# Motion Tracking with Velocity Update and Distortion Correction from Planar Laser Scan Data

Seungpyo Hong[1,2], Heedong Ko[1] and Jinwook Kim[1]
Imaging Media Research Center, Korea Institute of Science and Technology[1]
Dept. of HCI and Robotics, University of Science and Technology, Korea[2]
{junon, ko, jwkim}@imrc.kist.re.kr

## Abstract

*A rigid transformation can be estimated by comparing two laser scan data that are captured at different time instances. Laser scan data consists of a number of distance values, and usually be considered to be measured simultaneously. However for a certain type of laser scanner, it is not true. Specifically each of the distance in a same data set is measured in a sequential manner. Due to this reason, constructing geometry of surroundings from the scanned data includes a distortion in case that the scanning device moves. In this paper, we suggest a methodology to compensate the distortion by estimating the velocity of the scanning device and transforming the scan data. We also present performance and accuracy by demonstrating simulation and real-world experiment results.*

## 1. Introduction

Tracking is a key issue in many research areas such as Virtual Reality, Augmented Reality and Robotics. 2D laser scan sensor is one of the most widely used sensors for the indoor position tracking. Scan matching problem in the robotics area was defined as estimating rigid transformation between two scan data to track the sensor's motion[1]. And one of the solutions is using ICP(Iterative Closest Point) algorithm which was originally used for geometric alignment of three-dimensional data from 3D scanner[2].

In this paper, we introduce an innovative method to track the position of 2D laser scanning sensor by applying ICP algorithm. Our contribution is to estimate the velocity of scan device and compensate the distortion originated from scanning time difference inside a set of scan data, so that more accurate and up-to-date position can be tracked as a result. Next section, we first summarize several previous research related to scan matching problem. Next, we introduce an original approach to estimating rigid transformation(Section 3), and present novel method with velocity update(Section 4). After that, the original and the novel methods are compared in the simulated environment(Section 5) and in the real environment(Section 6).

## 2. Related Works

The ICP algorithm has become the dominant method for aligning three dimensional models based on the geometry[3]. Rigid transformation between model and data is acquired by iteratively finding the closest points. ICP algorithm is also used for localization of robot by matching current scan data with the scan data gained previously[1].

Javier Minguez, *et al.*[4] suggested a new distance metric which is suitable for minimizing rotational and translational error concurrently. Their new metric distance contributes better convergence rate and more accurate correspondence matching. A. Diosi and L. Kleeman[5] present improved scan matching method named *Polar Scan Matching*(PSM). Their work is based on the truth that laser scan data does not use Cartesian coordinate system but polar coordinate system natively. PSM improves processing speed and ability to converge to a correct solution.

Previous researches [1][4][5] are limited to finding closest point pairs effectively with the belief that scan data reflects surroundings correctly only with the white noise, although it is not true. Scan data distortion problem was addressed by O. Bezet and V. Cherfaoui[6] in terms of time error correction. Their solution was interpolating two values which are scanned at different frame. They assume that at a same angle $\theta$, the sensor measures the distance to the same object. The drawback becomes great in a case that laser scanner rotates and moves fast so that their assumption becomes false.

## 3. Estimating Rigid Transformation using ICP

ICP starts with two scan data and an initial guess for their relative rigid transformation, and iteratively refines the transformation by repeatedly generating pairs of closest

ICAT 2008
Dec. 1-3, Yokohama, Japan
ISSN: 1345-1278

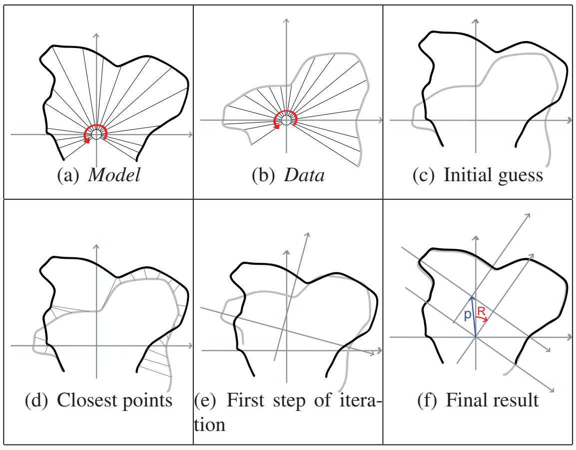



Figure 1. *Data*(b) iteratively converge to *Model*(a) through the processes in (c), (d) and (e). Finally, translation vector p and rotation matrix R are figured out with maximal correspondence(f).

points and minimizing an error metric[3] rather than finding corresponding points at once. ICP algorithm converges monotonically to the nearest local minimum of mean square distance metric[7]. If the initial guess of transformation is not close enough to ground truth, ICP algorithm will converge to local minimum which is not global minimum. In other word, if scanning device moves faster than a certain threshold so that an initial guess is not reasonable, ICP algorithm will fail to get right answer.

