

# Simplified EKF-SLAM by Combining Laser Range Sensor with Retro Reflective Markers for Use in Kindergarten

Takahiko Nakamura and Satoshi Suzuki  
Tokyo Denki University, Japan

**Abstract**—Recently in Japan, work contents imposed to childminders have increased due to several reasons such as variety of children's growth stage, requirement from parents, and administrative advices. To maintain good childcare service and to support kindergarten staffs, an objective monitoring system to watch over children is expected. In this paper, for the monitoring system, an automatic method to make an indoor map using the SLAM technique is discussed. To solve problem specific to the kindergarten use, using special landmark (LM) markers consisting of retro reflective material, an approach of a simplified EKF-SLAM which is effective by short-distance survey is proposed. Thorough simulation, the SLAM algorithm with an extraction method of the special LM from the laser range finder (LRF) sensor data was verified, and it was confirmed that the estimation error was as small as 0.57 % against total length of robot locomotion. Same algorithm was implemented to an actual robot that was mounted with the LRF sensor, and practical effectiveness was validated. As a result, the ability of the presented SLAM with a margin of estimation error of 5.3 % was also confirmed in the experiment.

**Keywords**—SLAM, EKF, laser range finder, retro reflective material.

## I. INTRODUCTION

RECENTLY, number of parents who work while entrusting their child to the day nursery is increasing by several reasons in Japan. So-called an issue of the children on the waiting list for a vacancy at the day nursery becomes an everyday affair. Moreover, through various revisions of administrative advices against the day nursery, the work of childminders in the nursery has been increasing [1]. In order to resolve this situation, utilization of objective system for assisting the work of the nursery and monitoring of children is one of solution. For instance, activity recognition of children using acceleration sensor and data-mining classifiers [2] was presented to analyze children's activity and history of their growth. In [3], a child monitoring system using the Kinect sensor to estimate mental status of the children from the

measurement was proposed. Although these techniques are useful to know some properties of children, they are not sufficient since children's nature of utilization of livelihood space is not taken into account. That is, in order to adequately estimate the growth and activity of children, it is necessary to know an indoor environmental map of the day nursery or kindergarten, since social meanings of areas in a house (for instance a play zone, corridor, and eating area) are significant for it. It is, however, difficult to obtain an environmental map of the day nursery because the room layout is changed frequently depending on the purpose of use, the number of children, and their age distribution. Therefore, automatic measurement of the indoor environment is desirable, and the SLAM (Simultaneous Localization and Mapping) by a mobile robot is an adequate approach.

Traditional SLAM methods utilize characteristic shape such as the corner of wall as feature points to identify environmental locations, and they usually utilizes an image processing and high speed computation, and large memory tend to be required. To overcome such drawbacks by simplification of computation, for instance, an orthogonal SLAM that limits orthogonal elements in the room is known [4]. Such technique, however, might not be suitable for modern kindergarten that is constructed with human-friendly and curved shape wall or furniture. Moreover, a swarm of children around the robot prevents the robot from correct measurement, and so-called kidnapped robot problem frequently occurs if the robot is not so heavy for children. Traditional methods managed to recover a certain amount of accuracy by repeating the SLAM survey motion. For this reason, almost all SLAM techniques assume implicitly that environmental elements of house and furniture are static [5]. Various dynamic SLAM techniques are presented; however, they also assume that architectural element are static at least [6]. In case of application in a kindergarten, it is feared that dynamic SLAM may not work well because there are many semi-statistical obstacles such as furniture. Therefore in this study, an alternative SLAM is presented by utilizing adroit LM sheets which are attached to the wall. The proposed SLAM algorithm is evaluated by a simulation, and the practical effectiveness was confirmed by experiment with actual mobile robot.

## II. METHOD OF ESTIMATING POSITION OF LM

### A. Virtual comb-shaped marker

When the LRF measures distance to a retro reflective material, a correct distance cannot be measured due to an effect of virtual reflection from the material. Therefore, data measured by the LRF shows a virtual comb-shape as shown in **Figure 1(b)**, if the LRF scans a LM sheet which consists of blank sheets and reflective sheets alternately. Our method utilizes this adroit LM to facilitate the detection of LM positions for the SLAM algorithm.

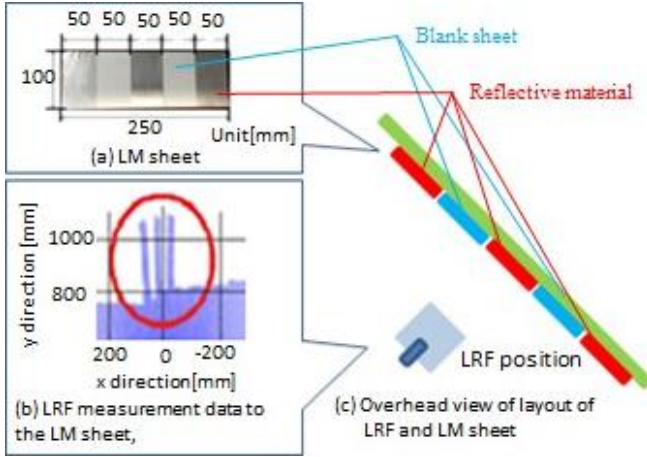


Figure 1 Landmark sheet measurement data by LRF

### B. Computation of LM's position using Hough transformation

In our previous work, a basic algorithm to derive a position of LM was presented [7]; however, the detection accuracy was insufficient due to error in extraction of LM sheets. Therefore, in this paper, a new method to improve the extraction is proposed by utilizing Hough transformation since it is known that a line detection method using Hough transformation has high tolerance ability against measurement noise.

