Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Planar Features and 6D-SLAM based on Linear Regression Kalman Filters with n-Dimensional Approximated Gaussians

Cihan Ulas*. Hakan Temeltas**

**The Scientific and Technological Research Council of Turkey (TUBITAK, BILGEM), Kocaeli, Turkey (Tel: 262-648-1882; e-mail: cihan.ulas@tubitak.gov.tr).*
*** Department of Control Engineering, Istanbul Technical University, Istanbul, Turkey (e-mail: hakan.temeltas@itu.edu.tr)*

Abstract: In this paper, a six-dimensional (6D) Simultaneous Localization and Mapping (SLAM) based on novel Linear Regression Kalman Filter (LRKF), called Smart Sampling Kalman Filter (S2KF), is proposed. While the conventional feature based SLAM methods use point features as landmarks, only a few take the advantage of geometric information like corners, edges, and planes. A feature based SLAM method using planar landmarks extracted from 3D Light Detection and Ranging (LiDAR) outdoor data is proposed. The method uses the LFKF with n-dimensional approximated Gaussians by addressing the data association problem based on semantic data of plane-features. Experimental results show the appropriateness of the approach, and the filter performance is compared with the traditional filters, such as Unscented Kalman Filters and Cubature Kalman Filters.

Keywords: LiDAR Perception, Plane Detection, 3D Feature Extraction, Outdoor S2KF-SLAM.

1. INTRODUCTION

Simultaneous Localization and Mapping (SLAM) is a fundamental problem of the autonomous systems in GPS (Global Navigation System) denied environments. The LiDAR optical sensor technology is primarily used in many SLAM applications. The data, given by range and bearing, obtained by the LiDAR measurement system is used in two different ways. While some SLAM methods use these raw data with a scan matching algorithm to obtain rigid body transformations (Borrmann et al.), the others use feature extraction algorithms from point clouds (Castellanos et al.). Weingarten and Siegward proposed an indoor 3D feature extraction method based on plane detection for EKF based 3D SLAM (Weingarten and Siegwart).

Feature based SLAM methods have a powerful theoretical background and applied to indoor and 2D case successfully in the last decade (S. Thrun). Neito et al. used artificial landmarks and natural features like tree for 2D SLAM using particle filters (Nieto et al.). Another method for feature extraction is to convert 3D point cloud into 2D range images and apply computer vision techniques (Yangming and Olson). Planes are mostly preferred geometric shapes in 3D environment modeling, and a plane extraction method for scan matching purpose is introduced for indoor and structured environment by Pathak (Pathak et al.).

The feature extraction from LiDAR data obtained in 3D outdoor is a difficult problem due to the complexity and unstructured nature of the environment. In addition, the distribution of the point cloud is directly related to the hardware setup, and it is mostly not homogenous as in camera (Wulf and Wagner); moreover, the vehicle and measurement models are mostly nonlinear. For this reason, the extended versions of the filters are used for linearization purposes like Extended Kalman Filters (EKF). The Unscented Kalman Filters (UKF) is another filtering technique using unscented transform rather than linearization (Martinez-Cantin and Castellanos). This provides better results than EKF if the nonlinear functions are in aggressive fashion. The filters EKF and UKF are known as Gaussian filters since they provide optimum solution under the Gaussian noise. Martinez and Castellanos experimentally validated the usage of UKF for large-scale outdoor environments in (Martinez-Cantin and Castellanos).

Filters based on statistical linearization are known as Linear Regression Kalman Filters (LRKFs). Approximation of Gaussians plays an important role for nonlinear system to improve state estimation. The Cubature Rule for approximation of Gaussians, which is also called Cubature Kalman Filters (CKF), is proposed (Arasaratnam and Haykin). In their paper, it is shown that CKF provides more accurate filtering results than the existing Gaussian filters and solves large spectrum of nonlinear problems. UKF has several limitations, which does not exist in CKF, such as numerical inaccuracy, unavailability of a square-root solution, and filter instability. A CKF-SLAM method based on point features is introduced by Pakki et al. in (Pakki et al.). A new version LRKF called Smart Sampling Kalman Filter (S2KF) is proposed by Steinbring (Steinbring and Hanebeck). S2KF is considered as generalization of all sample-based LRKFs like UKF and CKF since it covers the state space with an arbitrary number of samples. To be able to compute the samples based on the state mean and state covariance,

Localized Cumulative Distribution (LCD)-generated Dirac Mixture approximations are used (Hanebeck et al.).

