Proceedings of the Fifth IIEEJ International Workshop
on Image Electronics and Visual Computing 2017
Da Nang, Vietnam, February 28 - March 3, 2017

# AUTOMATIC, ACCURATE ESTIMATION OF THE POSITION AND POSE OF A LADDER IN 3D POINT CLOUD

Keishi Nishikawa†, Zhao Wang†, Jun Ohya†, Takashi Matsuzawa†, Kenji Hashimoto†, and Atsuo Takanishi†
†Waseda University, Dept. of Modern Mechanical Engineering

## ABSTRACT

This paper proposes a method for detecting a ladder, whose pose and position are unknown, in 3D point cloud data acquired by a laser range sensor. The proposed method consists of two steps: rough and fine estimations of the pose and position of the ladder. In the first step, Key-points, each of which concatenates three features based on 3D Vector Pair [1] and the Pair's positional information, are extracted from the 3D ladder model and scene, to which some pre-processes are applied. Rough estimation is obtained by applying the ICP (Iterative Closest Point) algorithm to the key-points of the scene and model. In the second step, the same processes as the first step are applied only to points in the volume determined based on the rough estimation so that fine estimation can be achieved. Experiments were conducted in a real scene including a ladder. It turns out that our method can detect the ladder at 2 meters away from the sensor with the mean error of less than 1cm.

## 1. INTRODUCTION

In recent years, there have been many serious disasters such as earthquakes, typhoons, and nuclear power plants accidents, but in many cases it is very dangerous or difficult for humans to rescue victims and suffers and to restore/recover artificial and/or natural things in these disaster sites. For these reasons, the actualization of robots that can do the above-mentioned dangerous and difficult tasks in disaster sites on behalf of humans is strongly desired. In response to this desire, the authors are researching on a disaster response robot [2].

Among all of our robot's functions, this paper deals with going up and down ladders of buildings or plants in disaster sites. In real disaster sites, our robot should be able to find ladders autonomously, but at present this function has not yet been achieved: our robot can go up or down a ladder whose position and specification are manually specified [2].

Chen et al. [3] developed a method for detecting a ladder in 3D point clouds obtained by a laser range sensor as a function of disaster recovery robots in DARPA Robotics Challenge. However, a problem of their method is that the pose and position of the ladder needs to be manually specified by the user in advance.

This paper aims at detecting a ladder in 3D point clouds automatically (without specifying the pose and position of the ladder). The method proposed by this paper consists of 2-step processes. The first and second steps conduct rough and fine estimations of the pose and position of the ladder in 3D point clouds acquired by a laser range sensor, respectively. The specific processes are detailed in Section 2.

## 2. PROPOSED METHOD

### 2.1. Basic idea and overview of the procedure

The processing flow of Chen et al.'s method [3] is described below. First, 3D point cloud data of a scene including a ladder is obtained with a range sensor. The 3D position of the ladder is manually specified by a user, and is inputted to the Pass Through Filter, which only passes the point cloud data within the specified range determined by the given 3D position of the ladder to the subsequent processes. After down-sampling the point cloud data outputted from the Pass Through Filter, planes corresponding to walls or floors are detected, and the point cloud data on the planes are removed. Iterative Closest Point (ICP)[4] algorithm is applied for registering the 3D ladder model to the point cloud data so that the ladder can be located in the scene's 3D point cloud. Note that Chen et al.'s method needs to specify the position and pose of the ladder in advance. This manual specification is a strong limitation and unpractical in real disaster sites.

