Proceedings of the 2000 IEEE
International Conference on Robotics & Automation
San Francisco, CA • April 2000

Robust Localization for 3D Object Recognition Using Local EGI and 3D Template Matching with M-Estimators

Kentaro Kawamura[1)], Kiminori Hasegawa[2)], Yasuyuki Someya[1)], Yoichi Sato[2)] and Katsushi Ikeuchi[2)]
1) Kyushu Electric Power Co. , Inc. Fukuoka, 815-8520, Japan
2) University of Tokyo Tokyo, 106-8558, Japan

Abstract

A tele-operated system in a robot greatly reduces the demands on the human operator, although some human intervention is still required to perform such tasks as insulator recognition, positional adjustments of the robot, and guidance toward electric lines and insulators. In order to automate some of the robot's capabilities, we have developed a 3D object-localization method for the robot's positional adjustment. The method is designed to be insensitive to noise, outliers and occlusions while, at the same time, it has optimal run-time efficiency. The main contribution of our algorithm is the use of an objective function which is specified to reduce the effect of noise and outliers in the range image and a method for minimizing this function. The objective function is efficiently minimized by dynamically recomputing correspondences as the pose improves.
Our algorithm is general enough to be applied not only to our dual-armed mobile robots but also to other teleoperation robots. This algorithm should greatly reduce the burden of operators when applied.
This paper first describes our algorithm, and then presents a performance evaluation.

1 Introduction

Figure 1 shows a mobile robot for maintaining electronic distribution systems [1] developed by Kyushu Electric Power Company. The robot is quite successful in reducing a number of tasks that human operators are presently required to perform under poor working conditions; in addition, use of the robot reduces the number of operator hot-line accidents.
Because this type of robot is controlled in the teleoperation mode, it still requires human assistance to perform such tasks as insulator recognition, positional adjustments of the robot, and guidance toward electric lines and insulators.
In order to reduce the amount of human assistance required, we have begun an additional project to automate some of the above tasks.

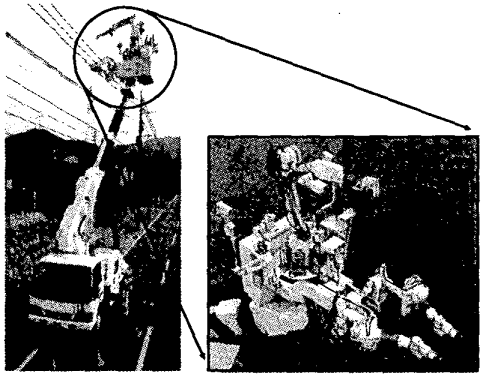

Figure 1: The mobile hot-line work robot

The flow of the robot work is classified into the 5 steps shown in Fig.2. With regard to step (1), currently a human operator drives the mobile robot from our base to the target electric pole and parks the robot near the pole. The robot moves almost horizontally and the approximate distance to the object is known. Therefore, the body of the electric pole will be recognized first. After that, going up along the pole, the arm and also the insulator will be recognized.
The required ability is to precisely determine the pose of the objects even under the following conditions; it should work under cluttered environments in which various objects, such as poles, arms, and tree leaves, appear in the same range image, it should work under various illumination conditions including sunlight, and it should be able to treat free form surfaces and unsymmetrical forms, because there are various types of objects in the electric distribution systems(Fig.2-(3)).
We have developed a method which has the above characteristics; the method consists of the following two points:

1. Segmentation of range images into small regions, and classification of these regions for estimating

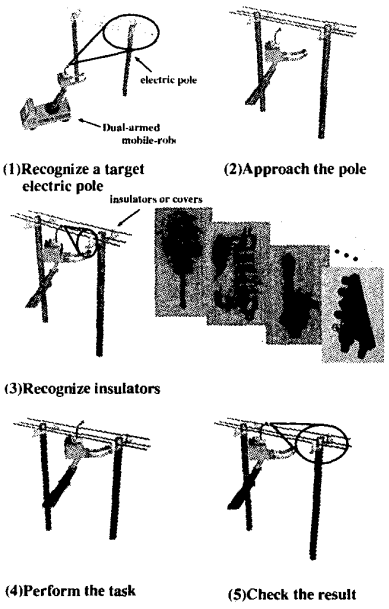

Figure 2: Flow chart of the robot work

the rough pose of the objects.

