Automatic Building Exterior Mapping Using Multilayer Feature Graphs

Yan Lu, Dezhen Song, Yiliang Xu, A. G. Amitha Perera, and Sangmin Oh

***Abstract*— We develop algorithms that can assist robot to perform building exterior mapping, which is important for building energy retrofitting. In this task, a robot needs to identify building facades in its localization and mapping process, which in turn can be used to assist robot navigation. Existing localization and mapping algorithms rely on low level features such as point clouds and line segments and cannot be directly applied to our task. We attack this problem by employing a multiple layer feature graph (MFG), which contains five different features ranging from raw key points to planes and vanishing points in 3D, in an extended Kalman filter (EKF) framework. We analyze how errors are generated and propagated in the MFG construction process, and then apply MFG data as observations for the EKF to map building facades. We have implemented and tested our MFG-EKF method at three different sites. Experimental results show that building facades are successfully constructed in modern urban environments with mean relative errors of plane depth less than 4.66%.**

I. Introduction

Our group is developing vision algorithms to assist building exterior survey using a mobile robot. This step can greatly assist building energy retrofitting. The task requires a robot to map building facades with its on-board camera when the robot travels. However, existing navigation methods often utilize low level landmarks, such as feature points and point clouds, and cannot directly provide information for build facades, which can be viewed as high level landmarks. Actually, the high level landmarks, such as primary planes and salient lines, have distinctive advantages over low level features. Bearing clear geometric meaning, high level landmarks are less sensitive to different lighting conditions and varying shadows where low level features are often challenged. High level landmarks are ubiquitous in modern urban areas where rectilinear objects dominate camera field of view. Humans are used to navigating in unknown environments by effectively using high level landmarks as reference. However, robots still have difficulty to utilize advantages of high level landmarks due to challenges in feature recognition and correspondence.

In 2012, we proposed a two-view based multilayer feature graph (MFG) as a scene understanding and knowledge representation method for robot navigation [1]. An MFG is constructed from overlapping and dislocated two views and contains five different features ranging from raw key points to planes and vanishing points in 3D. Here we build our high level landmark-based maps (see Fig. 1) by employing MFG as observations in an extended Kalman filter (EKF) framework. We analyze how errors are generated and propagated in the MFG construction process, which characterizes observation errors in the EKF. We derive closed form solutions for error distributions. Based on projective geometry, we derive the observation models to complete the EKF framework. We have implemented and tested our MFG-EKF method at three different sites. Experimental results show that high level landmarks are successfully constructed in modern urban environments with mean relative plane depth errors less than 4.66%.

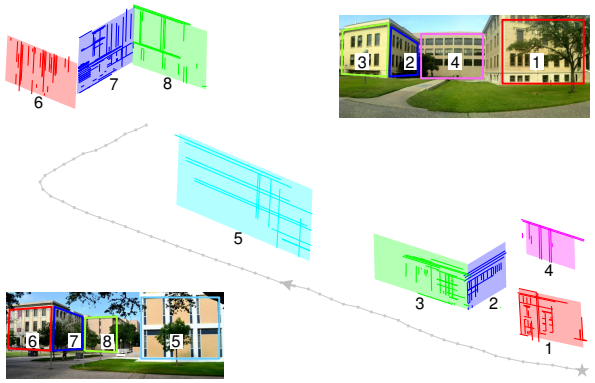


Fig. 1. A sample output of high level landmarks and the robot trajectory after the mapping process in 3D view. The system is able to recognize primary planes from building facades and their corresponding co-planar lines as high level landmarks. The numbered corresponding building facades are also color coded in the top right and bottom left images.

II. Related Work

Robotic mapping with high level landmarks relates to a broad body of research in simultaneous localization and mapping (SLAM) and visual odometry including different sensor configurations and different landmark selections.

Depending on costs, payload limitation, and navigation environments, the most common sensors for robot navigation include sonar arrays [2], laser range finders [3], [4], depth cameras [5], regular cameras [6]–[10], or their combinations [11], [12]. Mapping tasks are often conducted under the SLAM framework [13]. As a partially observable Markovian decision process, SLAM infers system states based on the sensory input using different filters and loop closure techniques. The system states usually include both landmarks and robot states whereas landmarks are the representation of the physical world. For example, landmarks are point clouds if a laser ranger finder or a depth camera is the primary sensor. In

Y. Lu, and D. Song are with Department of Computer Science and Engineering, Texas A&M University, College Station, TX 77843, USA. Emails: {*ylu, dzsong*}*@cse.tamu.edu*.

Y. Xu, A. G. A. Perera, and S. Oh are with Kitware, Inc., 28 Corporate Drive, Clifton Park, NY 12065, USA. Emails: {*yiliang.xu, amitha.perera, sangmin.oh*}*@kitware.com*.

vision-based SLAM, SIFT feature points or its variants [6], [7], [14], [15] and line features [8]–[10] are often employed as landmarks.

