doi:10.3772/j.issn.1006-6748.2015.01.005

Autonomous map query: robust visual localization in urban environments using Multilayer Feature Graph①

Li Haifeng (李海丰)*, Wang Hongpeng②**, Liu Jingtai**
(*College of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, P. R. China)
(**Insititute of Robotics and Automatic Information System, Nankai University, Tianjin 300071, P. R. China)

Abstract

When a vehicle travels in urban areas, onboard global positioning system (GPS) signals may be obstructed by high-rise buildings and thereby cannot provide accurate positions. It is proposed to perform localization by registering ground images to a 2D building boundary map which is generated from aerial images. Multilayer feature graphs (MFG) is employed to model building facades from the ground images. MFG was reported in the previous work to facilitate the robot scene understanding in urban areas. By constructing MFG, the 2D/3D positions of features can be obtained, including line segments, ideal lines, and all primary vertical planes. Finally, a voting-based feature weighted localization method is developed based on MFGs and the 2D building boundary map. The proposed method has been implemented and validated in physical experiments. In the proposed experiments, the algorithm has achieved an overall localization accuracy of 2.2m, which is better than commercial GPS working in open environments.

Key words: visual localization, urban environment, multilayer feature graph (MFG), voting-based method

0 Introduction

Localization is a key component in many mobile robot applications. GPS is popularly used for location-awareness. However the measurement error of low-cost GPS sensors for civil services may be up to tens of meters. Especially when working in an urban area, the GPS signal may be disrupted by high-rise buildings and becomes more unreliable. When a mobile robot equipped with a GPS sensor is traveling in urban environments, it can only obtain the inaccurate GPS data, which can only provide the robot with a rough location region in the 2D map, as shown in Fig. 1, where the dashed circle represents the potential location region obtained from inaccurate GPS data, triangle A in Fig. 1(b) from multi-pair of camera frames taken at two different locations of A and B in succession, given the inaccurate GPS data and a 2D top-down view building boundary map which is extracted from Google Maps in our experiments. Thus, it is needed to further determine the accurate location of robot A with the aid of other sensors. As cameras become small and cheap, the focus in this work is to develop an accurate visual localization method for mobile robots working in urban environments.

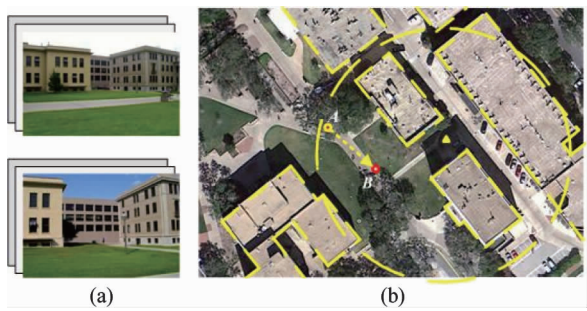

Fig. 1 Estimating camera location

Ref. [1] reported MFG to facilitate the robot scene understanding in urban area. MFG also connects the features in two views and the corresponding 3D coordinate system. 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. By constructing MFG, the 2D/

① Supported by the National High Technology Research and Development Program of China (No. 2012AA041403), National Natural Science Foundation of China (No. 60905061, 61305107), the Fundamental Research Funds for the Central Universities (No. ZXH2012N003), the Scientific Research Funds for Civil Aviation University of China (No. 2012QD23x).

② To whom correspondence should be addressed. E-mail: wanghp@robot.nankai.edu.cn; lihf_cauc@126.com

Received on Apr. 25, 2013

3D positions of features can be obtained including line segments, ideal lines, and all primary vertical planes.

It is an immediate application to employ MFG for localization applications. In this paper, MFG is applied to robot localization, given a 2D map with building outlines in top-down view with no 3D geometric information or appearance data. The 2D building outline map is extracted from Google Maps in our experiments. The proposed method has been implemented and validated in physical experiments. The localization error of the proposed algorithm in physical experiments is around 2m.

1 Related work

Visual localization[2,3] utilizes images taken from on-board camera(s) to estimate the robot location. The ability of accurate localization is an essential building block of robot navigation[4] and simultaneous localization and mapping (SLAM)[5].

Visual localization can have different camera configurations including omnidirectional camera and stereo vision systems. In Ref. [6], a fast indoor SLAM method using vertical lines from an omnidirectional camera was proposed. Nister et al. developed a visual odometry system to estimate the ego-motion of a stereo head[7]. In the proposed method, a regular pinhole camera is employed.

