2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

A Robust Stereo Semi-direct SLAM System Based on Hybrid Pyramid

Xiangrui Zhao[1], Renjie Zheng[1], Wenlong Ye[1], and Yong Liu[1,2]

***Abstract*— We propose a hybrid pyramid based approach to fuse the direct and indirect methods in visual SLAM, to allow robust localization under various situations including large-baseline motion, low-texture environment, and various illumination changes. In our approach, we first calculate coarse inter-frame pose estimation by matching the feature points. Subsequently, we use both direct image alignment and a multi-scale pyramid method, for refining the previous estimation to attain better precision. Furthermore, we perform online photometric calibration along with pose estimation, to reduce un-modelled errors. To evaluate our approach, we conducted various real-world experiments on both public datasets and self-collected ones, by implementing a full SLAM system with the proposed methods. The results show that our system improves both localization accuracy and robustness by a wide margin.**

I. Introduction

Visual SLAM technology relies on small-sized and inexpensive cameras to provide maps and local positioning results for mobile robots in unknown environments. It has made remarkable achievements in the past 30 years, and its positioning accuracy has reached a practical level. However, under challenging conditions, e.g., low-texture environment, large-baseline motion, and various illumination changes, performance degradation exists for most visual SLAM systems, which is the problem we seek to improve.

Specifically, in this work, we propose our method by exploiting the complementary characteristics of two well-developed visual SLAM methods, geometric feature based (indirect) ones [1]–[5] and photometric direct ones [6]–[11]. On one hand, feature based method performs better with large baseline motion, fast motion, and with large illumination changes. This is due to the fact that feature points are typically robust to scale, rotation, and illumination changes, compared to the direct gradient computation used in the direct methods. On the other hand, when performing localization in low-texture environments where the number of reliable feature points is limited, the accuracy of feature based method will be inevitably reduced. However, in this case the direct methods are less affected. In addition to accuracy, we also note that the map generated by the indirect methods is much sparser than the direct methods.

By analyzing the complementary properties of both indirect and direct methods, in this paper, we propose a novel hybrid approach to fuse both, to allow robust estimation under challenging conditions. We also note that, similarly to other robotic localization algorithms [11]–[16], we also perform online sensor model (photometric) calibration, which is the key to low-drift localization. To summarize, the main contributions of our work are as follows:

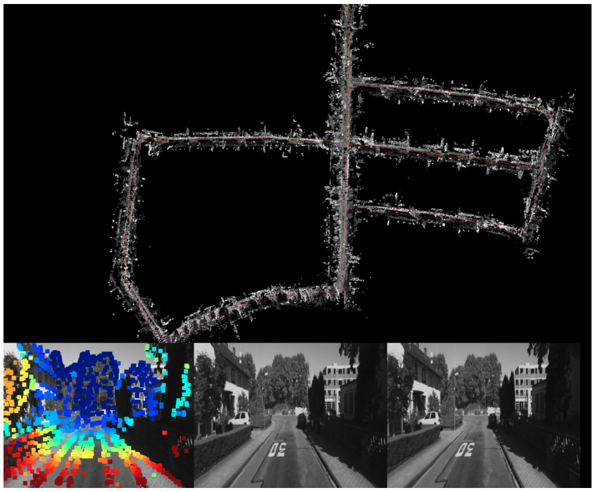


Fig. 1. Map and tracjectory on KITTI 05

- We design a hybrid pyramid method, to allow both direct and indirect based pose optimization for improving estimation accuracy and robustness.
- We also perform online photometric calibration together with pose estimation, to reduce un-modeled errors from photometric cost functions.
- We build a full stereo visual SLAM system with loop-close detection, pose graph optimization, and relocation, for performance evaluation. Our system outperforms state-of-the-art methods, e.g. ORB-SLAM2 [3], significantly.

