

# Vision-Based Robot Manipulator for Grasping Objects

Young Min Baek<sup>1,3</sup>, Ho Seok Ahn<sup>1,3</sup>, In-Kyu Sa<sup>2</sup>, Min Woo Lee<sup>3</sup>, Chang Kyun Lee<sup>3</sup>,  
Taehyun Kim<sup>3</sup>, Won Il Choi<sup>3</sup>, Byung Soo Lim<sup>4</sup>, Jin Young Choi<sup>1</sup>

<sup>1</sup> School of Electrical Engineering and Computer Science, ASRI, Seoul National University, Seoul, Korea,  
e-mail : {ymbaek, hsahn, jychoi}@neuro.snu.ac.kr

<sup>2</sup> Telecommunication R&D Center, Telecommunication Network Business Operations, Samsung Electronics., Suwon, Korea,  
e-mail : inkyu.sa@samsung.com

<sup>3</sup> Software Membership, Corporate Technology Operations, Samsung Electronics., Seoul, Korea,  
e-mail : {hsahn, ymbeak, mwlee, cklee, thkim, wichoi}@secsm.org

<sup>4</sup> Software Laboratories, Corporate Technology Operations, Samsung Electronics., Seoul, Korea,  
e-mail : byungsoo.lim@samsung.com

**Abstract**— Robot manipulator is one the most important features of a service robot. Manipulators are used to grasp objects or to gesture. Until now, there has been a lot of research about robot manipulators that can imitate the functions of a human being by recognizing and grasping objects. One example uses sensors and the other uses a predefined trajectory. But there are some drawbacks in manipulation without visual information. In this paper, we present a robot manipulator based on the object recognition system. We implemented closed-loop control that uses the feedback from visual information. In monocular camera system, we just can estimate approximate distance between the gripper and the object by comparing size of areas. So we use a sonar sensor to compensate estimation error and to improve the accuracy. We placed the web-camera on the top of the hand to recognize objects. We also present some vision-based manipulation issues and our system features.

## I. INTRODUCTION

Robots are used in various areas nowadays. They are used in factories, home and public places. According to this stream, we have developed a store-service robot named 'Part-Timer'. Robots that work in the stores need many technologies like SLAM, motor control, face recognition, object recognition, system integration, and so on. Using these skills, service robots can perform several tasks.

Among tasks, the most important thing is delivering the ordered item to customers. For delivery task, service robots should have three necessary modules; (1) Moving parts (SLAM Module), (2) High degree of freedom manipulator, (3) Recognition System. If robots fulfill these three conditions, it will be able to serve customers and replace a human part-timer.

We designed three essential systems for grasping objects. There are recognition and control system. The other one is a moving part, but in this paper, we would not explain about it in more detail. The recognition system interacts with outer environment and estimates how far the desired object is. The

control system actually moves the DC motors to locate the gripper on our desired position. We will explain the system structure and how the system work properly. We have designed flow-chart of delivery service first, and had experiments for our purpose.

In this paper, we focus on the manipulator for grasping objects using visual information. First, we present several issues about vision-based grasping manipulation, and show how we have developed the system. Next, we discuss several integration issues between the recognition system and the manipulation control system. Finally, we show the experimental results and conclude.

## II. RELATED WORKS

There are several researches in the use of visual information for manipulation. Using visual information is very useful for robots to interact with environments.

Smith presented vision-guided robotic issues [1]. The author mentioned 5 categories; (a) Open-Loop and Closed-Loop, (b) Blind and Visually-Guided Grasp, (c) Monocular, Stereo and Structured Light, (d) Camera Placement, (e) Object Geometry. This paper shows what we should consider about vision-guided grasping. We have considered these issues and trade-offs. This paper helps us to design the system architecture.

Sanz proposed the grasping of unknown objects based on computer vision [2]. He showed novel techniques to solve the problems under the imposed resource constraints. Computer vision technique played more important role to implement the system in real time.

