig. 12. Implementation to delivery task

Fig.11 shows grasp planning results in case of a cup as shown second example in Fig.9. Red points indicate 3D reconstructed points, and blue rectangles show pseudo finger planes. The most right graph shows a distribution of evaluation of grasp poses. Because low value indicates a good evaluation, grasp poses which touched to the side of the cup without a knob were judged to stable area to contact.

6.4 Integration to delivery task

An object carrying task was tried by using a mobile manipulator. The goal of this experiment was to hand an object to a person who stood another place from the object position. As described in section 2, we assumed that (i)there were no constraint on the object shape and no tags or marks on the object, (ii)relatively much natural texture could be found on the object surface, and (iii)the object has equivalent size that human could grasp it by one hand. Jaw Gripper hand which was a compact and light weight were used for grasping the object. Thin sponges were pasted up to the fingers to ensure area-based contact with the object.

Environment map which included the position of the object was given in advance. Moreover, the initial position of the robot and the position of the person were given in advance, too. In this condition, the robot planned its motion trajectory automatically by using artificial potential method (Connolly et al., 1990). As shown in Fig.12, the robot succeeded to picking up the object based on our automatic 3D modeling and grasp planning, and handed the object to a person who sit down on the side of a table.

7. Conclusion

In this chapter, a 3D modeling and grasp planning methods were described. Because the two methods were densely combined with the model representation 'oriented points', autonomous mobile manipulator implemented these methods can handle objects which are placed on real world without giving their shape and grasp information in advance. The authors showed the effectiveness of our approach through experiments by using a real robot.

8. References

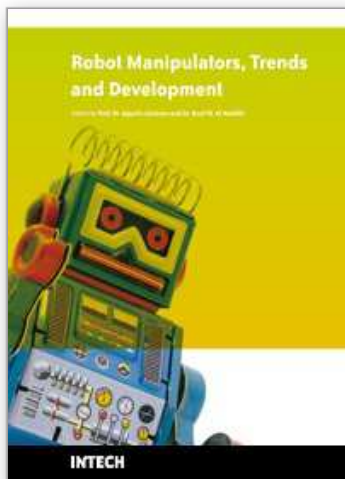
- N. Y. Chong & K. Tanie, *Object Directive Manipulation Through RFID*, Proc. of Int. Conf. on Control, Automation, and Systems, pp.22-25, 2003.
- C. Connolly, J. Burns and R. Weiss, *Path Planning using Laplace's Equation*, Proc. of the IEEE Intl. Conf. on Robotics and Automation, pp.2102-2106, 1990.
- R. Katsuki et al., *Design of Artificial Marks to Determine 3D Pose By Monocular Vision*, Proc. 2003 IEEE Int. Conf. Robotics and Automation, pp.995-1000, 2003.
- J. Miura et al., *Development of a Personal Service Robot with User-Friendly Interfaces*, 4th Int. Conf. on Field and Service Robotics, pp.293- 298, 2003.
- K. Nagatani & S. Yuta, *Autonomous Mobile Robot Navigation Including Door Opening Behavior-System Integration of Mobile Manipulator to Adapt Real Environment-*, Intl. Conf. on FSR, pp.208-215, 1997.
- K. Ikeuchi et al., *Determining Grasp Configurations using Photometric Stereo and the PRISM Binocular Stereo System*, The Int. Journal of Robotics Research, Vol. 5, No. 1, pp.46.65, 1986.

- K. Okada et al., *Vision based behavior verification system of humanoid robot for daily environment tasks*, Proc. of 6th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2006), pp 7-12, 2006.
- L. Petersson et al., *Systems Integration for Real-World Manipulation Tasks*, Proc. of IEEE Int. Conf. Robotics and Automation, pp.2500-2505, 2002.
- C. J. Poalman and T. Kanade. *A paraperspective factorization method for shape and motion recovery*, IEEE Trans. Pattern And Machine Intelligence, Vol.17, No.3, pp.206 – 217, 1997.
- J. Shi et al., *Good Features to Track*, IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp.593–600, 1994.
- C. Tomasi & T. Kanade. *The factorization method for the recovery of shape and motion from image steams*, Proc. Image Understanding Workshop, pp.459 – 472, 1992
- K. Yamazaki et al., *3D Object Modeling by a Camera mounted on a Mobile Robot*, Proc. of the 2004 IEEE Int. Conf. Robotics and Automation, 2004.
- K.Yamazaki et al., *A Grasp Planning for Picking up an Unknown Object for a Mobile Manipulator*, Proc. of the 2006 IEEE Int. Conf. Robotics and Automation, 2006.
- K.Yamazaki et al., *Motion Planning for a Mobile Manipulator Based on Joint Motions for Error Recovery*, Proc. of the 2006 IEEE Int. Conf. Intelligent Robots and Systems, pp. 7–12, 2006.

IntechOpen

IntechOpen

IntechOpen



Robot Manipulators Trends and Development

Edited by Agustin Jimenez and Basil M Al Hadithi

ISBN 978-953-307-073-5

Hard cover, 666 pages

Publisher InTech

Published online 01, March, 2010

Published in print edition March, 2010

This book presents the most recent research advances in robot manipulators. It offers a complete survey to the kinematic and dynamic modelling, simulation, computer vision, software engineering, optimization and design of control algorithms applied for robotic systems. It is devoted for a large scale of applications, such as manufacturing, manipulation, medicine and automation. Several control methods are included such as optimal, adaptive, robust, force, fuzzy and neural network control strategies. The trajectory planning is discussed in details for point-to-point and path motions control. The results in obtained in this book are expected to be of great interest for researchers, engineers, scientists and students, in engineering studies and industrial sectors related to robot modelling, design, control, and application. The book also details theoretical, mathematical and practical requirements for mathematicians and control engineers. It surveys recent techniques in modelling, computer simulation and implementation of advanced and intelligent controllers.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Yamazaki Kimitoshi, Masahiro Tomono and Takashi Tsubouchi (2010). Autonomous 3D Shape Modeling and Grasp Planning for Handling Unknown Objects, Robot Manipulators Trends and Development, Agustin Jimenez and Basil M Al Hadithi (Ed.), ISBN: 978-953-307-073-5, InTech, Available from:
<http://www.intechopen.com/books/robot-manipulators-trends-and-development/autonomous-3d-shape-modeling-and-grasp-planning-for-handling-unknown-objects>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen