Challenges in Monocular Visual Odometry: Photometric Calibration, Motion Bias, and Rolling Shutter Effect

Nan Yang, Rui Wang, Xiang Gao, and Daniel Cremers

***Abstract*—Monocular visual odometry (VO) and simultaneous localization and mapping (SLAM) have seen tremendous improvements in accuracy, robustness, and efficiency, and have gained increasing popularity over recent years. Nevertheless, not so many discussions have been carried out to reveal the influences of three very influential yet easily overlooked aspects, such as photometric calibration, motion bias, and rolling shutter effect. In this work, we evaluate these three aspects quantitatively on the state of the art of direct, feature-based, and semi-direct methods, providing the community with useful practical knowledge both for better applying existing methods and developing new algorithms of VO and SLAM. Conclusions (some of which are counterintuitive) are drawn with both technical and empirical analyses to all of our experiments. Possible improvements on existing methods are directed or proposed, such as a subpixel accuracy refinement of oriented fast and rotated brief (ORB)-SLAM, which boosts its performance.**

***Index Terms*—Localization, SLAM, performance evaluation and benchmarking.**

I. Introduction

MODERN visual SLAM systems usually have two basic components: VO and global map optimization. While the VO component incrementally estimates camera poses and builds up a local map, small errors are accumulated and over time the estimated camera poses start to drift away from their actual positions. If a previously visited location is detected, the drift can be eliminated by global map optimization using techniques like loop closure with pose graph optimization or bundle adjustment. VO, commonly considered as the system front-end, fundamentally determines the overall performance of a SLAM system. During the past few years, the VO community has seen significant progress in improving algorithm accuracy, robustness and efficiency [1]–[9]. Efforts have been made for different VO formulations, i.e., direct vs. feature-based methods, dense/semi-dense alternating optimization vs. sparse joint optimization. However, apart from these high-level diversities, it is still not clear how the performance can be influenced by the following low-level aspects:

a) Photometric calibration: Pixels corresponding to the same 3D point may have different intensities across images due to camera optical vignetting, auto gain and exposure controls.

b) Motion bias: Running a VO method on the same sequence forward and backward sometimes can result in significantly different performances.

c) Rolling shutter effect: Exposing pixels within one image at different timestamps can produce distortions that may introduce non-trivial errors into VO systems.

These three aspects can greatly affect the VO performance, yet their influences have not been systematically discussed and evaluated. In this work, we perform systematic and quantitative evaluations on the three most popular formulations of VO, namely direct, feature-based and semi-direct methods. Since evaluating all existing methods is not realistic, we select the state of the art of each family, *i.e.*, DSO [8], oriented fast and rotated brief (ORB)-SLAM [5] (with its loop closure and global bundle adjustment functionalities turned off) and SVO[9] (we use the updated version SVO 2.0). Our goal is to deliver practical insights for better applying existing methods and further designing new algorithms by giving insightful technical and empirical analyses to all of our experimental results. Our main contributions are summarized as follows:

1) While it has been shown in [8], [10] that photometric calibration can significantly improve the performance of direct methods, it is still unclear how or why it can influence other formulations. We complete this discussion by performing thorough evaluations on all the three selected methods, draw counter-intuitive conclusions and analyze the possible reasons.

2) Although motion bias was unveiled in [10], the problem was not studied there at all. In this work we exhaustively discuss the problem, analyze the reasons and perform experiments that support our conclusions.

3) In [8] the rolling shutter effect was tackled partly and indirectly by simply mimicking the effect using random pixel shifting. In this work we carry out evaluations on dataset that provides both global and simulated rolling shutter sequences. Besides, we further evaluate the selected methods on modern

Manuscript received February 24, 2018; accepted May 31, 2018. Date of publication June 13, 2018; date of current version June 25, 2018. This letter was recommended for publication by Associate Editor U. Frese and Editor C. Stachniss upon evaluation of the reviewers' comments. (Nan Yang and Rui Wang contributed equally to this work). *(Corresponding authors: Nan Yang and Rui Wang.)*

N. Yang, R. Wang, and D. Cremers are with the Chair for Computer Vision and Artificial Intelligence, Department of Informatics, Technical University of Munich, Garching 85748, Germany, and also with Artisense, Palo Alto, CA 94306 USA (e-mail: yangn@in.tum.de; wangr@in.tum.de; cremers@in.tum.de).

X. Gao is with the Chair for Computer Vision and Artificial Intelligence, Department of Informatics, Technical University of Munich, Garching 85748, Germany (e-mail: x.gao@in.tum.de).

Digital Object Identifier 10.1109/LRA.2018.2846813

industrial level cameras, which normally have rolling shutters but extremely fast readout speed.

4) In all the related experiments in [8], [10], only direct and feature-based methods were considered. In this work we add the popular representative of semi-direct methods [7], [9] to all our evaluations.

5) We propose possible improvements of existing methods, e.g., a sub-pixel accuracy refined version of ORB-SLAM delivering boosted performance.

II. Related Work

In this section we briefly introduce the principles of the three VO formulations together with their respective selected representatives. Afterwards we list the datasets used for our experiments.

A. Direct Methods

Direct methods use either all pixels (dense) [6], pixels with sufficiently large intensity gradient (semi-dense) [4], or sparsely selected pixels (sparse) [8] and minimize a photometric error obtained by direct image alignment on the used pixels, based on the brightness constancy assumption. Camera poses and pixel depths are estimated by minimizing the photometric error using non-linear optimization algorithms. Since much image information can be used, direct methods are robust in low-texture scenes and can deliver relatively dense 3D reconstructions. Consequently, due to the direct image alignment formulation, direct methods are very sensitive to unmodeled artifacts such as rolling shutter effect, camera auto exposure and gain control. More crucially, the brightness constancy assumption does not always hold in practice, which drastically reduces the performance of direct methods in environments with rapid lighting change.