A way of classifying visual localization methods is based on what kinds of features are used. Point features, such as Harris corners, scale invariant feature translation (SIFT)[8], and speed up robust feature (SURF) points[9] are the most popular and reliable ones. Many researchers developed their point feature-based visual localization methods[10,11]. However, compared with line features, point features usually contain more noise and result in high computation cost due to their large amount. Line features are easy to extract[12] more robust, and insensitive to lighting condition or shadows. Therefore, many visual localization applications employed line features and achieved quite accurate results[13-15]. Several recent works[16,17] reconstructed building facades to localize robots in urban scenes. Delmerico[18] proposed a method to determine a set of candidate planes by sampling and clustering points from stereo images with random sample consensus (RANSAC), using local normal estimates derived from principal component analysis (PCA) to inform the planar model. This method is a point-based method whose shortcomings have been discussed above. Cham[17] tried to identify vertical corner edges of buildings as well as the neighboring plane normals from a single ground-view omnidirectional image to estimate the camera pose from a 2D plan-view building outline map. However, this method is not robust for plane analysis due to missing vertical hypotheses. Those methods provide the inspiration that planes are important and robust features to be extracted in reconstruction and localization. Furthermore, a very recent work[18] developed a footprint orientation (FPO) descriptor, which is computed from an omnidirectional image, to match in 2D urban terrain model that is generated from aerial imagery to estimate the position and orientation of a camera.

A number of papers have addressed the problem of matching ground view images to aerial images[19], but they assume that 3D models in the aerial image are available, and focus on specific buildings rather than a broad search across the entire aerial image. Tracking using line correspondences between ground view video and an aerial image was carried out in Ref. [3].

The research group has worked on robot navigation using passive vision system in past years. A vertical line-based method for visual localization tasks[15] has been developed. In recent work[1], an multilayer feature graph (MFG) was reported to facilitate the robot scene understanding in urban area. Nodes of an MFG are features such as SIFT feature points, line segments, lines, and planes while edges of the MFG represented different geometric relationships such as adjacency, parallelism, collinearity, and coplanarity. MFG also connects the features in two views and the corresponding 3D coordinate system. The localization method based on MFGs will be shown.

2 System architecture and problem definition

2.1 System architecture and assumptions

Fig. 2 illustrates the system architecture. The proposed approach consists of off-line map generation and on-line robot localization. There are two main steps in off-line map generation: (1) Extracting an aerial image where the robot locates from the aerial image database based on the inaccurate GPS data, and (2) Generating a 2D map from the aerial image. On-line robot localization consists of three main steps: (1) Constructing MFG from each pair of overlapped camera images; (2) Conducting the perspective projection to obtain the 2D building facade outlines with line features on them from the top-down view; and (3) Estimating the robot location using a voting-based method based on the 2D map and the MFGs after perspective projection. These steps are illustrated in Fig. 2 and each step is described in detail in the following sections.

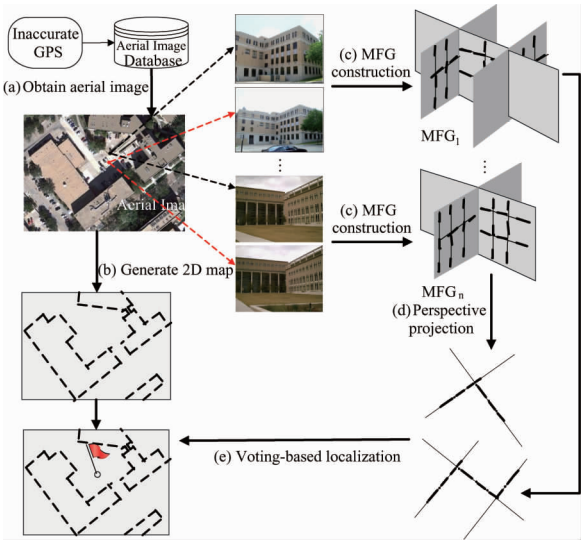

Fig. 2 System architecture

To formulate the problem and focus on the most relevant issues, the following assumptions are developed.

- The 2D map is up-to-date.
- The intrinsic parameters of the finite perspective camera are known by pre-calibration. The lens distortion has been removed.
- The robot knows its relative movements between places where two views are taken, which can be achieved with on-board inertial sensors or wheel encoders. These sensors are good at short distance measurement. This assumption is for the construction of MFG.