Our work belongs to the vision-based SLAM category where one or more cameras are the primary navigation sensor. Recently, many researchers realize landmark selections can make a difference in SLAM and visual odometry performance. Lower level landmarks [14], such as Harris corners and SIFT points, are relatively easy to use due to their geometric simplicity, which share many geometric properties with traditional point clouds used for laser range finders. However, point features are merely mathematical singularities in color, texture, and geometric space. They can be easily influenced by lighting and shadow conditions. Realizing the limitation, recent efforts focus on developing high level landmarks such as lines/edges/line segments [8], [9], [16]. Zhang et al. [17] use vertical lines and floor lines in a monocular SLAM and build a 3D line-based map in an indoor corridor environment.

More recent sophisticated methods combine multiple features such as points, lines, and planes. Gee et al. [18] incorporate 3D planes and lines into visual SLAM framework. Martinez et al. [19] propose a monocular SLAM algorithm that unifies the estimation of point and planar features. These works have demonstrated the robustness of high level landmarks and inspired this work. Observe that the existing works only treat different landmarks as isolated geometric objects, without exploring the inner relationship between them. The treatment simplifies the SLAM problem formulation but cannot fully utilize the power of high level landmarks.

Our build exterior work is built on visual navigation and mapping development over the past decade. We have developed appearance-based methods [20], investigated how depth error affects navigation [21], studied mirrored surface detection [22], and used vertical line segments for visual odometry tasks [23], [24]. In the process, we have learned that it is necessary to combine the benefits of different features to assist navigation. This leads to the MFG development [1] which captures four types of geometric relationships including adjacency, collinearity, coplanarity, and parallelism. Here we present our latest results on how to combine MFG in an EKF framework to utilize the feature relationships in the mapping process.

III. Problem Definition

A robot equipped with a single camera navigates in an unknown environment. The robot attempts to estimate high level landmarks such as building facades or salient edges from input image frames. The basic assumptions are,

- **a.1** The robot operates in a largely static modern urban environment with rectilinear structures, which is the prerequisite for MFG.
- **a.2** The onboard camera is pre-calibrated and has a known intrinsic matrix K.
- **a.3** The initial step of robot movement is known for reference. Otherwise, estimates would be up to scale.

In our approach, adjacent raw image frame pairs are first employed to construct MFG [1] sequence. Let I_k be the k-th ($k \in \mathbb{N}$) image frame and $\mathcal{M}_k$ ($k \geq 1$) be the MFG constructed from frames I_k and I_{k-1}. Fig. 2 illustrates that MFG is a data structure composed of five layers of feature nodes: *key points*, *line segments*, *ideal lines*, *primary planes* and *vanishing points*; edges between nodes of different layers represent geometric relationships including *adjacency*, *collinearity*, *coplanarity*, and *parallelism*.

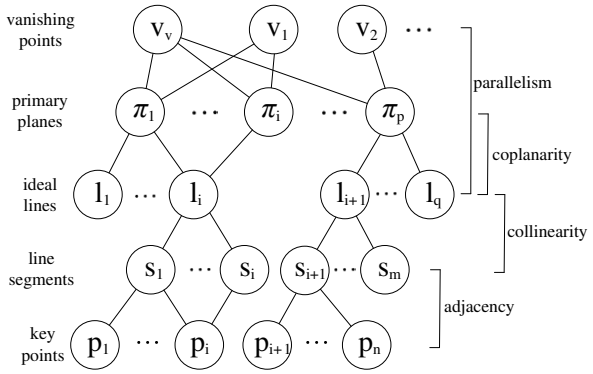


Fig. 2. The structure of an MFG from [1].

The resulting MFG sequence, $\{\mathcal{M}_k, k \geq 1\}$, is considered as the input to the problem. Denote $\{C_k\}$ the camera coordinate system (CCS) associated with I_k. MFGs assist us in identifying high level landmarks such as 3D planes and their associated coplanar lines in physical space. However, the planes and lines from $\mathcal{M}_k$ are represented w.r.t. $\{C_k\}$, which cannot be directly used as global landmarks. Define the world coordinate system (WCS), $\{W\}$, to coincide with $\{C_0\}$. Now we are ready to define our problem.

Problem 1: Given MFG sequence $\{\mathcal{M}_k : k \geq 1, k \in \mathbb{N}\}$, map high level landmarks including 3D planes and coplanar lines in $\{W\}$, and assess the uncertainty of the mapping process by deriving error covariance matrices for each landmark.

To solve the landmark mapping problem, we employ an EKF-based approach. In this approach, MFG $\mathcal{M}_k$ can be considered as a generalized observation at time k. Therefore, we need to understand how errors are distributed in the construction process of $\mathcal{M}_k$, which can serve as the observation error in the EKF. With the observation error derived, the landmark errors can be estimated by combining process errors using the EKF. Therefore, the problem is solved in two steps with the first step being the uncertainty analysis of MFG.