Direct Sparse Odometry (DSO): DSO performs a novel sparse point sampling across image areas with sufficient intensity gradient. Reducing the amount of data enables real-time windowed bundle adjustment (BA). Obsolete and redundant information is marginalized with the Schur complement [11], and the First Estimate Jacobians technique is involved in the non-linear optimization process [11], [12] to avoid mathematical inconsistency. As a direct method, DSO is fundamentally based on the brightness constancy assumption, thus the authors proposed a photometric camera calibration pipeline to recover the irradiance images [8], [10], which drastically increases the tracking accuracy [8]. An example of photometric calibration is shown in Fig. 1.

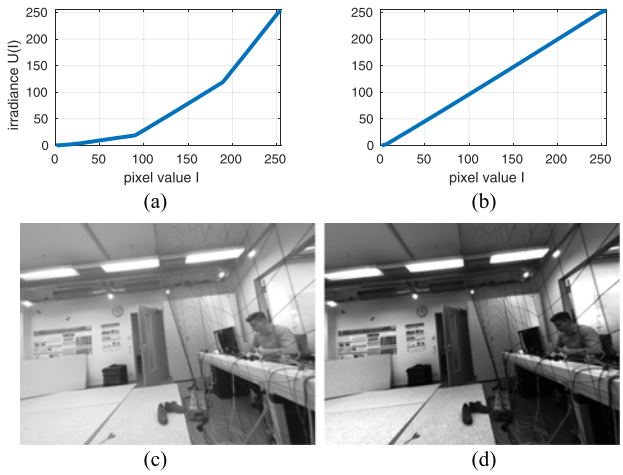


Fig. 1. Example of photometric calibration. Camera response functions with gamma correction (a) on andf (b) of, images (c) before and (d) after photometric calibration. (a) Response with gamma correction. (b) Response without gamma correction. (c) Original image. (d) Calibrated image.

B. Feature-Based Methods

Feature-based methods extract a sparse set of key-features and match them across multiple frames. Camera poses and feature depths are estimated by minimizing the reprojection errors between feature pairs. As modern feature descriptors are to some extent invariant to illumination and view-point changes, feature-based methods are more robust than direct methods to brightness inconsistencies and large view-point changes. However, feature extraction and matching bring additional computational overhead, which limits the number of features that can be maintained in the system. The reconstructed 3D maps therefore are much sparser and cannot be used directly for applications like obstacle avoidance and path planning. Moreover, in low-texture environments where not enough features can be extracted, tracking can easily get lost.

ORB-SLAM: ORB-SLAM has become one of the most popular feature-based methods and has been widely adopted for a variety of applications. It uses ORB features [13] for all the tasks including tracking, mapping, re-localization and loop closing. To track a new frame, motion-only BA is performed on its feature matches to estimate the initial pose, which is later refined by using all the feature matches in the local map and performing the pose optimization again. A covisibility graph is used to improve system efficiency by limiting the BA to a local covisible area. Unlike in DSO, in ORB-SLAM old points and keyframes are culled out directly from the active window without marginalization. To evaluate its VO performance, we disable its loop closure and global BA functionalities, and only focus on its tracking and local mapping components in this letter.

C. Semi-Direct Methods

Semi-Direct Visual Odometry (SVO): Semi-direct methods have been considered to be a hybrid of the two previously mentioned formulations. SVO [7] extracts Fast corners and perform direct image alignment on those areas for initial pose estimation. The feature extraction is later extended to also include edgelets [9] or line segments [14] to improve the robustness. The depths of the selected pixels are estimated from multiple observations using a recursive Bayesian depth filter. To reduce the drift caused by incremental estimations, poses and depths are refined by BA: Image patches from the reference frame and the current frame are aligned using an inverse compositional Lucas-Kanade algorithm, then the reprojection error is computed and minimized in BA using iSAM2 [15]. Due to its

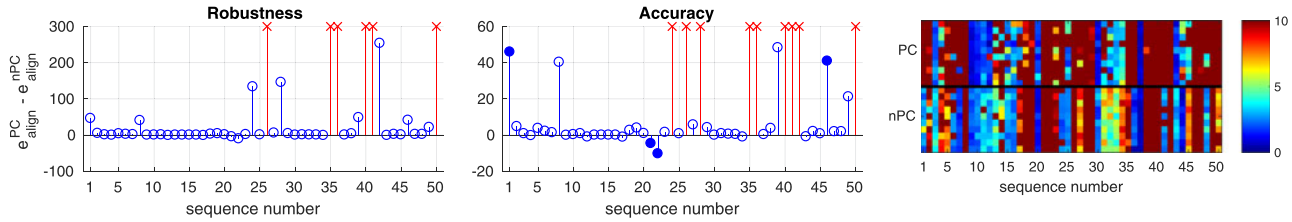


Fig. 2. Left: Performance difference of ORB-SLAM on the TUM Mono Dataset. The average of $e_{\text{align}}^{PC} - e_{\text{align}}^{nPC}$ is shown for each sequence, where e_{align}^{PC} and e_{align}^{nPC} stand for the alignment error with and without photometric calibration, respectively. The 6 sequences on which tracking completely fails after photometric calibration are marked in red. Middle: An enlarged view of the first plot. Sequences with performance differences larger than 50 are marked in red. Solid blue dots are used to mark the 4 sequences shown in Fig. 4. Right: Color-encoded errors of all runs. Data of ORB-SLAM without photometric calibration is obtained from [10].

exceptional high efficiency (around 400 fps on a laptop) and low cost, SVO can be transplanted to devices with limited computational resources, thus has gained a high popularity in a wide range of robotics applications.

