FAST INITIALIZATION FOR FEATURE-BASED MONOCULAR SLAM

A. Shaobo Zhang, B. Sheng Liu , C. Jianhua Zhang, D. Zhenhua Wang, E. Xiaoyan Wang

Zhejiang University of Technology
College of Computer Science and Technology
Hangzhou,Zhejiang, P.R.China

ABSTRACT

Initial map determines the effect of followed slam tracking. Most feature-based monocular slam initialize their map according to key points matching in close frames. Nevertheless, it will consume lots of computational resources and time. And it is easy to fail in some far scene or close scene. In this paper, we present a fast initialization method to reduce runtime and improve success rate of initialization for feature-based monocular slam. First, vanishing points detection based on line segment detector [1] is adopted. Second, we extract orb key points. And the coordinates of every key points are undistorted and normalized. Third, we generate the corresponding depth for each key point by normalizing its distance to the existing vanishing points or gaussian random number. We compare our method with state-of-the-art on public data sets and ours. The experiments show that our method outperforms on runtime and accuracy.

Index Terms— fast initialization, feature-based, monocular slam, vanishing points

1. INTRODUCTION

The simultaneous localization and mapping (SLAM) [2] [3] was firstly used in mobile robot, it helps mobile robot to build a map of unkown environment and compute it's position based on the map. Then the main task of auto location and navigation have the potential to achieve. Many compelling implementations of SLAM system [4][5][6][7] have been presented to improve computational efficiency while ensuring accurate estimation over the past decades.

Most initialization method of depth-known slam like rgbd slam [8], binocular slam, laser-based slam [9][10] fail on monocular slam system. Not all the information of the environment can be captured one time in a monocular slam system, particularly the depth. So most monocular slam adopt two-frames or multi-frames matching to build an initial map for subsequent tracking and mapping. In general, initial map based on these methods are not high-precision and efficient. And mass computation makes the process slowly.

The most famous feature-based monocular slam PTAM [11] is the first one to separate tracking and mapping into two parallel threads to speed up. Its map initialization is semi-automatic. The operator should move the camera and press key to obtain two frames in an angle. Pairs of key points in two frames are matched and used to solve rotation and transportation matrix. And the corresponding pairs of key points are calculated to map points to achieve initialization.

ORB-SLAM [12] almost fuses most of the best methods to combine a robust and efficient monocular slam system. Comparing to PTAM, its initialization method do not need artificial interference any more. However, the core theory to solve rotation and transportation matrix from pairs of matched key points is nearly the same as PTAM. During SLAM initializating, every frame is matched to the first frame until obtain enough matched pairs of key points and map points. So the strict parameter setting and external environment will affect the initializing process seriously. Time-comsuming and repeated initialization occur frequently.

In this paper, we propose a fast initialization algorithm based on the first captured frame of slam. To avoid computation cost and view distance limitation, we do not generate map points from key points matching. Based on vanishing points detection [13] and orb key points extraction procedure, we figure out the missing depth by normalizing according to the distance to vanishing points or gaussian random number.

The paper is organized as follows. Section one introduce related works. Section two describe our method in details. Experiments are presented in Section three and followed by conclusions at last.

2. PROPOSED FAST INITIALIZATION ALGORITHM

The proposed fast initialization algorithm of feature-based monocular SLAM can be divided into three parts. First, adopt MSAC [14][15] method to detect vanishing points in the first input frame based on LSD [1] results. Second, use ORB feature extraction in the same image. Then coordinates of all extracted key points are undistorted and normalized.

This work was supported by Zhejiang Provincial Natural Science Foundation of China under Grant number $LY15F020031$, $LQ16F030007$ and $LQ16F030007$, National Natural Science Foundation of China (NSFC) under Grant numbers 11302195 and 61401397.