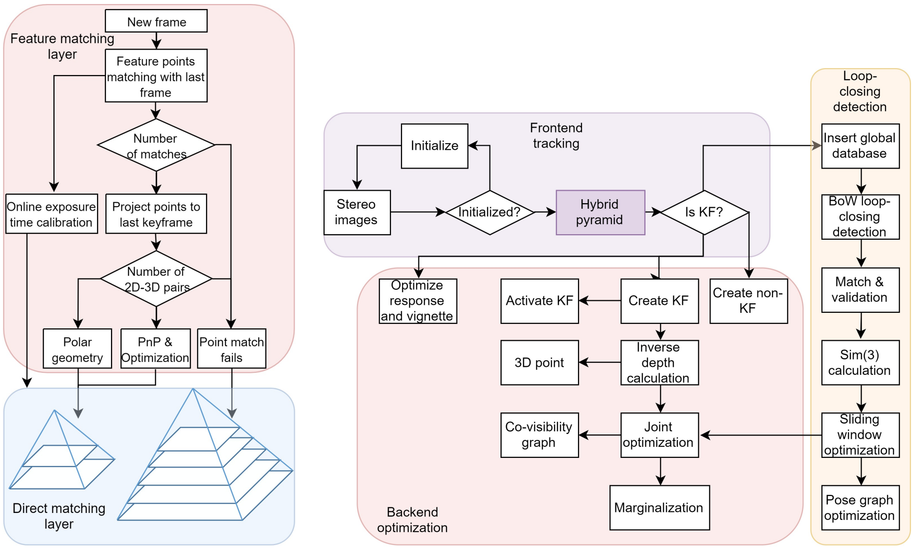
II. Related Work

In recent years, SLAM systems that combine direct and indirect methods have become popular. The SVO [17] proposed by Forster, Christian and Pizzoli belongs to the semi-direct method. It extracts feature points at the frontend, and uses optical flow to perform feature matching and pose estimation between consecutive frames. However, it still relies on geometric reprojection errors for pose optimization. Simiarly, [11] designed a method for extracting image patches at frontend, while utilizing photometric errors for backend optimization. Krombach [18], [19] used the pose obtained by a feature-based method as the initial value for the direct method. This can improve the stability of direct visual odometry. However, this method fails to reach high-precision pose estimation due to the lack of photometric sensor calibration. Kim [20] proposed a method of partitioned photometric

[1] Xiangrui Zhao ,Renjie Zheng and Wenlong Ye are with the Institute of Cyber-Systems and Control, Zhejiang University, Zhejiang, 310027, China.

[2] Yong Liu is with the State Key Laboratory of Industrial Control Technology and Institute of Cyber-Systems and Control, Zhejiang University, Zhejiang, 310027, China (Yong Liu is the corresponding author, email: yongliu@iipc.zju.edu.cn).

978-1-7281-4004-9/19/$31.00 ©2019 IEEE



(a) Pipeline of hybrid pyramid.

(b) Pipeline of the whole SLAM system.

Fig. 2. Overview

estimation, modified the photometric error function, and improved the partial illumination problem of the direct method. Younes [21] presented a method in combination of the direct and indirect methods, namely feature-based direct monocular odometry. Specifically, they presented a VO method that is based on DSO [22] but used feature-based tracking during optimization. However, in this method, when the direct method fails, it will begin to generate large calculation errors. Lee [23] proposed a loose-coupled method by combining ORB-SLAM2 and DSO to improve positioning accuracy. However, its frontend and backend are almost independent, which cannot share estimation information to further improve the pose precision.

III. Online Photometric Calibration Based on Feature Points

Our method is initially motivated by the work of [12]. Specifically, [12] proposed to extract the Shi-Thomasi corner points [24] by using gain-robust KLT algorithm and selecting good candidates for tracking. However, large exposure changes and small overlapping regions between frames are inevitable in real scenarios, which makes KLT not robust enough. Therefore, we propose to use ORB feature points. By performing descriptor based matching, the previously mentioned difficulties can be reduced.

After getting a set of points P tracked across images where point $p \in P$ is visible in frame F_p, the energy function is given by

$$E=\sum_{p\in P}\sum_{i\in F_p} w_i^p \left\| \underbrace{O_i^p - f\left(e_i V\left(x_i^p\right) L^p\right)}_{r(f,V,e_i,L^p)} \right\|_h \tag{1}$$

where w_i^P is a weighting factor for residual r, O_i^p is the output intensity of p in image i, e_i is the exposure time of image i, L_p is the radiance of p and x_i^p is the spatial location of the projection of p onto image i. We use the Huber norm $\|\cdot\|_h$ for robust estimation, parametrized by $h \in \mathbb{R}$.