D. Datasets

The following datasets are used for our experiments which cover a variety of real-world settings, e.g., indoor/outdoor, texture/textureless, global/rolling shutters.

The TUM Mono VO Dataset: [10] contains 50 sequences captured by a global shutter camera with two different lenses. Camera response function, dense attenuation factors and exposure time of each image are provided for photometric calibration.

The EuRoC MAC Dataset: [16] contains 11 sequences recorded by global shutter cameras mounted on a drone. Some of the sequences are quite challenging as they have extremely unstable motion and strong brightness change.

The ICL-NUIM Dataset: [17] has been extended by Kerl *et al.* [18] to provide both simulated rolling shutter and global shutter sequences of the same indoor environment. We use it for our experiments related the rolling shutter effect.

The Cityscapes Dataset: [19] provides a long street view sequence captured by industrial rolling shutter cameras, which is used to evaluate how the selected methods work against realistic rolling shutter effect.

III. Evaluation

A. Photometric Calibration

In the first experiment, we evaluate the influence of photometric calibration on the selected methods, focusing more on analyzing its impacts on formulations other than direct method. We use the 50 original sequences from the TUM Mono VO Dataset and their corresponding ones after photometric calibration, i.e., with the nonlinear camera response function G and pixel-wise vignetting factors V calibrated. Each method runs 10 times on each of these 100 sequences to account for non-deterministic behavior caused by e.g., multi-threading. The accumulative histogram (i.e., the number of runs that have errors less than the value given on the x axis) of the alignment error e_{align}[1] (meter),

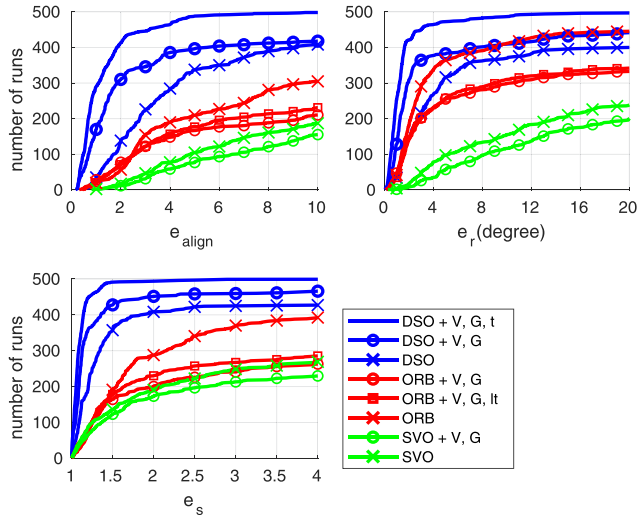


Fig. 3. Performance comparison of DSO, ORB-SLAM, and SVO on sequences with/without photometric calibration (camera response function G, vignetting factors V). The alignment error e_{align}, rotation drift e_r (in degree) and scale drift e_s are shown in the corresponding subplots. "lt" stands for loosed thresholds for ORB feature extraction. For reference, we also show results of DSO using camera exposure times t.

rotation drift e_r (degree) and scale drift e_s are calculated [10] and shown in Fig. 3. It is worth noting that integrating the exposure times t into the formulation of ORB-SLAM and SVO (not open-sourced) is not straightforward, therefore we do not use them for all three methods. For reference, we also show the results of DSO with all calibration information used, i.e., G, V and t.

In this experiment, with G and V calibrated the performance of DSO increases significantly, which is not surprising as direct methods are built upon the brightness consistency assumption. Interestingly, photometric calibration reduces the overall performance of SVO, and for ORB-SLAM the performance decline is even larger. As both SVO and ORB-SLAM extract FAST corners, we suspect the feature extraction and feature matching of these methods are influenced by the photometric calibration.

To better understand the results in Fig. 3, we further show the performance of ORB-SLAM on each sequence in Fig. 2, where the differences of the alignment errors with/without photometric calibration $e_{\text{align}}^{PC} - e_{\text{align}}^{nPC}$ are shown in the left and middle, alignment errors of all runs are shown in the right. ORB-SLAM

[1] e_{align} is the translational RMSE between the tracked trajectory when aligned to the start and the end segments of the ground truth trajectory. For details please refer to [10, eq. (14)].