Stanley described vision-based planar grasp planning [3]. A representation of the target and a quadtree expansion generate candidate grasps that are compared using a cost function. It is developed for a large number of planar or near planar objects.

Morales developed vision-based grasping using a humanoid robot with five-fingered hands [4]. The aim of this paper is to provide the humanoid robot with the ability of

grasping objects that appear in a kitchen environment. They used an object model database that contains the description of all the objects. The robot is able to grasp manifold types of objects using its five-fingered hands.

There is other approach for grasping. Saxena purposed a learning algorithm which predicts the position at which to grasp the object [5]. The robot is able to grasp differently shaped objects after the algorithm is trained via supervised learning.

These papers are commonly purposed on grasping objects. For this purpose, researchers designed various types of manipulators. We have chosen simple objects like beverage cans as target objects. Therefore, we have designed simple gripper. Object recognition system is very important for our system because that system recognizes objects in real-time manner, and estimates distance. We will discuss about it next section.

### III. SYSTEM OVERVIEW

We already have mentioned that there are some issues about vision-based grasping manipulation [1]. Smith introduced two kinds of control systems; Open-Loop and Closed-Loop. In case of the open-loop control, the vision system is used only to determine the object's position and orientation prior to a blind grasp. On the contrary, closed-loop control allows visual data to compensate for manipulator positioning inaccuracies and sensor noise. Typically, open-loop control requires more accurate calibration while closed-loop control requires faster vision system performance. For our system, we developed a high-performance object recognition algorithm choosing the close-loop control using SIFT. [7] It gives us more accurate results than open-loop systems because of compensating for noise data. Although object's position may be changed during grasping time, closed-loop system would be able to track transition of target object. Because vision system computation is quite burden, we use Pentium IV 2.4GHz only for vision system.

We placed a web-camera on the back of the hand in order not to hide objects while the manipulator is grasping object. We use a monocular camera that required no calibration and reduced hardware requirements. Unfortunately, monocular camera systems cannot provide three-dimensional location of target object. That is why we need to recover depth through some different way. We have implemented approximation distance calculator by comparing camera image size to database image size. We will discuss about this in section 5.

The shape and complexity of a gripper is different from the object that the manipulator would try to grasp. We have designed the gripper that can grasp beverage cans to work in practical services. The gripper has one degree of freedom, so it can work in two ways; closing, opening. It has one DC motor with a worm gear. Fig.1 shows mechanical design of the gripper.

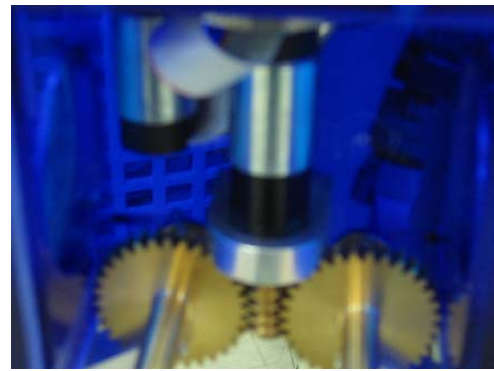


Fig.1 Mechanical Design of the Gripper

Fig.2 shows overall system architecture of the manipulation system. There are two main system parts; vision system and control system. The vision system takes charge of the object recognition, and the control system is in charge of positioning the robot manipulator on the desired position. Therefore, the vision system includes the object recognition system and depth finder to calculate the 3-dimensional position using the web camera. In the control system, angle value of the each shaft will be calculated by inverse kinematics rule. Then, the manipulator positioning controls the motors to each angle value. All of these two main parts of the manipulator is working in one computing system. Only manipulator positioning part is using three micro controllers. Each controller is able to control only two motors. The main system works in a different computer to manage other modules. It helps to improve the performance.