When a new frame arrives, it is corrected by the prior response and vignette function. The exposure time of the new frame can be calculated by the weighted least squares method.

$$E=\sum_{i=1}^{M}\sum_{p\in R_i} w_i^p \left(\frac{f^{-1}\left(O_i^p\right)}{V\left(x_i^p\right)} - e_i L^p\right)^2 \tag{2}$$

where P_i is the set of scene points visible in the i'th image and f^{-1} is the inverse of the response function. Each residual is now only dependent on the exposure time of its frame and the radiance of its scene point. So the exposure time can be calculated efficiently.

To model the camera response function, we use the Empiric Model of Response (EMoR) model. It applies a principal component analysis to find the mean response $f_0(x)$ and basis funciton $h_k(x)$. By choosing parameters $c_k \in \mathbb{R}$, we can get $f_G(x)$ following:

$$f_G(x) = f_0(x) + \sum_{k=1}^{n} c_k h_k(x) \tag{3}$$

When modelling vignette, we assume that its center falls together with the image center. It is modelled as a sixth-order polynomial:

$$V(x) = 1 + v_1 R(x)^2 + v_2 R(x)^4 + v_3 R(x)^6 \tag{4}$$

where $R(x)$ is the normalized radius of the image point x.

IV. Hybrid Pyramid-Based Semi-direct Method

Visual odometry based on direct methods does not contain feature point extraction and matching. Therefore, in order to improve the robustness to large-baseline motion in direct visual odometry, we add feature point matching into it and propose visual odometry based on the hybrid pyramid. The whole algorithm is based on the coarse-to-fine pyramid of DSO for improvements.

A. Feature Point Extraction

Feature point extraction is the same as LDSO [25].

- We first extract more corner points like the method in DSO by using the dynamic grid gradient threshold. The maximum number of corner points set here is 2000 so that we can extract more points with different gradients in the image.
- Further extraction is performed on the obtained corner points using the Shi-Tomasi corner extraction method. The corner points are filtered by a screening formula of $R = min(\lambda 1, \lambda 2)$ to extract points with higher repeatability.
- Calculate the ORB descriptors on the points extracted in the second step and convert them into Bag-of-Words vectors.
- Repeat the above steps in mesh with different sizes.

B. Hybrid Pyramid

The direct sparse odometry uses a five-layer pyramid framework. The highest level image is used for rough calculation, and then the upper layer's result is passed to the next layer, thereby achieving coarse-to-fine calculation, getting more accurate results. It can be known from the experiment that the first few layers can only get the approximate pose transformation result, and only the last layer obtains the most accurate results.

To get prior value in the first layer of the DSO, we use zero motion, uniform motion and semi-uniform motion hypothesis. It is possible to avoid being unable to iterate to the global optimal solution caused by the long displacement under some circumstances. However, in the case of large motion baselines, such as high-speed vehicle motion, the loss of pose calculation accuracy still exists. In this section, we propose a hybrid pyramids framework to improve its robustness to large motions.

1) Feature Matching Layer:

$Step$ 1 : Feature extraction and matching. When a new frame arrives, the feature points of the current frame are matched with the previous one. 2D-2D matches are obtained. If the number of matches is greater than N_1, the process goes to the next step. If it is less than N_1, the feature point matching fails due to the environment containing repetitive textures or too few textures. Then the process skips the following steps and goes to direct alignment layer.

$Step$ 2 : Getting 2D-3D matches. The matching feature points of the previous frame are projected onto the previous keyframe by the projection relationship. Since the previous keyframe calculated the depth, we select the matching feature points with depth (greater than 0 and depth within 100 times of the baseline)

$Step$ 3 : Choosing calculation method. We need to choose different calculation methods based on the number of 2D-3D matches.

- If the 2D-3D matches are less than N_2, the feature point matching fails. The process goes to direct alignment layer.
- If the 2D-3D matches are less than N_3 and more than N_2, we use the 2D-2D polar geometry method to calculate the pose relationship between the current frame and the previous one.
- If the currently obtained 2D-3D matches are more than N_3, we use the 2D-3D PnP method to calculate the pose relationship between the current frame and the previous one.

N_1, N_2, N_3 are set as fixed parameters and satisfy $N_2 < N_3$

After obtaining the feature point matching, we simultaneously calculate the exposure time of the latest frame and pass it together to the direct alignment layer.