IV. Observation Error: Uncertainty in MFG

Our previous work [1] has shown how to construct MFG using a feature fusion method. However, the uncertainty of each feature layer is yet to be analyzed. Here we detail the uncertainty for each layer of MFG in a bottom-up manner.

A. Error Modeling of Raw Features

The MFG construction algorithm takes two images I and I' as input and outputs a feature graph of five layers,

as illustrated in Fig. 2. In MFGs, key points and line segments are raw features directly detected from images using SIFT [25] and LSD [26], while ideal lines, primary planes and vanishing points represent high level features constructed from raw features. MFGs also include feature correspondences between two views.

Note that I and I' actually represent I_k and I_{k-1} in the continuous image sequence, respectively. Here we drop k and $k-1$ from notations for simplicity. Furthermore, we attach a superscript $'$ to variables associated with I'. As a convention, we use a $\sim$ on top of a homogeneous vector to denote its inhomogeneous counterpart throughout this paper.

For each key point $\mathbf{p}_i$ in I, we model its measurement error as an independent and identically distributed (i.i.d.) zero-mean isotropic Gaussian noise with variance σ^2 in each axis

$$\mathrm{Cov}(\tilde{\mathbf{p}}_i) = \sigma^2 \mathrm{I}_2, \quad \forall i \tag{1}$$

where I_2 is a 2×2 identity matrix.

For each line segment $\mathbf{s}_i$ in I, denote its two endpoints by $\mathbf{e}_{i1}$ and $\mathbf{e}_{i2}$. Define $\mathbf{u}_{i\parallel}$ and $\mathbf{u}_{i\perp}$ to be two unit vectors parallel and perpendicular to the line segment, respectively (see Fig. 3). We model the error of $\mathbf{e}_{i1}$ (the same for $\mathbf{e}_{i2}$)

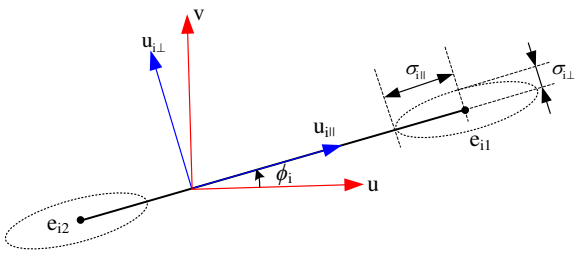


Fig. 3. Uncertainty of line segment endpoints.

as an independent 2D Gaussian with its covariance matrix to be diagonal in the coordinate system defined by $\mathbf{u}_{i\parallel}$ and $\mathbf{u}_{i\perp}$ as below

$$\Sigma_{i\parallel\perp} = \begin{bmatrix} \sigma_{i\parallel}^2 & 0 \\ 0 & \sigma_{i\perp}^2 \end{bmatrix}, \tag{2}$$

where $\sigma_{i\perp}$ and $\sigma_{i\parallel}$ are the standard deviations of $\mathbf{e}_{i1}$ in directions of $\mathbf{u}_{i\perp}$ and $\mathbf{u}_{i\parallel}$, respectively. $\sigma_{i\perp}$ is usually much smaller than $\sigma_{i\parallel}$. We have observed that $\sigma_{i\perp}$ usually is inversely correlated to the line segment length. Furthermore, $\sigma_{i\perp}$ also has a lower bound of σ_p due to pixelization error. Thus, we model the endpoint error as follows,

$$\sigma_{i\parallel} = \sigma_{\parallel}, \qquad \sigma_{i\perp} = \frac{\sigma_{i\parallel}}{\|\mathbf{s}_i\|} + \sigma_p, \quad \forall i, \tag{3}$$

where $\sigma_{\parallel}$ and σ_p are constant and independent of i, and $\|\mathbf{s}_i\|$ denotes the length of $\mathbf{s}_i$. The parameters for the models can be determined using Monte Carlo simulation. Projecting (2) back to the image coordinate system (ICS), we have

$$\mathrm{Cov}(\tilde{\mathbf{e}}_{i1}) = R(\phi_i)\Sigma_{i\parallel\perp}R(\phi_i)^{\mathsf{T}} \tag{4}$$

where ϕ_i is the angle between $\mathbf{u}_{i\parallel}$ (see Fig. 3) and u-axis, and $R(\phi_i) = \begin{bmatrix} \cos\phi_i & -\sin\phi_i \\ \sin\phi_i & \cos\phi_i \end{bmatrix}$. Note that the error model in (2-4) for line segments may differ for different line detectors. However, the rest of our analysis still applies.

With error distributions of raw features obtained, we are ready to analyze high level features such as ideal lines and primary planes.