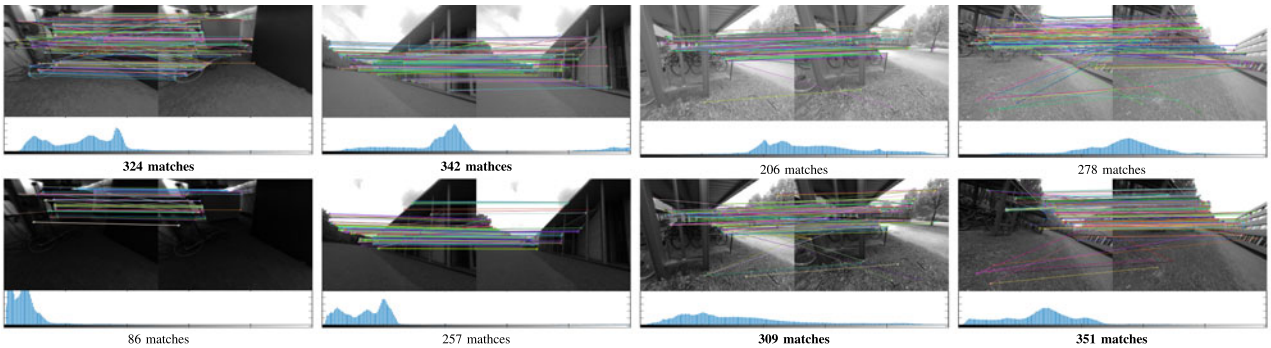


Fig. 4. Performance differences of ORB feature matching between consecutive frames before and after photometric calibration. Top: before calibration. Down: after calibration. Histogram of the left image is shown under each image pair. As can be seen here, after photometric calibration the numbers of matches decrease a lot on dark images, while increase significantly on bright images.

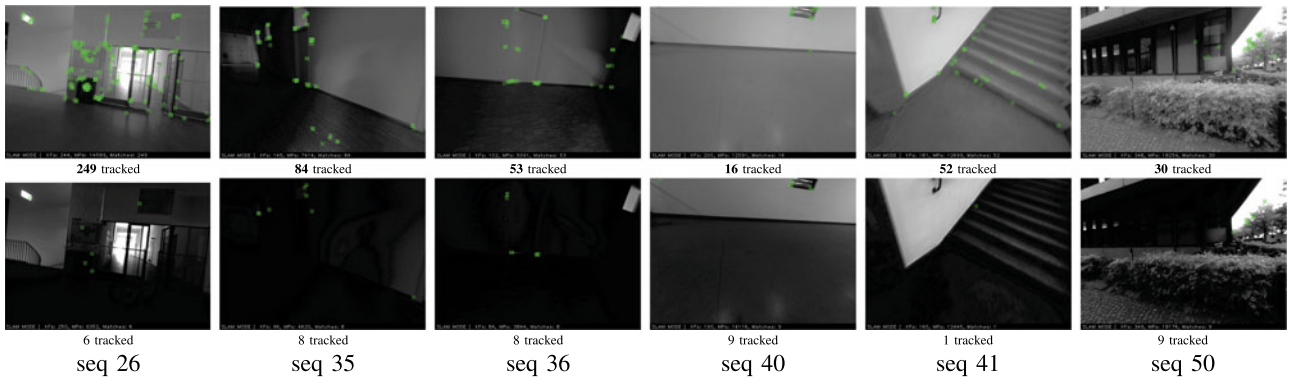


Fig. 5. Failure cases of the 6 sequences mentioned in the left of Fig. 2 where ORB-SLAM lost its tracking. Top: before photometric calibration. Down: after photometric calibration. Features shown in the images are the ones projected from the active local map to the current frame and successfully matched there. After photometric calibration, dark images become even darker and not enough features can be matched on them.

fails on 6 sequences and generally performs worse on the other sequences. However, the performance decline is not consistent over sequences. By rechecking the inverse camera response function G^{-1} in Fig. 1(a), we find the nonlinear function can be roughly divided into three linear parts with pixel values I belonging to $[0, 90)$, $[90, 190)$ and $[190, 255]$. Due to the different slopes, applying G^{-1} compresses intensities in $[0, 90)$ and stretches the ones in $[190, 255]$. In other words, it reduces contrast of dark areas while increases it for bright areas. As features like FAST and ORB generally work better on images with higher contrast (more evenly distributed intensity histogram), we further suspect that the performance declines of SVO and ORB-SLAM are mainly caused by dark frames.

To verify this another experiment is carried out: we extract ORB feature and match them on image pairs before and after being photometrically calibrated. Example results are shown in Fig. 4, where two of them are with dark images and the other two with bright ones. The numbers of feature matches and image histograms are shown under each image pair. As can be seen there, after photometric calibration the numbers of ORB feature matches decrease on the dark image pairs and increase on the bright ones. Although sometimes the drop of the numbers may not seem crucial (e.g., the second column of Fig. 4), the effect can be accumulated over multiples frames. In Fig. 5 we show how the number of matches can drastically decrease when the system projects all features within the local map to the newest frame to search for correspondences. As a result, only few features from the newest frame will be considered as inliers and added into the system, which is the main reason for the tracking failures in Fig. 2.

We also try to loose the threshold for FAST extraction on the calibrated images. As shown in Fig. 3, although it slightly improves the performance, it is still not comparable to the performance on the original images. The reason is that feature extraction with lower threshold delivers more noisy and unstable features. Moreover, as the images are internally represented by 8-bit unsigned integers, compressing the dynamic range of dark areas aggravates the discretization effect. The feature descriptors thus become less distinguishable, which corrupts the feature matching.