2. Localization by minimizing the distances between the 3D model and a range image, and confirmation of the class and the pose obtained by the previous segmentation results.

The class of the object as well as its position can be obtained by examining the various parameters (the area of the each region, the length of the edges, etc.) of the segmented regions. However, it is difficult to classify each region completely if various objects exist in the range image; the problem of the occlusion often occurs. There is a possibility that regions may affect the position adjustment of the objects. Therefore, we concomitantly use the localization method for the verification and the exact pose evaluation of the objects, which is based on the 3D template matching method, and the segmentation for checking the possibility that some objects exist in the range image.
Several segmentation algorithms have been proposed [9][10][11], and these may be categorized under two classes:

1. Region growing methods
2. Edge-based methods

Each method is divided into two groups:
those that use Quadric surfaces, and those that use the characteristics of the surface shape in a local area.

Here we integrate the regions and determine the class of each region by checking characteristics of the regions that are divided using region growing methods. First of all, we segment the range data into the small area on the basis of a planar fitting. After that, the extended Gauss image (EGI) is calculated to each local area. The class of each region is determined by checking the form of EGI. The use of EGI has the great advantage because EGI is able to express various types of surfaces[13][14].
As for the method of localization, we evaluate the pose of the objects by minimizing the errors between the points of the models and the range data.
Reviewing the existing systems, several localization algorithms have been proposed. Besl and McKay[2] presented the iterative closest-point(ICP) algorithm that iteratively computes nearest-neighbor correspondences between points on the model surface and points in the image data. Zhang[3] presented an improvement on the ICP algorithm to reduce the outlier sensitivity of ICP. He proposed a method for dynamic threshold selection to remove outliers. Haralick[4] et al. proposed and analyzed the use of M-estimators for robust pose estimation. Their algorithm used iteratively reweighted LS estimation. Lowe's algorithm[5] may be considered as a medium ground between pose estimation and pose refinement. But these methods seem to be sensitive to outliers and need a sufficiently large set of point correspondences. The random sample consensus (RANSAC) method of Fischler and Bolles[6] and the least-median-of-squares (LMedS) method of Mintz and Rosenfeld[7] are very useful for removing outliers.; unfortunately, these algorithms require an expensive combinatoric search.
We propose an algorithm which is robust and computationally inexpensive and also can recognize various objects. Our algorithm iteratively refines the pose of the object by optimizing an objective function defined over the range image data, model data and the object's pose. The main contribution of our algorithm is the use of an objective function which is specified to reduce the effect of noise and outliers in the range image and a method for minimizing this function. The objective function is efficiently minimized by dynamically recomputing correspondences as the pose improves.
In this paper, we present a method for recognizing 3D objects in 3D range images.
This paper is organized as follows. In Section 2, we describe how to obtain the initial pose of the objects. In Sections 3 to 5, we discuss how to estimate the pose of the objects. In Section 3, we explain how to efficiently set up the point visibility. in Section 4, we describe

a method for efficiently establishing point correspondences. In Section 5, we describe our pose optimization algorithm using M-estimators. In Section 6, we describe some experimental results for an insulator, in particular the effect on convergence of various M-estimators. We also describe the strategy of localizing objects in the target electric pole when applied. Finally, in Section 7, we offer our conclusions.

2 Model Pose Estimation

As we will show in the next section, it is possible to evaluate the fine pose of the object using our localization algorithm even if a large initial pose error exists. Although the correspondences in our localization algorithm are the nearest neighbors; it is preferable to project the model in the range image as closely as possible to the target object which corresponds to its model. A segmentation method of range data is used to solve this problem. In this method, we use the planar approximation for segmentation and classify the regions by utilizing the characteristic in each region. Most of the objects in the electric distribution systems are the symmetric and we use this useful characteristic for our purposes.