In this study, it is shown that planar features can be integrated into SLAM state vector and their infinite parameters can be estimated. Initially, the planar features are extracted from 3D LiDAR data, and then the extracted plane features are used in the SLAM problem. The SLAM state vector only holds the infinite plane parameters; however, the other extracted plane parameters such as convex-hull points, covariance matrix of the plane inlier points, and the others supported by the feature extraction method are kept in a feature table and updated after the measurement update step. The data association of the SLAM method is based on the semantic data since the infinite plane parameters cannot be used for correspondence. The performance of the S2KF-SLAM shows that it outperforms the UKF-SLAM and CKF-SLAM, respectively.

In the next section, the planar landmark extraction algorithm is explained. In Section III, a summary of S2KF theory is given, and the plane-feature based LRKF-SLAM is elucidated. In Section IV, the experimental results are presented. The conclusion is drawn in Section V.

2. PLANAR LANDMARK EXTRACTION

In this section, the steps of the planar landmark extraction algorithm are explained. Plane feature extraction from the point cloud for scan matching or environment modelling is not a new concept (Ulas and Temeltas), but in this study, it is shown that the planes can be used in a feature-based SLAM method. The planar feature properties are integrated into SLAM state vector and estimated recursively as in the point feature based SLAM.

The method starts with point cloud discretization by dividing the point cloud into regular cells and assigning each point to these cells. The well-known and simple discretization method is to use fixed cells; therefore, we used fixed cell sizes for the sake of simplicity. The other steps are presented in the following subsections.

1.1 Probabilistic Plane Fitting

After the discretization process, the planar segments are detected by first finding the inliers of a plane via RANSAC algorithm. Then if the number of inliers in a cell exceeds a threshold value, the least mean square plane fitting is applied to these inliers to obtain its parametric representation.

A general plane equation can be written by

$$\mathbf{n}\mathbf{x}^T + d = 0 \tag{1}$$

where $\mathbf{n}$ and d are infinite plane parameters that represents the normal vector and the plane minimum distance to origin, respectively.

1.2 Projection and Convex Hull Computation

After finding the infinite plane parameters of each cell, their convex hull is found. Convex hull computation is carried out in 2D space; therefore the 3D plane points are primarily projected to 2D space. Hence, the rotation matrix (R) and the translation vector (T) are computed by using the principal component analysis. First, the plane points are translated to the first element position by the subtraction of $B = \Gamma - \Gamma_o$, where Γ_o is the first point of the point cloud, and it is considered as the translation vector, T. The covariance matrix, C, of the plane points is computed. Then the singular value decomposition to the covariance matrix is applied as

$$C = UEV^T. \tag{2}$$

The rotation matrix, R, is obtained from the first two columns of the normalized eigenvector matrix, $V = [\mathbf{v}_1 \quad \mathbf{v}_2 \quad \mathbf{v}_3]$, which are the principal axes of the plane points.

$$R = [\mathbf{v}_1 \quad \mathbf{v}_2] \tag{3}$$

The translated plane points, B, are projected to principal axis plane by using this rotation matrix (R).

$$\Delta_{XY} = RB \tag{4}$$

The 3D plane points are represented in 2D space as $\Delta_{XY} \in \mathrm{R}^2$. Then the convex hull of the projected plane points is found. After that, the convex hull points H_p are translated back to 3D space with (5) and represented as $\Delta_{XYZ} \in \mathrm{R}^3$.

$$\Delta_{XYZ} = H_p R^T + \Gamma_0 \tag{5}$$

Finally, the merging step is applied to the plane patches.