B. Error Analysis of Ideal Lines

In the MFG construction process, an ideal line $\mathbf{l}_i$ is obtained by fitting a straight line through endpoints of a set of m_i collinear line segments $\{\mathbf{s}_j : 1 \le j \le m_i\}$. The i-th ideal line in I can be parameterized in terms of angle θ_i and intercept ρ_i with the following homogeneous format in ICS,

$$\mathbf{l}_i = [\cos\theta_i, \sin\theta_i, \rho_i]^{\mathsf{T}} \tag{5}$$

such that $u\cos\theta_i + v\sin\theta_i + \rho_i = 0$ holds for any point (u, v) on $\mathbf{l}_i$. Since the fitting process employs maximum likelihood estimation (MLE) to obtain optimal solution $[\theta_i^*, \rho_i^*]^{\mathsf{T}}$, we have the following lemma.

Lemma 1: Given collinear line segments set $\{\mathbf{s}_j\}$ with their endpoint covariance matrices in (4), if MLE is employed to estimate $[\theta_i^*, \rho_i^*]^{\mathsf{T}}$, the resulting $\mathbf{l}_i$ can be approximated by a Gaussian with a mean vector of $[\cos\theta_i^*, \sin\theta_i^*, \rho_i^*]^{\mathsf{T}}$ and a covariance matrix of,

$$\mathrm{Cov}(\mathbf{l}_i) = J\,\mathrm{Cov}(\theta_i^*, \rho_i^*)\,J^{\mathsf{T}}, \tag{6}$$

where $J = \begin{bmatrix} -\sin\theta_i^* & \cos\theta_i^* & 0 \\ 0 & 0 & 1 \end{bmatrix}^{\mathsf{T}}$ and $\mathrm{Cov}(\theta_i^*, \rho_i^*)$ is given by (21) in the online technical report [27].

Proof: See Appendix I in [27]. ■

C. Error Analysis of Primary Planes

In an MFG based on two views, a primary plane $\boldsymbol{\pi}_i$ is a 3D plane represented by a 4D homogeneous vector in the CCS associated with I. Furthermore, if $\boldsymbol{\pi}_i$ does not pass through the camera center (which is often the case in practice), we can have

$$\boldsymbol{\pi}_i = [\tilde{\boldsymbol{\pi}}_i^{\mathsf{T}}, 1]^{\mathsf{T}}, \tag{7}$$

where $\tilde{\boldsymbol{\pi}}_i$ is a 3×1 vector for the inhomogeneous representation of $\boldsymbol{\pi}_i$.

Based on the coplanarity relationship in an MFG, each plane $\boldsymbol{\pi}_i$ can be associated with p_i coplanar point correspondences $\{\mathbf{p}_{ij} \leftrightarrow \mathbf{p}'_{ij} : j = 1, \cdots, p_i\}$, and q_i coplanar line correspondences $\{\mathbf{l}_{i\kappa} \leftrightarrow \mathbf{l}'_{i\kappa} : \kappa = 1, \cdots, q_i\}$. These feature correspondences satisfy a homography induced by $\boldsymbol{\pi}_i$

$$\mathbf{p}'_{ij} = \mathrm{H}_i\mathbf{p}_{ij}\,, \quad \mathbf{l}_{i\kappa} = \mathrm{H}_i^{\mathsf{T}}\mathbf{l}'_{i\kappa}, \tag{8}$$

where

$$\mathrm{H}_i = K(\mathrm{R} - \mathbf{t}\tilde{\boldsymbol{\pi}}_i^{\mathsf{T}})K^{-1}, \tag{9}$$

and R and $\mathbf{t}$ are the rotation matrix and translation vector between two views, respectively.

Eqs. (8 and 9) suggest a method of computing $\tilde{\boldsymbol{\pi}}_i$ based on R and $\mathbf{t}$. However, if R and $\mathbf{t}$ are simply derived from epipolar geometry without considering the planar structure information, the solution is not optimal, and neither is $\tilde{\boldsymbol{\pi}}_i$. Inspired by the method from [28], we estimate all $\tilde{\boldsymbol{\pi}}_i$'s, R and $\mathbf{t}$ simultaneously by employing all geometric features

(i.e., key points and ideal lines) and constraints (i.e., epipolar constraint and homography) under an MLE framework. Define $\Theta_{P1} = [\tilde{\boldsymbol{\pi}}_1^\mathsf{T}, \cdots, \tilde{\boldsymbol{\pi}}_i^\mathsf{T}, \cdots]^\mathsf{T}$. Supposing Θ_{P1}^* is the MLE output of Θ_{P1}, we have the following lemma.

Lemma 2: Given that key point errors follow i.i.d. isotropic Gaussian distributions with covariance matrices in (1) and line segment endpoints follow independent Gaussian distributions with covariance matrices in (4), if MLE is employed to estimate all $\tilde{\boldsymbol{\pi}}_i$'s for primary planes, then the distribution of each $\tilde{\boldsymbol{\pi}}_i$ can be approximated by a Gaussian distribution with the following mean and covariance matrix,

$$\tilde{\boldsymbol{\pi}}_i^* = T_i\, \Theta_{P1}^* \tag{10}$$

$$\mathrm{Cov}(\tilde{\boldsymbol{\pi}}_i^*) = T_i\, \mathrm{Cov}(\Theta_{P1}^*)\, T_i^\mathsf{T} \tag{11}$$

where $T_i = [\mathbf{0}_{3,3i-3} : \mathrm{I}_3 : \mathbf{0}_{3,3(p-i)+6}]$, $\mathrm{Cov}(\Theta_{P1}^*)$ is derived in a way similar to that in (21) of [27], $\mathbf{0}_{a,b}$ is an $a \times b$ zero matrix, and I_3 is a 3×3 identity matrix.

