IOP Conf. Series: Materials Science and Engineering **428** (2018) 012056 doi:10.1088/1757-899X/428/1/012056

Optimized Landmark Distribution for Mobile Robot Visual Homing

Xun Ji[1,*] and Qidan Zhu[1]

[1]College of Automation, Harbin Engineering University, Harbin, China

[*]Corresponding author: grady_heu@126.com

Abstract. In this paper, we present an optimized landmark distribution method for mobile robot visual homing technology. To satisfy the equal distance assumption for feature-based homing algorithms, the proposed method eliminates the low-quality features according to the image pixel distance difference and the scale information. Finally, a near-ideal distribution of the landmarks can be obtained. We take the feature-based average landmark vector (ALV) algorithm as the example, and prove the effectiveness of the proposed method by the experiments.

1. Introduction

Visual homing is an attractive technology in the field of autonomous mobile robot navigation [1]. Using visual homing algorithms, the robot compares the panoramic image captured at its current location with the pre-stored goal image captured at the home location, and calculates the home vector to guide the robot to move towards the destination [2]. In contrast to traditional vision-based robot navigation methods (such as visual SLAM [3]), visual homing does not need any localization and mapping, but only the direction controlling the robot's movement. Therefore, visual homing can greatly reduce the calculation amount while maintaining high navigation precision [4].

According to the different input forms, visual homing can be broadly divided into two categories, including image-based homing and feature-based homing. Image-based homing usually uses the image pixels from the panoramic images as landmarks, such as warping [5] and descent in image distance (DID) [6]. Feature-based homing usually uses the scale-invariant features obtained by the feature detection and matching algorithms (SIFT [7] and SURF [8]) as landmarks, such as average displacement vector (ADV) [9] and average landmark vector (ALV) [10].

The above visual homing algorithms have proved their effectiveness to guide the robot to the destination, but almost all the homing methods should obey equal distance assumption, which describes the ideal landmark distribution for visual homing: If all the landmarks are uniformly distributed and located the same distance from the destination, the visual homing methods will exhibit the optimal performance. However, this assumption is unrealistic and always violated in practical applications, irregular landmark distribution will greatly reduce the performance of the visual homing algorithms, thus affecting the robot's actual trajectory.

To cope with the above problem, we propose an optimized landmark distribution method for the feature-based visual homing algorithms in this paper. We take the popular ALV method as the example and consider the SIFT features as the natural landmarks, the landmark distribution can be

improved to a nearly ideal state according to the image pixel distance difference and the scale information of SIFT, so that the better homing performance can be achieved.

2. Average Landmark Vector (ALV)

ALV is a classic visual homing algorithm inspired by biological navigation. In the ALV algorithm, the robot only memorizes the landmarks instead of the whole panoramic images, the landmarks are reduced to an average form (defined as the average landmark). By processing the average landmark, the two-dimensional home vector can be calculated, pointing from the robot's current location to the destination. Among all the visual homing algorithms, ALV can has the most simple model with the least amount of computation, in the meantime, ALV can also produce a very precise home vector.

The basic model of ALV can be described in figure 1. Taking the current image as an example, *C* is the robot's current location, L_1, L_2, …, L_n are the *n* landmarks (i.e. SIFT features) extracted in the scene. $\overrightarrow{CL_1}$, $\overrightarrow{CL_2}$, …, $\overrightarrow{CL_n}$ are defined as the unit landmark vectors pointing from the current location to the corresponding landmarks.

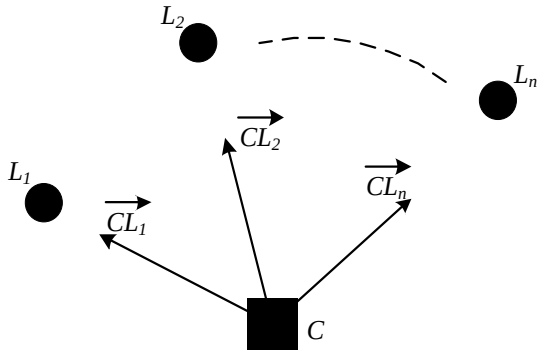


Figure 1. Basic model of ALV

We set the center of the current image as the origin of the image coordinate system, the coordinates of the robot's location can thus be $P_C = (0,0)$, the image coordinates of the *i*th landmark L_i are defined as $P_i = (x_i, y_i)$. Therefore, the *i*th landmark vector at *C* can be computed by:

$$\overrightarrow{CL_i} = \frac{P_i - P_C}{\|P_i - P_C\|} \tag{1}$$

The average landmark at *C* can then be calculated by averaging all the landmark vectors at *C*:

$$\overrightarrow{CL} = \frac{1}{n}\sum_{i=1}^{n}\overrightarrow{CL_i} \tag{2}$$

Similarly, the average landmark at *H* can be calculated by averaging all the landmark vectors at *H*:

$$\overrightarrow{HL} = \frac{1}{n}\sum_{i=1}^{n}\overrightarrow{HL_i} \tag{3}$$

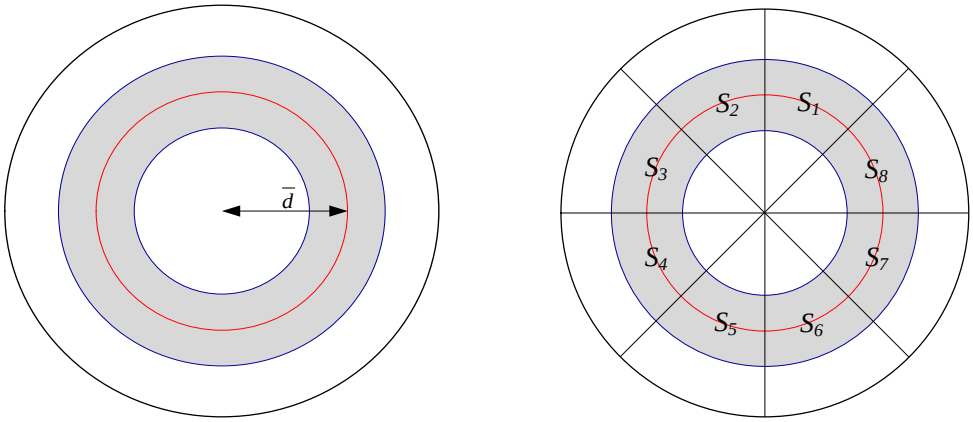
Finally, the home vector $\boldsymbol{h}$ can be generated by subtracting $\overrightarrow{CL}$ from $\overrightarrow{HL}$:

$$\boldsymbol{h} = \overrightarrow{CL} - \overrightarrow{HL} \tag{4}$$

3. Optimized Landmark Distribution

The proposed method consists of two principles. To make the actual distribution of the landmarks closer to the ideal state, the method optimizes the landmark distribution from two aspects: the image

pixel distance and the frequency where landmarks appear in different directions. The models of the two principles are shown in figure 2.



(a) (b)

Figure 2. The models of the principles. (a) The model of Principle 1. The distance from the image center to the red circle is $\bar{d}$. (b) The model of Principle 2.

3.1. Principle 1

As shown in figure 2(a), based on the imaging characteristics of panoramic images, most of the landmarks are extracted and located nearby the horizon circle [11]. However, the image positions of some landmarks are very close or far from the image center, these landmarks will impact on the homing performance. Therefore, Principle 1 is proposed to eliminate such types of the landmarks.

Assume that the image coordinates of the *i*th landmark are defined as (x_i, y_i). The image pixel distance of the *i*th landmark is defined as d_i:

$$d_i = (x_i^2 + y_i^2)^{-1} \tag{5}$$

The average image pixel distance of all the landmarks can be computed by:

$$\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i \tag{6}$$

Then, the distance difference of the *i*th landmark can be obtained by:

$$\Delta d_i = \left| d_i - \bar{d} \right| \tag{7}$$

Principle 1 only preserves the landmarks with $\Delta d_i < 0.3\bar{d}$ (the gray area), and removes the landmarks with $\Delta d_i \geq 0.3\bar{d}$ (the white area).

3.2. Principle 2

As shown in figure 2(b), Principle 2 evenly divides the goal image into N sector rings, denoted as S_1, S_2, …, S_N, respectively (we set N to 8). Assume that n landmarks are extracted, and the *i*th sector ring contains n_i landmarks, Principle 2 firstly calculates the average landmarks in each sector ring:

$$\bar{n} = \frac{n}{N} \tag{8}$$

Generally, if each sector ring contains $\bar{n}$ landmarks locating the same distance from the image center, the ideal distribution can be achieved. In Principle 2, the landmark amounts in each sector ring

is compared with $\bar{n}$. For the sector rings whose actual landmark amounts are smaller than $\bar{n}$, all the landmarks in these sector rings are preserved without further processing. Conversely, for the sector rings that contain more landmarks, Principle 2 will reduce the landmark amounts in these sector rings based on the scale information.