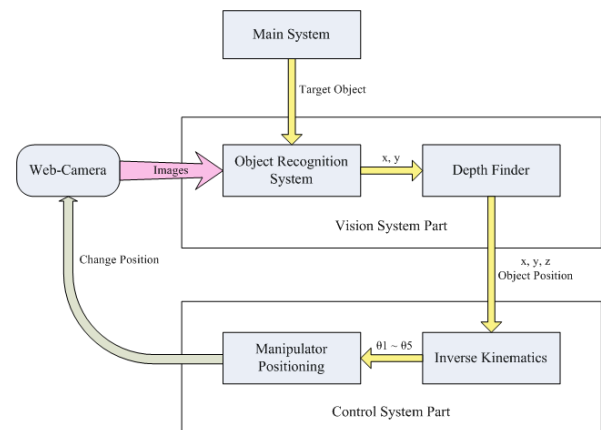


Fig.2 Overall System Architecture

Now we present a data flow of our system. First, the web-camera captures an image. The image coming from the web-camera goes to the object recognition system. The recognition system calculates a distance from the target object to the gripper. After that, the robot can get approximate coordinates (x, y, z) based on the web-camera. Depth value may not guarantee the accuracy, so we use sonar sensor to compensate errors. Next, the control system solves the inverse kinematics problem and moves motors to the desired position. The object recognition system is receiving the object position by changing the location of gripper. This con-

stitutes closed-loop system using the feedback of visual data.

Fig.3 shows the entire robot 'Part-Timer'. [8] Bottom level is a moving part that has two wheels. Middle level is a main system part where the main scheduler is running in. It has five sonar sensors to avoid obstacles. Upper level is a manipulator part. We already described system architecture. As you can see, it has one robot manipulator.



Fig.3 Store-Service Robot 'Part-Timer'

#### IV. DESIGN OF ROBOT MANIPULATOR

We have designed 6-DoF robot manipulator to grasp objects. There are two motors on the shoulder (Roll, Pitch), one motor on the elbow (Pitch), and two motors on the wrist (Roll, Pitch). The end effector also has one motor to drive the worm gear that controls the gripper.

We designed one manipulator and put it in front of the robot. Basically one manipulator is not enough to perform several motions. So we have considered about the efficient position of the manipulator. As each shaft has boundary in some area, entire workspace is quite small. To fulfill the delivery service, the robot has moving part. Moving parts can work concurrently with manipulator through main scheduling system.

We designed the manipulator using Solid Works. The CAD file of the manipulator is shown in Fig.4.

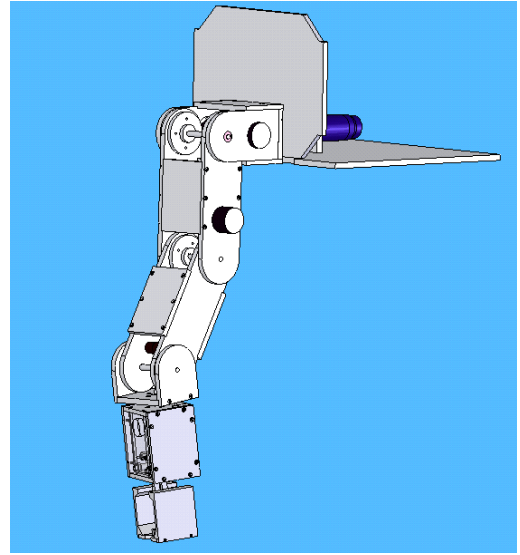


Fig.4 An example of CAD design of the manipulator

To grasp the objects, the robot should interact with its environment. For example, it should perceive where the desired object is. So the object recognition system is necessary. If there is a camera on the robot on same height as human eyes are, the robot cannot recognize objects very well because it could be too far from the object. We put the web-camera (Logitech Pro 4000) on the back of the hand so that it can see the object closer and move other position during searching objects. Even if there are no objects on the camera screen, the robot tries to find the object by locating its end effector to another position by controlling its robot manipulator. Placing the camera on the hand is more useful than placing it on the head.

