ICROS-SICE International Joint Conference 2009
August 18-21, 2009, Fukuoka International Congress Center, Japan

Real-Time Single Camera SLAM Using Fiducial Markers

Hyon Lim[1] and Young Sam Lee[2]

School of Electrical Engineering, Inha University, Incheon, Korea
E-mail: alex@alexlab.net[1], lys@inha.ac.kr[2]

Abstract: In this paper, a real-time single camera simultaneous localization and mapping (SLAM), that uses artificial landmarks is proposed. Proposed method uses the extended Kalman filter (EKF) to estimate robot pose and landmarks position. The core of the approach is the online creation of a map of fiducial markers in the environment within a probabilistic framework. Our key contributions include a development of measurement model of fiducial markers, solutions for global registration of fiducial markers, and the development of calibration-free indoor localization method. We present a detailed method to estimate the 3D location of fiducial markers from an image and how the robot is positioned. Simulation and experimental results for a self-developed mobile robot are both presented.

Keywords: Simultaneous Localization and Mapping (SLAM), EKF-SLAM, Fiducial marker, Monocular Vision

1. INTRODUCTION

The recent trends in visual SLAM has been towards the use of natural landmarks. However, one of the most critical limitation to the SLAM based on natural landmarks is the data association problem. The data association is the issue of matching current observations with previously obtained observations from a captured scene. Several techniques have been applied to the data association problem, such as the EKF based active region detection [1], nearest-neighbor method [2]. Nonetheless, the natural-landmark-based SLAM methods are vulnerable to moving objects located in a captured scene. Because the moving object causes miss data association. This crucial ability has been an obstacle of industrialization of SLAM into indoor robots.

In contrast, artificial landmarks have been guarantee better localization repeatability than natural landmarks thanks to their distinguishablity. Therefore, many commericial indoor positioning products have adopted artificial landmarks. However, the existing indoor localization methods based on artificial landmarks have traditionally relied on the location of landmarks measured before performing localization task. Complicated calibration and measurement process are involved to use these products. The set up and calibration of such devices is not trivial. Accuracy of location estimation is up to precision of beacons location measurements. Moreover, this process should be repeated when structure of the indoor changed.

To resolve these difficulties, we present the real-time single camera SLAM using fiducial markers. To summarize our complete localization algorithm, we take the image stream from a single camera, online creation of a map of fiducial markers in the environment within a probabilistic framework and use these to EKF to estimate robot's pose.

This paper has two main contributions. First, existing researches related to indoor localization require the location of landmarks in global coordinates to be stored a priori in the system. However, the proposed method does not require the location of landmarks to be stored a priori. We simultaneously estimate the location of landmark in global coordinates and robot based on bayesian probabilistic framework. Second, fiducial markers can be printed on a paper and attatched to a planar surface.(e.g., walls, ceilings, floors). This suggests that landmarks can be installed flexibly and randomly without any priori knowledge. Compared with existing techniques which use expensive laser range finders, the proposed method requires a camera, papers and a desktop printer to build an indoor localization system. Consequently, we can build SLAM system in a cost-effective way.

The next section is devoted to a literature review and comparison between existing approaches and proposed approach. Section 3 covers the algorithm of landmarks and the robot estimation. In section 3, measurement model of fiducial markers and motion model of self-developed mobile robot are presented. EKF solution of proposed method is also described. In Section 4, both simulation and experimental results are presented. The real experiment was conducted at corridor environment with self-developed mobile robot. Finally, conclusions are presented in Section 5.

2. RELATED WORK

Triangulation method has been applied to the landmark based localization research. The works of [3, 4] have used triangulation approach. It is used to estimate the robot's position based on observed bearings and ranges from sensor. The main disadvantage of triangulation method is that the landmark's location should be known a prior to localization. However, landmarks will typically be located in 3-D space, so it will be a hard work to measure the all location of landmarks.

Product use triangulation method to estimate robot's position already exist [5-7]. The weaknesses of mentioned products are complicated calibration procedure. [6] uses ultrasonic emitter/receiver to obtain distances from each landmarks to receiver. The landmark has power supply unit because it is ultrasonic emitter. Furthermore, estimation accuracy is varying with temperature because the speed of sound is different in condition of temperature. [5, 7] uses retro-reflective patches to po-

PR0002/09/0000-0177 ¥400 © 2009 SICE

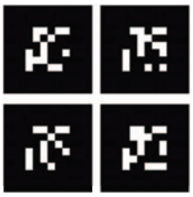


Fig. 1: Example fiducial markers used in this paper

sitioning robot. The [7] is designed for a single room and involves only a single landmark. Therefore it does not need data association. The landmark of [5] is an encoded patch for the data association. However, the calibration and mapping procedure should be performed prior to operation. SLAM with visual markers using monocular vision has not been proposed so far to the best of our knowledge.