2) Direct Alignment Layer:

According to the results of the feature matching layer, we get the following situations:

- When the feature point matching fails, we use the zero motion, uniform motion or semi-uniform motion hypothesis to give the initial value.
- When the 2D-2D pose estimation succeeds, the rotation in the relative pose of the current moment relative to the previous moment is obtained by the eight-point method, and the translation is obtained by the uniform velocity model estimation.
- When the 2D-3D pose estimation succeeds, the relative pose of the current moment relative to the previous moment is calculated by the feature point matching optimization.

The number of matching layer is also different when performing direct multi-layer iterative alignment. When the relative pose to the previous frame can be obtained, the direct alignment only uses a two-layer pyramid. When the feature point matching fails, we use a five-layer pyramid, which ensures that the direct alignment can be optimized to the global optimal solution under different conditions, and reduces computation to improve real-time performance.

3) Hybrid Residual Function:

In this section, we show the details of the residual function for the final optimization. Because we can get the reprojection error of the feature point matches and the direct alignment photometric error, we use the hybrid residual function to improve the accuracy of the final calculation. The total residual function is given by:

$$E(\xi) = w_{rep} \left\| E_{rep} \right\|^2 + w_{photo} \left\| E_{photo} \right\|^2 \tag{5}$$

Where w_{rep} and w_{photo} are weights of the two errors set as fixed parameters. We can rewrite the cost function as:

$$\begin{aligned} E(\xi) &= \begin{bmatrix} e_u \\ e_v \\ e_{photo} \end{bmatrix}^T \begin{bmatrix} w_{rep} & & \\ & w_{rep} & \\ & & w_{photo} \end{bmatrix} \begin{bmatrix} e_u \\ e_v \\ e_{photo} \end{bmatrix} \\ &= \boldsymbol{e}^T \mathbf{W} \boldsymbol{e} \end{aligned} \tag{6}$$

where e is joint residual, W is the information matrix and ξ is the camera's pose. Based on L-M iterations, we can get $\Delta\xi$

$$\Delta \boldsymbol{\xi} = -\left(\boldsymbol{J}^T \boldsymbol{W} \boldsymbol{J} + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{J}^T \boldsymbol{W} \boldsymbol{e} \tag{7}$$

where $\boldsymbol{J}$ is the Jacobin of $\boldsymbol{e}$ and λ is the damping factor.

V. Stereo Visual SLAM Based on Feature and Direct Method Fusion

In order to implement a full stereo visual SLAM system, we complete each module of the system. The frontend calculates the pose of each frame through the hybrid pyramid visual odometry, the backend receives poses, calculates stereo scales and optimizes keyframe poses. The loop-closing detection and optimization is performed in another thread. The global pose is optimized to obtain a globally consistent trajectory and map.

As is shown in Fig. 2(b), the frontend tracking module is similar to DSO. The whole system design resembles ORB-SLAM2, which is divided into three parallel threads: pose tracking thread, sliding window optimization thread and loop-closing detection thread.

- Frontend tracking thread. The image preprocessing section includes monocular image distortion correction, stereo image parallel correction, and corner points and feature points extraction of the stereo image. Then it uses the hybrid pyramid to track and finally determine whether it is a keyframe.
- Backend sliding window optimization thread. It first performs a stereo scale estimation, restores the inverse depth of the current frame 3D point and then uses the photometric error to optimize the points and poses in the sliding window. Finally, it judges whether the frames and points in the sliding window need to be marginalized and performs marginalization.
- Loop-closing detection and optimization thread. It first uses the bag-of-words model to perform loop-closing detection. When detecting a closed loop, it performs double matching verification of 2D-2D and 2D-3D. Then it projects the verified loop-closing frame to the current sliding window and calculates the Sim(3) transformation. Finally, it performs loop-closing optimization through the pose graph to correct the pose error and global map points.

VI. Evaluation

A. Online Photometric Calibration

In this experiment, the system uses a sliding window of length 7, extracts 1000 ORB feature points per frame and selects a keyframe every five frames. Each feature point is tracked at least three frames in order to be considered successful. The tracking image block size is set to 3*3, which avoids the use of only high-gradient pixels and reduce the amount of calculation. In this experiment, the whole image is divided into multiple meshes according to the image size to extract feature points. The size of each mesh is 32*32 so that the distribution of feature points in the image is uniform. In the backend optimization, the system sets 200 frames as a block. It optimizes the exposure time, response function, and vignette function parameters of the camera every 200 frames. The exposure time uses the value estimated by the frontend as the initial value. It is optimized again in this process. In each iteration of the backend optimization, the system uses an outlier culling strategy to eliminate some wrong points continuously, thereby improving the optimization speed and accuracy.