In order to relax the problem of Chen et al.'s method, this paper proposes the method shown in Figure 1. The proposed method consists of two steps: rough and fine estimations of the position and pose of the ladder, respectively. In the first step, the point cloud data of a scene including a ladder is inputted to the pass through filter, where this process is not always needed and does not require manual specification of the ladder's pose and position as opposed to Chen et al.'s method: i.e. this process is utilized for the computation efficiency in the first step. Next, like Chen et al.'s method, the scene point cloud data is down-sampled, and the plane removal is performed. To remove point cloud data that could exist on large non-planar surfaces, the point cloud data are processed by non-planar cluster removal. After another noise removal processes, Key-points are extracted using the 3D Vector Pair[1] proposed by

Akitsuki et al. and 3D ladder model, where the Key-point is built by concatenating a feature vector obtained by the 3D Vector Pair formed by three points selected randomly from the remaining point cloud data and the coordinates of the centroid of the three points. Rough estimation of the pose and position of the ladder is obtained by applying the ICP algorithm to the key-points of the scene and model. In the second step, the rough estimation by the first step and the point cloud data of the scene are inputted to the Pass Through Filter. Then, the data outputted by the Pass Through Filter are processed by the same processes as the first step. As a result of applying ICP to the key-points of the scene and model, the final fine estimation of the pose and positon of the ladder is obtained.

In this paper, the 3D ladder model is constructed by a CAD (Computer Aided Design) software. To generate the point cloud data of the ladder model, the CAD data is converted to the point cloud data through free software " CloudCompare"[5].

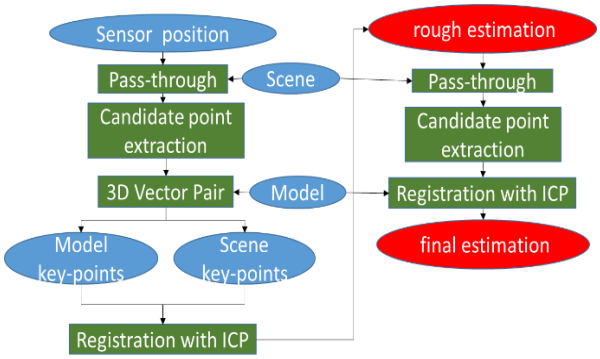


**Fig 1 Our Approach**

## 2.2. First step

### *2.2.1. Pass through filter*

Using a laser range sensor, point cloud data of the scene are acquired (Fig. 2 shows an example of the point could data) and inputted to the Pass Through Filter. This filter outputs the point cloud data within the range volume (half-cube) specified by the user. Figure 3 shows the result of applying this filter to the data in Fig. 2. As described in Section 2.1, this filter is not always needed, and is used only for computation efficiency in the first step, while this filter is significant for Step 2. In the first step of this paper, we specify so that the center of the range volume is centered at the center of the scene captured by the range sensor. The specific size of the range volume is 5[m] by 5[m] by 2.5[m], where only the length of the direction of the optical axis of the sensor is 2.5 [m].

**Fig. 2 Original scene's point cloud data**

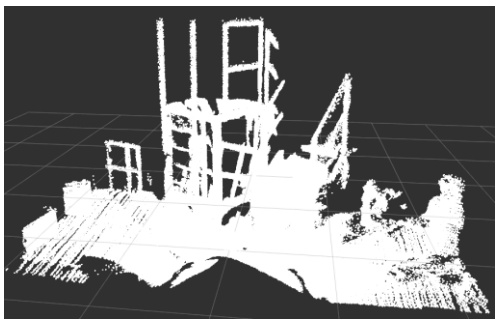

**Fig. 3 Point cloud data obtained by the Pass Through Filter**

### *2.2.2 Candidate point extraction*

In general, the number of points outputted from the Pass Through Filter is still very large: in other words, the outputted points include much various noise. These could cause computation inefficiency and inaccurate final estimation results. By reducing the number of points and suppressing the noise, candidate points that could correspond to the surface of the ladder should be extracted. Therefore, down-sampling (Fig. 4 shows the down sampling result for Fig. 3), removing points on large planes such as floors and walls (Fig. 5 shows the result for Fig. 4) and on large non-planar surfaces such as drums (Fig. 6 shows the result for Fig. 5), removing small planar patches that cannot be removed by the above-mentioned plane removal (Fig. 7 shows the result for Fig. 6) are performed for the points outputted from the Pass Through Filter. Consequently, candidate points, which are processed by Sec. 2.2.3, are obtained.