2.1 Segmentation using Planar Fitting

To divide the range data into several small regions, we used the method of region growing. Here we briefly explain the outline of the region growing method, which has been widely studied previously.

First of all, the kernel point at which to grow a region is selected. The kernel point is defined as the point(A) that is the closest to its local approximating plane, and this plane is obtained using its neighbor points($A_i, i = 1, ..., r$; r is the number of neighbor points. We used $r = 8$ here.). The normal of the plane is computed by building the inertia matrix of these points and taking the minimum eigenvectors. An error is evaluated as the distance between the point(A) and the local plane. The point which has the smallest error becomes the kernel.

The system expands the region(R_A) gradually by adding points(B_i) near the kernel. If the conditions below are satisfied, R_A integrates B_i.

$$||n_P \cdot n_{B_i}|| < Th_1 \quad (1)$$

where n_P is the normal of the plane(R_A), n_{B_i} is the normal of point B_i, and

$$d_{PB_i} < d_{max} \quad (2)$$

where d_{PB_i} is the distance between the plane(R_A) and B_i.

Both Th_1, d_{max} are thresholds. Whenever the region adds a new point B_i, the planar equation of regionR_A is renewed. If point B_i is included in this plane, the parameter of this plane is updated by using all the points inside the region R_A and point B_i. If there are no near points to add, the growth of this region(R_A) ends. This procedure is repeated until all points are included in any region.

2.2 Classification with Local EGI

Each region contains the points that are within the predefined distance from the plane approximated by the region growth method. This means that we can classify the regions in two cases: planes or other surfaces (cylinders, etc.). There are several methods for evaluating the errors between the plane and range data point or fitting quadric surfaces to each region to categorize regions[9][10][11][12]. However, these methods fail to classify the regions cases where unknown types of surfaces appear.

To avoid such a problem, we use the local EGI(LEGI) method[13][14]. It is possible to represent various types of surfaces by using many basis forms that are prepared beforehand. EGI can be obtained by translating unit normals of the points in the region to the same coordinate systems (Gauss Mapping) and weighting the amount of area that an original range data point has. The various surface expressions are possible if we use EGI rather than the primitive extraction method. Also, it is possible to obtain clear LEGIs for each region because we simply check the small regions (large enough to be classified). To express various forms more easily, we use stereographic projection of the LEGI; each point on the LEGI is projected onto the equatorial plane.

Using these images, we can classify each region. Based on the distribution of the images we can categorize them as follows.

1. Plane:all the points are on the origin.
2. Ellipsoid:the points are distributed in an ellipse state around the origin.
3. Cylinder:the points are distributed in a line around the origin.
4. Others:the images become arbitrary shapes.

The class of each region is checked based on such shapes of the points distribution.

2.3 Classification with Symmetric Character

We have studied two types of methods for categorizing these regions: the integration of regions by comparing the class of neighbors and the use of symmetric character. Performing these methods makes it possible to

categorize more exactly. In this paper we briefly explain the use of symmetric character. The procedure to use the symmetric is as follows.

1. compute the symmetric lines of the stereographic projected 2D image in each region.
2. the normal of the lines re-projected to the 3D image becomes the normal of the plane which will symmetrically divide the region in two.
3. compute the points in the range image which are close to this plane (in our experiments, this threshold was set to 1 [mm]).
4. compare the image on this plane with the tables. Tables are computed in the same manner for known objects in off-line.

We can estimate the rough pose of the model using this method. The next step is to evaluate the fine pose of the model. For this purpose, we have developed the template matching method which would be robust for large initial pose error, noise and occlusions.

3 Model Point Visibility

Before matching a surface point of a model with a point in a range image, it is prudent to first determine whether the model point is geometrically visible from the given viewing direction. In this section we present a local approximation method for predicting the visibility of points, given the pose of the object and the viewing direction.
Since the visibility computation will be performed many times as the pose improves, this computation must be as efficient as possible. For a computation of the visible portions of an object model, there are two standard algorithms from the field of computer graphics: ray-casting and z-buffering.
Since we use an appropriate model against the actual model, having a perfect visibility computation such as z-buffering is not only computationally expensive, but is also unnecessary. We now have to develop some other methods to select the visible points of the model efficiently. We can consider the problem in two cases: convex and concave surface visibility.
We here used triangulated models for localization. Mesh models are generally used to build models for 3d objects.