Proof: See Appendix II in [27]. ■

V. EKF BASED MAPPING WITH MFG

Two-view based MFGs only provide local information of high level features. In order to build a global map in $\{W\}$, EKF is employed to estimate the posterior of landmarks as well as a robot trajectory.

A. System State Representation

In the EKF framework, we maintain and keep updating a system state $\mathbf{y}_k$, which is composed of the robot state $\mathbf{x}_k$ and 3D landmarks.

The robot state is defined as

$$\mathbf{x}_k = [\mathbf{r}_k^\mathsf{T}, \mathbf{q}_k^\mathsf{T}, \boldsymbol{\nu}_k^\mathsf{T}, \boldsymbol{\omega}_k^\mathsf{T}]^\mathsf{T}, \tag{12}$$

where $\mathbf{r}_k$ is a 3D location in $\{W\}$, $\mathbf{q}_k$ is an orientation quaternion w.r.t. $\{W\}$, $\boldsymbol{\nu}_k$ is a velocity vector in $\{W\}$, and $\boldsymbol{\omega}_k$ is an angular velocity vector in $\{C_k\}$.

In $\mathbf{y}_k$, we use $\tilde{\boldsymbol{\pi}}_i^W$ to represent the i-th 3D plane landmark in $\{W\}$. To represent a 3D line, general methods like Plücker coordinates would need as many as 6 parameters. However, a 3D vector is sufficient in this work since our method is only interested in coplanar lines associated with landmark planes. Supposing a landmark line resides on plane $\tilde{\boldsymbol{\pi}}_i^W$, then there exists an one-to-one mapping between this line and its projection on image plane I_0, which is actually a 2D homography induced by $\tilde{\boldsymbol{\pi}}_i^W$. Let us denote $\mathbf{l}_j^k$ the projection of the j-th landmark line on I_k. Then, we can use $\mathbf{l}_j^0$ to fully represent the j-th landmark line in $\mathbf{y}_k$ since we already have $\tilde{\boldsymbol{\pi}}_i^W$ in $\mathbf{y}_k$.

As a result, the complete system state can be written as

$$\mathbf{y}_k = \left[\mathbf{x}_k^\mathsf{T}, \cdots, (\tilde{\boldsymbol{\pi}}_i^W)^\mathsf{T}, \cdots, (\mathbf{l}_j^0)^\mathsf{T}, \cdots\right]^\mathsf{T}. \tag{13}$$

B. EKF Formulation

In an EKF framework, a process model and an observation model need to be specified for the prediction and update steps, respectively.

1) Process Modeling: We follow the conventional assumptions of "constant velocity, constant angular velocity" in [14] for camera motion to formulate the process model as follows,

$$\mathbf{x}_{k+1} = \begin{bmatrix} \mathbf{r}_{k+1} \\ \mathbf{q}_{k+1} \\ \boldsymbol{\nu}_{k+1} \\ \boldsymbol{\omega}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_k + \boldsymbol{\nu}_k \Delta t \\ \mathbf{q}_k \times \mathbf{q}(\boldsymbol{\omega}_k \Delta t) \\ \boldsymbol{\nu}_k \\ \boldsymbol{\omega}_k \end{bmatrix},$$

where $\mathbf{q}(\boldsymbol{\omega}_k \Delta t)$ denotes the quaternion defined by the angle-axis rotation vector $\boldsymbol{\omega}_k \Delta t$, and Δt is the time interval between two steps. Note this is just a partial model for the system state in (13) while the rest of states of $\mathbf{y}_k$ are landmark states. Since landmarks are assumed to be static, their corresponding states remain unchanged in the prediction step.

2) Observation Modeling: An observation function maps the system state to landmark observations. For a plane landmark $\tilde{\boldsymbol{\pi}}_i^W$, the observation produced by $\mathcal{M}_k$ is its representation $\tilde{\boldsymbol{\pi}}_i^k$ in $\{C_k\}$. Define a matrix ${}_k^W T$ that transforms a 3D point (of homogeneous format) from $\{C_k\}$ to $\{W\}$ as

$${}_k^W T = \begin{bmatrix} R(\mathbf{q}_k) & \mathbf{r}_k \\ 0 & 1 \end{bmatrix}_{4\times 4}, \tag{14}$$

where $R(\mathbf{q}_k)$ represents the rotation matrix defined by $\mathbf{q}_k$. For a primary plane landmark $\boldsymbol{\pi}_i^k$, we know that $\boldsymbol{\pi}_i^k = {}_k^W T^\mathsf{T} \boldsymbol{\pi}_i^W$. This implies the observation to be

$$\tilde{\boldsymbol{\pi}}_i^k = \frac{\left({}_k^W T^\mathsf{T} \boldsymbol{\pi}_i^W\right)_{1:3}}{\left({}_k^W T^\mathsf{T} \boldsymbol{\pi}_i^W\right)_4} \tag{15}$$

where $(V)_a$ denotes the a-th element of vector V, and $(V)_{a:b}$ denotes the sub vector of V indexed from a to b.