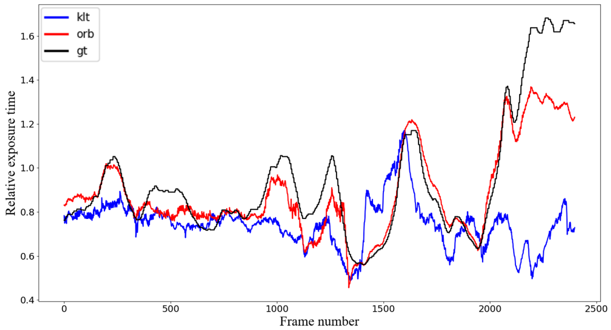
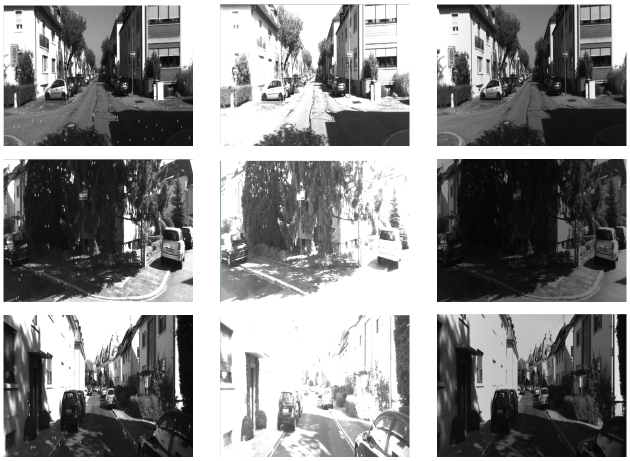


Fig. 3. Exposure time on Sequence 47.

Fig. 3 shows the estimation result and ground truth on Sequence 47. The red line is the exposure time estimation of our algorithm, the blue line is the exposure time of the KLT based photometric calibration and the black line is ground truth. It can be seen from the figure that the estimated exposure time of our algorithm is closer to the actual exposure curve. The KLT based calibration algorithm shows a large deviation in some cases with strong exposure, indicating that the photometric parameter calibration algorithm proposed in this paper has higher precision and robustness.



(a) raw (b) klt (c) orb

Fig. 4. Photometric rectification comparison

We test our online photometric calibration on KITTI dataset. Fig. 4 shows part of the rectified result.

In the original image, some parts of the picture are too intense and the pixels have been overexposed, especially at the top of the house in the distance. It is difficult to see the edge. When using the photometric correction algorithm based on KLT tracking, the photometric correction is wrong, and the entire image becomes overexposed. Using the photometric calibration algorithm proposed in this paper can reduce the image exposure so that the overexposure is well corrected and the texture of the whole image is clearer.

B. Hybrid Pyramid Tracking

We use the KITTI dataset to compare the pose proposition between frames in frontend. Most of the data is captured on a moving car in high-speed or highway environments. The motion speed is fast and the image frame rate is low. The motion baseline between two frames is large, which can verify the improvement of the large-baseline motion by the hybrid pyramid algorithm proposed in this paper. In order to reflect only the advantages of hybrid method tracking, without adding interference from other modules such as depth estimation, back-end optimization and other strategies. We accumulate the pose between two frames. The specific steps are as follows:

- Extract the FAST corner points of the current image.
- Use the stereo matching SGBM algorithm to estimate the depth of the image of the current frame, providing a corresponding depth for each extracted corner point.
- Use the 2D points of the current frame and the 3D points of the previous frame to perform pose estimation of different methods.
- Accumulate the estimated poses between the two frames obtained each time and draw the trajectories for comparison.

In this experiment, the original direct method uses a 5-layer pyramid. The parameter of the feature method refers to ORB-SLAM2. In the hybrid method, we use 300 ORB feature points in the first layer to match and use another two layers to perform the direct alignment. Other parameters are the same as the original direct method. This setting can rule out the influence of other parameters and algorithms on the pose estimation.