Furthermore, it is hard to calculate the distance from the camera because the robot has only one camera. Therefore, vision system roughly estimates the distance, and compensates the distance by using ultra sonic sensor. We will discuss compensating errors in the next section.

Fig.5 shows the robot arm with a web-camera and a sonar sensor.

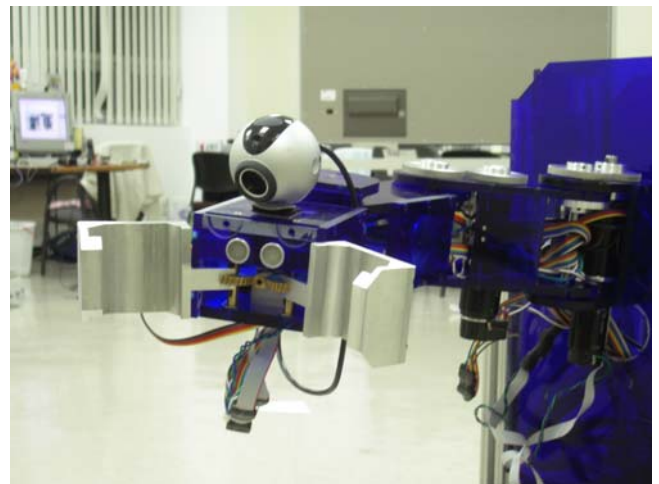


Fig.5 The gripper with a camera and a sonar sensor

## V. OBJECT RECOGNITION SYSTEM

There are many object recognition algorithms. We choose Scale-Invariant Feature Transform [6] to integrate our system. SIFT uses local features of the image, so it is robust to scale, rotation and change of illuminations. Our system needs robust and high performance vision system. Our vision system should process 5 frames per second at least for the closed-loop control. So we implemented basic SIFT algorithm and customized it for our robot system. Fig.6 shows an example of SIFT algorithm.

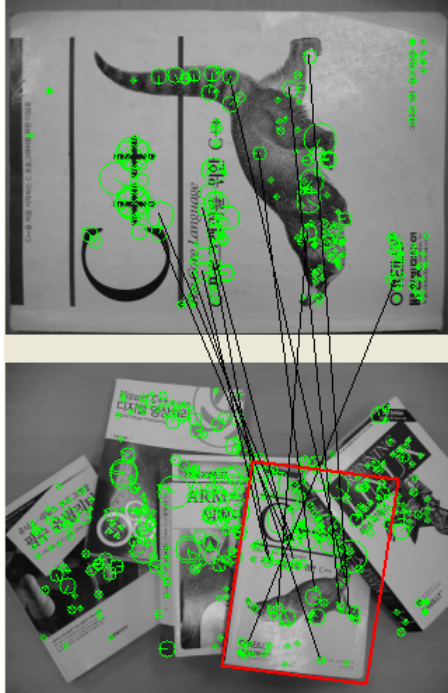


Fig.6 An example of the SIFT algorithm

Unfortunately, the robot has just one camera on the hand, so it is not able to estimate exact distance like when using stereovision. Therefore, more specific distance for the object database is required to calculate the distance using only one camera. When we make the object database, the robot should know the distance from the object to the camera. Then we calculate the distance comparing the area of object and the size of the database. The size of the object is inversely proportional to the square of the distance. Fig.7 shows a relationship between the area size and the distance.

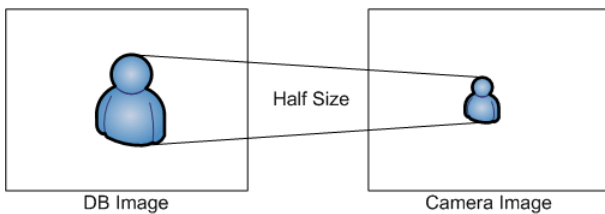


Fig.7 A relationship between the area and the distance

Like above, if the object is half the size of the databases', the area of the object will be quarter of the DB. Using this law, the robot can calculate the ratio between the area and the distance. The equations are like below.

$$d_a : d_b = a_a : \sqrt{a_b} \quad (1)$$

$$d_b = \frac{d_a * \sqrt{s_b}}{s_a} \quad (2)$$

$d$  indicates the distance, and  $s$  indicates the size(or area). Small  $a$  and  $b$  indicate input image and database image respectively. Eq. (1) shows the relationship between a distance and an area. According to the relation, Eq. (2) shows how we get the approximate distance from the difference of area.