*Data* is scan data which is captured next to the *Model*. The objective of ICP is to find rigid transformation that transforms *Data* to be maximally overlaid to *Model*. Objective function is defined to minimize the equation (1)($x_i$ is a point in the *Data*, $y_i$ is a point in the *Model* and is a corresponding point to $x_i$). By solving the objective function, rotation matrix $R$ and translation vector $p$ is computed.

$$f(R,p) = \sum_{i=1}^{n} \|Rx_i + p - y_i\|^2 \tag{1}$$

$$T = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \tag{2}$$

Because ICP uses closest points instead of corresponding points, the result comes from above process might be not close to ground truth. By applying iterative manner, the result converges to ground truth[7] as described in Figure 1. Finally, transformation $T = T_n ... T_1$ is estimated through n times repetition.

$$Model \approx T_n T_{n-1} ... T_2 T_1 \cdot Data \tag{3}$$

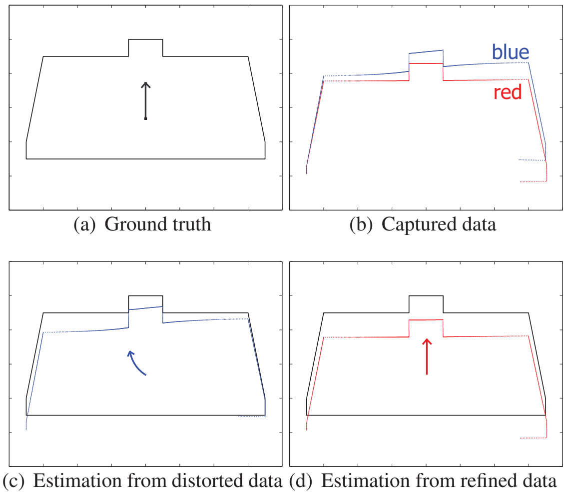



Figure 2. (a) is a given environment. Blue points in (b) shows distortion of scan data, and red points in (b) show compensated scan data. Transformation which is estimated from distorted data is shown in (c). Transformation which is estimated from compensated data is shown in (d).

## 4. Novel Method with Velocity Update

### 4.1. Distortion Appeared in Scan Data

Distance values in a same scan data are not scanned at the same time so that each distance value has significant time difference in its nature. Laser scan data might be distorted when scanning device moves during scanning. Figure 2 explains how scan data is distorted during scanning device moves. Blue points at Figure 2(b) is raw scan data from moving sensor. If we are trying to estimate the transformation without compensation, tracking is failed as shown in Figure 2(c). By inversely distorting the raw data, refined data(red points at Figure 2(b)) is obtained and motion is tracked well as shown in Figure 2(d).

### 4.2. Velocity Estimation and Data Compensation

$Data_i$ is scan data at time $t_i$. Time difference between $Data_i$ and $Data_{i-1}$ is $\Delta t$. There exist reference frame so that $Data_i$ has its own transformation to the reference frame, $T_i$ which is different from notation $T$ in previous section. Therefore, relation between $Data_i$ and $Data_{i-1}$ is represented in the equation (4) when two scan data are sorted along their correspondence. $V_i$ is a velocity based on the body coordinate frame of scanning device at time $i$.

$$Data_{i-1} = T_{i-1}^{-1} \cdot T_i \cdot Data_i \tag{4}$$

Velocity is assumed to be constant during a scanning time. First, $V_i$ is approximated from $T_{i-1}^{-1} T_i$ as shown in the equa-

ICAT 2008
Dec. 1-3, Yokohama, Japan
ISSN: 1345-1278

tion (5) which uses backward difference.

$$V_i = T_i^{-1} \cdot \dot{T}_i \approx \frac{1}{\Delta t} \log T_{i-1}^{-1} T_i \tag{5}$$

$Data_i$ consists of $n$ points, and time difference between adjacent points is $\Delta t_s$(=$\Delta t/n$). Each point belongs to $Data_i$ has its own transformation $T(t_i + j\Delta t_s)$ to reference frame is illustrated in the equation (6).

$$T(t_i + j\Delta t_s) = T_i \cdot e^{j\Delta t_s V_i} \tag{6}$$

Reflecting the equation (6) to the equation (4), $Data_i$ is converted into $Data_i^*$ which is compensated scan data with velocity as described in equation (7).