3.1 Convex Surface Visibility

We first discuss the simplest case, the convex case (e.g., an ellipse). In this case, the visibility of a point is computed by the following dot-product test:

$$visible_{convex}(c_i) = \begin{cases} true & n_{mi} \cdot v_i < 0 \\ false & otherwise \end{cases} \quad (3)$$

$i : 1, ..., h_m (h_m$:number of model patches)
where c_i denotes the ith triangle of model, n_{mi} denotes ith triangle's outward pointing normal and v_i is the viewing direction vector from the sensor center of projection to c_i.

3.2 Concave Surface Visibility

We have to compute point visibility for arbitrary shapes. Since the previous test (Equation (3)) for convex points is inexpensive, the test is first used to check whether the point is visible under the criterion described in the previous section. Once that has been determined, we can perform more expensive tests to determine whether the point is occluded by another part of the object.
We use lookup-tables (LUTs) to denote visible and invisible directions with respect to the local point coordinate (each point of the model has its own LUTs). The LUT is obtained by using the viewing hemisphere, which is tessellated into discrete bins. Each bin contains the binary value which indicates the visibility of each point of the model from viewing directions that map to the bin. The computation of the LUT for each point of the model is relatively expensive; we compute this off-line, a task which can be completed within a reasonable length of time and stored with the model. In on-line time, the system converts the current viewing direction to the local direction and consults the table for visibility of the model point from the current viewing direction.

4 Correspondence

Once the set of visible model points has been computed, we need to efficiently compute the correspondences between these model points and the points in the range image. Although it is difficult to find the correct correspondences without first knowing the pose of the object, we assume that the closest image point to a given model point is the correspondence.
The closest image point r_j to a given model point m_i, center of a mesh, can be defined as

$$r_j = \min_{r \in D} \| m_i - r \| \quad (4)$$

$i : 1, ..., h'_m (h'_m$:number of visible model patches)
where D is the set of three-dimensional data points in the range image.

5 Model Pose Optimization

The problem of pose optimization is how to deal with incorrect correspondences and noise. This closely resembles the classic pose estimation problem. We here discuss a solution to localization.

5.1 Pose Estimation

The pose estimation problem is to compute the pose which aligns the 3D model points m_i with their corresponding image points r_j. r_j is specified by the matrix-vector pair $< \mathbf{R}, t >$ where $\mathbf{R}$ is a 3×3 rotation matrix and t is a 3D translation vector. In general, the range points r_j will be contaminated by noise:

$$r_j = \mathbf{R}m_i + t + \beta \quad (5)$$

where β is a random 3D variable. Assuming that β follows a normal distribution, $P(\beta) \propto e^{-\frac{\beta^T\beta}{2\sigma^2}}$, then the optimal transformation is the least-squared error solution. The values$< \mathbf{R}, t >$ is obtained by minimizing the following equation,

$$f(\mathbf{R}, t) = \sum_i \| \mathbf{R}m_i + t - r_j \|^2 \quad (6)$$

The closed-form solution can be used to compute the model pose estimation. But if the errors in the observed data are not normally distributed or self occluded, LS estimation may be inappropriate. Since the closed form solution is no longer valid, an iterative approach is necessary to solve this problem, and it may be necessary to consider a different objective function which is the optimal estimator with respect to the error distribution of the data. Our localization relates to the pose estimation problem. Local minima are a problem, since we will have a number of incorrect correspondences to deal with. The appropriate objective function will clarify these problems.

5.2 Robust Pose Refinement

Suppose we are given a set of h'_m visible model points m_i and corresponding observed points r_j, and we want to compute the pose $< \mathbf{R}, t >$. The problem is that some of the h'_m correspondences will be incorrect; worse, we do not know which ones are incorrect. The errors for these incorrect correspondences do not fit a normal distribution. For a solution to this problem, we check the field of robust statistics.