2.2 Problem definition

In this paper, all the coordinate systems are right hand systems. The superscript denotes the corresponding notation in the second view. For example, notations in the format of (a, a') refer to a pair of corresponding features across two views.

- Define $\{W\}$ as a 3D Cartesian world coordinate system (WCS) with its x-z plane horizontal and y- axis pointing upwards.
- Define $\{C\}$ and $\{C'\}$ as two 3D Cartesian camera coordinate systems (CCS) at the first and second views, respectively. For each CCS, its origin is at the camera optical center, its z- axis coincides with the optical axis and points to the forward direction of the camera, its x- axis and y- axis are parallel to the horizontal and vertical directions of the CCD sensor plane, respectively.
- Define $\{I\}$ and $\{I'\}$ as two 2D image coordinate systems (ICS) at the first and second views, respectively. For each ICS, its origin is at the principal point and its u- axis and v- axis are parallel to x and y axes of $\{C\}$, respectively.
- Define $F = \{F_1, \cdots, F_n\}$ and $F' = \{F'_1, \cdots, F'_n\}$ as two image sets captured at two different positions, such as A and B in Fig. 1, respectively, with each element $F_i \in F$ and $F'_i \in F'$ being an image. F_i and F'_i, $i = 1, \cdots, n$ are one pair of camera frames with sufficient overlap.
- Define $\boldsymbol{X}$ as the estimated robot location in $\{W\}$ when taking F. Denote $\boldsymbol{X} = [x, z]^T$, where (x, z) is the robot location on the x-z plane of $\{W\}$.

With these notations defined, definition is the following.

Definition 1. MFG-based Localization: Given F and F', the inaccurate GPS data and a 2D building boundary map from top-down view, construct MFGs to estimate $\boldsymbol{X}$.

3 Approach

3.1 Aerial image extraction and map generation

The publicly available Google Maps are chosen as the proposed aerial image database. Based on the GPS data, it can be easily to obtain the aerial image where the robot locates from the database.

Building boundaries are good features to be used for localization applications in urban environments because they can be detected in both aerial and ground images. The aerial image used in this paper is at a resolution of approximately 3.5 pixels/m. It allows our localization system to obtain sub-meter position accuracy. In the previous work[20], an automated method to create 2D building boundary map from an aerial image was introduced. However, the 2D maps generated from the automated methods are not perfect for the following reasons. (1) The obstruction from trees, streets and other things in aerial images; (2) The aerial image that we used are not exact orthographic images, so there are errors due to the perspective. Thus, after the automated map generation, we modify the results manually to obtain higher accuracy. A 2D map, denoted as M, consists of a set of m building facades, $s_1, s_2, \cdots, s_m$, with $s_i = (p_i^0, p_i^1)$. Here, (p_i^0, p_i^1) denotes the 2D points of the facade's projection onto the ground plane (the point coordinates in the aerial image). We do not consider the heights of the facades because they cannot be observed from aerial images.

The building facades in a 2D map M can be classified into three types according to their visibility. Define p^c as the camera center's projection onto the ground plane (the point coordinate in the aerial image). As shown in Fig. 3, the camera's field of view is the re-

gion between the two rays starting from p^c. A building facade $s_i = (p_i^0, p_i^1)$ is visible if both p_i^0 and p_i^1 are in the camera's field of view, and either the line segment $\overline{p^c p_i^0}$ or line segment $\overline{p^c p_i^1}$ does not intersect with any $s_j \in M, j \neq i$; Building facade s_i is partly visible if there exists a sub-line segment s_i^v on s_i such that s_i^v is visible; Otherwise, s_i is defined to be invisible. As shown in Fig. 3, p^c is the projection of camera center, then s_1 is visible, s_2 is partly visible, and s_3 is invisible.

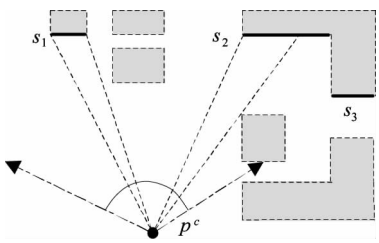

Fig. 3 Visibility of building facades in a 2D map

3.2 Multilayer feature graph construction

Since the proposed visual localization algorithm is based on MFGs, it will be to start with a brief review of MFG, which was firstly presented in the previous work[1]. Fig. 4 illustrates how MFG organizes different types of features according to their geometric relationships. MFG is a data structure consisting of five layers of features, i.e., key points, line segments, ideal lines, vertical planes and vanishing points. Edges between nodes of different layers represent geometric relationships such as adjacency, collinearity, coplanarity, and parallelism. MFG also connects the features in two views and the corresponding 3D coordinate system.