3. METHODS

3.1 Fiducial Marker as a Landmark

In this paper, we use fiducial markers as landmarks. Fiducial markers are frequently used in augmented reality researches to estimate the camera pose for overlaying of virtual model to a captured video frame [8, 9]. But mentioned researches used fiducial markers in a single local coordinate. For example, estimation space is 1-2 meter wide and estimation process only involves about one to four markers. However, the proposed method has over 6m estimation space and involves more than 15 markers. These markers are not attahced on a single desktop, but distributed along the wall. Unlike augmented reality implementation, proposed method can recover the position of markers in the global frame from integration of every single observation. The number of markers are bounded by the 10 bits of identification code, which is 1024(=2^{10}). This paper considers basic fiducial markers shown in Fig. 1 for the simplicity [9]. Fiducial markers consist of four vertices and an identification code at the center of a marker. The identification code contains 10-bit binary number and CRC. The typical experiment environment of the proposed method is illustrated in Fig. 8b.

Fiducial markers have good properties to extract markers from a complicated and mixed scene. First, its tone and shape is easy to detect. The tone of fiducial markers is black and white, so it is easy to distinguish in the binary image. Second, the vertices of fiducial markers can be easily extracted from line crossings thanks to its square shape. However, black and white fiducial markers are visually unpleasure. For the visual pleausre, various form of another type of fiducial markers exist. In [10], the fiducial markers are optimized for visual pleasure with the aim to unobtrusively integrate them into homes. In [11], the fiducial markers are invisible to human eyes to avoid undesirable visual effects using retroreflector paint. The mentioned fiducial markers can be applied to the proposed method for the visual pleasure.

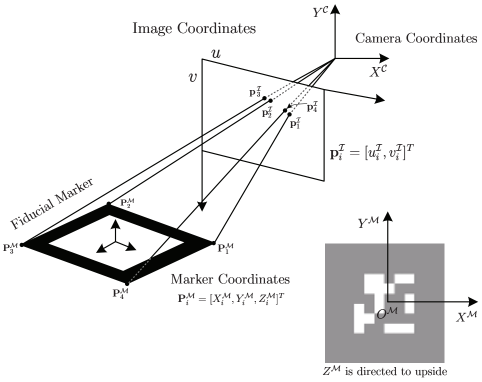


Image Coordinates
$Y^{\mathcal{C}}$
Camera Coordinates
u
v
$X^{\mathcal{C}}$
$\mathbf{p}_3^{\mathcal{I}}$
$\mathbf{p}_2^{\mathcal{I}}$
$\mathbf{p}_4^{\mathcal{I}}$
$\mathbf{p}_1^{\mathcal{I}}$
$\mathbf{p}_i^{\mathcal{I}} = [u_i^{\mathcal{I}}, v_i^{\mathcal{I}}]^T$
Fiducial Marker
$\mathbf{P}_2^{\mathcal{M}}$
$\mathbf{P}_3^{\mathcal{M}}$
$\mathbf{P}_1^{\mathcal{M}}$
$\mathbf{P}_4^{\mathcal{M}}$
Marker Coordinates
$\mathbf{P}_i^{\mathcal{M}} = [X_i^{\mathcal{M}}, Y_i^{\mathcal{M}}, Z_i^{\mathcal{M}}]^T$
$Y^{\mathcal{M}}$
$O^{\mathcal{M}}$
$X^{\mathcal{M}}$
$Z^{\mathcal{M}}$ is directed to upside

Fig. 2: Marker, Camera, Image coordinates system

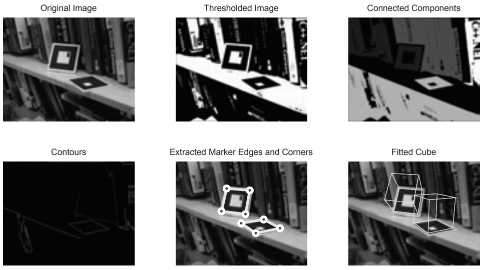


Fig. 3: Image processing algorithm

3.2 Calculation of Position of Markers

The fiducial marker coordinates is shown in Fig. 2. The position of a marker from camera center is computed from transformation of coordinates between marker and camera. We use $\mathbf{P_i} = [X, Y, Z]^T$ to denote vertices on the marker in the global coordinates. If there exists four points on a plane, we can assume that the 3D coordinates of known four points are on the $Z = 0$ plane without lack of generality. By abuse of notation, we still use $\mathbf{P_i}$ to denote vertices on the marker, but $\mathbf{P_i} = [X, Y]^T$ since Z is always equal to 0.