1.3 Merging Step

After finding the convex hull points of the planes, the horizontal planes are discarded and the plane merging method is applied. To be able to merge two planes, they must satisfy the three conditions. These conditions are given as follows;

Orientation test: $cos^{-1}\left\{\mathrm{n_i}^{\mathrm{T}}.\mathrm{n_j}\right\} < \mathrm{t_O}$

Translation test: $|\,\mathrm{n_i}^{\mathrm{T}}(c_i - c_j)| < \mathrm{t_T}$

Closeness test: $(c_i - c_j)^T \Sigma_p^{-1} (c_i - c_j) < \mathrm{t_P}$

where c_i is the center of the gravity of a plane, Σ_p is the pooled covariance matrix as defined by

$$\Sigma_p = \frac{\mathrm{k}_i}{\mathrm{k}_i + \mathrm{k}_j}\Sigma_i + \frac{k_j}{\mathrm{k}_i + \mathrm{k}_j}\Sigma_j. \tag{6}$$

where k_i is the number of inliers in a plane, and Σ_i is the covariance matrix of the plane points.

The first two conditions are obvious and applied to indoor case (Weingarten et al., 2004) by Weingarten; however, the closeness test is a must for the outdoor case. The plane segments may satisfy the first two tests, but they might be actually far away from each other and belong to different surfaces. For that reason, the usage of the Mahalanobis

distance with pooled covariance matrix Σ_p (6) is proposed in the third constraint to measure closeness.

The merged finite planes are accepted as landmarks if they contain sufficient number of points and their area is larger than a threshold value. As a result, if a cell contains a plane, then its normal vector (n_i), center of gravity (c_i), convex hull points (XYZ_i), covariance matrix (Σ_i), and the number of inliers (k_i) is computed. Here, i denotes an non-empty cell identity number. Fig. 1 shows the extracted landmarks from the first scan of the data set (Wulf). This scan belongs to an outdoor environment and can be considered as semi-structured.

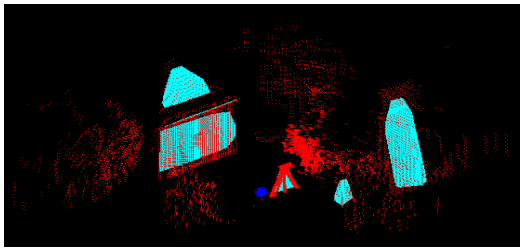

Fig. 1 Extracted landmarks from the first scan. There are totally 6 landmarks. Robot position is shown by star mark.

In the next section, the plane-feature based localization and mapping method details are explained.

3. PLANE-FEATURE BASED S2KF-SLAM

In this section, the plane-feature based S2KF-SLAM method is explained. Firstly, deterministic approximations of N-dimensional standard normal distributions based on S2KF theory and its application to conventional SLAM problem is briefly described. Then the proposed modifications are discussed.

3.1 Probabilistic Plane Fitting

The key point of the S2KF theory (Steinbring and Hanebeck) is based on deterministic Dirac Mixtures which utilize Localized Cumulative Distribution (LCD) described in (Hanebeck et al.). The LCD approach solves the density approximation problem, by turning it to an optimization problem, and systematically minimizing a modified Cramer-von Mises distance between the Dirac Mixture approximation and an arbitrary Gaussian distribution. Although the LCD approach can approximate any random Gaussian distribution, it is computationally expensive and not feasible for online usage. Therefore, offline-generated Dirac Mixture approximations (DMA) stored in a Cache are used. DMA

$$f_{LCD}(\zeta) = \frac{1}{L}\sum_{i=1}^{L}\delta(\zeta - \zeta_i) \approx N(\zeta, I) \tag{7}$$

of a standard normal distribution with L equally weighted and optimally placed samples ζ_i, where L might be chosen randomly. Here, $\hat{\zeta} = \frac{1}{L}\sum_i \zeta_i = 0$ and unit sample covariance

$$P^s = \frac{1}{L}\sum_i (\zeta_i - \hat{\zeta})(\zeta_i - \hat{\zeta})^T = \mathbf{I}_{\mathrm{L}} \tag{8}$$

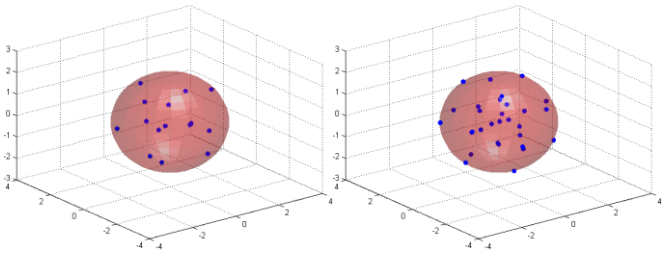

Fig. 2 LCD sampling of a 3D standard normal distribution with 15 (left) and 25 (right) samples.