We consider M-estimation, which is a generalization of least squares (we believe that the pure least-squares and thresholding methods are sensitive to outliers).

The general form of M-estimators is

$$E(z) = \sum_i \rho(z_i) \quad (7)$$

where $\rho(z)$ is an arbitrary function of the errors, z_i, in the observation. The equivalent probability distribution to $E(z)$ is $P(z) = e^{-E(z)}$. The M-estimate is the maximum-likelihood estimate of $P(z)$ and our choice of $\rho(z)$ determines $P(z)$. The errors, z_i, are a function of model pose p:$z_i(p)$. We use a unit 4-vector q (quaternion representation[8]) to denote the rotation instead of rotation matrix $\mathbf{R}$ because of its attractive mathematical properties. Now the 7-vector $p = [q^{\mathbf{T}}\, t^{\mathbf{T}}]$ denotes the set of pose parameter/rigid transformation.

Table 1: Table of weight functions for M-estimation

Function	$w(z)$
Huber	$w = \begin{cases} 1 & \lvert z\rvert \le \sigma \\ \frac{\sigma}{\lvert z\rvert} & \lvert z\rvert > \sigma \end{cases}$
Tukey	$w = \begin{cases} (1 - (\frac{z}{\sigma})^2)^2 & \lvert z\rvert \le \sigma \\ 0 & \lvert z\rvert > \sigma \end{cases}$
Lorentz	$w = (\frac{1}{1+\frac{1}{2}(\frac{z}{\sigma})^2})$
Gauss	1

We can find the parameters p that minimize $E(z)$ by taking the derivative of $E(z)$

$$\frac{\partial E}{\partial p} = \sum_i \frac{\partial \rho}{\partial z_i} \cdot \frac{\partial z_i}{\partial p} = 0 \quad (8)$$

By substituting $w(z) = \frac{1}{z}\frac{\partial \rho}{\partial z}$, we get

$$\frac{\partial E}{\partial p} = \sum_i w(z_i) z_i \frac{\partial z_i}{\partial p} \quad (9)$$

If we forget that $w(z)$ is a function of z,

$$\frac{\partial \rho}{\partial z} = wz \quad (10)$$

$$\rho = \frac{1}{2} w z^2 \quad (11)$$

And it is recognized as weighted least-squares(WLS). In this case, the term $w(z)$ measures the weight of the contribution of errors of magnitude z toward a WLS estimate.

There are many other possible choices of $\rho(z)$ to reduce the sensitivity to outliers on the estimation. Table 1 lists several possible functions that are used for M-estimation in this paper. Gaussian function (LS) is used for comparison.

Finally, we use a robust M-estimator to solve for p. We minimize

$$E(p) = \frac{1}{|V(p)|} \sum_{i \in V(p)} \rho(z_i(p)) \quad (12)$$

where $V(p)$ is the set of visible model points for the model pose parameters $p(|V(p)|$ is the number of visible model points). $z_i(p)$ is the distance(errors) between the ith pair of correspondences. To correctly minimize the desired function $E(p)$, we use the gradient-descent search.

6 Experimental Results

6.1 Experimental Procedure

We performed experiments using range images of an insulator in electric distribution systems. The insulator's height and width were $180\,mm$ and $110\,mm$,

respectively. To build the models for the insulator, we used a range sensor that consisted of a light-striping projector and a camera[15].
We took several range images of the insulator from various viewing directions, and converted those images to triangulated models. Using the Zipper program[16], those mesh models were aligned manually and merged into a uniform mesh model. Then each model was decimated from the original model using the Schroeder algorithm [17]. At each mesh, the visibility lookup table was constructed(section 3).
Figure 3 is a typical initial starting point for the convergence tests(50 mm transition here) and an overlay of the insulator model at its estimated actual location. The lines in this figure are the correspondences.
Each experiment consisted of 100 trials of our localization algorithm from a randomly generated initial pose estimate. We verified the results of each trial by checking the percentage within a distance (2 mm in our experiments) of visible model points that matched an image point, and determined the number of correct trials. To compare the performances of the various weight functions, we repeated each experiment using the Huber, Tukey and Lorentzian function. Gaussian function was used for comparison.