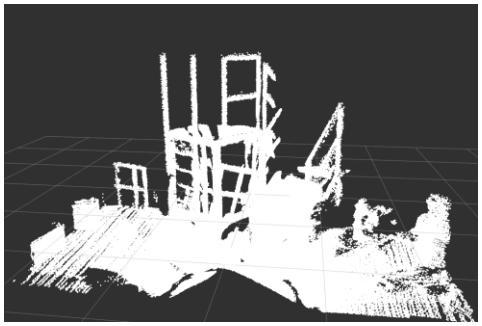

**Fig 4 down-sampled point cloud data**

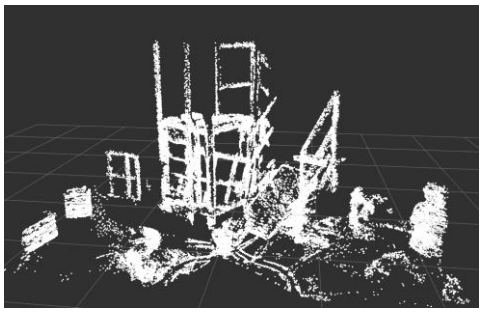

**Fig 5 point cloud data processed with plane removal**

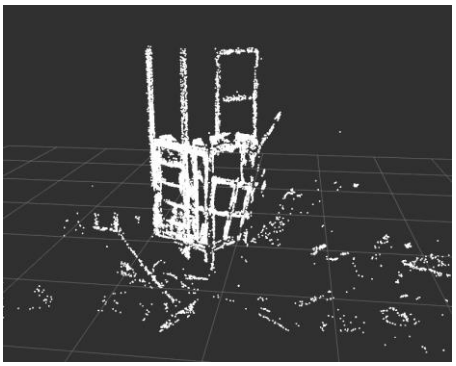

**Fig 6 point cloud data whose points on non-planar surfaces are removed**

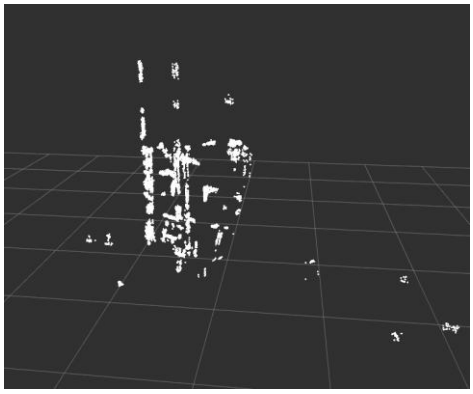

**Fig 7 point cloud data whose points on small patches are removed**

*2.2.3 Key-point extraction based on 3D Vector Pair*

As described in Sec. 2.1, Key-points are obtained from 3D Vector Pairs, which are obtained from randomly selected three candidate points (in the following, candidate points are simply called points). The Key-points are utilized for registering the ladder model and scene. More details are described in the following.

*A. Vector pair generation*

As Shown in Fig 8, three points P, Q1, and Q2 are randomly selected from the given point cloud data, and two vectors, $\boldsymbol{d}_{Q1}$, and $\boldsymbol{d}_{Q2}$ are formed, where their starting point is P, and ending point are Q1and Q2. Suppose that $l_1$ is the target distance for the distance between P and Q1, and similarly $l_2$ is for P and Q2. The values for $l_1$ and $l_2$ are constant and are manually given in advance. Let θ be the angle between the two vectors $\boldsymbol{d}_{Q1}$, and $\boldsymbol{d}_{Q2}$. If the 3D Vector Pair does not satisfy Eq. (1), the three points are discarded, and another three points are randomly selected.