When a fiducial marker is captured to a camera, the four vertices of a marker are projected to the 2D image. The parameters involved in this process are called intrinsic parameters. This parameters are obtained prior to SLAM by applying camera calibration techniques [12]. Extrinsic parameters represent the camera pose relative to marker coordinates that can be recovered from the homography between a marker and imaged vertices. Homography is a 2D to 2D linear transform between a marker in the global coordinates and projected vertices in the image coordinates. The marker vertices point denoted by $\mathbf{P_i}$ and its image $\mathbf{p_i}$ is related by a homography $\mathbf{H}$:

$$s\tilde{\mathbf{p}} = \mathbf{H}\tilde{\mathbf{P}} \tag{1}$$

where

$$\mathbf{K} = \begin{bmatrix} s_u & \gamma & u_0 \\ 0 & s_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{H} = \mathbf{K} \begin{bmatrix} r_{11} & r_{21} & t_1 \\ r_{12} & r_{22} & t_2 \\ r_{13} & r_{23} & t_3 \end{bmatrix} \tag{2}$$

The $\mathbf{K}$ is a camera intrinsic matrix, with (u_0, v_0) the coordinates of the principal point, s_u, s_v the scale factors in image u and v axes, and γ the parameter describing the skew of the two image axes. The following equation represent the relation between marker and image coordinates.

$$\begin{bmatrix} \tilde{\mathbf{P}}_1^T & \mathbf{0^T} & -u_1\tilde{\mathbf{P}}_1^T \\ \mathbf{0^T} & \tilde{\mathbf{P}}_1^T & -v_1\tilde{\mathbf{P}}_1^T \\ \vdots & \vdots & \vdots \\ \tilde{\mathbf{P}}_4^T & \mathbf{0^T} & -u_4\tilde{\mathbf{P}}_4^T \\ \mathbf{0^T} & \tilde{\mathbf{P}}_4^T & -v_4\tilde{\mathbf{P}}_4^T \end{bmatrix} \mathbf{b} = \mathbf{Ab} = \mathbf{0} \tag{3}$$

where $\tilde{\mathbf{P}} = [X, Y, 1]^T$, $\mathbf{b} = [\bar{\mathbf{h}}_\mathbf{1}^\mathbf{T}, \bar{\mathbf{h}}_\mathbf{2}^\mathbf{T}, \bar{\mathbf{h}}_\mathbf{3}^\mathbf{T}]^\mathbf{T}$, (u, v) are image coordinates. And $\bar{\mathbf{h}}_i^T$, the ith row of $\mathbf{H}$.

The matrix $\mathbf{A}$ can be initially solved by singular value decomposition(SVD). Using SVD, $\mathbf{A}$ can be decomposed as $\mathbf{A} = \mathbf{UDV^T}$ with D diagonal with positive diagonal entries, arranged in descending order down the diagonal. The last column of V is the solution of the parameter set $\mathbf{b}$ in a least square manner. In most cases, the condition number of $\mathbf{A}$ is poor, because of some elements are constant, some are in pixels, some are in *mm* unit. Much better results can be obtained by applying pixel coordinates normalization, described in [13], prior to computation. The Algorithm 1 describes the overall process of marker extraction and pose estimation.

Algorithm 1 Calcuation of Position of Marker (Fig. 3)

```
Thresholding of the image with threshold
Get connected components from the thresholded image
Get contours from the connected components image
Extract four vertices from the contour image
Solve the linear equation (3) applying SVD with known marker geometry and extracted image coordinates.
Set the cost function $J$ to be minimized in order to determine $\mathbf{b}$ for over-determined solutions [13].
while cost function $J$ above threshold do
    do Levenberg-Marquardt iteration
end while
```

3.3 Robot and Sensor model

3.3.1 Notations

In this paper, we attached three additional letters enclosded in brackets and denoting, by order, the directions of the three coordinates axes. For example, $\mathcal{O}^{\mathcal{W}}\{NWU\}$ would be a world frame with the X-axis pointing 'N'orthwards, the Y-axis pointing 'W'estwards and the Z-axis pointing 'U'pwards (See Fig. 4). In the same way, $\mathcal{O}^{\mathcal{R}}\{FLU\}$ would be a 'F'ront, 'L'eft, 'U'pper by order. This notation is developed in the thesis by Ortega [14]. This can be very helpful when reasoning about the frame axes order.