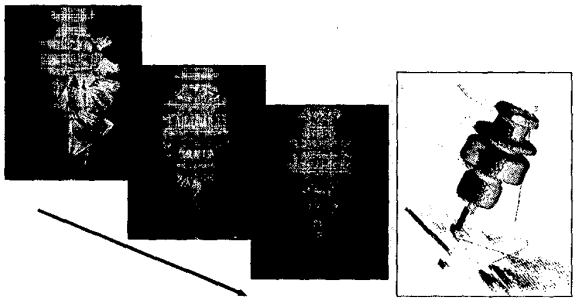

Figure 3: Left: A typical initial starting point (translation 50 mm) and an overlay of the insulator model at the estimated location. Right: An off-center view

6.2 Effects of M-estimators

Four versions of our localization algorithm were tested. Table 2. shows that we began with $\sigma = 4mm$ and reduced it to $\sigma = 2mm$ and then to $\sigma = 1mm$ as the algorithm progressed(σ appears in Table 1.).
The results of our experiment show that the Lorentzian and Huber weight functions perform better than the two other functions and have much better convergence properties. The Tukey function performs reasonably well if the initial error is less than $50\,mm$. In any case of Gaussian function, we cannot get better results than those obtained by using the Lorentzian and Huber functions.

Table 2: Results of Localization Performance

Initial pose Error(mm)	Correct (%)			
	Huber	Tukey	Lorentz	Gauss
25	100	99	100	99
50	86	87	93	76
75	81	55	76	61
100	66	39	59	49

6.3 Pose Estimation Using LEGI

In this experiment, we estimated the initial pose of the objects (the body, the arm and the insulator) using the segmentation and classification results. Figure 4 shows the results of classification based on the shapes of the Local EGI. In this experiment, we can easily categorize regions which belong to the arm and the pole. For the other type, the symmetric planes are computed in each region. In our case, most of the objects on electric poles which have a large amount of area in the segmented regions are symmetric. The black colored lines in Figure 5 show the intersections of the range data in the other-type regions and their symmetric planes. The images on the symmetric plane are checked to determine whether they have correspondences to the data-base images. In this experiment

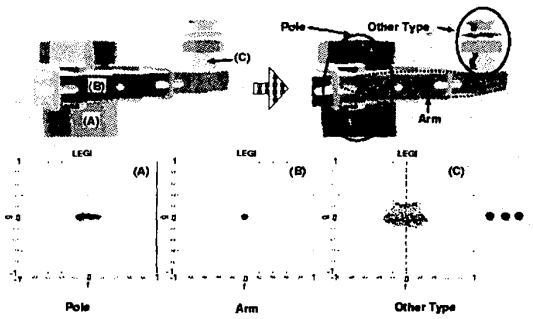

Figure 4: Segmentation Results: Planar fitting(up-left), and Classification(up-right). Stereographic projection of Local EGIs onto a 2D plane(bottom).

we were able to find the pole, the arm and the insulator. We were also able to estimate the pose of the objects using the position of each region.
Then the initial pose for each class was given to our localization algorithm to evaluate the precise pose. The

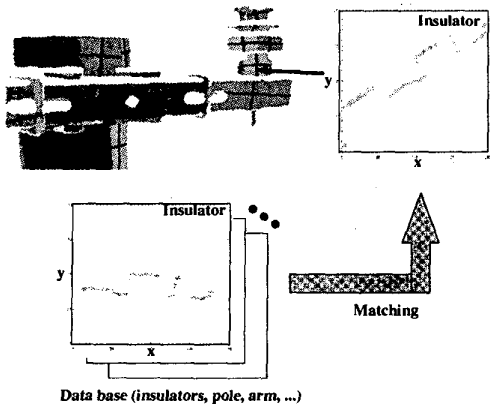

Figure 5: Symmetric Character of the insulator.