where $\mathbf{I}_{\mathrm{L}}$ denotes the L-dimensional identity matrix. Fig. 2 shows generated Dirac Mixture approximations for two different numbers of samples in case of a 3D normal distribution. Given an random Gaussian distribution

$$h(y) = N(y, \hat{y}, \mathbf{P}^z) \tag{9}$$

the matrix square root $\sqrt{\mathbf{P}^z}$ of $\mathbf{P}^z$ is computed by Cholesky decomposition. The transformed DMA, which are computed state samples, are obtained by the Mahalanobis transform (Härdle and Simar) as

$$y_i = \sqrt{\mathbf{P}^z}\zeta_i + \hat{y} \qquad \forall i \in \{1,...,L\}. \tag{10}$$

The number of samples is not fixed as in UKF and CKF; thus, the estimation quality can be easily improved by simply increasing the number of employed samples. The DMA generation for any desired number of dimension and sample is carried out offline and can be considered as a core for state estimation. In fact, this issue is problematic in terms of SLAM problem since the state vector constantly augmented whenever a new feature observed. However, we solve this by limiting number of feature in the state vector. The SLAM state vector starts with 6D robot position and the state vector size increases up to 15 planar feature namely 66 dimensions (6 + 15*4). Then a dimension reduction is processed with eliminating the first 10 observed feature. This approach makes the SLAM method forgetful but the experimental studies show very satisfactory results.

3.2 Feature Based SLAM with Nonlinear Filtering

Feature based S2KF-SLAM algorithm uses the infinite plane parameters as features. However, other semantic data information of the planes found in landmark extraction is used when solving the data association problem. Thus, every plane feature is considered as an infinite plane in SLAM posterior; however, in fact they are plane patches in the correspondence decision. First, the vehicle, landmark, and the sensor models are elucidated as follows. Then S2KF-SLAM motion and measurement update steps are explained.

Vehicle Model The vehicle model is given by

$$\mathbf{x}_{v_k} = f(\mathbf{x}_{v_{k-1}}, \mathbf{u}_{k-1}) + w_{k-1} \tag{11}$$

where w_{k-1} denotes the zero mean Gaussian distribution noise vector with the covariance matrix $\mathbf{Q}$, and the control signal or odometry data is provided as $\mathbf{u} = [\delta x, \delta y, \delta z, \delta\alpha, \delta\beta, \delta\gamma]$. The vehicle state vector is represented by $\mathbf{x}_v = [\mathbf{x}_{v_p} \quad \mathbf{x}_{v_o}]$ where $\mathbf{x}_{v_p} = [x, y, z]$ and $\mathbf{x}_{v_o} = [\alpha, \beta, \gamma]$ denote the robot

position and the orientation, respectively. The vehicle model function given by f and can be disclosed as in (11) for the odometry data having the relative rigid body transformation parameters.

$$\begin{bmatrix} x_k \\ y_k \\ z_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ z_{k-1} \end{bmatrix} + Rot(x_{v_{o,k}}) \begin{bmatrix} \delta x_{k-1} \\ \delta y_{k-1} \\ \delta z_{k-1} \end{bmatrix}$$
$$\begin{bmatrix} \alpha_k \\ \beta_k \\ \gamma_k \end{bmatrix} = \begin{bmatrix} \alpha_{k-1} \\ \beta_{k-1} \\ \gamma_{k-1} \end{bmatrix} + \begin{bmatrix} \delta\alpha_{k-1} \\ \delta\beta_{k-1} \\ \delta\gamma_{k-1} \end{bmatrix} \quad (12)$$

where Rot matrix represent the three successive rotations defined by the Euler angles in x, y, and z axes.