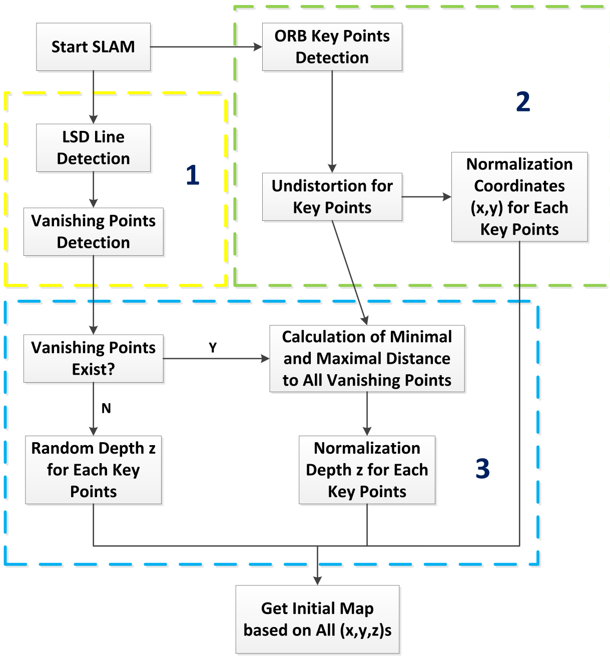

Fig. 1. Flow chart of our fast initialization algorithm.

Third, we present a gaussian random number method and vanishing points normalization method according to the number of detected vanishing points. We integrate coordinate (x, y) from modular 2 and z from modular 3 to structure $3D$ coordinate (x, y, z) lastly. All the $3D$ coordinates of key points made up the initial map for followed slam tracking. The main flow chart of our method is shown as Fig.1.

2.1. Vanishing Points Detection

2.1.1. Line Segment Detector

The first input frame of SLAM is the unique image we needed in our fast initialization algorithm. Lines of the image are detected by line segment detector method in the first step. Comparing with global line detector like hough method [16], LSD extracts lines in local region. Though long line may be split into several parts, the advantage of quick calculating speed is the main reason why we adopt LSD to detect lines. And the weakness can be fixed by parameters adjustment partly. Further more, it does not affect the result in our applicated scenario in the next step.

2.1.2. M-estimatior Sample and Consensus

M-estimatior Sample and Consensus is a robust framework to calculate parameter model. In our application, we parameterize all the lines detected by LSD and estimate the most possible less than or equal to 3 cross points as our vanishing points by MSAC. We show the results of LSD and vanishing points detection in Fig.3.

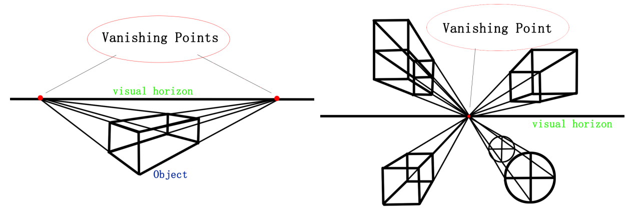

Fig. 2. Sketch of the possible vanishing points in images.

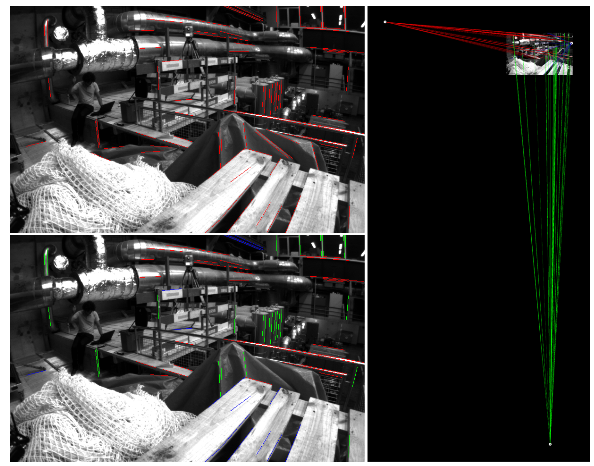

Fig. 3. The image in top of the left column shows results of LSD method. And we depict the lines which determine the corresponding vanishing points in colors red, green and blue respectively in bottom. Image in the right column represents the detected vanishing points.

2.2. Key Points Procedure

2.2.1. ORB Feature Extraction

ORB [17] is a feature method that can extract and match key points very fast. It is consisted by FAST corner detector, intensity centroid and Brief descriptor. They are known as efficient computation. And rotation invariance makes it much more suitable for real-time slam system. The ORB results and structure of used Brief descriptor are shown in Fig.4.