Recall that SVO extracts FAST corners and edgelets, the photometric calibration used in our experiments can reduce the number of successful extractions on dark image areas. On the other hand, instead of matching features, SVO matches image patches around those corner or edgelets, which is similar to direct image alignment and thus is less sensitive to the reduced intensity

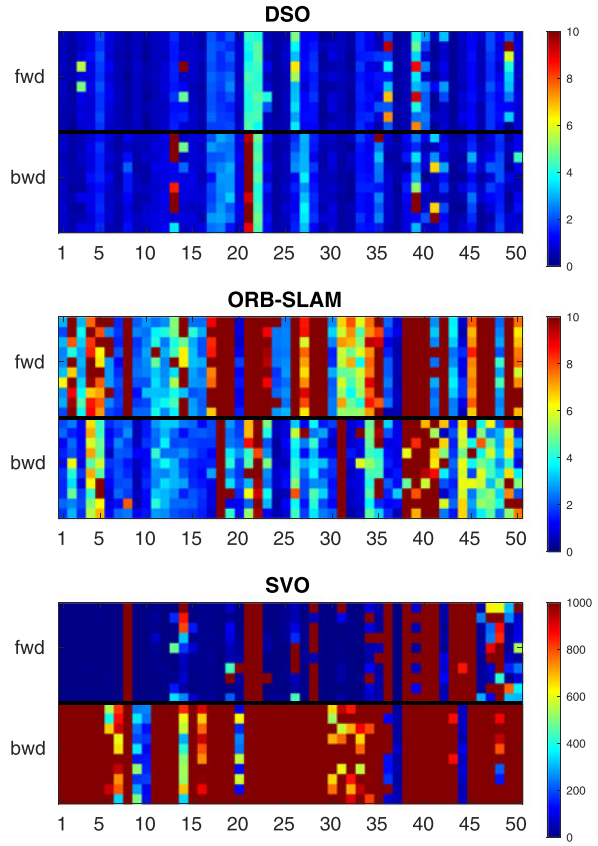


Fig. 6. Results on TUM Mono VO Dataset running forward and backward. Each method runs 10 times forward and 10 times backward on each sequence. The alignment errors e_{align} are color coded and shown as small blocks. For the first two plots we use the results obtained in [8]. Note that for the results of SVO we use a different scale on the errors.

contrast. Moreover, SVO performs direct image alignment for initial pose estimation, to which photometric calibration is supposed to be beneficial. We believe these are the reasons for the reduced performance declines compared ORB-SLAM.

B. Motion Bias

The term motion bias here refers to the difference of VO performance caused by running the same sequence forward and backward. Note that this is different from those studied in [20]–[22]. As shown in the top two plots in Fig. 6, experiments in [8], [10] demonstrate that DSO does not suffer from such bias, but ORB-SLAM performs better when running backward. While this issue was raised there, no analysis, conclusion or possible remedies were given. To get a more thorough understanding of motion bias, we first perform the same experiment for SVO and show the result in Fig. 6. Surprisingly, SVO does not perform very well on this dataset and it gets very large alignment errors for all backward runs (note that we already use the settings recommended by one author of SVO 2.0). Generally speaking, the TUM Mono VO is a quite challenging dataset for monocular VO as it contains a lot of poorly textured areas. We suspect this is the main reason for the obtained results. However, as SVO 2.0 is not open-sourced, we cannot analyze further.

Fig. 7. Sample sequences from the TUM Mono VO Dataset, on which ORB-SLAM has the largest motion bias [10]. The first 5 images show scenarios where motion bias can happen. The end part of *sequence 31* is shown in the last image. Such high frequency textures (leaves) are challenging for initialization when running backward.

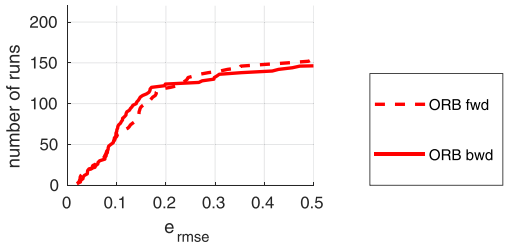


Fig. 8. Performance of ORB-SLAM on EuRoC MAV Dataset running forward and backward.

We exclude SVO from the remaining experiments on the TUM Mono VO Dataset in this section.

Both for direct and feature-based methods, triangulation is a necessary step for estimating depths of newly observed 3D points. Despite the cases with pure rotational camera motions, better depth estimation usually can be achieved with larger disparity between an image pair. When the camera is moving forward in a relatively open area, new points will emerge from the image center and have relatively small motions among consecutive frames. This pattern of optical flow introduces poorly initialized depths into the system. On the contrary, when moving backward, points close to the camera come into the field of view with large parallaxes, thus their depths are better initialized. We claim this is the main reason for the improved performance of ORB-SLAM when running backward.

To verify this, we check the sequences on which ORB-SLAM performs significantly better running backward and show them in the first 5 subfigures in Fig. 7. It can be seen that all of them fulfill our description above. We also check the two counter examples, *sequence 31* and *44*, on which ORB-SLAM performs better running forward. At the end part of *sequence 31* there is a large amount of high frequency textures (leaves) as shown in the last image in Fig. 7, which makes ORB-SLAM not able to initialize or fail directly after the initialization when running backward. In *sequence 44*, interestingly, the camera is moving most of the time backward, which in fact verifies our conclusion.

We also run ORB-SLAM on the EuRoC MAV Dataset, where the sequences are captured in relatively closed indoor