Landmark Model The planar landmarks are assumed as stationary $\mathbf{x}_{m_k} = \mathbf{x}_{m_{k+1}}$ and represented in world (W) frame. The SLAM map is augmented with the following state vector representation,

$$\mathbf{x}_{m_k} = \begin{bmatrix} \mathbf{n}_{F_k}^W & d_{F_k}^W \end{bmatrix} \quad (13)$$

$$\mathbf{x}^a{}_{k+1} = \begin{bmatrix} \mathbf{x}_{v_k} & \mathbf{x}_{m_{k+1}} \end{bmatrix} \in R^{6+4N}. \quad (14)$$

Measurement Model Measurement model parameters, z_k are provided by the feature extraction method and stated as

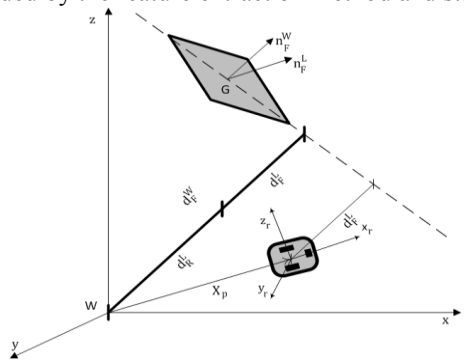

Fig. 3 Infinite Plane representation in local and world frame.

$$z_k = h(\mathbf{x}, \mathbf{u}_{k-1}) + v_{k-1}$$
$$z_k = \begin{bmatrix} n_{F_k}^L \\ d_{F_k}^L \end{bmatrix} = \begin{bmatrix} Rot^T(x_{v_{o,k}}) n_{F_k}^W \\ d_{F_k}^W + (n_{F_k}^W)^T x_{v_{p,k}} \end{bmatrix} + v_{k-1} \quad (15)$$

where n_F^L and d_F^L are the plane normal and its minimum distance to origin in local frame, respectively. In the formulations superscripts, W and L, indicate the World frame and the Local frame, and T represents the matrix transpose. As shown in Fig. 3, the proposed feature extraction method provides the plane properties in local frame, and the plane parameters are transformed to the world frame by

$$n_F^W = Rot(x_{v_o}) n_F^L \quad (16)$$

$$d_F^W = d_F^L - (n_F^W)^T x_{v_p} \quad (17)$$

The other plane properties such as center of gravity of the planes G_F^L, the covariance matrix C_F^L of the plane points and convex hull points $\Delta^L{}_{XYZ,F}$ given by (5) are also transferred to the world frame by using the estimated robot position, x_{v_p}, and orientation, x_{v_o}.

$$C_F^W = Rot(x_{v_o}) C_F^L \quad (18)$$

$$G_F^W = Rot(x_o) G_F^L + x_{v_p} \quad (19)$$

$$\Delta^W{}_{XYZ,F} = Rot(x_{v_o}) \Delta^L{}_{XYZ,F} + x_{v_p} \quad (20)$$

After the measurement update, two associated landmarks are merged again to increase the consistency, and in this case these values are again obtained through the projection and convex hull computation procedures. Motion and measurement updates processes are explained as the following.

Motion Update Motion update step is based on the vehicle model (11) and the state and covariance matrix is augmented as

$$\bar{\mathbf{x}}_{k-1} = \begin{bmatrix} \bar{\mathbf{x}}_{v_{k-1}} & \mathbf{u}_{k-1} \end{bmatrix}$$
$$\mathbf{P}_{k-1} = \begin{bmatrix} \mathbf{P}_{k-1} & 0 \\ 0 & Q \end{bmatrix} \quad (21)$$

where $\bar{\mathbf{x}}_{k-1}$ is the state vector in the (k-1)th time step. $\mathbf{u}_{k-1}$ is the applied control signal at this time. $\mathbf{P}_{k-1}$ denotes the state covariance matrix. Then DMA, ζ, is evaluated with (10), and the square root of the P matrix is obtained with Cholesky decomposition, $\Sigma = chol(\mathrm{P})^{\mathrm{T}}$. The time index ($k$) is reduced for convenience, and state sample points are evaluated by

$$y_i^- = \bar{\mathbf{x}} + \Sigma\zeta_i. \quad (22)$$

Then these points are propagated through the vehicle model as $\mathbf{X}_i = f(y_i^-, \mathbf{u})$.

The estimated state and corresponding state vector is obtained

$$\bar{\mathbf{x}}^- = \frac{1}{2n}\sum_{i=1}^{2n} \mathbf{X}_i$$
$$\mathbf{P} = \frac{1}{2n}\sum_{i=1}^{2n} \mathbf{X}_i \mathbf{X}_i^T - (\bar{\mathbf{x}}^-)(\bar{\mathbf{x}}^-)^T. \quad (23)$$

Measurement update step is applied as follows.