For a line landmark $\mathbf{l}_j^0$, its observation from $\mathcal{M}_k$ is $\mathbf{l}_j^k$. Supposing $\mathbf{l}_j^0$ lies on plane $\tilde{\boldsymbol{\pi}}_i^W$, $\mathbf{l}_j^k$ can be computed from $\mathbf{l}_j^0$ via a homography [29]

$$\mathbf{l}_j^k = \frac{(\mathrm{H}_i^k)^{-\mathsf{T}} \mathbf{l}_j^0}{\left\| \left((\mathrm{H}_i^k)^{-\mathsf{T}} \mathbf{l}_j^0\right)_{1:2} \right\|} \tag{16}$$

where $\mathrm{H}_i^k = K \left[R^{-1}(\mathbf{q}_k) + R^{-1}(\mathbf{q}_k) \mathbf{r}_k \left(\tilde{\boldsymbol{\pi}}_i^W\right)^\mathsf{T} \right] K^{-1}$ and $\|\cdot\|$ is L^2 norm.

Eqs. (15 and 16) fully determine the observation function. It is worth noting that the covariance matrices of observation noise have been presented in Lems. 1 and 2. EKF also provides covariance of landmarks in its covariance update and prediction steps. Since this is a standard EKF procedure, we skip details here.

C. Landmark Initialization and Management

Since two views are needed to establish an MFG, the system should start at $k = 1$ when $\mathcal{M}_1$ is constructed and landmark planes and lines are added to $\mathbf{y}_1$. Starting from $\mathbf{y}_1$, the system enters the prediction and update loops. As the robot travels farther, new landmarks may be discovered and added to the system state. Because the MFG output is the landmark representation in the current CCS, it needs to be transformed to $\{W\}$ before augmenting the system state.

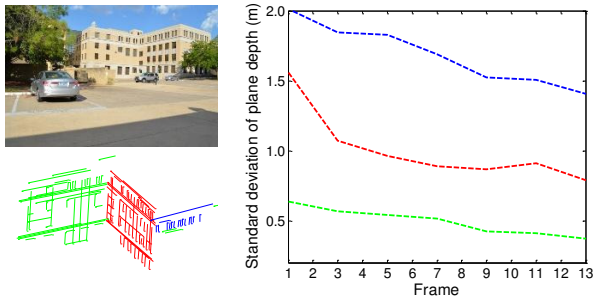
This coordinate transformation is represented by ${}^W_k T$ or the inverse of H^k_i as shown in (14-16).

VI. Experiments

We have implemented the proposed method using Matlab on a Desktop PC. The camera used in the experiment is a pre-calibrated Nikon D5100 camera equipped with a 18 mm lens, which ensures a horizontal field of view of 60°. Images are down-sampled to a resolution of 800×530 pixels. We have conducted two experiments: uncertainty test and field mapping test.

A. Uncertainty Test

The purpose of this experiment is to verify how the estimation uncertainty of landmarks changes as more images entering our system. A sequence of 14 images has been taken while the camera was carried by a person walking towards a building. The starting point is around 40 meters away from the building. Images have been captured every $1 \sim 2$ meters approximately with the first step length known to be 1.5 meters.



(a) (b)

Fig. 4. (a) A sample view (upper) and constructed 3D landmarks (lower), (b) Standard deviations of plane depth vs #frames.

The upper image in Fig. 4(a) shows a sample of the image sequence and the lower line drawing in Fig. 4(a) shows the 3D landmarks constructed from the image sequence. Each plane and its coplanar line segments are coded in the same color. Fig. 4(b) demonstrates that the standard deviation of the depth of each landmark plane (using the same color coding as that in the lower line drawing in Fig. 4(a)) decreases as the frame number increases.

B. Field Mapping Test

Fig. 5. Experiment sites.

In the second experiment, we have tested our method in the field including three sites on Texas A&M University campus as shown in Fig. 5. At each site, the camera follows a pre-defined route. Images are taken every 4 meters approximately while the first step has been known to be exact 4.0 meters as a reference. The distance traveled and the numbers of frames collected at each site are shown in columns 2 and 3 of Tab. I. As shown in the table, our method was able to successfully recognize high level landmarks including primary planes (col. 3) and their coplanar line segments (col. 4). Fig. 1 actually shows a 3D visualization of the map of high level landmarks constructed from data of site 1, where coplanar lines are color coded according to underlying planes.