$$\left|arccos\left(\boldsymbol{d}_{Q1}\cdot\boldsymbol{d}_{Q2}/\left\|\boldsymbol{d}_{Q1}\right\|\left\|\boldsymbol{d}_{Q2}\right\|\right)-\theta\right|\le\theta_{th}$$
$$\left|\left\|\boldsymbol{d}_{Q1}\right\|-l_1\right|\le l_{th},\left|\left\|\boldsymbol{d}_{Q2}\right\|-l_2\right|\le l_{th}\quad(1)$$

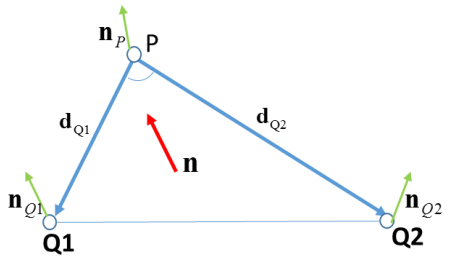


**Fig 8 Generation of 3D Vector pair**

In Eq. (1), $\theta_{th}$is the threshold of the angle, and $l_{th}$ is the threshold for the length difference. Three points are chosen from the given point cloud data to generate $\boldsymbol{d}_{Q1}$and $\boldsymbol{d}_{Q2}$that satisfy Eq.(1). Three features $s_P$, $s_{Q1}$, and $s_{Q2}$ are computed as follows. First, the normal to each of the three points is computed as the normal to the plane fitted to more than three neighboring points. The normals to the three points are denoted as $\boldsymbol{n}_P, \boldsymbol{n}_{Q1}, \boldsymbol{n}_{Q2}$. Also, each normal is a unit vector. In Eq. (2), $\boldsymbol{n}$ is the normal (unit vector) to the plane the two vectors span, as shown in Fig 8. Thus, the three features, $s_P$, $s_{Q1}$, and $s_{Q2}$, are computed by Eq. (2).

$$\boldsymbol{n}=\boldsymbol{d}_{Q1}\times\boldsymbol{d}_{Q2}/\left\|\boldsymbol{d}_{Q1}\right\|\cdot\left\|\boldsymbol{d}_{Q2}\right\|$$
$$s_P=\boldsymbol{n}\cdot\boldsymbol{n}_P,\ s_{Q1}=\boldsymbol{n}\cdot\boldsymbol{n}_{Q1},\ s_{Q2}=\boldsymbol{n}\cdot\boldsymbol{n}_{Q2}\quad(2)$$

Also, the 3D coordinates of the centroid of the three points are recorded. The above-mentioned three features and the centroid's 3D coordinates are concatenated in a vector, which is defined as the Key-point in this paper.

*B. Occurrence Probability histogram of 3D Vector Pairs*

From the key-point, a histogram for the three features $s_P$, $s_{Q1}$, and $s_{Q2}$ is built. Suppose that the number of 3D Vector Pairs that have $s_P$, $s_{Q1}$, and $s_{Q2}$ is represented by h($\boldsymbol{S}$), where $\boldsymbol{S}$ = ($s_P$, $s_{Q1}$, $s_{Q2}$). Then, the occurrence probability histogram is defined by Eq. (3).

$$\mathrm{Ph}(\boldsymbol{S})=\mathrm{h}(\boldsymbol{S})/\sum_{s_P=1}^{L}\sum_{s_{Q1}=1}^{L}\sum_{s_{Q2}=1}^{L}\mathrm{h}(\boldsymbol{S})\quad(3)$$

where $L$ is the number of bins of the histogram. In this paper, the bin width of the histogram is 0.20. If Ph($\boldsymbol{S}$) is

small, that means the occurrence probability of $\boldsymbol{S}$ is small: in other words, its uniqueness is large.