3.3.2 Motion model of self-developed robot

The developed robot is shown in Fig. 8a. We applied proposed method to this robot with motion equation described in this section. We assumes the robot motion model is nonlinear time-varying equation denoted by

$$\mathbf{x}_{k+1} = \mathbf{f}_v(\mathbf{x}_k, \mathbf{u}_k, \mu_k) \tag{4}$$

where $\mathbf{f}(\cdot)$ is nonlinear function, μ_k is a control noise vector which has $\mathbf{Q_k}$ as a covariance. Control vector is $\mathbf{u}_k = [V_k, \Omega_k]^T$ which means velocity and steering angle respectively. The process model of used robot is described by

$$\begin{bmatrix} \dot{x}_v \\ \dot{y}_v \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} V\cos(\theta) \\ V\sin(\theta) \\ V\tan(\Omega)/L \end{bmatrix} \tag{5}$$

where L is the distance between wheel axles as shown in Fig. 4. To simplify the equation in the update stage, the kinematic model of the vehicle is designed to represent the trajectory of the center of the camera as follows :

$$\begin{bmatrix} x_c \\ y_c \end{bmatrix} = \begin{bmatrix} x_v + a\cos\theta - b\sin\theta \\ y_v + a\sin\theta + b\cos\theta \end{bmatrix}. \tag{6}$$

The derivation of this equation is given by

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \end{bmatrix} = \begin{bmatrix} \dot{x}_v - (a\sin\theta + b\cos\theta)\dot{\theta} \\ \dot{y}_v + (a\cos\theta - b\sin\theta)\dot{\theta} \end{bmatrix}. \tag{7}$$

Finally, the full state representation can be written as

$$\mathbf{x_{k+1}} = \mathbf{f_v}(\mathbf{x_k}, \mathbf{u_k}, \mu_\mathbf{k}) = \begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}, \tag{8}$$

where

$$\begin{aligned} f_1 =& x_k + \Delta t \cdot V_k \cos\theta_k \\ & - \Delta t \frac{V_k}{L} \tan\Omega_k \cdot (a\sin\theta_k + b\cos\theta_k) \\ f_2 =& y_k + \Delta t \cdot V_k \sin\theta_k \\ & + \Delta t \frac{V_k}{L} \tan\Omega_k \cdot (a\cos\theta_k - b\sin\theta_k) \\ f_3 =& \theta_k + \Delta t \frac{V_k}{L} \tan\Omega_k \end{aligned}$$

and Δt denotes sampling interval. Reader should note that the center of camera is regarded as the location of robot. The velocity of the enter of the axle can be obtained from encoders equipped in a shaft. If the linear velocity of vehicle's wheel is denoted by V_e, the V_k is given by

$$V_k = \frac{V_e}{1 - \tan(\Omega) \cdot \frac{H}{L}}. \tag{9}$$

3.3.3 Measurement model of fiducial markers

The observation equation relating the vehicle states to the observation is

$$\mathbf{z}_{i,k} = \mathbf{h}_i(\mathbf{x}_k, \mathcal{M}, \eta_k) = \begin{bmatrix} x_i^{\mathcal{C}} \\ y_i^{\mathcal{C}} \\ z_i^{\mathcal{C}} \end{bmatrix} = \begin{bmatrix} \Phi_{11} \\ \Phi_{21} \\ z_i^{\mathcal{C}} \end{bmatrix}, \tag{10}$$

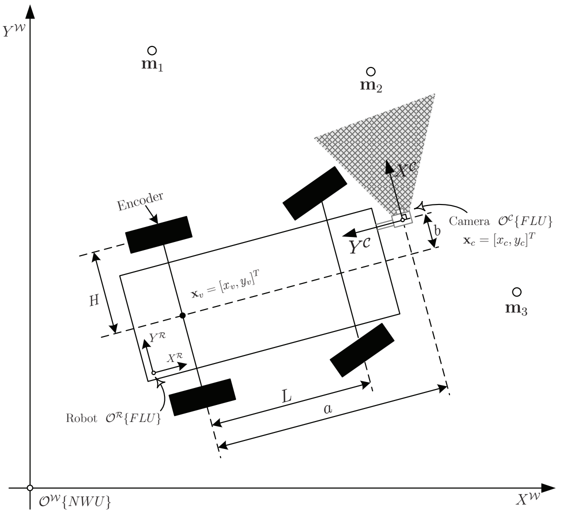

$Y^{\mathcal{W}}$
$\mathbf{m}_1$
$\mathbf{m}_2$
$\mathbf{m}_3$
$X^{\mathcal{C}}$
$Y^{\mathcal{C}}$
Encoder
Camera $\mathcal{O}^{\mathcal{C}}\{FLU\}$
$\mathbf{x}_c = [x_c, y_c]^T$
b
H
$\mathbf{x}_v = [x_v, y_v]^T$
$Y^{\mathcal{R}}$
$X^{\mathcal{R}}$
Robot $\mathcal{O}^{\mathcal{R}}\{FLU\}$
L
a
$\mathcal{O}^{\mathcal{W}}\{NWU\}$
$X^{\mathcal{W}}$

Fig. 4: World, robot, sensor coordinates system

where

$$\Phi_{11} = -(x_i^{\mathcal{W}} - x_k)\sin\theta_k + (y_i^{\mathcal{W}} - y_k)\cos\theta_k$$
$$\Phi_{21} = -(x_i^{\mathcal{W}} - x_k)\cos\theta_k - (y_i^{\mathcal{W}} - y_k)\sin\theta_k,$$

and η_k is a measurement noise vector which has $\mathbf{R}_k$ as a covariance.