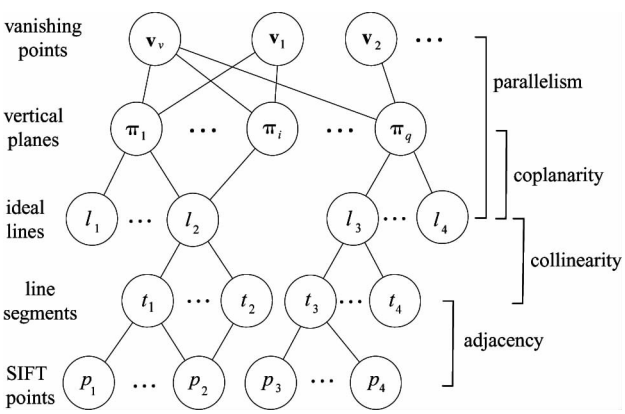

Fig. 4 The structure of an MFG

In an MFG, key points and line segments are raw features extracted from images using methods like SIFT[8] and line segment detector (LSD)[12], while other layers of features are estimated based on them. In Ref. [1], a feature fusion algorithm is presented to construct an MFG based on two views by verifying the geometric relationships incrementally, iteratively, and extensively. As an important part of MFG, the algorithm is able to detect all primary vertical planes and line features in them with a reasonable accuracy. In this work, the localization application using MFGs will be focused on.

3.3 Perspective projection

Since the building boundary map obtained from the aerial image is a 2D map, it also needs to project MFGs to the 2D ground plane to prepare for the following matching. The vertical planes in MFG are parallel to y- axis in $\{W\}$ (and therefore also the ground plane normal), thus, the problem reduces to a 1D perspective projection.

MFG contains the 3D formats of line segments and vertical planes in $\{W\}$. By projecting all entities, such as vertical planes and line segments, to the ground plane, we can obtain the perspective projection of an MFG, denoted as P. Under the projection, each vertical plane π_i becomes a line from top-down view, denoted as ℓ_i. Note that since MFG cannot provide the boundary of vertical plane, the projection of a vertical plane is a line instead of a line segment. Define $L_i^h = \{\iota_j^h\}$, $j \in I_i^h$ and $L_i^v = \{\iota_r^v\}$, $r \in I_i^v$ as the projections of 3D horizontal and vertical line segment sets lying in vertical plane π_i, respectively, where I_i^h and I_i^v are the index sets with which the horizontal and vertical line segments lie in π_i. After the perspective projection, the horizontal line segments are still line segments while the vertical line segments become points, as shown in Fig. 5, the thin lines denote the vertical planes' projections, the thick line segments and points denote the horizontal and vertical line segments lying in the vertical planes, respectively.

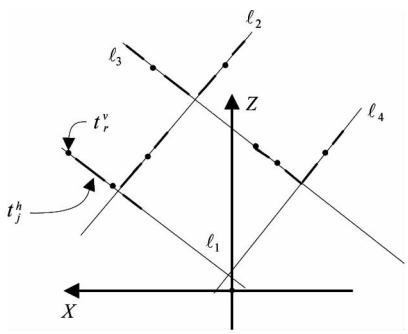

Fig. 5 Perspective projection of an MFG

3.4 Feature-weighted localization using a single MFG

After the perspective projection step, the localization problem using a single MFG converts into matching

P into M to find the accurate camera location X.

The matching criterion is: the total overlapped length between vertical/horizontal line segment features in P and building facades in M is maximum. Fig. 6 gives an illustration of matching evaluation. Due to the construction error of MFG after perspective projection, ι_j^h may be not exactly lying on ℓ_i, so ι_j^h is projected onto ℓ_i. The overlapped part between the projection and s_t is defined as $\iota_{j,t}^+$, while the non-overlapped part is defined as $\iota_{j,t}^-$. The vertical line segments ι_1^v and ι_2^v become points after perspective projection. These points are projected onto ℓ_i. If the projection of ι_r^v lies on s_t, it is called that ι_r^v is on s_t. In Fig. 6, ι_1^v is on s_t, while ι_2^v is not on s_t.