Measurement Update The measurement update step is based on the observation model (15). The DMA points are evaluated by the estimated state and covariance matrix as $y_i^+ = \bar{\mathbf{x}}^- + \sqrt{\mathbf{P}}\zeta_i$, and then they are propagated through the observation model as $Z_i = h(y_i^+, \mathbf{u}_{k-1}) + v_{k-1}$, where v_{k-1} is the zero mean measurement Gaussian noise with $\mathbf{R}$ covariance matrix. The predicted measurement mean vector $\bar{\mathbf{z}}_i^-$ and related covariance matrices, $\mathbf{P}_{zz}$ and $\mathbf{P}_{xz}$ computed as

$$\bar{\mathbf{z}}_i^- = \frac{1}{2n}\sum_{i=1}^{2n} Z_i \quad (24)$$

$$\mathbf{P}_{zz} = \frac{1}{2n}\sum_{i=1}^{2n} Z_i Z_i^T - (\bar{\mathbf{z}}_i^-)(\bar{\mathbf{z}}_i^-)^T + R$$
$$\mathbf{P}_{xz} = \frac{1}{2n}\sum_{i=1}^{2n} X_i Z_i^T - (\bar{\mathbf{x}}^-)(\bar{\mathbf{z}}_i^-)^T \quad (25)$$

The Kalman Gain (**G**), updated state vector and covariance matrix are evaluated by

$$\mathbf{G} = \mathbf{P}_{xz}\mathbf{P}_{zz}^{-1}$$
$$\bar{\mathbf{x}}^+ = \bar{\mathbf{x}}^- + \mathbf{G}(z - \bar{z}) \quad (26)$$
$$\mathbf{P} = \mathbf{P} - \mathbf{G}\mathbf{P}_{zz}\mathbf{G}^T$$

If a new landmark is observed, it is added to the state vector with the state augmentation model.

State Augmentation The state augmentation is based on the landmark model definition (13) and the augmentation given by (14) and applied in two steps. Firstly, the state vector and covariance matrix is augmented with the new observations as follows. The augmented state model $g^a = [\mathbf{x}_v \quad \mathbf{x}_m]$ is constructed, and then the augmented state vector and covariance matrix are computed by following the same procedure in motion update step with (22) - (23).

$$\bar{\mathbf{x}}^+ = [\bar{\mathbf{x}}_v \quad z]$$
$$\mathbf{P} = \begin{bmatrix} \mathbf{P} & 0 \\ 0 & R \end{bmatrix} \quad (27)$$

Data Association The data association is the fundamental and critical part of any SLAM method. The conventional data association methods for point features are based on statistical measurement methods. For the planar features, one cannot use the classical approaches since the features are represented by infinite plane parameters. Therefore, semantic information based on plane properties such as the convex hull points, number of plane points, and covariance matrix of the plane inlier points can be used to decide correspondence decision. In this work, the same criteria proposed in the merging step are used in the data association as well. First, the two conditions which are translations and orientations tests are investigated. Then if these tests are satisfied, the closeness test is checked. In this data association approach, there is no need to store any neighboring cell structure or tree representation.

4. EXPERIMENTAL RESULTS

In the performance analysis of the proposed method, the experimental dataset provided by Oliver Wulf is used (Wulf). This data set was recorded at the Leibniz University Campus where each scan has approximately 20,000 data points. The initial pose estimates are given by xv, yv as position, and θv as orientation in 3D. The ground truth pose data is available in 6D, $x_v = [x_{v_p} \quad x_{v_o}]$. The error norm of the robot position in three dimensions is computed, where the process noise covariance matrix $Q = diag(\sigma_x^2, \varepsilon^2, \sigma_z^2, \varepsilon^2, \sigma_\theta^2, \varepsilon^2)$ is set by $\sigma_x = \sigma_y = 10\,\text{cm}$ $\sigma_\theta = 0.1°$, and $\varepsilon = 10^{-3}$.