$$Data_i^* = \{e^{j\Delta t_s V_i} \cdot p_j \mid j = 0, ..., n\} \tag{7}$$

Through the equation (6) and (7), $Data_i$ is transformed to be at the moment when the first point of $Data_i$ is scanned. Tracked motion from the compensated data will always be delayed, because the last point is scanned latest. By changing the reference time($t_i$) of $Data_i$ as a moment when the last point is scanned, equation (6) is changed to

$$T(t_i - (n-j)\Delta t_s) = T_i \cdot e^{(n-j)\Delta t_s(-V_i)} \tag{8}$$

with negative velocity. Equation (7) is changed to

$$Data_i^* = \{e^{(n-j)\Delta t_s(-V_i)} \cdot p_j \mid j = 0, ..., n\} \tag{9}$$

As a result from the above processes, velocity($V_i$) and compensated data($Data_i^*$) are computed. However estimated $V_i$ is not close to ground truth, because $V_i$ is computed from distorted scan data. By iteratively update $V_i$ and $Data_i^*$, $V_i$ will converge.

### 4.3. Accelerating Convergence Speed

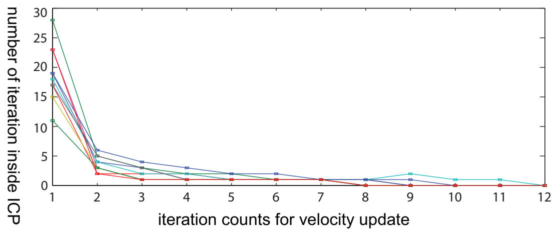
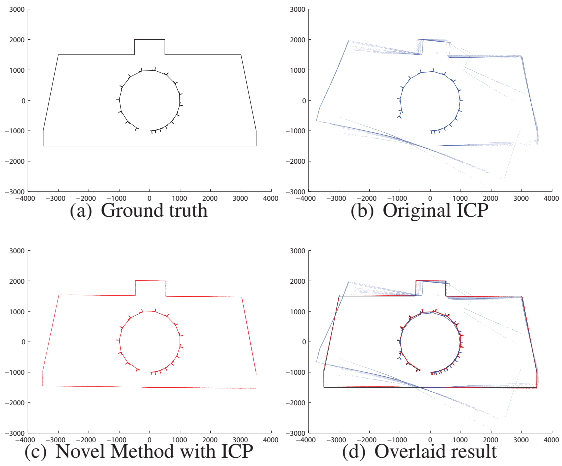



Figure 3. Number of iteration inside ICP decreases as the velocity converges. This data is sampled from the simulation at Figure 4(c).

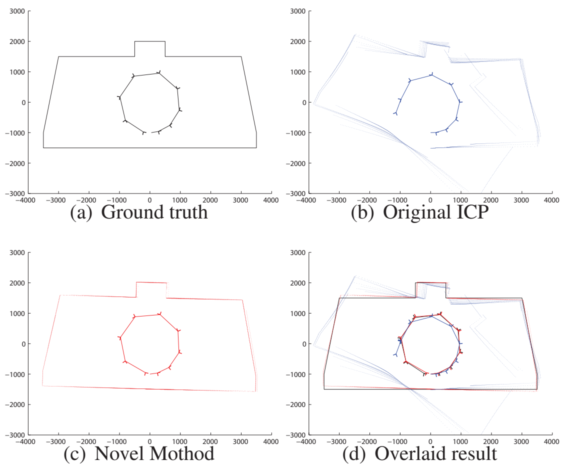
Compensation process requires nested loop outside ICP iteration. Fortunately, convergence inside ICP algorithm is accelerated dramatically by the reason of reusing previous result as an initial guess of current estimation, accordingly total complexity of novel algorithm increases less than twice of original one. Moreover, effective outlier rejection is possible during the iteration of velocity update. At every step of the iteration, the estimated rigid transformation from the previous step is used to be an initial guess of current step. By using the initial guess and sensor's effective range, the points which are included in $Data_i$ and expected to have no corresponding point in $Data_{i-1}$ can be rejected.

## 5. Simulation



(a) Ground truth (b) Original ICP (c) Novel Method with ICP (d) Overlaid result

Figure 4. Scanning device moves along the arc line, equation (10) and $\theta = 2\pi(1 - sinc(t/2))$.



(a) Ground truth (b) Original ICP (c) Novel Method (d) Overlaid result

Figure 5. Scanning device moves along the arc line, equation (10) and $\theta = 2\pi(1 - sinc(t))$.