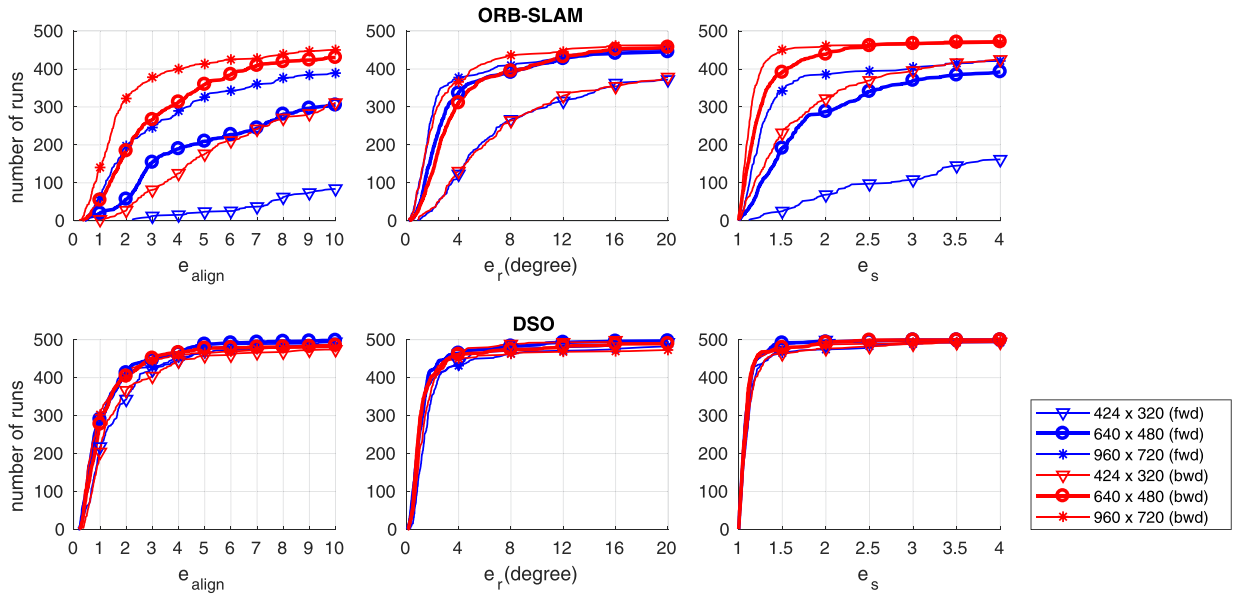


Fig. 9. Performance differences of ORB-SLAM (top) and DSO (down) on the TUM Mono VO Dataset due to motion bias at different image resolutions. While DSO delivers similar results under different settings, ORB-SLAM performs consistently better when running backward and the performance gaps increase with reduced image resolutions.

environment and the camera motion is rather diverse without any clear pattern. We assume ORB-SLAM should deliver similar results running forward and backward. The result is shown in Fig. 8 and it verifies our assumption.

The analysis above does not explain why DSO performs consistently running forward and backward. We claim the performance gain of DSO mainly comes from its implementation and feature-based or semi-direct methods can be improved taking into consideration the following issues:

a) Depth representation: Instead of using depth directly like ORB-SLAM, DSO uses an inverse depth parametrization that affects the validity range of linearization and can better cope with distant features [23]. We thus claim that the distant points, which are poorly initialized from the image center, have less impact on DSO.

b) Point sampling strategy: DSO samples points evenly across the entire image, which can be beneficial to avoid selecting many points from locations that only give poor initializations (e.g., image center).

c) Point management: In ORB-SLAM, features extracted from a new frame will be added into the system, if they can match those features that are already in the window but haven't been matched before. If all these features gather together at the image center, they will be added with inaccurate depth estimations. In contrary, DSO only samples candidate points from the new frame but does not add them to the system immediately. The depth estimations of these points keep being refined (outliers are removed) before they are activated and added. Moreover, points are only selected to be activated if they can keep the uniform spatial distribution of all activated points. All these strategies prevent problematic points from being added into the system.

d) Discretization artifacts: In direct methods, the depth of a newly observed point is initialized by searching for its correspondence in the reference frame along the epipolar line using sub-pixel accuracy. In feature-based methods, however, a new feature is extracted and matched to a previously observed feature with both of them at discretized image locations. Thus feature-based methods suffer one time more from pixel discretization artifact. The effect becomes more severe when matching those distant features emerging from the image center running forward. To verify our analysis, we first perform the experiment in Fig. 9 where we run DSO and ORB-SLAM forward and backward on sequences sampled to different resolutions. The performance of DSO drops a little on low resolution sequences, but overall it is robust to such artifact. In contrast, the performance gaps of ORB-SLAM between running forward and backward increase significantly with reduced resolutions (thus severer discretization artifact).

In our second experiment, we adopt a sparse optical flow algorithm to refine the feature matching step of ORB-SLAM to achieve sub-pixel precision. We use the iterative Lucas-Kanade method implemented in OpenCV and run the refined ORB-SLAM on the first 5 sequences shown in Fig. 7. The result is shown in Table I. ORB-SLAM performs similarly running backward as before but much better (more than 50% on average) running forward, which supports our analysis. For reference we also show the results on all the sequences in Fig. 10.

C. Rolling Shutter Effect

In the first experiment of this section, we run DSO, ORB-SLAM and SVO 10 times on each of the 4 *Living Room* sequences of the ICL-NUIM Dataset, as well as on their simulated rolling shutter correspondences. Although currently this is the only dataset that provides global shutter and rolling shutter sequences at the same time, it is worth noting the simulated rolling shutter effect is relatively strong. The results are shown in Fig. 11(a). All three methods are influenced by rolling

TABLE I
RESULTS OF ORIGINAL ORB-SLAM AND OUR REFINED ORB-SLAM

	e_{align}^{fwd}		e_{align}^{bwd}	
Seq.	Original	Refined	Original	Refined
17	12.29	**3.05**	1.50	**1.33**
23	10.33	**5.52**	3.18	**1.94**
29	21.84	**10.52**	**2.24**	2.58
46	27.18	**14.89**	**5.10**	5.27
47	20.57	**10.85**	**4.58**	5.80
mean	18.44	**8.97**	**3.32**	3.38

We run both methods 10 times on each of the selected sequences from Fig. 7. The data of original ORB-SLAM is obtained from [10]. Sub-pixel refinement significantly improved the performance of forward run, while did not help so much with the backward run, which verifies our conclusion.