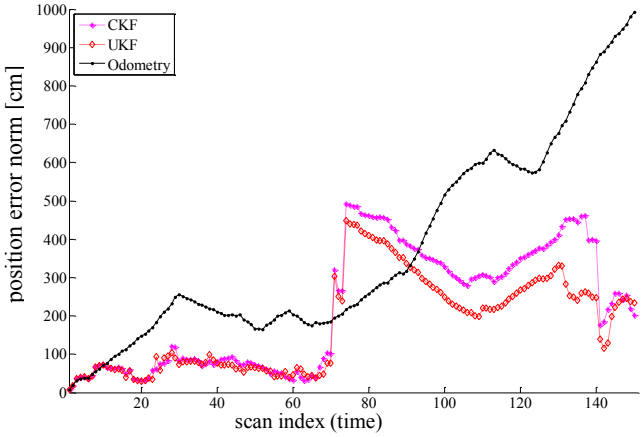

Fig. 4 Error norm of the robot pos. for full-state SLAM representation.

The S2KF does not use full state representation, but first, we would like to present the full state experimental results based on CKF and UKF. In Fig. 4, the error norm of the robot position in three dimensions shows that UKF and CKF provide almost the similar performance for full-state filter representation. Obviously, around the 70th observation (just after loop closing) the filters become instable. On the other hand, under the same conditions, the limited state filter representation provides superior results, especially for UKF and S2KF as shown in Fig. 5.

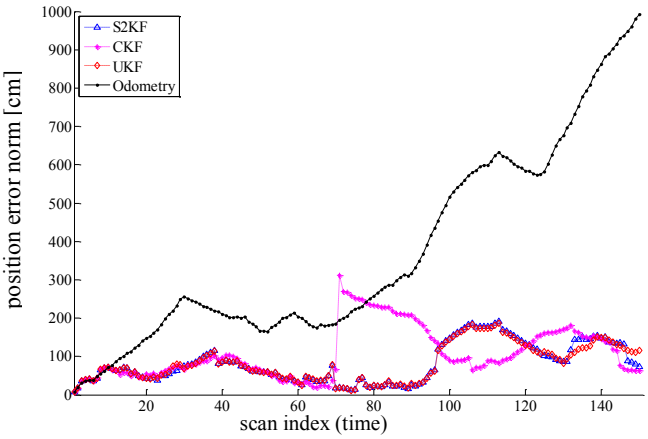

Fig. 5 Error norm of the robot pos. for limited-state SLAM representation.

Moreover, an advanced limited filter representation, which keeps the first 5 features in the state vector constantly and the whole state vector for 10 features, is proposed. Namely, the filter dimension is reduced to 10 features whenever it reaches above 15 features. The error norm of the robot positions are given in Fig. 6. The results show that the S2KF performs better than the other two filters. Besides the given results, in our studies, we observed that the S2KF is more stable than UKF and CKF.

The estimated map of the environment is given in Fig. 7, where the planar features (patches), sparse point cloud (scatter points), true robot positions (dot), and the robot position error ellipsoids on its mean are shown. One has to note that the uncertainty ellipsoids are getting smaller when the loop is closed.

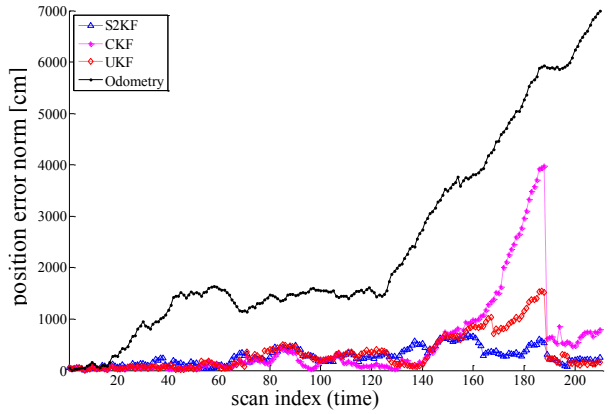

Fig. 6 Error norm of the robot pos. for second type limited-state SLAM representation.