A phenomenon is stated in [12] that if a certain SIFT feature has a larger value, the distance between the robot and the corresponding landmark is relatively closer. Therefore, when the two images are matched, the two SIFT features in the each matching pair have different scale values, this implies that the represented landmark has different distances to the two capture positions. Although we cannot obtain the exact distance values, we can qualitatively compare the difference of the above two distances.

Since visual homing is a local navigation strategy for a small scale experimental area, for most landmarks, their distances to the robot will not change too much, resulting in small scale changes. Hence, For the sectors containing more landmarks, Principle 2 sorts all the preserved landmarks according to the scale difference from small to large, the landmarks with small scale difference are removed in sequence, and the removed number is the difference between the actual number of the landmarks and $\bar{n}$.

4. Experiments

4.1. Panoramic Image Databases

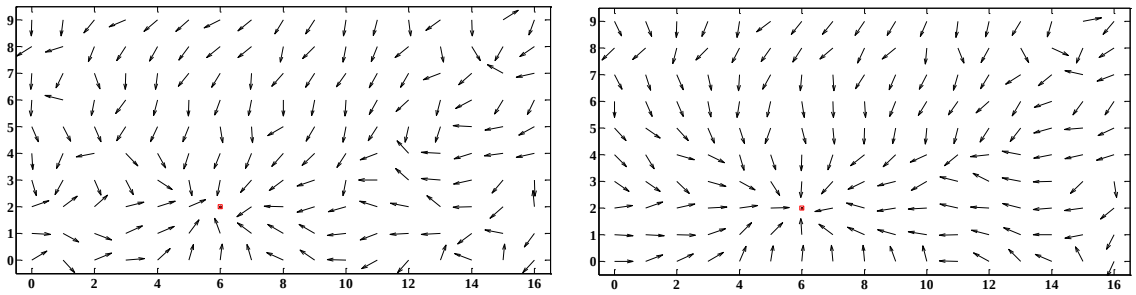
The experiments were performed based on the panoramic image databases collected at Bielefeld University. The capture environment of the databases is a 4.8m×2.7m indoor spaces, including 170 capture positions, which are spaced 30cm apart. The robot equipped with a panoramic vision system captured the panoramic image at each capture position. We selected two versions of the database, including Original and Chairs. The Original version is the default condition of the environment. The curtains and doors are closed, the lights in the scene are turned off. The Chairs version is based on the Original version, three chairs are randomly set in the environment. Two samples of the panoramic image versions are shown in figure 3.

(a) (b)

Figure 3. Two samples of the panoramic image versions.

4.2. Homing Performance

We test the homing performance between the initial and improved ALV algorithms. To intuitively exhibit the effect of the algorithms, we arbitrarily selected a panoramic image from the Original version as the goal image, and the remaining 169 images were considered as all the possible current images. The home vectors were calculated between the goal image and these current images. Figure 4 shows the visualization results of the initial and the improved ALV algorithm. (6,2) was set as the goal location.



(a) (b)

Figure 4. Visualization results. Red square represents the goal location (6,2). Arrows represent the calculated home vectors. (a) Initial ALV algorithm. (b) Improved ALV algorithm.

4.3. Average Angular Error and Return Ratio

To quantitatively analyze the performance of the above two algorithms, we adopted two widely used performance metrics, called average angular error (*AAE*) and return ratio (*RR*).

AAE indicates the average error between the ideal home vector and the actual home vector. Assume that the direction of the ideal home vector is denoted as α_{ide}, pointing from the current location to the goal location. The direction of the calculated home vector is denoted as α_{act}. The angular error (*AE*) can be computed by:

$$AE(H,C) = \left|\alpha_{ide} - \alpha_{act}\right| \tag{9}$$

AAE can thus be calculated by:

$$AAE = \frac{1}{pq}\sum_{k=1}^{pq} AE(H,C_k) \tag{10}$$

Where C_k represents the *k*th current location. *p* and *q* respectively denotes the rows and columns of the capture location. According to the image databases, *p* is set to 17 while *q* is set to 10.

RR shows the proportion of all the possible current locations that can successfully return to the destination. A dummy robot is created to simulate the movement according to the calculated home vector, the standard to decide whether a certain current location will become a successful homing location can be presented as follows:

Step 1: Control the robot to move towards its nearest location according to the calculated home vector.