The original and the novel algorithms are tested in the fully simulated environment. Scanning sensor and environment are implemented virtually, and sensor's motion is predefined to compare with estimated trajectories. HOKUYO

| | Original ICP | | Novel Method | |
|---|---|---|---|---|
| | R. Error | T. Error | R. Error | T. Error |
| Test1 | 58.14° | 2191mm | 7.28° | 177mm |
| Test2 | 79.98° | 2014mm | 17.06° | 65mm |
| Test3 | 16.80° | 1490mm | 6.88° | 408mm |
| Test4 | 54.59° | 2942mm | 3.28° | 210mm |

Table 1. Measured drift errors through the tests. (R. Error is rotational error, T. Error is translational error)

URG04-LX[8] scan sensor's functionality is implemented with 4 meter range, 240° scan angle, 0.36° angular resolution and 100Hz sampling rate(10 scan data/sec). We design a simple environment as shown in Figure 4(a). Sensor moves along the arced line with parametric equation described in equation (10). Sensor in second simulation(Figure 5) moves twice faster than first simulation(Figure 4).

$$T = \begin{bmatrix} sin(\theta) & cos(\theta) & 1000 * sin(\theta) \\ -cos(\theta) & sin(\theta) & -1000 * cos(\theta) \\ 0 & 0 & 1 \end{bmatrix} \quad (10)$$

By comparing Figure 4 and 5, the faster sensor moves, the more accurately novel algorithm tracks the motion than the original algorithm.

## 6. Experiment

Experiments were made in office-size real environment with real sensor. Algorithm and framework are implemented in c++ language, and tested in laptop with Intel dual core 2.0Gz CPU and HOKUYO URG04-LX sensor connected via USB interface. Experiments are accomplished by returning to start point and measuring the drift errors. Drift error is accumulated and becomes lager as time passes because motion is only tracked frame by frame. Sensor and laptop are carried by cart and driven by person along the predefined path.

Test1(Figure 6(a)) and Test2(Figure 6(b)) are tested by going around a room while Test3(Figure 6(d)) and Test4(Figure 6(d)) are tested by going straight, rotating and coming back. The novel algorithm shows more stable tracking than the original algorithm as listed in Table 1.

## 7. Conclusion

We suggest an innovative motion tracking method from planar laser scan data by updating the velocity and compensating the scan data. Our contribution is that the distortion which comes from scanning time difference during scanning is compensated by iteratively refining the velocity. From the results of simulation and experiment, the novel algorithm provides more accurate and delay-less position tracking, and is more tolerant for the faster motion than the original ICP algorithm.

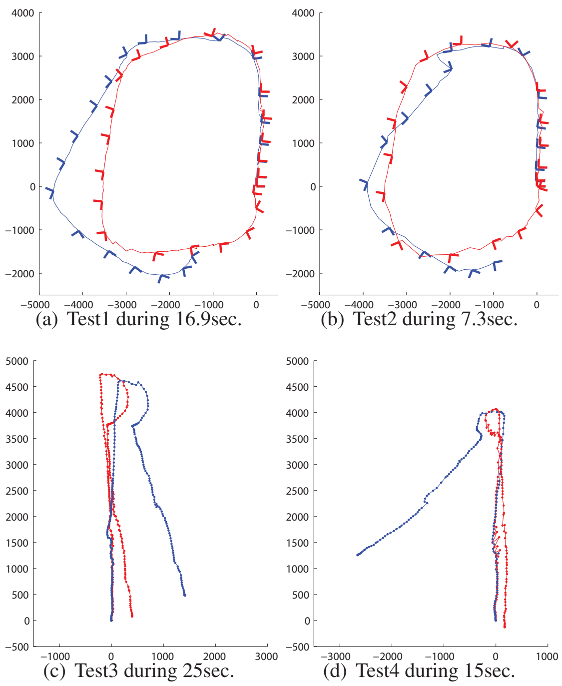



Figure 6. Experimental Results

## References

[1] F. Lu and E. Milios, "Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans," *Journal of Intelligent and Robotic Systems*, vol. 18, no. 3, pp. 249–275, 1997. 1

[2] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," *Robotics and Automation, IEEE International Conference on*, pp. 2724–2729, 1991. 1

[3] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in *Proc. Intl. Conf. on 3D Digital Imaging and Modeling*, 2001, pp. 145–152. 1, 2

[4] J. Minguez, L. Montesano, and F. Lamiraux, "Metric-based iterative closest point scan matching for sensor displacement estimation," *Robotics, IEEE Transactions on*, vol. 22, no. 5, pp. 1047–1054, Oct. 2006. 1

[5] A. Diosi and L. Kleeman, "Fast laser scan matching using polar coordinates," *Journal of Intelligent and Robotic Systems*, vol. 26, no. 10, pp. 1125–1153, 2007. 1

[6] O. Bezet and V. Cherfaoui, "Time error correction for laser range scanner data," in *Information Fusion, International Conference on*, July 2006, pp. 1–7. 1

[7] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. Pattern Anal.*, vol. 14, no. 2, pp. 239–256, 1992. 2

[8] H. Kawata, A. Ohya, S. Yuta, W. Santosh, and T. Mori, "Development of ultra-small lightweight optical range sensor system," in *Proc. Intelligent Robots and Systems, IEEE/RSJ International Conference on*, 2005, pp. 1078–1083. 4