*C. Searching for corresponding points using observability*

To compute the observability $\mathrm{Obs}(A)$ of point $A$ [1] in the point cloud, unit vectors are randomly generated in the 3D space. As shown in Fig. 9, for each unit vector $\boldsymbol{o}_k$, the plane $\pi_1$ perpendicular to the unit vector can be considered. The point $A$ is projected to the plane $\pi_1$ in parallel to the vector $\boldsymbol{o}_k$. and the projected point is defined as $A_1$. Using some points in the neighborhood of point $A$, the normal $\boldsymbol{a}_n$ of A is obtained. Then, the observability $\mathrm{Obs}(A)$of point $A$ in the point cloud is given by

$$\mathrm{Obs}(A) = \sum_{k=1}^{K} \delta(A, \boldsymbol{o}_k)/K \qquad (4)$$

where $\delta$ is the function that returns 1 if a point $A$ is visible from $A_1$, where the visibility can be checked by computing the angle between $\boldsymbol{a}_n$ and $\boldsymbol{o}_k$; otherwise, $\delta$ returns 0. In Fig. 9, $A$ is visible from $A_1$, but not visible from $A_2$. In Eq. (4), K denotes the number of the normals randomly generated.

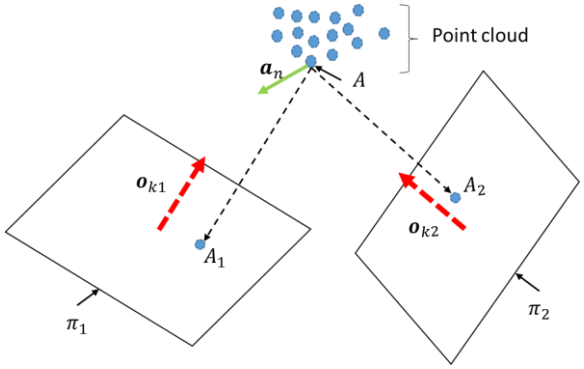


**Fig 9 Observability of the point A**

In this paper, $K$=500. This equation is applied to the 3D Vector Pair. If the Vector Pair is observable, the three points shown in Fig 8 are observable at the same time. Thus, Eq. (4) can be rewritten as follows.

$$\mathrm{Obs}(\boldsymbol{v}) = \sum_{k=1}^{K} \delta(P, \boldsymbol{o}_k)\delta(Q1, \boldsymbol{o}_k)\delta(Q2, \boldsymbol{o}_k)/K \qquad (5)$$

where $\boldsymbol{v}$ = ( P, Q1, and Q2).

*D. Selecting Vector Pairs considering occurence probability and observability*

A Vector Pair is selected using the occurrence probabilistic histogram $\mathrm{Ph}(\boldsymbol{S})$ and observability $\mathrm{Obs}(\boldsymbol{v})$. Specifically, $\mathrm{Ph}(\boldsymbol{S})$ and $\mathrm{Obs}(\boldsymbol{v})$ are linearly combined as

$$I(\boldsymbol{v}) = w_1 \mathrm{Obs}(\boldsymbol{v}) + w_2(1 - \mathrm{Ph}(\boldsymbol{S})) \qquad (6)$$

where $w_1$, and $w_2$ are the weights. If $I(\boldsymbol{v})$ in Eq. (6) is larger than $I_{th}$,the Vector Pair is selected; otherwise, discarded. This paper uses $I_{th}$=0.9, $w_1$=0.12, $w_2$=0.88. Suppose that $\boldsymbol{G}$ is the centroid of the selected Vector Pair. Key-point $\boldsymbol{K}$ is computed as follows.

$$\boldsymbol{K}=\boldsymbol{G}+c\boldsymbol{S} \qquad (7)$$

where $c$ is the coefficient that converts dimensionless $\boldsymbol{S}$ to a quantity whose unit is length. The parameter $c$ is needed to combine the two quantities with different units. As a result of experimental comparative study presented in Sec. 3.2, this paper uses $c$ = 1. Note that not only $\boldsymbol{S}$ but also $\boldsymbol{G}$ is used, because if only $\boldsymbol{S}$ is used, the registration accuracy described in Sec. 2.2.4 tends to get low according to preliminary studies. In this paper, 800 key-points are extracted from both the model data and the scene data.