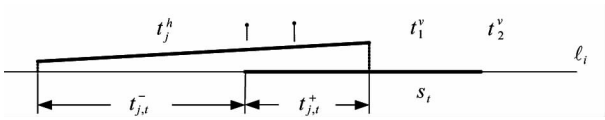

Fig. 6 An illustration of matching evaluation

Thus, a matching evaluation function between $\iota_j^h \in L_i^h$ and $s_t \in M$ is defined as

$$Q^h(\iota_j^h, s_t) = \begin{cases} 0, & \text{if } s_t \text{ is invisible} \\ \| \iota_{j,t}^+ \| - \| \iota_{j,t}^- \|, & \text{otherwise} \end{cases} \tag{1}$$

where $\iota_{j,t}^+$ and $\iota_{j,t}^-$ are two line segments, representing the overlapped and non-overlapped parts between the projection of ι_j^h and the visible part of s_t, respectively, as shown in Fig. 6, where it shows the case that s_t is visible. $\| \cdot \|$ denotes the length of a line segment.

Similarly, as shown in Fig. 6, a matching evaluation function between $\iota_r^v \in L_i^v$ and $s_t \in M$ is defined as

$$Q^v(\iota_r^v, s_t) = \begin{cases} \| \iota_r^v \|, & \text{if } \iota_r^v \text{ is on } s_t \\ - \| \iota_r^v \|, & \text{otherwise} \end{cases} \tag{2}$$

$\mathcal{I}^+$ is defined as an index set for $\ell_i \in P$ such that $\forall i \in \mathcal{I}^+$, ℓ_i has at least one correspondence in M^h. Similarly, $\mathcal{I}^-$ is defined as an index set for $\ell_j \in P$ such that $\forall j \in \mathcal{I}^-$, ℓ_j has no correspondence in M.

Note that the baseline distance between two views obtained from onboard sensors is inaccurate. Thus P is up to scale by a scalar $\lambda \in [\underline{\lambda}, \bar{\lambda}]$, where $\underline{\lambda}$ and $\bar{\lambda}$, determined by the error range of measured baseline distance, are the lower and upper bounds of λ. ℓ_i^λ, $\iota_j^{h,\lambda}$, $j \in I_i^h$ and $\iota_r^{v,\lambda}$, $r \in I_i^v$ are defined as the line ℓ_i, line segments ι_j^h and ι_r^v at scale level λ, respectively.

The total matching evaluation function is defined as

$$f(x,z,\lambda) = \sum_{a \in \mathcal{I}^+} \sum_{\substack{\iota_j^h \in L_a^h \\ s_t \in s(\ell_a^\lambda)}} Q^h(\iota_j^{h,\lambda}, s_t) + \sum_{c \in \mathcal{I}^+} \sum_{\substack{\iota_r^v \in L_c^v \\ s_t \in s(\ell_c^\lambda)}} Q^v(\iota_r^{v,\lambda}, s_t)$$
$$- \sum_{b \in \mathcal{I}^-} \sum_{\substack{\iota_j^h \in L_b^h \\ \iota_r^v \in L_b^v}} (\| \iota_j^{h,\lambda} \| + \| \iota_r^{v,\lambda} \|),$$
$$s.t. \ \underline{\lambda} \leqslant \lambda \leqslant \bar{\lambda} \tag{3}$$

where (x,z) is the camera location, and $s(\ell_a^\lambda)$ denotes the corresponding building boundary set of $\ell_a^\lambda \in M$.

In Eq. (3), the first two terms are to evaluate the overlapping between horizontal/vertical line segments in P and building boundaries in M, and the last term is to demonstrate the case that there is no building boundary in M corresponding to vertical plane π_b.

Therefore, the localization problem using a single MFG based on the map query can be converted into the following optimization problem,

$$\arg\max_{x,z,\lambda} f(x,z,\lambda) \tag{4}$$

The above optimization problem can be solved using the Levenberg-Marquardt algorithm[21].

By now, the camera location can be obtained from an MFG and a 2D building boundary map by solving the above optimization problem. However, this method can not guarantee the correctness of solution. In the 2D building boundary map M, if there exist more than one group of similar building boundaries that can match with P, maximizing Eq. (4) directly may lead to the wrong solution. The case will happen more likely when the number of vertical planes in P is small. To solve this problem, a voting-based camera position estimation method is proposed as follows.