First, a linear part detected by LRF against a flat wall surface (the linear part is called wall line hereafter) is searched from the measurement data by utilizing Hough transformation. Second, the virtual comb-shape is found by checking an offset difference from the detected wall line. Finally, a position of LM is computed as a center of the comb-shape. Details of the procedure to obtain the LM position are explained in the following sections.

In the following experiment, URG-04-LX-UG01 (HOKUYOU DENKI corp., JAPAN) is used and the measurable range and resolution of the sensor are plus-minus 120 degree and 0.36 degree, respectively. Hence, the measurement data of the LRF sensor are described as  $(l_i, \theta_i)$   $i = 1 \dots 686$ , where  $i$ ,  $l_i$ , and  $\theta_i$  are an angle index, a measurement distance, and the corresponding laser irradiation angle, respectively. The raw LRF data are converted into two-dimensional local coordinate value as  ${}^r(x_i, y_i)$  by  ${}^r x_i = l_i \cos \theta_i$ ,  ${}^r y_i = l_i \sin \theta_i$ .

### C. Detection of the LM maker sheet

After the wall lines are detected by Hough transformation, the comb-shape markers are found by checking offset between each data point measured by LRF and each wall line which was detected. The detection algorithm is explained below.

#### Step 1.

Values of slope  $a$  and an intercept  $b$  which correspond to candidates of the wall lines are computed after applying an orthodox Hough transformation. Distance  $d_i$  between each local coordinate value computed from raw LRF data and the segment of each wall line is computed by

$$d_i = \frac{{}^r x_i a + {}^r y_i b}{\sqrt{a^2 + b^2}} \quad (1)$$

#### Step 2.

After boundary points between the reflective sheet and the blank sheet are found, the LM position is determined by detecting three boundary points that are caused by the virtual comb-shape. The boundary point is found as a point where a sequence of the distance  $d_i$  changes rapidly because the reflective material gives virtual longer measurement data than actual distance. To tell the boundary point from others, perpendicular distances from the wall line between adjoined pairs at indexes  $i-1$  and  $i$  are compared, and the discrimination is performed by comparing the predetermined threshold  $\tau_1$ . That is, all angle indexes  $i$  which satisfy  $d_{i-1} - d_i > \tau_1$  are stored in a mutable array which is denoted by  ${}^* \alpha_1$ . Similarly, other indexes  $i'$  which satisfy  $d_{i-1} - d_{i'} < \tau_2$  are stored into other mutable array  ${}^* \alpha_2$ . Assuming that new integer  $\beta$  is an index to refer a component of array  ${}^* \alpha_1$  and  ${}^* \alpha_2$ , all index numbers that are registered in those mutable arrays and satisfy  ${}^* \alpha_{1[\beta]} - {}^* \alpha_{2[\beta]} > \tau_3$  simultaneously, are searched for all  $\beta$ , where  $\tau_3$  is a threshold and notation  $\alpha_{*[\beta]}$  means  $\beta^{\text{th}}$  element in the array  $\alpha_*$ . Using such  $\beta$ , value of  $\frac{{}^* \alpha_{1[\beta]} + {}^* \alpha_{2[\beta]}}{2}$  is adopted as an index corresponding to the position of reflective material parts. Such all indexes indicating reflective material parts are stored in other mutable array  ${}^* g$ .

#### Step 3.

Indexes showing the LM marker's position are extracted by checking indexes numbers registered in  ${}^* g$ . When  ${}^* g_{[\beta']} - {}^* g_{[\beta'-1]} > \tau_4$  and  ${}^* g_{[\beta'+1]} - {}^* g_{[\beta']} > \tau_4$  hold simultaneously for  $\beta'$  and such  $\beta'$  are found three time continuously, the  $\beta'$  is determined as an index of the LM marker's position, where  $\tau_4$  is a threshold. Finally, the LM marker's coordinate values  $(m_x, m_y)$  are computed as follows.