3.4 The EKF-based SLAM

3.4.1 The robot and landmarks models

The SLAM problem is that of moving through an environment containing a population of features or landmarks with a given vehicle with a known kinematic model, starting at an unknown location. The vehicle is equipped with a sensor that can take measurements of the relative location between any individual landmark and the vehicle itself. The absolute locations of the landmarks are not available. For an unified framework to solve the SLAM problem, we stack the state of vehicle and landmark as

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{f}(\tilde{\mathbf{x}}_\mathbf{k}, \mathbf{u}_\mathbf{k}, \mathbf{w}_\mathbf{k}) = \begin{bmatrix} \mathbf{f}_\mathbf{v}(\mathbf{x}_\mathbf{k}, \mathbf{u}_\mathbf{k}, \mathbf{w}_\mathbf{k}) \\ \hat{\mathbf{m}} \end{bmatrix} \quad (11)$$

$$\mathbf{P}_\mathbf{k} = \begin{bmatrix} \mathbf{P}_\mathbf{vv} & \mathbf{P}_\mathbf{vm} \\ \mathbf{P}_{\mathbf{vm}^\mathbf{T}} & \mathbf{P}_\mathbf{mm} \end{bmatrix} \quad (12)$$

3.4.2 Prediction step

At the prediction step, the robot state is time-updated without sensor information. At the same time, landmarks in the environment are observed. State prediction is described as

$$\tilde{\mathbf{x}}_{k+1|k} = \mathbf{f}(\tilde{\mathbf{x}}_{\mathbf{k}|\mathbf{k}}, \mathbf{u}_\mathbf{k}, \mathbf{w}_\mathbf{k}). \quad (13)$$

The covariance update is performed applying equation below.

$$\mathbf{P}_{\mathbf{k+1}|\mathbf{k}} = \nabla\mathbf{f}_{\tilde{\mathbf{x}}}\mathbf{P}_{\mathbf{k}|\mathbf{k}}\nabla\mathbf{f}_{\tilde{\mathbf{x}}}^\mathbf{T} + \nabla\mathbf{f}_\mathbf{u}\mathbf{Q}_\mathbf{k}\nabla\mathbf{f}_\mathbf{u}^\mathbf{T}. \quad (14)$$

Jacobian matrix $\nabla\mathbf{f}_{\tilde{\mathbf{x}}}$ is obtained by

$$\nabla\mathbf{f}_{\tilde{\mathbf{x}}} = \frac{\partial\mathbf{f}}{\partial\tilde{\mathbf{x}}_{\mathbf{k}|\mathbf{k}}} = \begin{bmatrix} \frac{\partial f_1}{\partial(x,y,\theta,\mathbf{m}_\mathbf{i})} \\ \frac{\partial f_2}{\partial(x,y,\theta,\mathbf{m}_\mathbf{i})} \\ \frac{\partial f_3}{\partial(x,y,\theta,\mathbf{m}_\mathbf{i})} \end{bmatrix} = \begin{bmatrix} \nabla\mathbf{f}_{\mathbf{v}_\mathbf{x}} & \mathbf{0}_\mathbf{vm} \\ \mathbf{0}_\mathbf{vm}^\mathbf{T} & \mathbf{I}_\mathbf{m} \end{bmatrix}, \quad (15)$$

where $\mathbf{0}_\mathbf{vm}$ is $\mathbb{R}^{\dim(\mathbf{x}_\mathbf{v})\times\dim(\mathbf{m})}$ and $\mathbf{I}_\mathbf{m}$ is $\mathbb{R}^{\dim(\mathbf{m})\times\dim(\mathbf{m})}$. The notation $\dim(\mathbf{x})$ means the length of a given vector. The Jacobian matrices in Eqn. (14) are $\nabla\mathbf{f}_{\mathbf{v}_\mathbf{x}} = \frac{\partial\mathbf{f}_\mathbf{v}}{\partial\mathbf{x}_{\mathbf{k}|\mathbf{k}}}$ and $\nabla\mathbf{f}_{\mathbf{v}_\mathbf{u}} = \frac{\partial\mathbf{f}_\mathbf{v}}{\partial\mathbf{u}_\mathbf{k}}$ respectively.