However, there are some errors about 3~4cm within work space because of the object shape and environment. Even if very small error occurs, the manipulator has many chances to fail to grasp objects. That is why we use ultra sonic sensors to compensate errors. The ultra sonic sensor that we used is a module type that constitutes both a transmitter and a receiver. Fig. 8 shows the ultra sonic sensor, SRF-04. (Robot Electronics Ltd.) Now the robot computes the interval by measuring the ultra sonic returning time. This sensor fusion reduces most failure ratio.



Fig.8 A sonar sensor module

We use SIFT transformation matrix to locate the position of the object in the scene. We can get the transformation matrix if there are three matching points at least [7]. The transformation matrix indicates the object's location and its orientation. Then the manipulator control system moves motors to locate the end effector at the center position of the object.

## VI. EXPERIMENTAL RESULTS

It is not possible to grasp the object only using the manipulator system. It is required that the integration of entire robot system. Before we explain the experimental results, we present the overall robot system and flowchart. Grasping strategy plays an important rule in system integration. We assumed some cases and started to integrate systems based on the scenario. In practical environment, there are many exceptional cases that we could not imagine. We thought that the main purpose of the system integration is to solve the problems that we face. Fig. 9 shows the flowchart of grasp processing that we assumed.

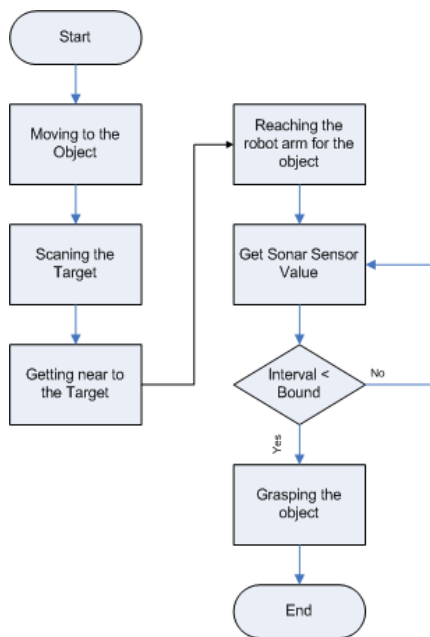


Fig.9 A flowchart of processing

First, the robot goes to the pre-defined position where the desired object is near by. Here, we assumed that the robot knows approximately the place where the object located. After moving, the robot searches the object by using its manipulator. Fig. 10 shows a searching state of the robot.



Fig.10 Searching for a desired object

The robot is finding the object in real time. If the robot finds the desired object, the robot moves itself to the location of the object in the workspace. That is why the scanning process is necessary as the web camera is able to search further range than the manipulation workspace. The moving part of the robot is using different computing resources, so we can process the main scheduler and the object recognition in parallel. Fig. 11 shows the movement of the robot when the object is outside of the workspace.

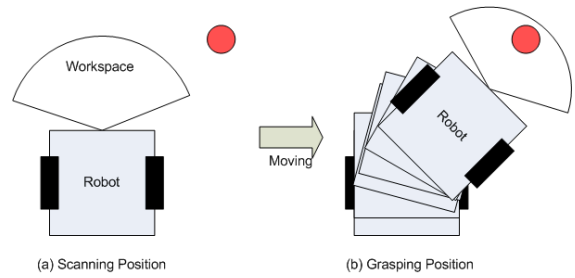


Fig.11 Movement of the robot after scanning

After that, the robot moves the manipulator, so that the object is at the center of the camera by solving inverse kinematics problem. In this time, the image data will be captured and continually used for the vision processing. If the object is in the workspace, the robot holds out its manipulator while the sonar sensor is checking whether the robot can grasp the object or not. If the robot decides that it is enough distance to grasp the object, the gripper would be closed to grasp. Using the processing described above, the robot can grasp the object.