We employ three error metrics to assess landmark mapping accuracy. ε_d and ε_a are defined for evaluating planes, and ε_L is defined for assessing lines. Suppose plane $\tilde{\boldsymbol{\pi}}^W_i$ is introduced to the map since the k_i-th frame I_{k_i}. Let d_i denote the true plane depth of $\tilde{\boldsymbol{\pi}}^W_i$ in $\{C_{k_i}\}$ obtained using a BOSCH GLR225 laser distance measurer with a range up to 70 m and measurement accuracy of ± 1.5 mm. Define $\hat{d}_i$ as the estimated value of d_i from EKF output. Then a relative metric for plane depth error is defined as

$$\varepsilon_d = \frac{1}{N} \sum_{i=1}^{N} \frac{\|d_i - \hat{d}_i\|}{d_i}, \tag{17}$$

where N is the number of landmark planes. Similarly, define ε_a to be the angular error metric for plane normal. It is worth noting that there exists global drifting error between $\{C_{k_i}\}$ and $\{W\}$, which will be addressed in future loop closure stage. Here we focus on ε_d and ε_a after the plane landmark appears in the camera field of view.

To evaluate a line landmark's estimation accuracy, we consider a re-projection error in ICS. Suppose $\mathbf{l}^0_j$ is added to the map since the k_j-th frame. Let $\hat{\mathbf{l}}^k_j$ be the re-projection of $\mathbf{l}^0_j$ in I_{k_j}, and $\mathbf{e}^{(j)}_h$ be the h-th observed endpoint of line segment in I_{k_j} associated with $\hat{\mathbf{l}}^k_j$. Then the error metric for lines is defined based on the distance between observed line segment endpoints and re-projected line in local frame:

$$\varepsilon_L = \frac{1}{M} \sum_{j=1}^{M} \frac{1}{N_j} \sum_{h=1}^{N_j} d_\perp(\mathbf{e}^{(j)}_h, \hat{\mathbf{l}}^k_j), \tag{18}$$

where $d_\perp(\cdot)$ represents the distance from a point to a line, M is the number of line landmarks and N_j is the number of line segment endpoints associated with $\hat{\mathbf{l}}^k_j$.

Tab. I shows the mapping results based on the three metrics. It is clear that our method successfully maps the high level landmarks. However, since loop closure has not been performed, the estimated camera trajectory inevitably suffers from drifting error, which will be addressed in the future work.

VII. Conclusion and Future Work

We developed a method to allow a mobile robot to perform mapping of building facades by enabling high level landmark mapping. The method incorporated a multiple layer feature graph (MFG) into an EKF framework. We analyzed how errors are generated and propagated in the MFG construction

TABLE I

FIELD MAPPING TEST RESULTS.

site	dist. (m)	#frames	#planes	#lines	ε_d (%)		ε_a (°)		ε_L (pixel)	
					mean	std. dev.	mean	std. dev.	mean	std. dev.
1	216	55	8	197	3.48	2.91	1.77	2.34	0.52	0.26
2	156	40	6	231	4.66	3.27	0.83	3.75	0.39	0.22
3	180	36	7	225	4.09	3.96	1.65	3.09	0.47	0.31

process, which are used as observation error models in the EKF. We derived closed form solutions for error distribution to quantify the observation errors. Based on projective geometry, we derived observation models to complete the EKF framework. We implemented and tested the system at three different sites. Experiment results have shown that high level landmarks are successfully constructed in a modern urban environment with mean relative plane depth error less than 4.66%.

In future, we will develop a new loop closure algorithm by utilizing feature relationships in MFG to deal with the drifting issue and improve system robustness. We will incorporate the MFG framework with different sensors such as depth cameras and IMU to further increase system performance. We also plan to investigate how to utilize GPU to accelerate the computation process. More large scale experiments for both indoor and outdoor applications are planned as well.

ACKNOWLEDGMENT

Special thanks to H. Ge for his assistance with experimental data collection. Thanks are also due to C. Kim, W. Li, M. Hielsberg, J. Lee, and X. Liu for their inputs and contributions to the Networked Robots Lab at Texas A&M University.

REFERENCES

[1] H. Li, D. Song, Y. Lu, and J. Liu, "A two-view based multilayer feature graph for robot navigation," in *Robotics and Automation, 2012. ICRA'12. IEEE International Conference on*. St. Paul, MN, USA: IEEE, May 2012, pp. 3580–3587.

[2] A. Elfes, "Sonar-based real-world mapping and navigation," *Robotics and Automation, IEEE Journal of*, vol. 3, no. 3, pp. 249 –265, june 1987.

[3] A. Diosi and L. Kleeman, "Laser scan matching in polar coordinates with application to slam," in *Intelligent Robots and Systems, 2005.(IROS 2005). IEEE/RSJ International Conference on*. Alberta, Canada: IEEE, Aug. 2005, pp. 3317–3322.