2.2.2. Key Points Undistortion

In general, distortion is unavoidable when mapping scene in real world to a digital image by camera. Radial and tangential distortion are two main errors and must be corrected. The radial distortion can be described as follows

$$\begin{cases} |x^{'} = x * (1 + k_1 * r^2 + k_2 * r^4) \\ |y^{'} = y * (1 + k_1 * r^2 + k_2 * r^4) \end{cases}. \tag{1}$$

Here x, y are original image coordinates and $x^{'}$, $y^{'}$ are corrected coordinates by radial distortion parameter k_1, k_2.

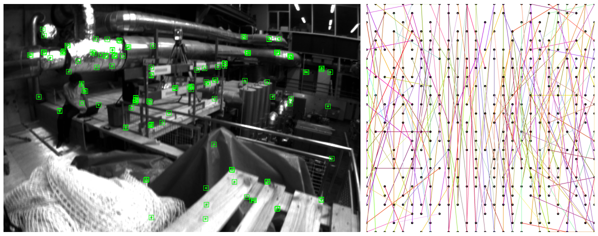

Fig. 4. The ORB result shows on the left and the structure of used Brief descriptor exhibits on the right.

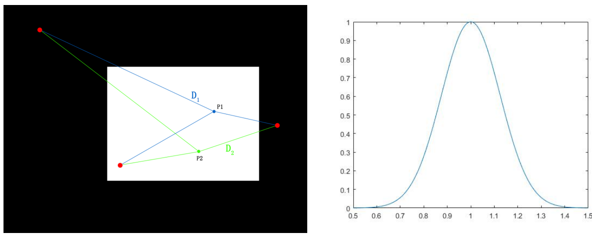

Fig. 5. In the first image, $P1$, $P2$ represent two key points, the corresponding blue lines and green lines are their initial depth D_1 and D_2. The second image shows the adopted gaussian probability distribution.

And $r^2 = x^2 + y^2$.

The tangential distortion shows below

$$\begin{cases} |x^{''} = x^{'} + [2 * p_1 * y + p_2(r^2 + 2 * x^2)] \\ |y^{''} = y^{'} + [2 * p_2 * x + p_1(r^2 + 2 * y^2)] \end{cases}. \tag{2}$$

$x^{''}$, $y^{''}$ denote the final corrected points coordinates, p_1, p_2 are tangential distortion parameter.

2.2.3. *Coordinates Normalization*

We normalize key points coordinates in a standardized form. The formula of normalization can be defined as

$$\begin{cases} |P_{w\_x} = (x^{''} - C_x)/f_x \\ |P_{w\_y} = (y^{''} - C_y)/f_y \end{cases}. \tag{3}$$

Where C_x, C_y are main point and f_x, f_y are focal length of the camera which capture the image. All C_x, C_y, f_x, f_y, k_1, k_2, p_1, p_2 constitute camera internal parameters. We obtain them from camera calibration method of Zhang [18]. P_{w_x}, P_{w_y} will be used in the last section

2.3. Depth Generation

2.3.1. *Min and Max of Distance*

Our depth generation method depends on the results of vanishing points detection. If we get vanishing points greater than zero, a depth normalization strategy based on the distance between key points and vanishing points is activated.

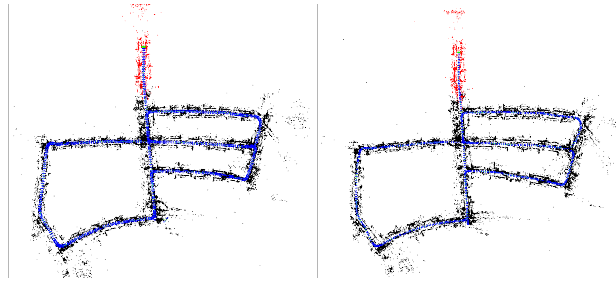

Fig. 6. Final maps of our method (left) and ORB-SLAM (right) on KITTI dataset [19]. They are nearly the same.

We assume that vanishing points number we obtained is i, and the extracted key points number is j. For each key point, the total distance for every vanishing points is calculated

$$D_j = \sum_{n=1}^{i} \sqrt{(x_j - v_i)^2 + (y_j - v_i)^2}. \tag{4}$$

Finally in this step, we select the maximal and minimal D_j from all the D. And D_j is the initial depth of the corresponding key point, shows in Fig.5.