We performed experiments of the store service robot ‘Part-Timer’ by sending it on an errand of bringing the object. Fig. 12 shows the store service robot ‘Part-Timer’.

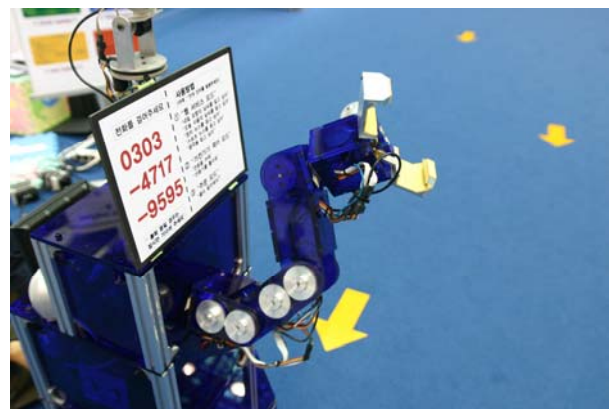


Fig. 12 The robot arm of the ‘Part-Timer’

‘Part-Timer’ has several intelligent modules; moving parts, pan-tilt camera, recognition system, home appliance controller, and the manipulator. The purpose of the overall system is to serve the customer in a store. If one customer says the desired object to the robot, the robot recognizes the speech, and it moves to the position to try to grasp the object. Then the robot brings the desired object to the customer. We optimized the gripper to grasp beverage cans, so we did the experiments using cylinder shape objects. The result of the experiment showed above 90% success ratio in 5 by 5 square meters area when we use the integrated robot system.

Fig. 13 presents a processing after the robot has found the desired object. First, the robot arm is in an initial state. If the robot receives a scanning command from the main scheduler, the object recognition system starts to work and the robot locates its manipulator to other position. If the robot finds the object, the manipulator will be reached out. The sonar sensor is used in this state. Reaching the robot manipulator, the sonar sensor is checking the distance from the object. Finally, the gripper is closed to grasp.