3.4.3 Update step

A posterior state is computed by

$$\tilde{\mathbf{x}}_{k+1|k+1} = \tilde{\mathbf{x}}_{k+1|k} + \mathbf{K}_{\mathbf{k}+1}\left[\mathbf{z}_\mathbf{k+1} - \mathbf{h}\left(\hat{\mathbf{x}}_{\mathbf{k+1}|\mathbf{k}}\right)\right]$$
$$\mathbf{P}_{\mathbf{k+1}|\mathbf{k+1}} = \mathbf{P}_{\mathbf{k+1}|\mathbf{k}} - \mathbf{K}_{\mathbf{k}+1}\mathbf{S}_{\mathbf{k}+1}\mathbf{K}_{\mathbf{k+1}}^\mathbf{T}$$
$$\mathbf{S}_\mathbf{k+1} = \nabla\mathbf{h}_{\tilde{\mathbf{x}}_\mathbf{a}}\mathbf{P}_{\mathbf{k+1}|\mathbf{k}}\nabla\mathbf{h}_{\tilde{\mathbf{x}}_\mathbf{a}} + \mathbf{R}_{\mathbf{k}+1}$$
$$\mathbf{K}_\mathbf{k+1} = \mathbf{P}_{\mathbf{k+1}|\mathbf{k}}\nabla\mathbf{h}_{\tilde{\mathbf{x}}_\mathbf{a}}\mathbf{S}_\mathbf{k+1}^{-\mathbf{1}}$$

where $\nabla\mathbf{h}_{\tilde{\mathbf{x}}_\mathbf{a}} = \frac{\partial\mathbf{h}_\mathbf{i}}{\partial\tilde{\mathbf{x}}_{\mathbf{a},\mathbf{k+1}|\mathbf{k}}}$.

3.4.4 State augmentation

If a new landmark measurement is available, the state have to be augmented. The measurement must be converted to global coordinates to be added to existing state. Because the measurement is relative location between a landmark and a vehicle. Let $\hat{\mathbf{x}}_{aug}$ be the augmented state with measurement.

$$\hat{\mathbf{x}}_{aug} = \begin{bmatrix} \tilde{\mathbf{x}}_k \\ \mathbf{z} \end{bmatrix}$$
$$\mathbf{P}_\mathbf{aug} = \begin{bmatrix} \mathbf{P}_\mathbf{vv} & \mathbf{P}_\mathbf{vm} & \mathbf{0} \\ \mathbf{P}_\mathbf{vm}^\mathbf{T} & \mathbf{P}_\mathbf{mm} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R} \end{bmatrix}$$

The measurement is then converted to global coordinates by

$$\begin{bmatrix} m_{x,i} \\ m_{y,i} \\ m_{z,i} \end{bmatrix} = \begin{bmatrix} x_i^{\mathcal{W}} \\ y_i^{\mathcal{W}} \\ z_i^{\mathcal{W}} \end{bmatrix} = \mathbf{g}_\mathbf{i}(\mathbf{x}_\mathbf{k}, \mathbf{z})$$
$$= \begin{bmatrix} x_k - x_i^{\mathcal{C}}\sin\theta_k - y_i^{\mathcal{C}}\cos\theta_k \\ y_k + x_i^{\mathcal{C}}\cos\theta_k - y_i^{\mathcal{C}}\sin\theta_k \\ z_i^{\mathcal{C}} \end{bmatrix}$$

We then add the measurements by

$$\tilde{\mathbf{x}}_k^+ = \mathbf{f}_\mathbf{i}(\hat{\mathbf{x}}_\mathbf{aug}) = \begin{bmatrix} \tilde{\mathbf{x}}_k \\ \mathbf{g}_\mathbf{i}(\mathbf{x}_\mathbf{k}, \mathbf{z}) \end{bmatrix}. \quad (16)$$

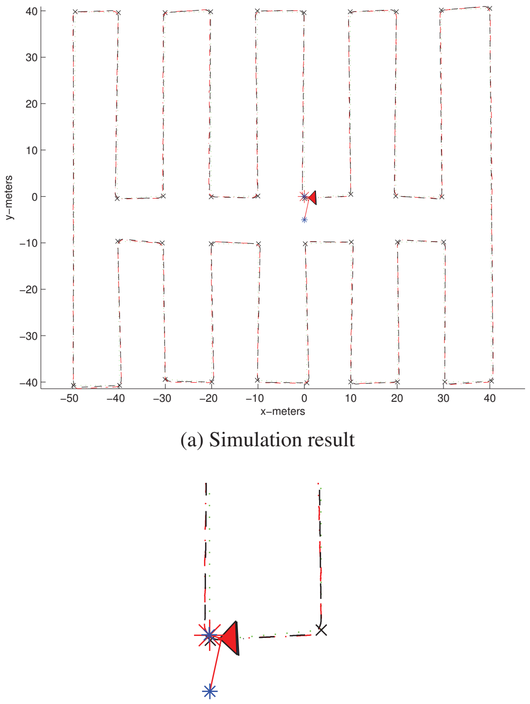
Covariance is updated by

$$\mathbf{P}_\mathbf{a} = \nabla\mathbf{f}_{\mathbf{x}_\mathbf{aug}}\mathbf{P}_\mathbf{aug}\nabla\mathbf{f}_{\mathbf{x}_\mathbf{aug}}^\mathbf{T} \quad (17)$$

where

$$\nabla\mathbf{f}_{\mathbf{x}_\mathbf{aug}} = \frac{\partial\mathbf{f}_\mathbf{i}}{\partial\mathbf{x}_\mathbf{aug}} = \begin{bmatrix} \mathbf{I}_\mathbf{v} & 0 & 0 \\ 0 & \mathbf{I}_\mathbf{m} & 0 \\ \nabla\mathbf{g}_{\mathbf{x}_\mathbf{v}} & 0 & \nabla\mathbf{g}_\mathbf{z} \end{bmatrix}$$