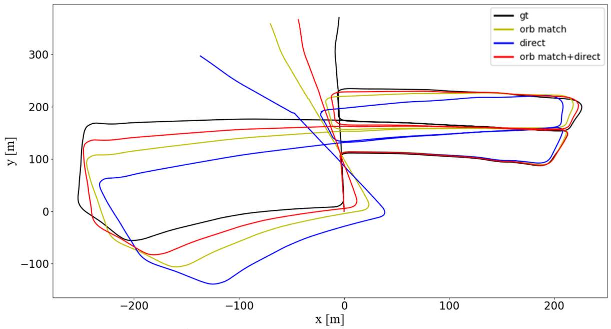


Fig. 5. Pose accumulation comparison

In Fig. 5, the yellow line is the result of the ORB feature matching method, the blue line is the original direct method and the red line is the hybrid pyramid method. The black line is ground truth in Sequence 05. It can be seen that the original direct method is less effective because of the great movement. After adding feature matching in the uppermost layer to obtain the initial value, the accuracy of direct alignment can exceed the feature matching method. The experiments show that under the same algorithm strategy without other optimizations, the pose calculated by the hybrid pyramid method proposed in this paper is more accurate. It improves the robustness of the direct method to large baseline motion.

TABLE I

AVERAGE RUNNING TIME

Method	Time
Direct method	30.3ms
Feature-based method	24.2ms
Hybrid method	35.7ms

The processor of the computing platform in the test is Intel i7-4790@3.60GHz. Table I shows the time consumption of the three methods. The hybrid pyramid method proposed in this paper takes more time than others. But it still meets the requirements of real-time.

C. Stereo Visual SLAM Based on Hybrid Pyramid

In order to verify the accuracy and robustness of the overall system, this section tests the trajectory accuracy of the entire system on public dataset KITTI and our dataset collected by a mobile robot in the campus. We compare the result with the state-of-the-art ORB-SLAM2 system.

TABLE II

COMPARISON WITH DIRECT METHOD

	Hybrid method		Direct method	
	Mean (m)	RMSE (m)	Mean (m)	RMSE (m)
Seq00	**1.02973**	**1.11851**	12.0296	13.3774
Seq01	**8.42158**	**10.9853**	46.5368	57.9955
Seq02	**3.62857**	**4.11444**	10.8594	12.7483
Seq03	**0.93416**	**1.07167**	8.32629	9.35551
Seq04	**0.42946**	**0.46587**	2.96395	3.38809
Seq05	**0.86280**	**0.98589**	7.41651	8.02779
Seq06	**1.28865**	**1.39560**	7.49934	8.34913
Seq07	**0.82309**	**0.89792**	X	X
Seq08	**2.76125**	**3.38392**	23.6450	25.6290
Seq09	**2.80135**	**3.40864**	22.0984	24.2153
Seq10	**0.66203**	**0.71849**	2.20243	2.67760

TABLE III

KITTI RESULT WITH LOOP-CLOSING

	Hybrid method		ORB-SLAM2	
	Mean (m)	RMSE (m)	Mean (m)	RMSE (m)
Seq00	**1.02973**	**1.11851**	1.13575	1.26814
Seq01	**8.42158**	**10.9853**	10.9960	11.6609
Seq02	**3.62857**	**4.11444**	5.83154	6.98676
Seq03	0.93416	1.07167	**0.68673**	**0.80080**
Seq04	0.42946	0.46587	**0.30890**	**0.34692**
Seq05	0.86280	0.98589	**0.73178**	**0.81351**
Seq06	1.28865	1.39560	**0.88633**	**0.90682**
Seq07	**0.82309**	**0.89792**	1.31268	1.49593
Seq08	**2.76125**	**3.38392**	3.42435	3.85576
Seq09	**2.80135**	**3.40864**	2.89656	3.53948
Seq10	**0.66203**	**0.71849**	1.11236	1.24230

Sequences 0-10 in KITTI provide ground truth pose. But they are not completely accurate, especially in the vertical direction. So we only compare x and y on the translation error. We run our method, ORB-SLAM2 and direct method on 11 sequences. Table II shows the average ATE and RMSE of the hybrid and direct methods proposed in this paper after six runs. The result shows that the direct method has lower precision, most of the root mean square errors are above