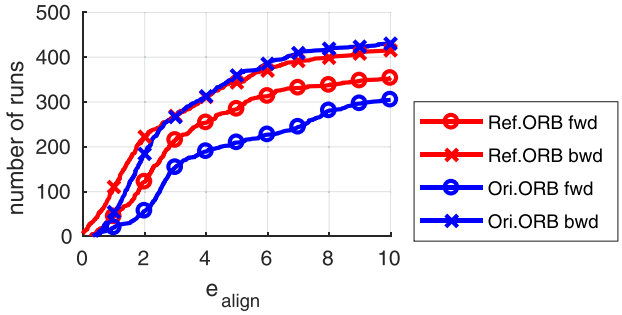
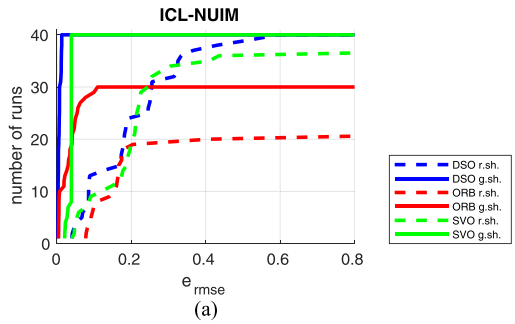


Fig. 10. Performance comparison between the original ORB-SLAM and our refined version on the full TUM Mono VO Dataset. With sub-pixel accuracy refinement of feature matching, ORB-SLAM performs better running forward and similarly running backward.

shutter effect, yet the performance declines of DSO and SVO are apparently larger than ORB-SLAM. This result verifies that feature-based methods are more robust to the rolling shutter effect than direct methods. To show the influence of the rolling shutter effect on direct methods, we show examples of the reconstructed scene by DSO in Fig. 11(b) and Fig. 11(c). It can be seen that the delivered reconstruction has very large scale drift on the rolling shutter sequence (the big structure in the background of Fig. 11(c) is the drifted reconstruction of the painting in the foreground).

Although SVO performs feature matching followed by BA for refining structures and poses, in the initial pose estimation for each frame it uses direct image alignment, thus does not use the correspondences estimated by feature matching. This explains its performance decline is larger than that of ORB-SLAM. It is also worth mentioning, as can be seen in Fig. 11(a), that the overall performances of DSO and SVO on this dataset significantly transcend the one of ORB-SLAM under both global and rolling shutter settings. The main reason is that the scenes in this dataset are indoor environments with low textured structures such as walls, floors and doors and thus are very challenging for corner-based feature extraction. Due to the fact the SVO is able to use image information on edges, it gains certain robustness on this dataset. As a result, the selected direct and semi-direct methods outperform the feature-based method even on rolling shutter images.



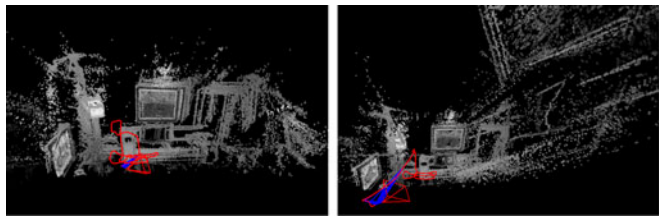


Fig. 11. Results on the extended ICL-NUIM Dataset with original global shutter setting and the simulated rolling shutter setting. (a) Performances of the selected methods. (b) DSO, *lr kt0* global shutter. (c) DSO, *lr kt0* rolling shutter.

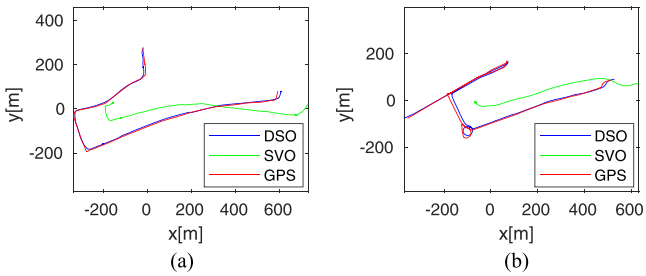


Fig. 12. Estimated trajectories of the segments of the *Frankfurt* sequence from the Cityscapes Dataset. The used frames are shown below each plot. Estimated poses are aligned to GPS coordinates with 7D similarity transformation. Note that the provided GPS coordinates are not accurate. (a) Frame 27001-33000. (b) Frame 87001-93000.

While the results above coincide with our intuition, it sometimes can be misleading. One may easily draw the conclusion that on sequences with enough texture and captured using rolling shutter cameras, feature-based methods should be preferable than direct or semi-direct methods. This is not always the case. Recall that the rolling shutter effect in the extended ICL-NUIM Dataset is artificially simulated. On modern industrial level cameras, pixel read-out speeds are usually extremely fast such that the rolling shutter effect is to some extent neglectable for many applications. In the second experiment, we aim at comparisons on images with such realistic rolling shutter effect. As there is no such dataset that provides both real global shutter and rolling shutter sequences, we only compare the VO accuracies on realistic rolling shutter sequences. For this purpose, we use the *Frankfurt* sequence of the Cityscapes Dataset and split it into smaller segments (each with around 6000 frames). To our surprise, ORB-SLAM always fails on the selected segments: whenever the camera rotates strongly at street corners or large

occlusion occurs due to moving vehicles, which has also been reported by other users of ORB-SLAM. We thus suspect the failures are not related to the rolling shutter effect. In Fig. 12 we show the estimated camera trajectories of DSO and SVO. Although SVO suffers more from scale drift, both the direct and semi-direct methods are able to track on the entire selected segments. The last thing to point out is, without a proper dataset, it is still difficult to analyze the exact influence of the rolling shutter effect on existing VO methods.