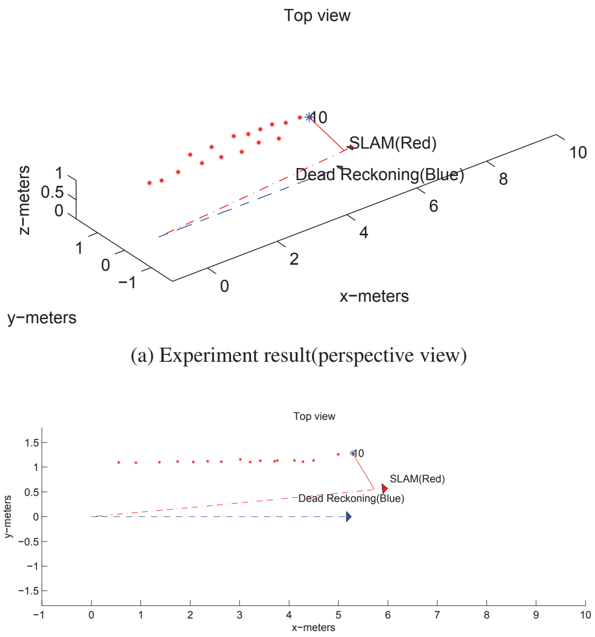
and $\nabla\mathbf{g}_{\mathbf{x}_\mathbf{v}} = \frac{\partial\mathbf{g}_\mathbf{i}}{\partial\mathbf{x}_\mathbf{v}}$ and $\nabla\mathbf{g}_{\mathbf{x}_\mathbf{v}} = \frac{\partial\mathbf{g}_\mathbf{i}}{\partial\mathbf{z}}$.



(a) Simulation result

(b) Zoom of the path

Fig. 5: The simulation result



(a) Experiment result(perspective view)

(b) Experiment result(top view)

Fig. 6: The experiment result. Video illustrating the proposed SLAM results is available from `http://www.alexlab.net/ICCAS2009_HyonLim.avi`

4. EXPERIMENT RESULTS

This section shows simulation and experimental results of the algorithm presented. The simulation was done in an area of 90 by 80 m with a sensor field of view of 4 m (See Fig. 5a). The vehicle travels at a constant speed of 1.5 m/s. The sensor observations are corrupted with Gaussian noise with standard deviations of 0.1 m in range.

The simulation results for the proposed SLAM algorithm are shown in Fig. 5a. Here the dashed line depicts the dead-reckoning for the robot pose and the dash-dot line the estimated vehicle path.

Fig. 5b shows a zoom of the path of the simulation which depicts in more detail the sensory information that is added to the simulation environment obtained by the algorithm. The small difference between the simulated true path and estimated path illustrate the accuracy of the results obtained by the approach.

The algorithm was also tested using experimental data. In the experiment a lab-built robot was fitted with dead reckoning and single camera. The testing environment was the corridor of same floor of laboratory. Fig. 8a shows the robot used for the experiments and 8b shows a picture of experimental area. We attached 17 markers on the side wall randomly.

Fig. 6 illustrates the result obtained with the algorithm. The dashed line denotes the dead-reckoning for the robot pose and the dash-dot line the robot's path estimated. The points on the plot represent an observed marker in the global frame. We set the vehicle twisted 0.0914 rad intentionally to verify the estimation accuracy. In reality, robot was twisted but dead-reckoning result shows that the odometry does not aware it. However, captured image reflects the twisted angle. Therefore, the SLAM result shows that it is very similiar trajectory that robot travels in real. Due to the twist angle, robot was slipped about 0.55 m along y-axis. The estimated vehicle result shows that the estimated slipped distance is very close to the measured value. The travel distance is also same as measured value. The estimation result is shown in Fig. 7.