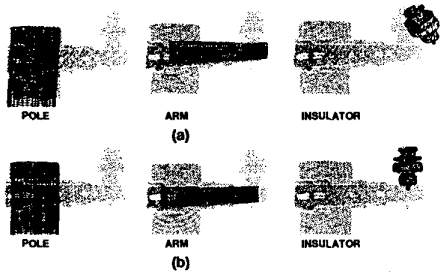

Figure 6: Localization Results: Initial pose(a) estimated by segmentation, and final pose(b) evaluated by our localization method.

models projected on the range data are shown in Figure 6. In this picture, we also show the final results, refined poses. We can see that the final poses of the models are well localized. The error between the pose estimated by the segmentation results and our algorithm is approximately $8-47\,mm$. Of course, we know the relative position of the objects on a pole (the distance between the body and the insulator, etc.), although it contains some translation and rotation errors. This is because a few centimeters in errors are allowed when workers attach arms to the pole, and the robot installed on the vehicle has comparatively low location precision. We believe that using segmentation results with local EGI for the initial pose estimation is effective for our purpose.

6.4 Progressive Strategy for Global Localization

Finally, we did outdoor experiments to test the practical use of this method using a PULSTEC Range Sensor, which can accurately obtain outdoor images.
A structure like the electric pole is composed of various parts. In other words, it is possible to compose the whole by recognizing each part. Therefore, we can reconstruct the whole parts by using the method of pose estimation[18].
Because this robot has been installed on a vehicle, its location precision is comparatively low. Thus, we followed a progressive strategy for global localization of the electric pole.
The range data that were obtained by this strategy and the results of localization in each step are shown in Figure 7 (up). The progressive strategy is very effective in the case that the surrounding environment of the objects is approximately known and the targets are very huge. The objects in the figure 7 (bottom) are other type of insulators and their covers. The covers are not symmetrical types of objects. Though the rough pose of this object can be estimated by using the position of the adjacent insulator, our localization algorithm can also refine the pose of the object.

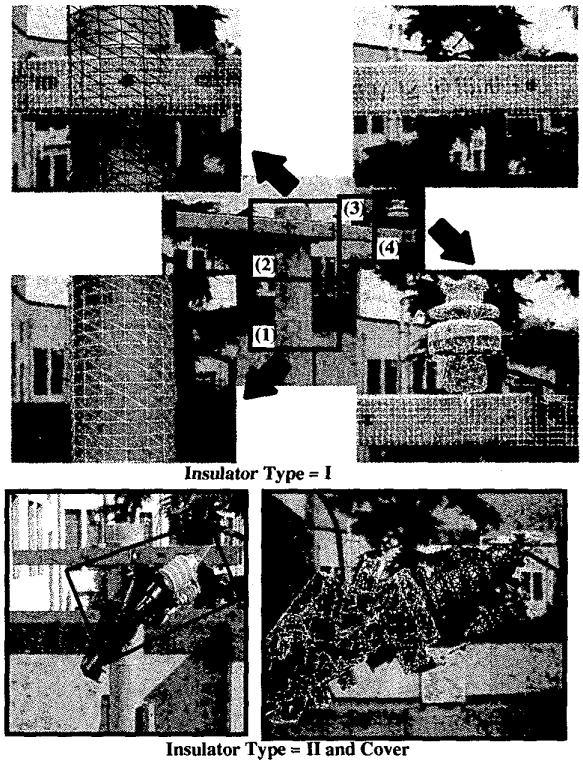

Figure 7: Progressive Strategy for Localizing Objects

7 Conclusion

We have developed a localization algorithm for a dual-armed mobile robot. The key components of our algorithm are: a point visibility table for efficient visible computation, a k-d tree-based point correspondence algorithm, and a robust pose estimator, an M-estimator, with dynamic correspondences. We have verified the effectiveness of our algorithm under various initialization errors. We have also examined the effect of M-estimators. We also discussed the use of the local EGI of a segmented region for classification, and the results of this usage are used for the initial pose in the localization.