$$m_x = l_i \cos \theta_i \quad (2)$$

$$m_y = l_i \sin \theta_i, \quad (3)$$

$$i = {}^* g_{[\beta']} \quad \text{for all } \beta'$$

## III. SLAM ALGORITHM

A two wheeled differential drive mobile robot is assumed to be used for the following SLAM algorithm. The state of the

robot is described as  $p_t := [x, y, \theta]_t^T \in R^3$ , where  $x$ , and  $y$  are coordinate values of a center of the robot,  $\theta$  is the robot's orientation, and an integer subscript  $t$  represents the sampling time index and the same notation is used below. An input vector  $u$  is defined as  $u := [v, \omega]^T$ , where  $u$  is a translational velocity, and  $\omega$  is an angular velocity. Then the kinematics of the robot can be described as follows by considering two cases of  $\omega \neq 0$  and  $\omega = 0$ :

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}_{t+1} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}_t + \begin{cases} \begin{bmatrix} -\frac{v}{\omega} \sin\theta + \frac{v}{\omega} \sin(\theta + \omega \cdot \Delta) \\ \frac{v}{\omega} \cos\theta + \frac{v}{\omega} \cos(\theta + \omega \cdot \Delta) \\ \omega \cdot \Delta \end{bmatrix}, & \omega \neq 0 \\ \begin{bmatrix} v \cos\theta \cdot \Delta \\ v \sin\theta \cdot \Delta \\ 0 \end{bmatrix}, & \omega = 0 \end{cases} \quad (4)$$

where  $\Delta$  is a constant sampling interval. For latter discussion, the following expression is used instead of (4).

$$p_{t+1} = p_t + g(\theta_t, u_t) \quad (4)$$

where function  $g$  expresses the second term of RHS in (4). Moreover, Jacobians of  $g$  concerning  $p$  and  $u$  for  $\omega \neq 0$  and  $\omega = 0$  are obtained as follows, respectively.

$$\frac{\partial g(\theta_t, u_t)}{\partial p_t} = \begin{cases} \begin{bmatrix} 0 & 0 & -\frac{v}{\omega} c_\theta + \frac{v}{\omega} c_\gamma \\ 0 & 0 & -\frac{v}{\omega} s_\theta + \frac{v}{\omega} s_\gamma \\ 0 & 0 & 0 \end{bmatrix}, & (\omega \neq 0) \\ \begin{bmatrix} 0 & 0 & -v s_\theta \\ 0 & 0 & v c_\theta \\ 0 & 0 & 0 \end{bmatrix}, & (\omega = 0) \end{cases} \quad (5)$$

$=: G_t$

$$\frac{\partial g(\theta_t, u_t)}{\partial u_t} = \begin{cases} \begin{bmatrix} \frac{(-s_\theta + s_\gamma)}{\omega} & \frac{v_t \{s_\theta - s_\gamma\}}{\omega^2} + \frac{v_t \cdot \Delta c_\gamma}{\omega} \\ \frac{(-c_\theta + c_\gamma)}{\omega} & \frac{v_t \{c_\theta - c_\gamma\}}{\omega^2} + \frac{v_t \cdot \Delta s_\gamma}{\omega} \\ 0 & \Delta \end{bmatrix}, & (\omega \neq 0) \\ \begin{bmatrix} c_\theta \cdot \Delta & 0 \\ s_\theta \cdot \Delta & 0 \\ 0 & 0 \end{bmatrix}, & (\omega = 0) \end{cases} \quad (6)$$

$=: V_t$

where  $s_\theta := \sin\theta$ ,  $c_\theta := \cos\theta$ ,  $s_\gamma := \sin(\theta + \omega \cdot \Delta)$ , and  $c_\gamma := \cos(\theta + \omega \cdot \Delta)$ .

#### A. Derivation of sensor model

In this section, a sensor model of LRF is derived for a use of the measurement update in the SLAM algorithm. The model is computed under the assumption that true values of geometrical information are known. That is, distance  $r$  and relative angle  $\phi$  between the LRF sensor and a certain LM are assumed to be measured. The details are as follows.

First, true values of each robot's variable are assumed to be given, and they are denoted as  $[\tilde{x}, \tilde{y}, \tilde{\theta}]^T$ . Moreover, true positions of all LMs are also known and the  $x$  and  $y$  coordinate

values are described as  ${}^{(j)}\tilde{m}_x$  and  ${}^{(j)}\tilde{m}_y$  for  $j^{\text{th}}$  LM. Then the LRF sensor model function, say  ${}^{(j)}h(p, m)$ , that calculates the distance  ${}^{(j)}\tilde{r}$  and relative angle  ${}^{(j)}\tilde{\phi}$  between the robot and the  $j^{\text{th}}$  LM can be expressed as

$${}^{(j)}h(p, m) := \begin{bmatrix} {}^{(j)}\tilde{r} \\ {}^{(j)}\tilde{\phi} \end{bmatrix} = \begin{bmatrix} \sqrt{{}^{(j)}\tilde{m}_x - \tilde{x}}^2 + {}^{(j)}\tilde{m}_y - \tilde{y}}^2 \\ \text{atan2}({}^{(j)}\tilde{m}_y - \tilde{y}, {}^{(j)}\tilde{m}_x - \tilde{x}) - \tilde{\theta} \end{bmatrix} \quad (7)$$