TABLE IV

MOBILE ROBOT DATASET

Num	Date	Time	Scene	Illumination	Weather	Dynamic scene	Direction	length(m)
1	6.1	9:18	A	Daytime	Sunny	A little	Counterclockwise	341.990
2	6.1	9:32	A	Daytime	Sunny	A little	Clockwise	325.425
3	6.5	18:16	A	Dusk	After raining	Abundant	Counterclockwise	361.765
4	6.5	18:38	B	Dusk	After raining	Abundant	Clockwise	777.568
5	6.12	17:48	A	Dusk	Cloudy	Several	Counterclockwise	406.877

TABLE V

TEST RESULT ON MOBILE ROBOT DATASET

	Hybrid method		ORB-SLAM2	
	Mean (m)	RMSE (m)	Mean (m)	RMSE (m)
Seq1	**0.35860**	**0.43696**	0.75548	0.86192
Seq2	**0.42058**	**0.45972**	1.17225	1.29146
Seq3	**0.62313**	**0.75241**	1.24543	1.33656
Seq4	**0.90563**	**1.01496**	2.28772	2.49190
Seq5	**0.36995**	**0.48760**	0.61075	0.69430

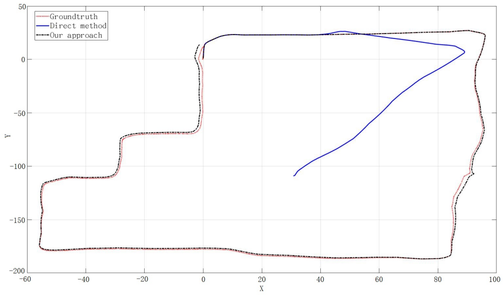


Fig. 7. The result of direct tracking failure

10m. It is because the KITTI dataset has a lower frame rate and large motion baseline. The direct method doesn't have good adaptability. The accuracy of our hybrid method is much higher than that of the direct method. Table III shows the comparison results between the proposed algorithm and ORB-SLAM2 on KITTI with loop-closing detection module. Our method proposed in this paper is equivalent to the accuracy of the state-of-the-art ORB-SLAM2 system. The accuracy on sequence 00, 01, 02, 07, 08, 09 and 10 even exceeds ORB-SLAM2.

We use five sequences of data collected by mobile robots for testing. The sequence 1, 2, 3, 5 runs under the same scene A and the sequence 4 runs under scene B. Datasets are collected at different time under different weather conditions, such as daytime, cloudy days, and even rainy days. The Table IV illustrates the specifics of the data set. As we can see from the table, great changes have taken place in different data sets, such as changes in illumination, dynamic car body movement, dynamic pedestrian movement, and weather conditions. Moreover, in order to verify whether the proposed algorithm works in different environments, we collected data in different directions of movement. Sequences 1, 3, and 5 are collected in a counterclockwise direction, and sequences 2 and 4 are collected in a clockwise direction.

From Table V we can know that the proposed algorithm in the mobile robot dataset has a much smaller ATE than ORB-SLAM2.

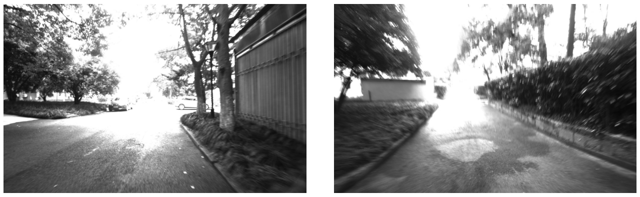


Fig. 6. Direct method fails due to light interference

As is shown in Fig. 6, the data collected by the mobile robot has a large amount of light interference. There are many cases of excessive darkness or overexposure. Some of the pedestrians and cars are overexposed because of the sunlight. In Fig. 7, the blue line is the trajectory of the hybrid algorithm proposed in this paper and the red line is the trajectory of the direct method that does not use feature point tracking. Because of the online photometric estimation algorithm based on ORB feature point tracking, the whole system can still estimate the pose normally, while the ordinary direct method runs under the influence of illumination and finally lead to the program crash. It shows that the hybrid algorithm proposed in this paper has higher illumination robustness than the original direct method.

VII. CONCLUSION

In this paper, we propose a robust stereo visual SLAM system based on the hybrid pyramid to for robust pose estimation under challenging environments. We also perform online photometric calibration to obtain better photometric parameters and apply it into visual odometry. Furthermore, We extend stereo visual SLAM system based on our hybrid pyramid frontend for evaluation. The photometric rectification test, hybrid pyramid tracking test and SLAM test on KITTI and mobile robot dataset indicate that our method is accurate and robust in challenging real-world scenarios.