[4] V. Nguyen, A. Harati, A. Martinelli, R. Siegwart, and N. Tomatis, "Orthogonal slam: a step toward lightweight indoor autonomous navigation," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. Beijing, China: IEEE, Oct. 2006, pp. 5007–5012.

[5] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "Rgb-d mapping: Using depth cameras for dense 3d modeling of indoor environments," in *the 12th International Symposium on Experimental Robotics, New Delhi & Agra, India*, Dec. 2010.

[6] E. Eade and T. Drummond, "Scalable monocular slam," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2006*, vol. 1. New York, NY, USA: IEEE Computer Society, June 2006, pp. 469–476.

[7] K. Konolige and M. Agrawal, "Frameslam: From bundle adjustment to real-time visual mapping," *Robotics, IEEE Transactions on*, vol. 24, no. 5, pp. 1066–1077, 2008.

[8] P. Smith, I. Reid, and A. Davison, "Real-time monocular slam with straight lines," in *British Machine Vision Conference*, vol. 1, Edinburgh, UK, Sep. 2006, pp. 17–26.

[9] T. Lemaire and S. Lacroix, "Monocular-vision based SLAM using line segments," in *IEEE International Conference on Robotics and Automation, ICRA 2007*. Roma, Italy: IEEE, April 2007, pp. 2791–2796.

[10] Y. Choi, T. Lee, and S. Oh, "A line feature based SLAM with low grade range sensors using geometric constraints and active exploration for mobile robot," *Autonomous Robots*, vol. 24, no. 1, pp. 13–27, 2008.

[11] L. Chen, T. Teo, Y. Shao, Y. Lai, and J. Rau, "Fusion of lidar data and optical imagery for building modeling," *International Archives of Photogrammetry and Remote Sensing*, vol. 35, no. B4, pp. 732–737, 2004.

[12] C. Rasmussen, "A hybrid vision + ladar rural road follower," in *IEEE International Conference on Robotics and Automation, Orlando, Florida*, May 2006, pp. 156–161.

[13] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.

[14] A. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, oct. 2003, pp. 1403 –1410 vol.2.

[15] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *9th European Conference on Computer Vision (ECCV)*, Graz, Austria, May 2006, pp. 404–417.

[16] E. Eade and T. Drummond, "Edge landmarks in monocular slam," *Image and Vision Computing*, vol. 27, no. 5, pp. 588 – 596, 2009.

[17] G. Zhang and I. H. Sun, "Building a partial 3d line-based map using a monocular slam," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, may 2011, pp. 1497 –1502.

[18] A. Gee, D. Chekhlov, A. Calway, and W. Mayol-Cuevas, "Discovering higher level structure in visual slam," *Robotics, IEEE Transactions on*, vol. 24, no. 5, pp. 980 –990, oct. 2008.

[19] J. Martinez-Carranza and A. Calway, "Unifying planar and point mapping in monocular slam," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2010, pp. 43.1–43.11.

[20] D. Song, H. Lee, J. Yi, and A. Levandowski, "Vision-based motion planning for an autonomous motorcycle on ill-structured roads," *Autonomous Robots*, vol. 23, no. 3, pp. 197–212, Oct. 2007.

[21] D. Song, H. Lee, and J. Yi, "On the analysis of the depth error on the road plane for monocular vision-based robot navigation," in *The Eighth International Workshop on the Algorithmic Foundations of Robotics, Guanajuato, Mexico, Dec. 7-9*, 2008.

[22] Y. Lu, D. Song, H. Li, and J. Liu, "Automatic recognition of spurious surface in building exterior survey," in *IEEE International Conference on Automation Science and Engineering (CASE), Madison, Wisconsin*, Aug. 2013.

[23] J. Zhang and D. Song, "On the error analysis of vertical line pair-based monocular visual odometry in urban area," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*. St. Louis, USA: IEEE, Oct. 2009, pp. 3486–3491.

[24] J. Zhang and D.Song, "Error aware monocular visual odometry using vertical line pairs for small robots in urban areas," in *Special Track on Physically Grounded AI, the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)*, Atlanta, Georgia, USA, July 2010.

[25] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 4, pp. 91–110, Nov. 2004.

[26] R. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "Lsd: A fast line segment detector with a false detection control," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 4, pp. 722 –732, april 2010.

[27] Y. Lu, D. Song, Y. Xu, A. G. A. Perera, and S. Oh, "Automatic building exterior mapping using multilayer feature graphs," Department of Computer Science and Engineering, Texas A&M University, Tech. Rep. 2013-6-1, Jun. 2013. [Online]. Available: http://www.cse.tamu.edu/media/18395/2013-6-1.pdf

[28] P. Chen and D. Suter, "Simultaneously estimating the fundamental matrix and homographies," *Robotics, IEEE Transactions on*, vol. 25, no. 6, pp. 1425–1431, 2009.

[29] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge Univ Pr, 2003.