2.3.2. *Depth Normalization*

To estimate an effective depth coordinate, we normalize our depth around 1 in section $[0.5, 1.5]$, which is adopted by most slam system after [20] present inverse depth. The deviation 0.5 is obtained by lots of experiments. So the formula can be described as

$$p_{w\_z} = 0.5 + D_j/(D_{max} - D_{min}). \tag{5}$$

Here p_{w_z} is the final depth of the corresponding key point p_j, and D_{max} and D_{min} denote the maximal and minimal distance mentioned in the last step.

2.3.3. *Random Depth Creation*

Our fast initialization method is set to generate the initial map with one frame. Otherwise, fast initialization would make no sense. But it's not easy to detect vanishing points in some natural environment without many lines. So when the number of vanishing points from modular 2 is zero, we generate depth by gaussian random number generator. The gaussian probability distribution equation shows as

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma}(-\frac{(x-\mu)^2}{2\sigma^2}). \tag{6}$$

Where we set $\mu = 1$ and $\sigma = 0.125$. From the second image of Fig.5, we can see the distribution is almost close to section $[0.5, 1.5]$. We set p_{w_z} obtained here as supplement of the final depth.

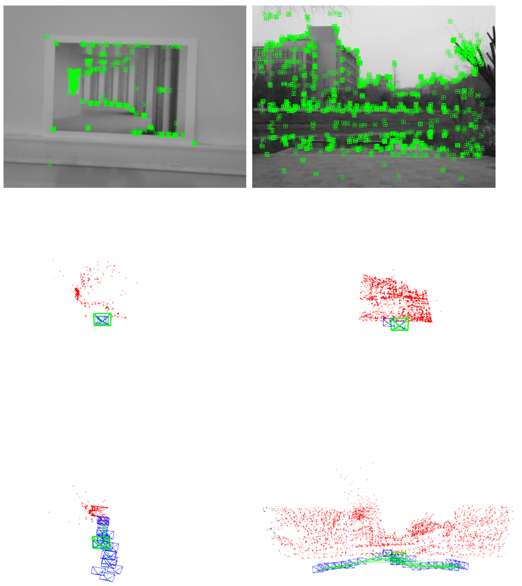

Fig. 7. Left images shows results of our fast initialization method on our indoors data sets, and the right is results of outdoors. Images in the first row represent extracted key points on the first frame. And the second is their corresponding map points. From the last row images, we can see the slam system run well based on our fast initialization method.

3. EXPERIMENTS AND ANALYSIS

We have illustrated the mechanism of our fast initialization for monocular slam. In this section, we test our algorithm on available public data sets and our data sets. And to analysis the advantage of our method, we compare the effect of fast initialization with state-of-the-arts. All the experiments are under system ubuntu 14.04 and intel core i5-4210M 2.60GHz.

3.1. Experiments on Public Data Sets

We compare our results with the most excellent ORB-SLAM on KITTI dataset [19] and EuRoC data sets [21], using the implementations from the respective authors. The whole procedure of initialization time in Table.1 indicate our method is much more efficient. And the tracked map points number for per frame after initialization shows in Table.2 prove our method outperforms ORB-SLAM in accuracy. Comparison of the final map exhibits in Fig.6, they are nearly the same.

Table 1. Time consuming and frame number of the initialization procedure. In this table, EuRoC01 is the easiest data set and EuRoC05 corresponds to the most difficult data set. Our method costs less time and shows much more stable.

	EuRoC01	**EuRoC03**	**EuRoC05**
Our Method (s)	0.191721	0.186739	0.186671
Number of Frames	1	1	1
ORB-SLAM (s)	0.388882	2.681946	3.018167
Number of Frames	7	55	81

Table 2. The number of matched points between map points and frames after initialization. We can see our method maintain lots of built map points of initialization. It proves our fast initialization runs in high accuracy.

	Frame Number	**Our Method**	**ORB-SLAM**
EuRoC01	1	2009	147
	2	2009	147
	3	2009	357
	4	1980	363
	5	1980	363
EuRoC03	1	2008	122
	2	2008	122
	3	2008	595
	4	1983	602
	5	1983	602
EuRoC05	1	1988	104
	2	1988	104
	3	1928	400
	4	1928	369
	5	1726	461

3.2. Experiments on our data sets

When the moving camera is far from the objects, the captured images will not change a lot. So it is hard to initialize if we adopt frames matching methods. Our fast initialization will never be bothered by this situation. It will always initialize in a frame no matter outdoors or indoors data sets. The results in Fig.7 expresses the ability of our method.