3.5 Voting-based localization using multiple MFGs

In the voting-based localization stage, first, the 2D building boundary map is divided into a $N_a \times N_a$ grid G and define a zero-initialized $N_a \times N_a$ accumulator array Acc correspondingly. Denote (x_i, z_j) as the center of $G(i,j)$. In the proposed voting-based method, each MFG does not only determine one solution from Eq. (4). Instead, each MFG can provide multiple candidate solutions. Traverse G, and set $G(i,j)$ as a candidate solution region if

$$\frac{g(x_i, z_j)}{f_{\max}} > T_r \tag{5}$$

where $f_{\max}$ is the maximum value obtained from Eq. (4), T_r is a specific ratio threshold, and

$$g(x_i, z_j) = \arg\max_{\lambda} f(x_i, z_j, \lambda)$$

Correspondingly, $Acc(i,j)$ increments by 1 if $G(i,j)$ is selected to be a candidate solution region. In order to obtain the correct and optimal camera position, the combination of candidate solutions with the best consensus obtained from different MFGs must be determined. Thus, the candidate solution region with the largest score (number in Acc) is selected as the

correct solution region. The final optimal solution is determined based on Eq. (4), and the only difference is that multiple MFGs are utilized here and the searching region is within the sub-region with the largest score.

The proposed voting-based camera position estimation method using multiple MFGs is described as Algorithm 1.

Algorithm 1: Voting-based Localization using MFGs

Input : F and F'
Output: Camera position $\mathbf{X}$

```
Generate G in the building boundary map;
Initialize a 2D accumulate array Acc;
for each pair of camera frames F_i ∈ F and F'_i ∈ F' do
    Construct MFG from F_i and F'_i;
    Perspective projection for the MFG;
    Determine the candidate solution region from the
    MFG by Eq.(5);
Find largest scoring bin Acc(i, j) in Acc to get X based
on Eq.(4) using all MFGs by searching within G(i, j);
return X.
```

4 Experiments

The proposed visual localization method has been implemented by using Matlab 2008b on a laptop PC. In the physical experiments, a BenQ DCE1035 camera with a resolution of 1095×821 pixels is used. It is to run 7 tests $(A_i, B_i), i = 1, \cdots, 7$ on a university campus, as shown in Fig. 7, where points $A_i, i = 1, \cdots, 7$ denote the first positions in each test to take pictures, respectively, and points $B_i, i = 1, \cdots, 7$ are the second positions to capture pictures, respectively. For the five tests $(A_1, B_1) - (A_5, B_5)$, 4 pairs of camera frames are taken with significant overlapping in each test. For the other two tests (A_6, B_6) and (A_7, B_7), 3 pairs of camera frames are taken in each test. The baseline distance between two positions in each test is measured with a tape measure. The orientation settings of the camera are set to ensure a good overlapping between each pair of images. In order to determine the ground truth of camera positions, the relative distances from the camera center to the surrounding building facades are measured using a BOSCH GLR225 laser distance measurer with a range up to 70m and measurement accuracy of ± 1.5mm. Considering the localization error of GPS in urban environments, the whole searching region is set to be $150\text{m} \times 150\text{m}$, centered at GPS data. G is set to be 60×60, with the size of each sub-region being $2.5\text{m} \times 2.5\text{m}$. Threshold T_r is set to be 0.7.

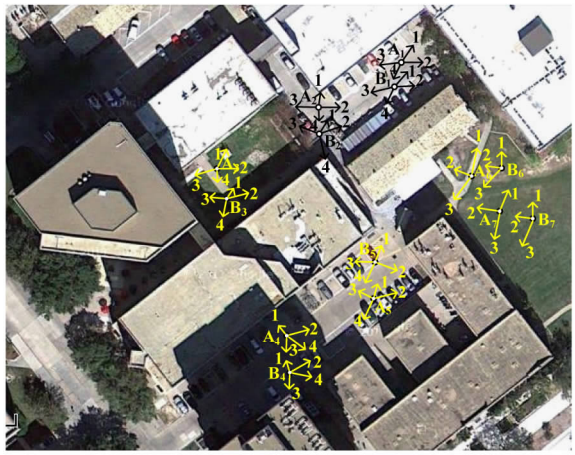

Fig. 7 Positions and orientations of camera in 7 tests

4.1 MFG construction results

The proposed algorithm has successfully constructed MFGs. As a sample output, Table 1 shows the vertical plane identification results and the accuracy of vertical plane reconstruction in test (A_1, B_1). Denote $\hat{\pi}_i$ and $\bar{\pi}_i$ as the estimation from the MFG construction and the ground truth of vertical plane π_i, respectively. Ground truth $\bar{\pi}_i$ is obtained by using three non-collinear 3D points lying in π_i.