*2.2.4 Rough estimation using ICP registration*

By applying ICP registration to the key-points extracted from model data and scene data, rough estimation of the position and pose of the ladder is performed. The ICP registration computes the following cost.

$$Cost = argmin_{\mathbf{R},\mathbf{t}} \sum_{m=0}^{M} \|\mathbf{R}\boldsymbol{p}_m + \mathbf{t} - \boldsymbol{q}\| \qquad (8)$$

where **R** is the rotation matrix, and **t** is the translation vector. In Eq. (8), $\boldsymbol{p}_m$ is the m-th point in the model key-points, M is the number of key-points, and $\boldsymbol{q}$ is the key-point in the scene and the nearest neighbor of $\boldsymbol{p}_m$. For each **R** and **t**, *Cost* in Eq. (8) is computed so that **R** and **t** that minimizes *Cost* is obtained.

### 2.3 Second step

The **R** and **t** obtained by Sec. 2.2.4 are rough estimation of the position and pose of the ladder. Based on the rough estimation, the range volume for the Pass Through Filter is determined. In this paper, the length of the volume's side in the direction of the optical axis of the sensor is 0.90[m], the length of the horizontal side orthogonal to the optical axis is 0.90[m], and the vertical length is 5[m]. Note that the volume of the second step is smaller than the first step, so that fine estimation is obtained by the second step.

The subsequent processes are same as the first step.

## 3. EXPERIMENTAL RESULTS AND DISCUSSION

### 3.1 Experimental environment

Experiments are conducted in the environment shown in Fig 10. For the experiments, we employed the packages belonging to Point Cloud Library[7] to process the point cloud data. To obtain point cloud data, the sensor MultiSense SL developed by Carnegie Robotics

Corporation was used. The PC used to process the data is Panasonics' Let's Note, CF-SX2, Intel Corei7.

As shown in Fig 11, two patterns for the geometry between the sensor and ladder are tested experimentally. In pattern 1, the ladder is placed at 2[m] away from the sensor in its optical axial direction, where in Fig 11, y-axis is the optical axis, x-axis is the perpendicular to y axis and parallel to the ground, and z-axis is perpendicular to x and y axes. In pattern 2, the ladder is placed at 2[m] away from the sensor in the direction of optical axis and rotated clock wise by 60[deg] about z-axis.

**Fig 10 Experimental environment**

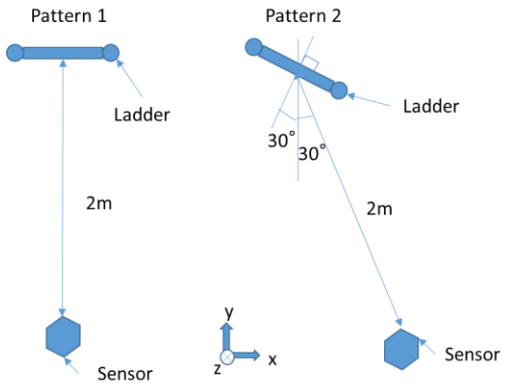


**Fig 11 Experimental requirements**

### 3.2 Results

Table1~3 show the mean error between the estimated results and ground truth of the position and pose of the ladder for the three values for the parameter *c*, which is explained in Sec. 2.2.3 D. The position error is described as the Euclidean distance between ground truth position of a ladder and estimated position.

Table 1 mean error between ground truth and estimation in $c$=0.5

| Pattern | error in x m | error in y m | position error m | direction error deg |
|---|---|---|---|---|
| 1 | 0.02639 | 0.04771 | 0.05452 | 1.936 |
| 2 | 0.1959 | 0.05163 | 0.2026 | 2.396 |

Table 2 mean error between ground truth and estimation in $c$=1.0

| Pattern | error in x m | error in y m | position error m | direction error deg |
|---|---|---|---|---|
| 1 | 0.006905 | 0.001800 | 0.007136 | 2.344 |
| 2 | 0.09370 | 0.01346 | 0.09466 | 1.186 |

Table 3 mean error between ground truth and estimation in $c$=1.5

| Pattern | error in x m | error in y m | position error m | direction error deg |
|---|---|---|---|---|
| 1 | 0.008007 | 0.04823 | 0.04889 | 0.2631 |
| 2 | 0.1594 | 0.02456 | 0.1613 | 123.8 |

The projected model data of the estimation results when $c$=1.0 is shown in Fig 12, because $c$=1.0 gives the smallest error. In Fig 12, the red colored data is the ladder model data, while the green colored data is the estimation result.