4. CONCLUSION

In this paper, we propose a novel fast initialization algorithm for feature-based monocular slam. The missing depth for key points are not generated by image pairs matching, but by the normalizing distance between key points and vanishing points or gaussian random number. Comparing with state-of-the-art, our method shows good achievement on accuracy and time consuming. Especially on our data sets, our method works well on close scene and far scene no matter outdoors and indoors data sets. In future work, we plan focus on restricting the number of used key points based on detected lines to reduce runtime of the initialization procedure further.

5. REFERENCES

[1] Rafael Grompone Von Gioi, Jrmie Jakubowicz, Jean Michel Morel, and Gregory Randall, "Lsd: A line segment detector," *Image Processing on Line*, vol. 2, no. 4, pp. 35–55, 2012.

[2] Sebastian Thrun and John J. Leonard, *Simultaneous Localization and Mapping*, Springer Berlin Heidelberg, 2008.

[3] H Durrant-Whyte and Tim Bailey, "Simultaneous localization and mapping: Part i," *IEEE Robotics & Amp Amp Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[4] Jakob Engel, Thomas Sch?ps, and Daniel Cremers, "Lsd-slam: Large-scale direct monocular slam," 2014.

[5] Michael Montemerlo, Sebastian Thrun, Daphne Koller, and Ben Wegbreit, "Fastslam:a factored solution to the simultaneous localization and mapping problem," *Archives of environmental contamination and toxicology*, vol. 50, no. 2, pp. 240–248, 2015.

[6] Philipp Tiefenbacher, Timo Schulze, and Gerhard Rigoll, "Off-the-shelf sensor integration for mono-slam on smart devices," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 15–20.

[7] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse, "Monoslam: real-time single camera slam.," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 29, no. 6, pp. 1052, 2007.

[8] F Endres, J Hess, J Sturm, and D Cremers, "3-d mapping with an rgb-d camera," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 177–187, 2014.

[9] R. Kurazume, Y. Tobata, Y. Iwashita, and T. Hasegawa, "3d laser measurement system for large scale architectures using multiple mobile robots," in *International Conference on 3-D Digital Imaging and Modeling*, 2007, pp. 91–98.

[10] Gerald F Marshall, "Handbook of optical and laser scanning," *Optical Engineering*, 2011.

[11] Georg Klein and David Murray, "Parallel tracking and mapping for small ar workspaces," in *IEEE and ACM International Symposium on Mixed and Augmented Reality*, 2007, pp. 1–10.

[12] Ral Mur-Artal, J. M. M. Montiel, and Juan D. Tards, "Orb-slam: A versatile and accurate monocular slam system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[13] J. M. M Montiel and A. J Davison, "A visual compass based on slam," in *IEEE International Conference on Robotics and Automation*, 2006, pp. 1917–1922.

[14] Hoang Hon Trinh and Kang Hyun Jo, "Image-based structural analysis of building using line segments and their geometrical vanishing points," in *SICE-ICASE, 2006. International Joint Conference*, 2006, pp. 566–571.

[15] M Nieto and L Salgado, "Non-linear optimization for robust estimation of vanishing points," in *IEEE International Conference on Image Processing*, 2010, pp. 1885–1888.

[16] M. C. K Yang, Jong Sen Lee, Cheng Chang Lien, and Chung Lin Huang, "Hough transform modified by line connectivity and line thickness," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 19, no. 8, pp. 905–910, 1997.

[17] E Rublee, V Rabaud, K Konolige, and G Bradski, "Orb: An efficient alternative to sift or surf," vol. 58, no. 11, pp. 2564–2571, 2011.

[18] Zhengyou Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[19] A Geiger, P Lenz, C Stiller, and R Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[20] J. Civera, A. J. Davison, and J. M. M. Montiel, "Inverse depth parametrization for monocular slam," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 932–945, 2008.

[21] Michael Burri, Janosch Nikolic, Pascal Gohl, Thomas Schneider, Joern Rehder, Sammy Omari, Markus W Achtelik, and Roland Siegwart, "The euroc micro aerial vehicle datasets," *International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.