We used the pole, the arm and the insulator for our experiment, and to verify the performance for practical use, we also conducted some experiments in outdoor environments. Experimental results demonstrate that using dynamic correspondences within a gradient-descent search of a robust objective function is a key component in achieving this capability. In particular, our algorithm achieves a wide degree of convergence for object localization in noisy range image data and large initial pose errors.

Our algorithm can be applied not only to our dual-armed mobile robots, but also to other robots. We believe that our algorithm should greatly reduce the efforts of operators in tele-operation by automatically aligning a robot with respect to an object. We plan to explore such applications of our algorithm in the future.

References

[1] M. Nakashima, H. Yakabe, M. Maruyama, K. Yano, K. Morita and H. Nakagaki, "Application of Semi-Automatic Robot Technology on Hot-Line Maintenance Work," *IEEE Conf. on Robotics and Automation*, 843-850, 1995.

[2] Paul J. Besl and Neil D. McKay, "A method for registration of 3-d shapes," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 14(2):239-256, 1992.

[3] Zhengyou Zhang, "Iterative point matching for registration of free-form curves and surfaces," *International Journal of Computer Vision*, 13(2):119-152, 1994.

[4] Robert M. Haralick, Hyonam Joo, Chung-Nan Lee, Xinhua Zhuang, Vinay G. Vaidya, and Man Bae Kim, "Pose estimation from corresponding point data," *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1426-1446, 1989.

[5] David G. Lowe, "Robust model-based motion tracking through the integration of search and estimation," *International Journal of Computer Vision*, 8(2):113-122, 1992.

[6] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, 24(6), 1981.

[7] Peter Meer, Doron Mintz, and Azriel Rosenfeld, "Robust regression methods for computer vision: a review," *International Journal of Computer Vision*, 6(1):59-70, 1991.

[8] O. D. Faugeras and Martial Hebert, "The representation, recognition, and locating of 3-d objects, " International Journal of Robotics Research, 5(3):27–52, 1986.

[9] M. Oshima and Y. Shirai, "Object Recognition Using Three-Dimensional Information, " IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. PAMI-5, no. 4, pp. 353–361, 1983.

[10] R. Hoffman and A. K. Jain, "Segmentation and Classification of Range Images, " IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. PAMI-9, no. 5, pp. 608–620, 1987.

[11] P. J. Besl and R. C. Jain: "Segmentation Through Variable-Order Surface Fitting, " IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 10, no. 2, pp. 167–192, 1988.

[12] P. J. Flynn and A. K. Jain, "BONSAI:3-D Object Recognition Using Constrained Search," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 13, no. 10, 1991.

[13] K. Ikeuchi, "Recognition of 3-D Objects Using the Extended Gaussian Image," Proc. of the Intern. Joint Conf. on Artificial Intelligence, pp. 595–600, 1981a.

[14] Horn, B.K.P., "Extended Gaussian Images," Proc. of the IEEE, vol. 72, no. 12, pp. 1671–1686, 1984b.

[15] K. Sato and S. Inokuchi, "Range-imaging system utilizing nematic liquid crystal mask," *Proceedings of 1st ICCV, London*, 657-661, 1987.

[16] Greg Turk and Marc Levoy, "Zippered polygon meshes from range images," *Proceedings of SIGGRAPH 94*, 311-318. ACM, 1994.

[17] William Schroeder, Jonathan Zarge, and William Lorensen, "Decimation of triangle meshes," *Proceedings of SIGGRAPH 92*, 65-70, 1992.

[18] Alex P. Pentland, "Perceptual Organization and the Representation of Natural Form," Journal of Artificial Intelligence, vol. 28, pp. 293–331, 1986.

[19] Mark D. Wheeler, "Sensor Modeling, Probabilistic Hypothesis Generation, and Robust Localization for Object Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 3, pp. 252–265, 1995.

[20] K. Kawamura, M. D. Wheeler, O. Yamashita, Y. Sato and K. Ikeuchi, "Localization of Insulators in Electric Distribution Systems by Using 3D Template Matching from Multiple Range Images," Proc. of IROS , vol. 3, pp. 1534–1540, 1998.