Another Jacobian  ${}^{(j)}H$  of  ${}^{(j)}h(p, m)$  concerning the state  $p$  is computed as follows.

$${}^{(j)}H := \frac{\partial {}^{(j)}h(p, m)}{\partial p} = \begin{bmatrix} -\frac{{}^{(j)}\tilde{m}_x - \tilde{x}}{\sqrt{q}} & -\frac{{}^{(j)}\tilde{m}_y - \tilde{y}}{\sqrt{q}} & 0 \\ \frac{{}^{(j)}\tilde{m}_y - \tilde{y}}{q} & -\frac{{}^{(j)}\tilde{m}_x - \tilde{x}}{\sqrt{q}} & -1 \end{bmatrix} \quad (8)$$

where  $q = \delta_x^2 + \delta_y^2$ , and  $\delta_x := {}^{(j)}\tilde{m}_x - \tilde{x}$ ,  $\delta_y := {}^{(j)}\tilde{m}_y - \tilde{y}$ .

#### B. Estimation of positions using EKF

In order to perform the SLAM computation, an extended Kalman filter (EKF) procedure is applied to estimate status of robot variable and the LM's positions. To utilize the procedure, the state of the map, that are  ${}^{(j)}\tilde{m}_x$  and  ${}^{(j)}\tilde{m}_y$ , and the robot status  $p$  are combined into the following augmented state vector  $y$  as

$$y := [x, y, \theta, {}^{(1)}\tilde{m}_x, {}^{(1)}\tilde{m}_y, \dots, {}^{(N)}\tilde{m}_x, {}^{(N)}\tilde{m}_y]^T \in R^{(2N+3) \times 1}$$

where  $N$  is the number of all LMs and  $N$  is assumed to be already known. Then, the augmented state vector  $y$  at time  $t + 1$  is estimated using  $g(\theta_t, u_t)$  with status at time  $t$  as follows.

$$y_{t+1} = y_t + F^T \cdot g(\theta_t, u_t) \quad (10)$$

where  $F := [I_{3 \times 3} \quad 0_{3 \times 2N}]$  is a coefficient matrix to extract components of the robot status  $p$  from  $y$ . Then a Jacobian of (10) concerning  $y$ , that is described as  $G_t$ , is computed as follows using  $G_t$  which was described by (5).

$$G_t = I_{2N+3} + F^T G_t F \in R^{(2N+3) \times (2N+3)} \quad (11)$$

EKF-SLAM is an algorithm to express a belief  $\text{bel}(y_t)$  by using an average of the estimation for  $y_t$  and an estimated covariance  $\Sigma_t$  of  $y_t$ . The former, that is the average, is denoted as  $\mu_t$  and the size and physical meaning in the element are the same as  $y_t$ . On EKF-SLAM, the robot state vector  $p$  and positions of LMs ( ${}^{(j)}m_x$  and  ${}^{(j)}m_y$ ) are estimated via an estimation of  $\mu_t$ . In order to update  $\mu_t$  and  $\Sigma_t$  in an iterative computation, the input  $u_t$  to the robot kinematics, an actual measurement value  $z_t$  from the sensor, the robot kinematics model described by (4), and the LRF sensor measurement model defined by (7) are used. Specifically, an estimated average at the time  $t$ , that is denoted by  $\bar{\mu}_t$ , is computed by

using the robot kinematics model from the previous average  $\mu_{t-1}$ . Subsequently, the covariance matrix  $\Sigma_t$  for  $\mu_t$  is updated by projecting other covariance of noise perturbation expressed in a control space domain into the state space domain. (This procedure is called a prediction phase.) Next, a difference between an actual measurement value  $z_t$  and a predicted measurement  $\hat{z}_t$  is computed, where  $\hat{z}_t$  is obtained using the LRF sensor model from the predicted value  $\bar{\mu}_t$ . The difference ( $z_t - \hat{z}_t$ ) is reflected into the update of prediction of  $\bar{\mu}_t$  according to other updated covariance matrix  $\Sigma_t$ . (This procedure is called a measurement update phase.) All the details are explained below.

#### Step 0. Preparation

A new coefficient matrix  ${}^{(j)}F'$  that extracts both state variables of robot ( $x, y, \theta$ ) and  $j^{\text{th}}$  LM ( ${}^{(j)}m_x, {}^{(j)}m_y$ ) from the augmented vector  $\mathbf{y}_t$  is defined by

$${}^{(j)}F' = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{5 \times (2j-2)} \\ \mathbf{0}_{2 \times 3} & \mathbf{I}_{2 \times 2} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{3 \times 2} \\ \mathbf{0}_{5 \times (2N-2j)} \end{bmatrix} \in \mathbb{R}^{5 \times (2N+3)} \quad (12)$$

It is assumed that a constant covariance matrix  $R$  of the robot status  $p$  is known as  $\text{diag}(\sigma_x^2, \sigma_y^2, \sigma_\theta^2)$ . Similarly, other covariance matrix  $Q$  concerning the measurement value  $z$  is predetermined as  $Q := \text{diag}(\sigma_r^2, \sigma_\phi^2)$ .