Step 2: If the following two cases happen, jump to Step 4:

Case 1: The robot finally reach the pre-set destination.

Case 2: The robot moves more than 17+10=27 steps, this means the robot has moved more than half the circumference of the environment but still not reach the destination.

Step 3: Continue to perform Step 1.

Step 4: If Case 1 happens, we declare the tested current location is valid. Conversely, if Case 2 happens, we declare the tested current location is failed.

After all the current locations are tested according to the above steps. *RR* can be computed by:

$$RR = \frac{T_s}{T} \tag{11}$$

Where T_s indicates the number of the valid current locations. *T* indicates the total number of the current locations. According to the image databases, *T* is set to 169.

Table 1 shows the corresponding *AAE* values, we set (6,2) from the Original version and (8,5) from the Chairs version as the home locations.

Table 1. *AAE* Results ($^{\circ}$).

Version	Goal	ALV	Improved
Original	(6,2)	34.504	23.417
Chairs	(8,5)	47.036	34.147

Table 2 and Table 3 shows the *RR* results for the two algorithms based on the two image versions. Three evenly distributed locations are selected as the goal location to obtain a general conclusion.

Table 2. *RR* Results for Original Version

Goal	ALV	Improved
(2,8)	0.663	0.781
(8,3)	0.805	0.828
(15,1)	0.586	0.639

Table 3. *RR* Results for Chairs Version

Goal	ALV	Improved
(2,8)	0.680	0.787
(8,3)	0.799	0.834
(15,1)	0.568	0.598

5. Conclusions

In this paper, we propose two principles to optimize the landmark distribution for visual homing. The landmark ditribution is optimized from the two aspects, including the image pixel distance and the frequency at which landmarks appear in different directions. After the landmarks are filtered, the optimized distribution are more close to the ideal state, satisfying the equal distance assumption. We take the popular ALV algorithm as the example, and perform a series of the experiments based on the panoramic image databases. Results reveal that the landmarks optimized by the proposed principles can make the ALV algorithm exhibit better homing performance than the initial state, the robot can return to the specified destination from more possible locations with more accurate and robust home vectors.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (61673129, 51674109).

References

[1] Zhu Q, Liu C and Cai C, A novel robot visual homing method based on sift features, *Sensors*, 2015. vol 15, no.10: pp 26063-26084.
[2] Lee C and Kim D, Local homing navigation based on the moment model for landmark distribution and features, *Sensors*, 2017. vol 17, no.11: pp 2658.
[3] Gamallo C, Mucientes M and Regueiro C V, Omnidirectional visual slam under severe occlusions, *Robotics and Autonomous Systems*, 2015. vol 65, no. C: pp 76-87.
[4] Zhu Q, Liu X and Cai C, Feature optimization for long-range visual homing in changing environments, *Sensors*, 2014. vol 14, no.2: pp 3342.
[5] Franz M O, Schölkopf B, Mallot H A and Bülthoff H H, Where did I take that snapshot? Scene-based homing by image matching, *Biological Cybernetics*, 1998. vol 79, no.3: pp 191-202.

[6] Zeil J, Hoffmann H I and Chal J S, Catchment areas of panoramic images in outdoor scenes, *Journal of the optical society of America A*, 2003. vol. 20, no.3: pp 450-469.
[7] Lowe D G, Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision*, 2004. vol. 60, no.2: pp 91-110.
[8] Bay H, Ess A, Tuytelaars T, and Gool L V, Speeded-up robust features, *Computer Vision and Image Understanding*, 2008. vol. 110, no.3: pp 346-359.
[9] Hong J, Tan X, Pinette B, Weiss R and Riseman E M, Image-based homing, *IEEE Control Systems*, 1992. vol. 12, no.1: pp 38-45.
[10] Lambrinos D, Möller R, Labhart T and Pfeifer R, A mobile robot employing insect strategies for navigation, *Robotics and Autonomous Systems*, 2000. vol. 30, no.1-2: pp 39-64.
[11] Zhu Q, Liu C and Cai C, A robot navigation algorithm based on sparse landmarks, *Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics. IEEE*, 2014: pp 188-193.
[12] Churchill D and Vardy A, An orientation invariant visual homing algorithm, *Journal of Intelligent and Robotic Systems*, 2013. vol. 71, no.1: pp 3-29.