In future work, we plan to integrate inertial measurement unit to deal with stronger inter-frame motion and apply local photometric calibration to eliminate regional luminosity changes.

VIII. ACKNOWLEDGEMENT

This work is jointly supported by the national natural science foundation of China under Grant U1509210, 61836015 and Alibaba-Zhejiang university joint research institute of Frontier Technologies. Also, thanks for Mingyang Li for the helpful discussion in this work.

References

[1] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 1052–1067, 2007.

[2] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pp. 225–234, IEEE, 2007.

[3] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[4] M. Li and A. I. Mourikis, "High-precision, consistent ekf-based visual-inertial odometry," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, 2013.

[5] S. Lynen, T. Sattler, M. Bosse, J. A. Hesch, M. Pollefeys, and R. Siegwart, "Get out of my lab: Large-scale, real-time visual-inertial localization.," in *Robotics: Science and Systems*, 2015.

[6] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European Conference on Computer Vision*, p-p. 834–849, Springer, 2014.

[7] J. Engel, J. Stückler, and D. Cremers, "Large-scale direct slam with stereo cameras," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 1935–1942, IEEE, 2015.

[8] D. Caruso, J. Engel, and D. Cremers, "Large-scale direct slam for omnidirectional cameras," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 141–148, IEEE, 2015.

[9] V. Usenko, J. Engel, J. Stückler, and D. Cremers, "Direct visual-inertial odometry with stereo cameras," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pp. 1885–1892, IEEE, 2016.

[10] R. Wang, M. Schwörer, and D. Cremers, "Stereo dso: Large-scale direct sparse visual odometry with stereo cameras," in *International Conference on Computer Vision (ICCV)*, vol. 42, 2017.

[11] X. Zheng, Z. Moratto, M. Li, and A. I. Mourikis, "Photometric patch-based visual-inertial odometry," in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3264–3271, 2017.

[12] P. Bergmann, R. Wang, and D. Cremers, "Online photometric calibration of auto exposure video for realtime visual odometry and slam," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 627–634, 2018.

[13] M. Li and A. I. Mourikis, "Online temporal calibration for camera–imu systems: Theory and algorithms," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 947–964, 2014.

[14] A. Censi, A. Franchi, L. Marchionni, and G. Oriolo, "Simultaneous calibration of odometry and sensor parameters for mobile robots," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 475–492, 2013.

[15] T. Schneider, M. Li, M. Burri, J. Nieto, R. Siegwart, and I. Gilitschenski, "Visual-inertial self-calibration on informative motion segments," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6487–6494, IEEE, 2017.

[16] M. Li, H. Yu, X. Zheng, and A. I. Mourikis, "High-fidelity sensor modeling and self-calibration in vision-aided inertial navigation," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 409–416, IEEE, 2014.

[17] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Proceedings of the IEEE international conference on computer vision*, pp. 1449–1456, 2013.

[18] N. Krombach, D. Droeschel, and S. Behnke, "Combining feature-based and direct methods for semi-dense real-time stereo visual odometry," in *International Conference on Intelligent Autonomous Systems*, pp. 855–868, Springer, 2016.

[19] N. Krombach, D. Droeschel, S. Houben, and S. Behnke, "Feature-based visual odometry prior for real-time semi-dense stereo slam," *Robotics and Autonomous Systems*, vol. 109, pp. 38–58, 2018.

[20] P. Kim, H. Lee, and H. J. Kim, "Autonomous flight with robust visual odometry under dynamic lighting conditions," *Autonomous Robots*, pp. 1–18, 2018.

[21] G. Younes, D. Asmar, and J. Zelek, "Fdmo: Feature assisted direct monocular odometry," *arXiv preprint arXiv:1804.05422*, 2018.

[22] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2018.

[23] S. H. Lee and J. Civera, "Loosely-coupled semi-direct monocular slam," *arXiv preprint arXiv:1807.10073*, 2018.

[24] J. Shi and C. Tomasi, "Good features to track," tech. rep., Cornell University, 1993.

[25] X. Gao, R. Wang, N. Demmel, and D. Cremers, "Ldso: Direct sparse odometry with loop closure," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2198–2204, IEEE, 2018.