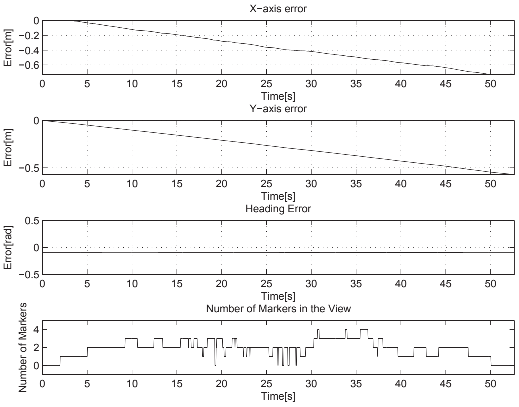
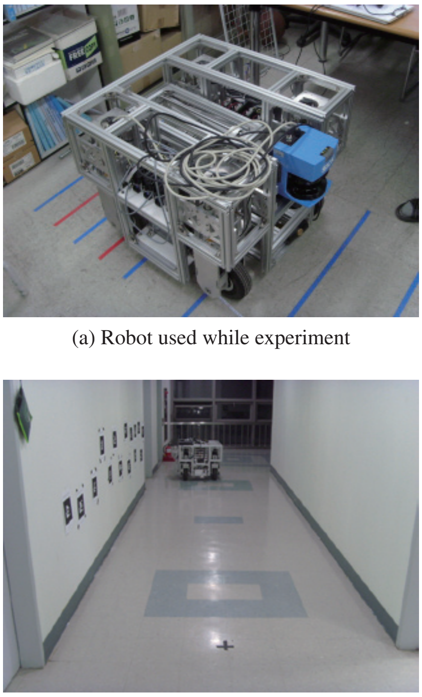


Fig. 7: The error between dead-reckoning and SLAM.

5. CONCLUSIONS

In this paper, we have proposed an indoor SLAM using fiducial markers. The proposed method is based on



(a) Robot used while experiment

(b) Experiment Setup

Fig. 8: Experiment setup and used robot

pose estimation of fiducial markers and we combine it with the EKF to estimate robot pose and landmark positions simultaneously in global coordinates. The major advantages of this approach can be summarized as follows: First, we can build SLAM system in a cost-effective way because the proposed system requires a single monocular camera and fiducial markers easily generated using desktop printers. Second, landmarks can be installed flexibly and randomly without any priori knowledge. This means that there is no complicated calibration and measurement process involved. Third, the proposed method has features of easy data association and good robustness.

Using natural features to perform indoor SLAM is still not mature enough. It has too many exceptions and assumptions. Moreover, existing methods which use natural landmarks or features are easily fragile due to data association failure. However, thanks to the proposed landmark model which includes identification code on the surface, data association is highly successful. Moreover, 3D position of each landmark can be obtained from one frame of image.

We used the EKF for the experiment in this paper. As described in this paper, the EKF involves complicated Jacobian matrices to linearize the motion model and measurement model around current states. However, mean and covariance of estimated states are slightly different from true states. This causes inconsistent behaviour of the EKF. To address this problem, recently, [15] proposes new Kalman filter framework for nonlinear system without linearization procedure. This is called unscented Kalman filter (UKF). We expect that the application of UKF will make the proposed method more accurate.

REFERENCES

[1] A. J. Davison, I. Reid, N. Molton, and O. Stasse, "MonoSLAM: Real-time single camera slam," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, June 2007.

[2] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, June 2001.

[3] I. Shimshoni, "On mobile robot localization from landmark bearings," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 6, pp. 971–976, Dec 2002.

[4] M. Betke and L. Gurvits, "Mobile robot localization using landmarks," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 2, pp. 251–263, April 1997.

[5] "Stargazor$^{\text{TM}}$," Hagisonic, Daejeon, South Korea.

[6] "U-sat$^{\text{TM}}$," Korea LPS, Pusan, South Korea.

[7] "Northstar," Evolution Robotics, Pasadena, CA.

[8] W. Daniel and S. Diester, "Artoolkitplus for pose tracking on mobile devices," in *Proceedings of 12th Computer Vision Winter Workshop (CVWW'07)*, 2007.

[9] M. Fiala, "Artag, a fiducial marker system using digital techniques," in *Proceedings of the Computer Vision and Pattern Recognition (CVPR'05)*, 2005.

[10] S. Saito, A. Hiyama, T. Tanikawa, and M. Hirose, "Indoor marker-based localization using coded seamless pattern for interior decoration," in *IEEE Virtual Reality Conference 2007*, 2007, pp. 67–74.

[11] Y. Nakazato, M. Kanbara, and N. Yokoya, "Localization of wearable users using invisible retro-reflective markers and an ir camera," in *Proceedings of SPIE Electronic Imaging*, vol. 5664, 2005, pp. 563–570.

[12] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1330–1334, 1998.

[13] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[14] J. S. Ortega, "Towards visual localization, mapping and moving objects tracking by a mobile robot: a geometric and probabilistic approach," Ph.D. dissertation, Institut National Polytechnique de Toulouse, 2007.

[15] S. Julier and J. Uhlmann, "A new extension of the kalman filter to nonlinear systems," in *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls, Orlando, FL*, 1997.