#### Step 1. Prediction phase

##### 1-1. Extraction

Temporal variable  $\bar{\theta}$  is extracted from an estimates average vector  $\mu_{t-1}$  that was computed at time  $t-1$ . This extraction operation is described as  $\bar{\theta} \leftarrow \mu_{t-1}[\theta]$ , and similar expression is used below.

##### 1-2. Update of the average of predicted variable

The average value  $\bar{\mu}_t$  is computed using the robot kinematics model from the temporal variable  $\bar{\theta}$  and input  $u_t$  as follows.

$$\bar{\mu}_t = \mu_{t-1} + F^T \cdot g(\bar{\theta}, u_t) \quad (13)$$

##### 1-3. Update of the covariance matrix

Covariance matrix  $\Sigma$  is obtained as a sum of two variances: one is a variance  $\mathbf{G}_t$  that comes from a change in  $\mathbf{y}_t$ , and the other is a projection of a variance in a change of the robot status  $p$  into the state space domain. Therefore, a predicted covariance  $\bar{\Sigma}_t$  is calculated by

$$\bar{\Sigma}_t = \mathbf{G}_t \Sigma_t \mathbf{G}_t^T + F^T R F \quad (14)$$

#### Step 2. Measurement update phase

Measurement information of all LMs cannot be always obtained by the LRF sensor due to an existence of environmental obstacles. Therefore, only estimated states of LMs that can be found from the sensor data are updated. And each LM is registered in series that the LM is newly identified by the detection method of virtual comb-shape. Specifically, each LM's positional information based on the measured data is stored into the augmented state vector  $\mathbf{y}$  defined by (9). Herein

after, steps 2-1 to 2-5 are applied to each measurement status  ${}^{(j)}z_t = {}^{(j)}[r, \phi]_t$  for all LMs which are found at time  $t$ .

##### 2-1. Reference of numbering of individual LM

Mutable array  $c$  to register a correspondence relation between a numbering of each LM,  $j$ , and the order of discovering of LM,  $k$ , is prepared, that is the index  $j$  of LM is referred as  $j = {}^{(k)}c$ .

##### 2-2. Addition of new LM entry

If  $j^{\text{th}}$  LM is discovered for the first time from the measured LRF sensor data  ${}^{(k)}[r, \phi]_t^T$ , a new component corresponding to the new LM is added as new  $j^{\text{th}}$  entry  ${}^{(j)}[\bar{m}_x, \bar{m}_y]_t^T$  in the predicted vector  $\bar{\mu}_t$  using the following computation.

$${}^{(j)} \begin{bmatrix} \bar{m}_x \\ \bar{m}_y \end{bmatrix} = \begin{bmatrix} {}^{(k)}r_t \cdot \cos({}^{(k)}\phi_t + \bar{\theta}) \\ {}^{(k)}r_t \cdot \sin({}^{(k)}\phi_t + \bar{\theta}) \end{bmatrix} \quad (15)$$

##### 2-3. Update of known LMs

When  $j^{\text{th}}$  LM is already in the mutable array  $c$ , an estimated state of measurement  ${}^{(j)}\hat{z}_t$  is computed by the following equation.

$${}^{(j)}\hat{z}_t = [\sqrt{q'}, \arctan 2(\delta'_y, \delta'_x) - \bar{\theta}]^T \quad (16)$$

where  $q' = \delta'_x{}^2 + \delta'_y{}^2$ ,  $\delta'_x := {}^{(j)}\bar{m}_x - \bar{x}$ ,  $\delta'_y := {}^{(j)}\bar{m}_y - \bar{y}$ ,  $\bar{x} \leftarrow \mu_{t-1}[x]$ , and  $\bar{y} \leftarrow \mu_{t-1}[y]$ .

##### 2-4. Calculation of Kalman gain

In order to update a Kalman gain for EKF, it is necessary to compute a Jacobian concerning the robot status  $p$  and  $j^{\text{th}}$  LM's status. This Jacobian is denoted by  ${}^{(j)}H_t$  and derived as

$${}^{(j)}H_t = \frac{1}{q} \begin{bmatrix} -\sqrt{q}\delta_x & -\sqrt{q}\delta_y & 0 & \sqrt{q}\delta_x & -\sqrt{q}\delta_y \\ \delta_y & -\delta_x & -1 & -\delta_y & \sqrt{q}\delta_x \end{bmatrix} \cdot {}^{(j)}F' \quad (17)$$

Kalman gain is updated by the following equation.

$${}^{(j)}K_t = \bar{\Sigma}_t ({}^{(j)}H_t)^T \cdot \left\{ {}^{(j)}H_t \bar{\Sigma}_t ({}^{(j)}H_t)^T + Q \right\}^{-1} \quad (18)$$

This equation means a modification of the gain by considering a sum of two kinds of covariance: a projection of augmented vector's covariance into a measurement space which was spanned by  $(r, \phi)$ , and covariance  $Q$  in measurement signal.