According to Ref. [1], directly comparing $\hat{\pi}_i$ with $\bar{\pi}_i$ is not meaningful because the result depends on the coordinate system and unit selections. To avoid the problem, the 3D point reconstruction error is utilized in comparison. Define x_j as a 2D image point lying in π_i. With the aid of camera intrinsic parameters and plane equations, this point can be reconstructed from $\bar{\pi}_i$ and $\hat{\pi}_i$, respectively. Let $\bar{X}_j$ and $\hat{X}_j$ be the corresponding results. A relative error metric is defined as $\dfrac{\| \bar{X}_j - \hat{X}_j \|}{\| \bar{X}_j \|}$ where $\| \cdot \|$ represents the Euclidean distance. For each vertical plane, 20 image feature points are selected manually as even as possible to cover the whole plane region in the image. The mean value and standard deviation of the relative errors are shown in Table 1 for the four image pairs in test (A_1, B_1).

Table 1 Percentile relative errors of the reconstructed 3D points

No.	π_1		π_2		π_3	
	mean	std. dev.	mean	std. dev.	mean	std. dev.
1	2.37	0.28	3.29	0.91	—	—
2	2.88	0.33	3.14	0.59	—	—
3	4.76	1.14	3.93	0.96	5.05	0.62
4	4.12	0.81	2.49	0.54	—	—

Table 1 gives a sample output where the MFG construction algorithm has identified vertical planes in

the images, which results in the different numbers of vertical planes for the image pairs in test (A_1, B_1). The relative errors of points on planes are reasonably small which indicates that the estimated planes are reasonably accurate.

4.2 Voting-based localization results

To informally evaluate the effectiveness of our voting-based localization method, solution uniqueness is inspected by visualizing scores in a 2D robot-position version of the accumulator array. An example result is shown in Fig. 8. Given only one bin with the highest score, it is evident that final solution is unique.

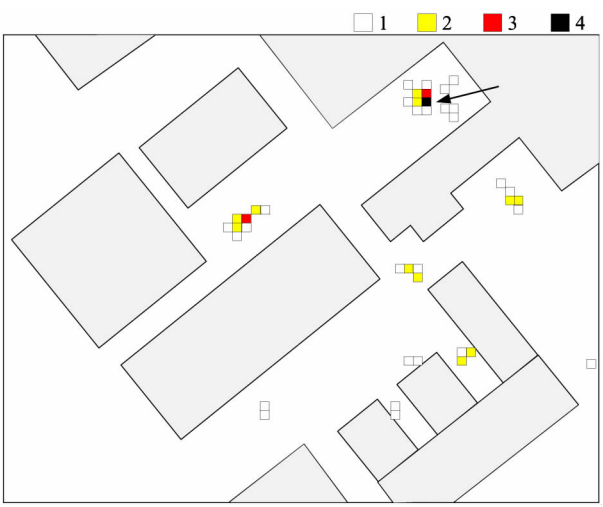

Fig. 8 Example distribution of camera position scores in a 2D position version of the accumulator array, with 2D map overlay. Arrow shows ground truth position.

The MFG-based localization method is compared with the line-based method[15]. Table 2 shows the localization errors using the two methods, respectively. From the table we can conclude that, both methods can localize the camera correctly in all tests. The localization errors of the MFG-based method are obviously smaller than those of line-based method. And in comparison with the ground truth, all the localization errors using the proposed MFG-based method are no more than 2.8m, and the average error is 2.2m. This result is superior to that of the standard positioning service by GPS. The localization error is caused by many factors, such as MFG construction error and map generation error.

Table 2 Comparison of localization errors between MFG-based and line-based methods

Test ID.	MFG-based (m)	Line-based (m)
(A_1, B_1)	2.1	4.3
(A_2, B_2)	1.6	3.4
(A_3, B_3)	2.6	4.4
(A_4, B_4)	1.7	3.5
(A_5, B_5)	2.3	3.9
(A_6, B_6)	2.4	4.1
(A_7, B_7)	2.8	4.6

5 Conclusions

A robust visual localization method is reported based on MFGs and a 2D top-down view building boundary map. By constructing MFGs from camera frames, the 2D/3D positions of multiple features, including line segments, ideal lines, and all primary vertical planes are obtained. A voting-based map query method has been proposed to find the accurate location of camera in the 2D map. The localization method has been implementedand tested in the physical experiments. Results showed that the localization error of the proposed method is around 2m, which is better than commercial GPS working in open environments. More experiments will be done in the following, and it is also planed to integrate the proposed approach with other localization methods and sensors.