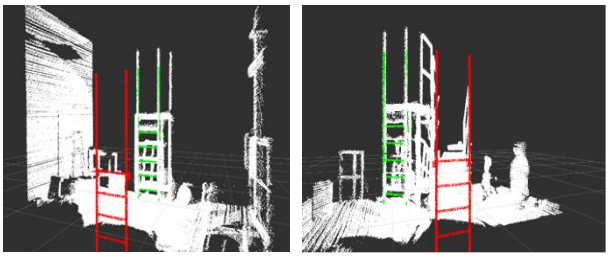

**Fig 12 Estimation results of the position and pose of a ladder**

In Fig. 12, the Left image is the estimated result of the pattern 1, and the right one is the estimated result of the pattern 2. The processing time for the whole processes is 966, 614[mesc].

### 3.3 Discussion

As shown in tables, the best result is aquired when the parameter $c$=1.0. Especially, as shown in Table 2, in case of pattern 1, the mean errors in x and y are less than 1[cm], while the mean error of the direction is less than 3 [deg]. This results is accurate enough for the robots to do tasks such as climing the ladder. In the pattern 2, the position error is less than 10[cm], which is larger than that in the pattern 1. The reason is that the ladder in the pattern 2 does not confront the sensor. Therefore, the observability of the ladder in the pattern 2 is smaller than that in the pattern 1. In our experiments, as described in 2.4, the weights of the uniquness and observability are constant in both patterns. In the future, it is necessary to develop a method that can adaptively change these weights.

## 4. CONCLUSION

This paper has proposed a method that can estimate the pose and position of a ladder in 3D point cloud. The method consists of rough and fine estimations. The

rough estimation is obtained by ICP registration for key-points extracted from the ladder model and scene. The fine estimation repeats the same processes as the rough estimation only for points in the limited volume determined by the rough estimation. As a result of experiments in a real scene including a ladder, the mean error of less than 1 cm is achieved.

## 5. REFERENCES

[1] Shuichi Akizuki, Manabu Hashimoto, "Position and Pose Recognition of Randomly Stacked Objects using Highly Observable 3D Vector Pairs", Proceedings of the 40th Annual Conference of the IEEE Industrial Electronics Society, pp.5266-5271 (2014).
[2] Takashi Matsuzawa, Kenji Hashimoto, Shinya Hamamoto, Tomotaka Teramachi,Xiao Sun, Kazuhito Uryu, Ayanori Koizumi and Atsuo Takanishi, “End-effector for Disaster Response Robot with Commonly Structured Limbs and Experiment in Climbing Vertical Ladder,” Proceedings of the 21st CISM IFToMM Symposium on Robot Design, Dynamics and Control, pp.311-319 (2016).
[3] Xiaopeng Chen, et al. ,”3D Model based Ladder Tracking Using Vision and Laser Point Cloud Data*”, in Proceeding of the 2015 IEEE Conference on Robotics and Biomimetics, Zhuhai, China, pp.1365-1370 (2015).
[4] P. J. Besl and N. D. McKay, “A Method for Registration of 3-D Shapes,” IEEE Pattern Analysis and Machine Intelligence, vol.14, pp.239–256 (1992).
[5] CloudCompare http://www.danielgm.net/cc/ (2016).
[6] M. A. Fischler, R. C. Bolles, “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,” Comm. of the ACM, Vol 24, pp 381-395 (1981).
[7] Point Cloud Library http://pointclouds.org/ (2016).