##### 2-5. Update of estimated variables

The estimated augmented vector  $\bar{\mu}$  is modified according to a difference between an actual measurement variable  $z$  and an estimated quantity of measurement  $\hat{z}$  by

$$\mu_{t+1} \leftarrow \bar{\mu}_t + {}^{(j)}K_t \cdot ({}^{(j)}z_t - {}^{(j)}\hat{z}_t) \quad (19)$$

Estimated covariance is also updated for a next loop operation as follows.

$$\Sigma_{t+1} = (I - {}^{(j)}K_t \cdot {}^{(j)}H_t) \cdot \bar{\Sigma}_t \quad (20)$$

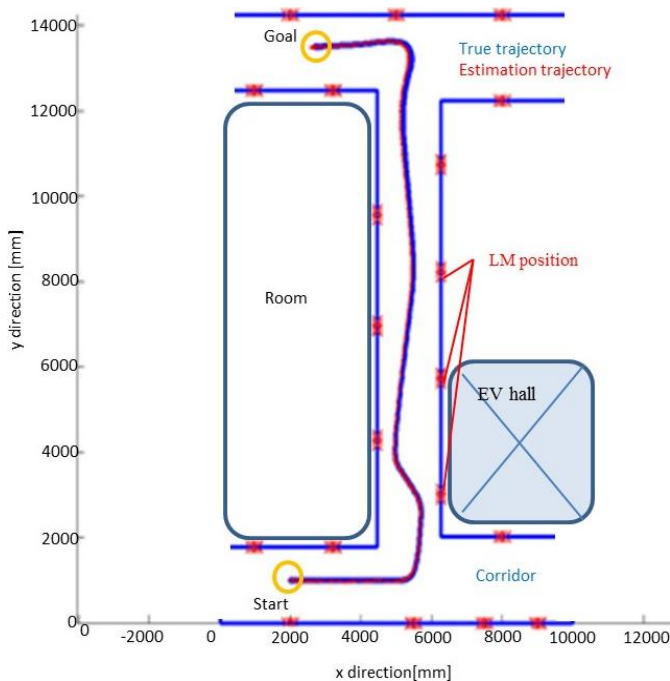
By repeating an above-mentioned procedure, an estimation of augmented vector  $\mu_t$  is updated successively, and the robot state  $(x, y, \theta)$  and all LMs' state  $(m_x, m_y)$  that are included in  $\mu$  can be estimated. In short, an estimation of the robot position and an update of the environmental map are realized simultaneously.

#### IV. VERIFICATION OF THE PRESENTED SLAM METHOD

##### A. Simulation

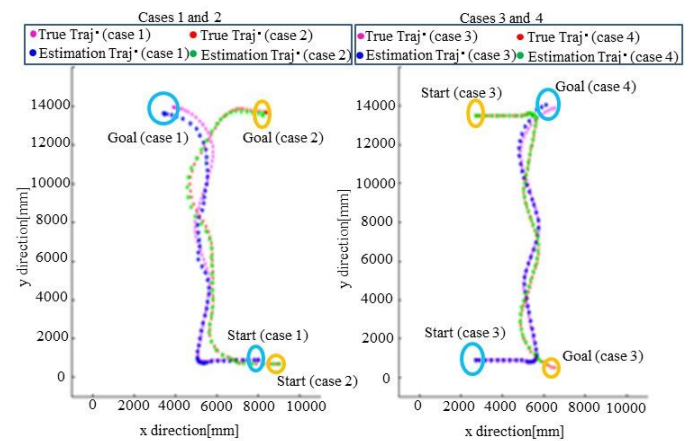
First, the simplified EKF-SLAM algorithm was solely confirmed by simulation with fixed point LMs not with virtual comb-shape LM's. The simulation environment for a virtual SLAM robot was designed, as shown in **Figure 2**. The environment was assumed as an H-shape corridor in a building, where the latter experiment was performed, and total 19 land markers' point were put on the virtual space. For the virtual robot motion control, an input to the robot was given by an operator on the simulator. The maximum of input was specified as 0.5 [m/s] and 0.5 [rad/s] for translational and angular velocities. To check the robustness of the EKF effect, the LRF measurement data was intentionally contaminated by random measurement noise (the maximum is 3 [cm]).

In **Figure 2**, the trajectories obtained by simulation were drawn. The blue and red trajectories show the true trajectory of the virtual robot and the estimated one by the presented EKF-SLAM algorithm. It was confirmed that the estimated trajectory coincides with the true one pretty well. The mean error of the estimation of the robot's position was 112 [mm] under the intentional 3 [cm] perturbation measurement noise. This error rate corresponds to 0.82 [%] against the total length of the robot trajectory. It can be confirmed that mechanism of the simplified EKF-SLAM worked sufficiently.