IV. Conclusion

We present a thorough evaluation for state-of-the-art direct, semi-direct and feature-based methods on photometric calibration, motion bias and the rolling shutter effect, with the aim of providing practical inputs to the community for better applying existing methods and developing new VO and SLAM algorithms. Our main conclusions are:

1) With photometric calibration, the performance of direct methods gets improved significantly, while for semi-direct and feature-based methods, it depends on the used feature, the camera response function and the overall brightness of the scene. Ideally active camera control [24] should be applied to deliver the feature extractor and matcher with images of good quality. For direct methods, when photometric calibration information is not available, online calibration methods [25] should be used.

2) Compared to direct methods, feature-based methods have a relatively large performance bias when running forward and backward. Possible reasons are discussed: depth representation, point selection and management, discretization artifact. When adopting existing feature-based methods for applications like autonomous driving, more effort should be taken to address the motion bias.

3) Direct and semi-direct methods are more sensitive to the rolling shutter effect. But when the rolling shutter effect is not strong, or the environment is low textured, the rolling shutter effect might not be the deciding factor on performance anymore. When the pixel readout speed is fast enough, even direct methods can deliver satisfying results. Besides, a specific dataset is needed for getting a better understanding on the rolling shutter effect.

4) The used feature-based methods are more sensitive to pixel discretization artifact. When possible, images with higher resolutions are preferable. Moreover, sub-pixel accuracy refinement on feature extraction and matching can boost their performance, which is verified by our sub-pixel refined version of ORB-SLAM.

Acknowledgment

The authors would like to thank J. Engel and J. Stückler for the insightful discussions, and also thank Z. Zhang to recommend the settings of SVO 2.0 for the evaluation on the TUM Mono VO dataset.

References

[1] M. Irani and P. Anandan, "All about direct methods," in *Proc. Int. Workshop Vis. Algorithms*, 1999, pp. 267–277.
[2] P. H. Torr and A. Zisserman, "Feature based methods for structure and motion estimation," in *Proc. Int. Workshop Vis. Algorithms*, 1999, vol. 1883, pp. 278–294.
[3] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Proc. 6th IEEE ACM Int. Symp. Mixed Augmented Reality*, Nara, Japan, Nov. 2007, pp. 225–234.
[4] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2014, pp. 834–849.
[5] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, 2015.
[6] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2320–2327.
[7] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 15–22.
[8] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 611–625, Mar. 2018.
[9] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "SVO: Semidirect visual odometry for monocular and multicamera systems," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 249–265, Apr. 2017.
[10] J. Engel, V. Usenko, and D. Cremers, "A photometrically calibrated benchmark for monocular visual odometry," 2016. [Online]. Available: arXiv:1607.02555
[11] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 314–334, 2015.
[12] G. Huang, A. Mourikis, and S. Roumeliotis, "A first-estimates jacobian EKF for improving SLAM consistency," in *Proc. Exp. Robot.*, 2009, pp. 373–382.
[13] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. 2011 Int. Conf. Comput. Vis.*, Washington, DC, USA, 2011, pp. 2564–2571. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2011.6126544
[14] R. Gomez-Ojeda, J. Briales, and J. Gonzalez-Jimenez, "Pl-SVO: Semi-direct monocular visual odometry by combining points and line segments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 4211–4216.
[15] M. Kaess, H. Johnnsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *Int. J. Robot. Res.*, vol. 31, pp. 217–236, 2012.
[16] M. Burri *et al.*, "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, pp. 1157–1163, 2016.
[17] A. Handa, T. Whelan, J. McDonald, and A. Davison, "A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM," in *Proc. IEEE Intl. Conf. Robot. Autom.*, Hong Kong, China, May 2014, pp. 1524–1531.
[18] C. Kerl, J. Stueckler, and D. Cremers, "Dense continuous-time tracking and mapping with rolling shutter RGB-D cameras," in *Proc. IEEE Int. Conf. Comput. Vis.*, Santiago, Chile, 2015, pp. 2264–2272.
[19] M. Cordts *et al.*, "The Cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.
[20] G. Dubbelman and F. C. Groen, "Bias reduction for stereo based motion estimation with applications to large scale visual odometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2222–2229.
[21] G. Dubbelman, P. Hansen, and B. Browning, "Bias compensation in visual odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 2828–2835.
[22] S. Farboud-Sheshdeh, T. D. Barfoot, and R. H. Kwong, "Towards estimating bias in stereo visual odometry," in *Proc. Can. Conf. Comput. Robot Vis.*, 2014, pp. 8–15.
[23] J. Civera, A. J. Davison, and J. M. Montiel, "Inverse depth parametrization for monocular SLAM," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 932–945, Oct. 2008.
[24] Z. Zhang, C. Forster, and D. Scaramuzza, "Active exposure control for robust visual odometry in HDR environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 3894–3901.
[25] P. Bergmann, R. Wang, and D. Cremers, "Online photometric calibration of auto exposure video for realtime visual odometry and SLAM," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 627–634, Apr. 2018.