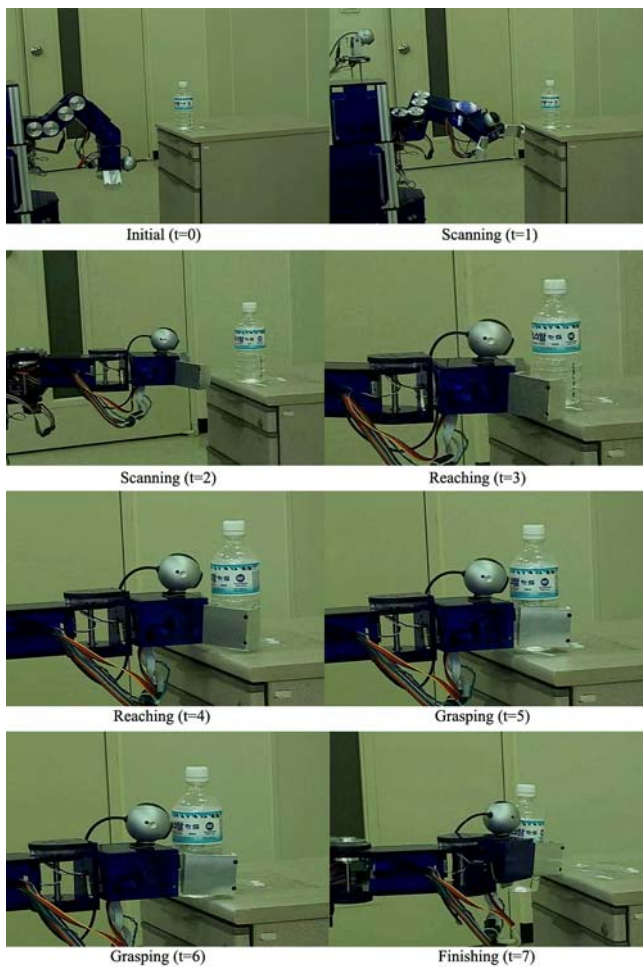


Fig. 13 Time progressing of a grasping

However, there are some unexpected errors during processing. Most of the errors occur due to the robot's environment like slips of the floor or obstacles in the robot path. If the slip errors are accumulated, the robot moves to the position where it cannot find the object.

Especially, if the object is too near to the gripper, then the robot cannot find the object using the camera. The ultra sonic sensor is needed in that case. After the gripper is closed, the robot does not know whether the object is in the gripper or not. A further study of our project is compensating for these errors. So we will use pressure sensors or some other sensors to receive feedback data of grasping objects.

## VII. CONCLUSION

We have shown several issues of vision-based grasp manipulation and explained the implementation of our robot. For the robotic manipulator to perform like human's arm, it is necessary to use vision information. So we put the camera on the gripper so that the robot can recognize the object nearside. Our experimental results show a high success ratio, although we did encounter some problems. We use only particular objects like cylindrical shape objects, but it is required to learn the grasp points of object to use it [5]. We also do not use the feedback data whether the object is

grasped or not. If the robot fails to grasp the object, the robot does not recognize that it has failed. Therefore, the robot changes its status to the next stage even through there is nothing in the gripper. In the future, we intend to consider more manifold states for the robot to cognize the current situation. And we will design two manipulators and fingers to improve the grasping performance.

## VIII. REFERENCES

- [1] Smith.C.E. and Papanikolopoulos.N.P "Vision-Guided Robotic Grasping: Issues and Experiments" *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol.4, no.22-28 , Apr. 1996, pp.3203-3208.
- [2] Sanz.P.J., Pohil.A.P., Inesta.J.M., "Real-time grasping of unknown objects based on computer vision " *Advanced Robotics, 1997. ICAR '97. Proceedings., 8th International Conference on*, Jul. 1997, pp.319-324
- [3] Stanley.K., Wu.Q.M.J., Gruver.W.A., "Implementation of vision-based planar grasp planning " *Systems, Man and Cybernetics, Part C, IEEE Transactions on*, Nov. 2000, pp.517-524
- [4] Morales.A., Asfour.T., Azad.P., Knoop.S. and Dillmann, R., "Integrated Grasp Planning and Visual Object Localization For a Humanoid Robot with Five-Fingered Hands" *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, Oct. 2006, pp.5663-5668
- [5] Ashutosh.S, Justin.D, Justin.K, Chioma.O, Andrew.Y, "Learning to Grasp Novel Objects using Vision" *International Symposium on Experimental Robotics (ISER), 2006.*
- [6] D.G.Lowe, "Object recognition from local scale-invariant features" *In Proc. of the 7th ICCV, 1999*, pp.1150-1157
- [7] D.G.Lowe, "Distinctive image features from scale invariant key-points" *IJCV, 2004*, pp.91-110
- [8] Ho Seok Ahn, In-Kyu Sa, Young-Min Baek, Jin Young Choi, "System Model Architecture of Intelligent Service Robot for Unmanned Store Management" *Annual Workshop of Korea Robotics Society, Jun.2006* (Korean Paper)
- [9] Ho Seok Ahn, In-Kyu Sa, Young-Min Baek, Jin Young Choi, "The implementation of Intelligent Service Robot for Unmanned Store Management in Ubiquitous" *The annual workshop of The institute of Electronics Engineers of Korea, Jun.2006* (Korean Paper)