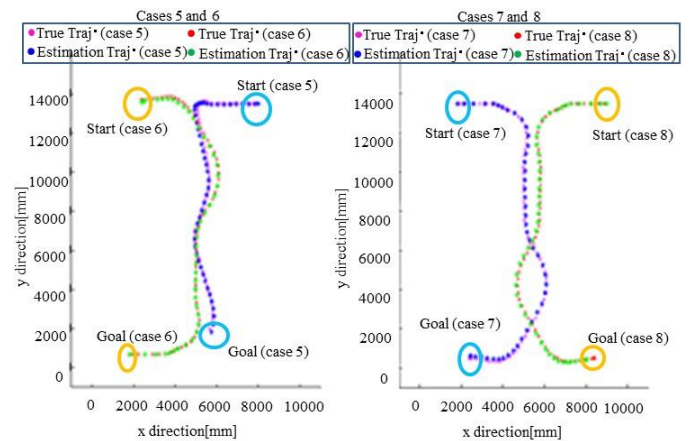


**Figure 2** Simulation result of the simplified EKF-SLAM using fixed points LMs under intentional measurement noise.

Second, in order to check an effectiveness of the same EKF-SLAM algorithm with the virtual-comb shape maker detection approach, another simulation was executed. Using same virtual corridor environment, total 19 virtual comb-shaped LMs were placed in wall part. Virtual measurement data from a robot to virtual comb-shaped LMs was obtained by a lay-tracing computation in the simulation. At that time, random measurement noise (maximum 5 [mm] and 0.01 [rad] for distance and angle respectively) was added to imitate the actual measurement noise. The command to the robot motion was similarly given by an operator during the simulation. Changing a combination of the start and goal of virtual robot locomotion, total eight cases of the simulation were executed and each result is drawn in **Figure 3** and **Figure 4**. As shown in those figures, it was confirmed that each estimated trajectory matched well to true one. **TABLE I** summarizes the mean error with variance for eight cases.



**Figure 3** SLAM simulation result (cases 1, 2, 3, and 4)



**Figure 4** SLAM simulation result (cases 5, 6, 7, and 8)

It can be found that the mean error was almost same small value in spite of existence of disturbance noise in the virtual sensor measurement. Average of the error in eight cases was  $103.1 \pm 7.28$  [mm] which corresponds to 0.57 % accuracy against total length of robot locomotion. It can be said that the presented EKF-SLAM with virtual comb-shaped makers can obtain sufficient presumption of the robot's self-position and environmental geometry position.

TABLE I Average error of self-position estimation

Case	Average error
1	$122.6 \pm 79.3$
2	$97.7 \pm 86.2$
3	$97.9 \pm 86.1$
4	$109.8 \pm 44.1$
5	$102.6 \pm 56/4$
6	$85.7 \pm 42.6$
7	$105.1 \pm 108.4$
8	$122.7 \pm 79.4$

### B. Experiment

The presented EKF-SLAM approach was implemented to actual robot that was designed in authors' previous study [8]. The maximum speed of the robot was specified in similar way to previous simulation conditions. Eight real virtual comb-shaped markers were attached to wall of a real indoor corridor as shown in **Figure 5 (a)**, and size and sensor spec were the same as in the simulation. **Figure 5 (b)** shows the layout of the LMs and the planned trajectory for the actual robot.