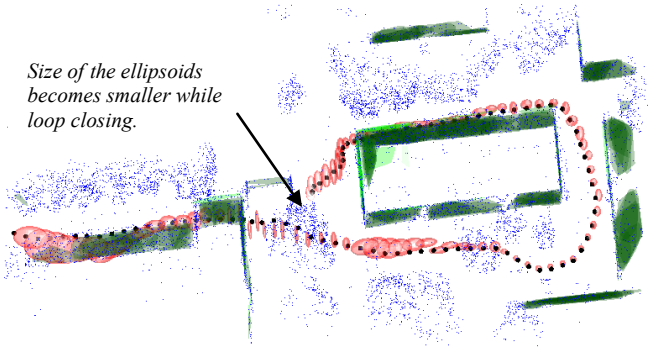

Fig. 7 Estimated planar map of the environment.

5. CONCLUSIONS

In this study, a plane-landmark based S2KF-SLAM is proposed. The S2KF-SLAM performance is compared to the CKF and UKF-SLAM methods. The experimental results show that S2KF-SLAM performance is better than UKF and CKF and can be used in 3D outdoor SLAM successfully.

REFERENCES

ARASARATNAM, I. & HAYKIN, S. 2009. Cubature Kalman Filters. *Automatic Control, IEEE Transactions on,* 54**,** 1254-1269.

BORRMANN, D., ELSEBERG, J., LINGEMANN, K., NUCHTER, A. & HERTZBERG, J. 2008. Globally consistent 3D mapping with scan matching. *Robot. Auton. Syst.,* 56**,** 130-142.

CASTELLANOS, J. A., MONTIEL, J. M. M., NEIRA, J. & TARDOS, J. D. 1999. The SPmap: a probabilistic framework for simultaneous localization and map building. *Robotics and Automation, IEEE Transactions on,* 15**,** 948-952.

HANEBECK, U. D., HUBER, M. F. & KLUMPP, V. Dirac mixture approximation of multivariate gaussian densities. Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on, 2009. IEEE, 3851-3858.

HÄRDLE, W. & SIMAR, L. 2007. *Applied multivariate statistical analysis*, Springer.

MARTINEZ-CANTIN, R. & CASTELLANOS, J. A. Unscented SLAM for large-scale outdoor environments. Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on, 2-6 Aug. 2005 2005. 3427-3432.

NIETO, J., GUIVANT, J. & NEBOT, E. FastSLAM: Real Time Implementation in Outdoor Environments. Australasian Conference on Robotics and Automation, 2002 Auckland.

PAKKI, K., CHANDRA, B., GU, D. & POSTLETHWAITE, I. Cubature Kalman Filter based Localization and Mapping. IFAC WC, 2011 2011. 2121-2125.

PATHAK, K., BIRK, A., VASKEVICIUS, N., PFINGSTHORN, M., SCHWERTFEGER, S. & POPPINGA, J. 2010. Online three-dimensional SLAM by registration of large planar surface segments and closed-form pose-graph relaxation. *J. Field Robot.,* 27**,** 52-84.

S. THRUN, W. B., AND D. FOX 2005. *Probabilistic Robotics*, MIT Press.

STEINBRING, J. & HANEBECK, U. D. S 2 KF: The Smart Sampling Kalman Filter. Information Fusion (FUSION), 2013 16th International Conference on, 2013. IEEE, 2089-2096.

ULAS, C. & TEMELTAS, H. A fast and robust scan matching algorithm based on ML-NDT and feature extraction. Mechatronics and Automation (ICMA), 2011 International Conference on, 7-10 Aug. 2011 2011. 1751-1756.

WEINGARTEN, J. & SIEGWART, R. EKF-based 3D SLAM for structured environment reconstruction. Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on, 2-6 Aug. 2005 2005. 3834-3839.

WEINGARTEN, J. W., GRUENER, G. & SIEGWART, R. Probabilistic plane fitting in 3D and an application to robotic mapping. Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on, 26 April-1 May 2004 2004. 927-932 Vol.1.

WULF, L. & WAGNER, B. Fast 3D Scanning Methods for Laser Measurement Systems. Proceedings of the International Conference on Control Systems and Computer Science, 2003. 312-317.

WULF, O. 2010. *Hannover, Leibniz University Campus* [Online]. Available: http://kos.informatik.uni-osnabrueck.de/3Dscans/.

YANGMING, L. & OLSON, E. B. Extracting general-purpose features from LIDAR data. Robotics and Automation (ICRA), 2010 IEEE International Conference on, 3-7 May 2010 2010. 1388-1393.