References

[1] Li H F, Song D Z, Lu Y, et al. A two-view based multilayer feature graph for robot navigation. In: Proceedings of the IEEE International Conference on Robotics and Automation, Saint Paul, USA, 2012. 3580-3587

[2] Royer E, Lhuillier M, Dhome M, et al. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision*, 2007, 74(3): 237-260

[3] Leung K, Clark C, Huissoon J. Localization in urban environments by matching ground level video images with an aerial image. In: Proceedings of the IEEE International Conference on Robotics and Automation, Pasadena, USA, 2008. 551-556

[4] Song D Z, Lee H, Yi J G. On the analysis of the depth error on the road plane for monocular vision-based robot navigation. In: Proceedings of the 8th International Workshop on the Algorithmic Foundations of Robotics, Guanajuato, Mexico, 2008. 301-315

[5] Davison A, Reid I, Molton N, et al. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, 29(6): 1052-1067

[6] Wongphati M, Niparnan N, Sudsang A. Bearing only Fast-SLAM using vertical line information from an omnidirectional camera. In: Proceedings of the IEEE International Conference on Robotics and Biomimetics, Guilin, China, 2009. 1188-1193

[7] Nister D, Naroditsky O, Bergen J. Visual odometry for ground vehicle applications. *Journal of Field Robotics*,

2006, 23(1): 3-20
[8] Lowe D. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004, 60(2): 91-110
[9] Bay H, Tuytelaars T, Van G L. Surf: Speeded up robust features. In: Proceedings of European Conference on Computer Vision, Graz, Austria, 2006. 404-417
[10] Se S, Lowe D, Little J. Vision-based mobile robot localization and mapping using scale-invariant features. In: Proceedings of the IEEE International Conference on Robotics and Automation, Seoul, Korea, 2001. 2051-2058
[11] Wolf J, Burgard W, Burkhardt H. Robust vision-based localization by combining an image-retrieval system with Monte Carlo localization. *IEEE Transactions on Robotics*, 2005, 21(2): 208-216
[12] Gioi R V, Jakubowicz J, Morel J, et al. LSD: A fast line segment detector with a false detection control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010, 32(4): 722-732
[13] Lemaire T, Lacroix S. Monocular-vision based SLAM using line segments. In: Proceedings of the IEEE International Conference on Robotics and Automation, Roma, Italy, 2007. 2791-2796
[14] Elqursh A, Elgammal A. Line-based relative pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Colorado Springs, USA, 2011. 3049-3056
[15] Li H F, Liu J T, Lu X. Visual localization in urban area using orthogonal building boundaries and a GIS database. *ROBOT*, 2012, 34(5): 604-613
[16] Delmerico J, David P, Adelphi M, et al. Building façade detection, segmentation, and parameter estimation for mobile robot localization and guidance. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems, San Francisco, USA, 2011. 1632-1639
[17] Cham T, Ciptadi A, Tan W, et al. Estimating camera pose from a single urban ground-view omnidirectional image and a 2D building outline map. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, USA, 2010. 366-373
[18] David P, Ho S. Orientation descriptors for localization in urban environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, USA, 2011. 494-501
[19] Lee S, Jung S, Nevatia R. Automatic pose estimation of complex 3d building models. In: Proceedings of the 6th IEEE Workshop on App of Computer Vision, Orlando, USA, 2002. 148-152
[20] Li H F, Xiang J L, Liu J T. An automatic building extraction method from high resolution satellite image. In: Proceedings of the China Control Conference, Hefei, China, 2012. 4884-4889
[21] More J. The Levenberg-Marquardt algorithm: implementation and theory. *Numerical analysis*, 1978, 630: 105-116

Li Haifeng, born in 1984. He is the lecture of Civil Aviation University of China. He received his Ph. D degree in Institute of Robotics and Automatic Information System of Nankai University in 2012. He also received his B. S. degree from Nankai University in 2007. His research interests include robot navigation and computer vision.