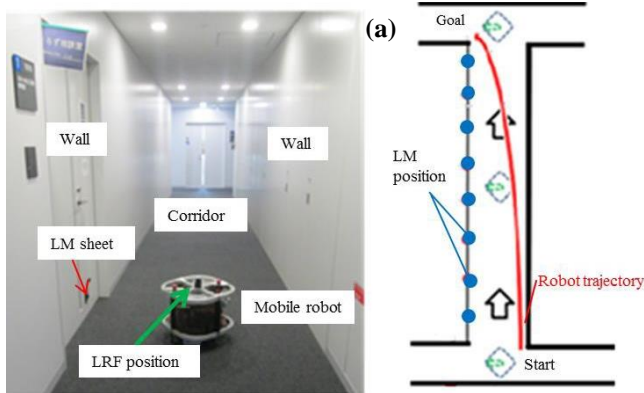


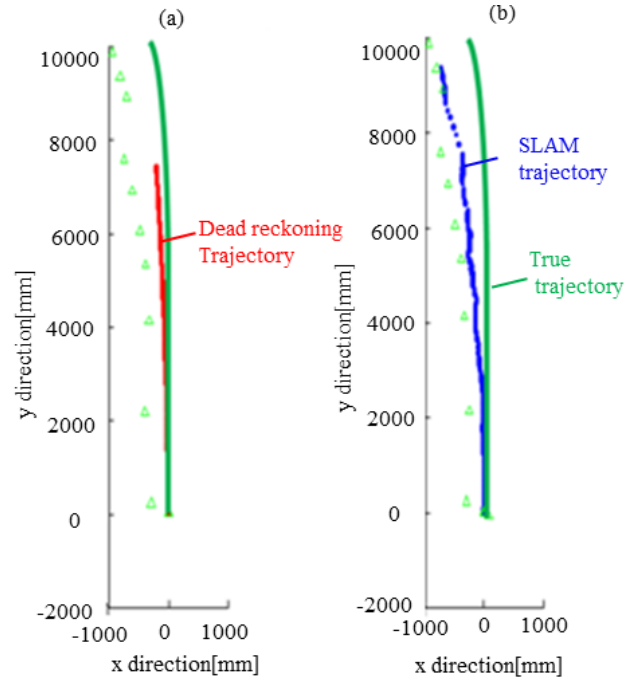
Figure 5

- (a) Experimental environment with the actual mobile robot  
 (b) Layout of the environment and robot's trajectory

In the experiment, the robot was controlled by manual operation along the line of the corridor to about 10 [m] length. **Figure 6 (a)** shows the obtained trajectory by a dead reckoning form the encoders embedded in the robot's motors. **Figure 6 (b)** shows the estimated result obtained by presented method.

In case of dead reckoning, the error rate against true trajectory was as large as 40.7 [%]. This large error was caused by a slip of the wheels of the robot. On the other hand, the presented method shows a 5.3 [%] error which was smaller than the case of dead reckoning. This value is better than dead-reckoning but is worse as about ten times than the result of simulation. Although the conditions in real and in simulation are not the same, the reason comes from difficulty in the detection of virtual comb-shaped in real environment. That is, the detection by the LRF sensor is weakened depending on the

length of the measurement points. This issue remains as a future work.



(b) **Figure 6 Self-position estimation result: (a) by dead reckoning (b) by presented EKF-SLAM with virtual comb-shaped markers.**

### V. CONCLUSION

Number of parents who work while entrusting their child to the day nursery is increasing in Japan.

From this reason, objective monitoring system to watch over children is expected. In this study, to solve problem specific to the kindergarten use, an approach of a simplified EKF-SLAM which is effective by short-distance survey is proposed. The proposed SLAM algorithm was evaluated by a simulation, and the practical effectiveness was confirmed by experiment with actual mobile robot. Through simulation, the SLAM algorithm with an extraction method of the special LM from LRF sensor data was verified. For this result, it was confirmed that the estimation error was as small as 0.57 % against total length of robot locomotion. Finally, same algorithm was implemented to an actual robot that was equipped with the LRF sensor, and practical effectiveness was validated. As a result, the presented SLAM with a margin of estimation error of 5.3 % was also confirmed in the experiment.

As remained issue, we have to confirm whether the presented SLAM technique operates well when children surround the robot in real situation.

### ACKNOWLEDGMENT

Experiment in kindergarten was supported by many children, their parents, and childminders in Benesse Childcare Centor Hiyosi. The present authors really appreciate many participants who embraced our requests kindly.

## REFERENCES

- [1] Japan Economics Information Division, "Current of Robot Industry of Japan", JETRO Japan Economic Monthly, March 2006.
- [2] Jun Goto, Takuya Kidokoro, Tomohiro Ogura, and Satoshi Suzuki, "Activity Recognition System for Watching over Infant children," in Proc. of the 22nd IEEE International Symposium on Robot and Human Interactive Communication RO-MAN, 2013.
- [3] Xiaoyi Yu, Lingyi Wu, Qingfeng Liu, Han Zhou, "Children tantrum behaviour analysis based on Kinect sensor", Intelligent Visual Surveillance (IVS) 2011, pp. 49-52, 2011.
- [4] Mirco Alpen, Klaus Frick, Joachim Horn "A Real-Time On-Board Orthogonal SLAM for an Indoor UAV", 5<sup>th</sup> International Conference, ICIRA 2012, Montreal, Proceedings, Part III, pp. 542-551, 2012.
- [5] Juan Nieto, Tim Bailey and Eduardo Nebot, "Scan-SLAM: Combining EKF-SLAM and Scan Correlation", Field and Service Robotics, Results of the 5<sup>th</sup> International Conference, Part 4, pp. 167-178, 2006.
- [6] A.Kawewong, N.Tongprasit, S.Tangruamsub, O.Hasegawa, "Online and Incremental Appearance-based SLAM in Highly Dynamic Environments", International Journal of Robotics Research (IJRR), June 23, 2010.
- [7] Takahiko Nakamura, Satoshi Suzuki, "Kindergarten SLAM utilizing Laser range sensor and retro-reflective markers", Industrial Electronics Society, IECON 2013 – 39<sup>th</sup> Annual Conference of the IEEE, pp. 8300-8305, 2013.
- [8] Shiro Matsunaga, Satoshi Suzuki "Human assistive control on tele-operated human-robot cooperative carrying task," IECON 2012, pp. 4165-4170, 2012.