16th International Conference on Machine Vision Applications (MVA)
National Olympics Memorial Youth Center, Tokyo, Japan, May 27-31, 2019.

Revisiting Visual Odometry for Real-Time Performance

Gaurav Singh, Meiqing Wu and Siew-Kei Lam
Nanyang Technological University Singapore
{ gaurav012@e.ntu.edu.sg, meiqingwu@ntu.edu.sg
and siewkei_lam@pmail.ntu.edu.sg }

Abstract

Visual Odometry (VO) is a key component in modern driver assistance systems and robotics. Meeting the real-time requirements is mandatory for VO in such applications. Previous works have primarily focused on improving accuracy at the cost of longer runtime. In this work, we propose novel strategies for feature correspondence setup, outlier removal and robust pose optimization in the VO pipeline to achieve real-time performance of close to 30 frames-per-seconds (fps) on a dual-core 3.5 GHz CPU while maintaining high accuracy. In particular, computationally efficient strategies are introduced to obtain an initial set of good features and rapidly filter out the outliers to minimize the computational overhead in later stages. In addition, we propose a depth based weighting and saturated-residual scheme during pose optimization to increase the robustness of VO. Experimental results show that the proposed VO achieves the fastest speed among all the top-ranked VO and SLAM systems on KITTI leader-board. Specifically, the proposed VO is 47% faster than state-of-the-art ORB-SLAM2 with comparable accuracy on KITTI dataset.

1 Introduction

Visual Odometry (VO) can be regarded as motion estimation of an agent based on images that are taken by the camera/s attached to it [10]. VO is the key component of modern driver assistance systems and autonomous driving systems [21, 11]. The two main requirements of VO are pose accuracy and speed. Real-time performance of VO is equally important as the pose accuracy; particularly for safety-critical applications such as collision avoidance. Achieving real-time performance is challenging as VO is typically deployed on embedded systems with tight computational constraints. State-of-the-art systems have achieved a pose error rate around 1% [16, 6, 4], but at the cost of high runtime. None of the existing top-performing VO in the KITTI leader-board [2] are capable of achieving real-time performance[1].

1.1 Related Work

We focus our discussion on feature based motion-estimation using stereo-camera due to its suitability to outdoor environments, low computation complexity and a wide range of applicability [19, 5, 21]. Feature-based methods use distinct image features such as SIFT [15], ORB [18], etc., to estimate the pose by minimizing reprojection errors. They usually consist of feature correspondence setup, outlier removal and pose optimization [10]. Feature correspondence setup extracts features from each incoming frame and establishes the feature correspondences/matches between frames. It is the most crucial step because inaccurate matches largely affect the pose estimates. It is also the most time-consuming part of VO. Some methods rely on large number of feature matches or complex features to improve the accuracy such as MFI [3]. Viso2 [13] observed that the distribution of features is very important and thus, it employs bucketing to obtain better-distribution. Viso2 utilized simple features and achieved fast tracking time (0.05 secs/frame on 1 core-2.5 GHz), but with large translation errors (2.4%).

Feature matches are affected by image noise, false matching and moving objects, which are usually referred to as outliers. Outliers can degrade the pose estimation accuracy significantly. To identify and remove the outliers, existing systems [3, 13, 16] rely on outlier removal methods such as RANSAC [17, 9], MLESAC [1] and iterative outlier removal [3, 16]. These methods are iterative and involve several operations; hence, they are computationally complex. Also, they rely on fixed thresholds for removing outliers which limits their effectiveness when scene changes. The RotROCC [4] indicated that outlier detection based on a fixed threshold is inappropriate and proposed optical flow based scaling of the reprojection errors. Although RotROCC [4] and SOFT [6] have better pose accuracies ($< \%1$), their computational cost is very high i.e, runtime of 0.1 secs/frame on 2 core-2.0GHz and 0.3 secs/frame on 2 core-2.0GHz [12]. Finally, the pose optimization stage uses the correspondences to estimate the relative poses.

Simultaneous Localization and Mapping (SLAM) [5] is also used to estimate the camera motion. But unlike VO, it jointly estimates poses and map of the scene. SLAM systems [16], [8] use loop closure and bundle adjustment (BA) to correct the pose drifts. However, BA is computationally expensive and is therefore, executed in a separate thread for map-optimization [16]. It achieves very low translation errors ($< \%1$) and takes around 0.065 secs/frame using 2 cores-3.5GHz. In terms of runtime and pose accuracy, ORB-SLAM2

[1] We define real-time performance to be around 30fps. Our evaluations are performed on current standard desktop CPU.

is one of the best-performing methods [20].

1.2 Main Contributions

The main contributions of our work are as follows:

1. ***Accurate and efficient feature correspondences setup***. We used an adaptive threshold and bucketing based feature extraction to obtain uniformly distributed features that overcome the challenges of low contrast and textureless regions. To obtain better quality correspondences, a novel similarity measure utilizing the feature's location information is proposed. This resulted in more accurate matches than existing approaches which rely on descriptors only.

2. ***Adaptive outlier removal***. An adaptive outlier removal strategy based on prior pose information is proposed to eradicate outliers. Unlike RANSAC etc. [3, 1], the proposed strategy does not rely primarily on feature set for hypothesis generation, thereby enabling it to successfully remove most outliers. This leads to improved pose accuracy. Our method is also computationally more efficient than other techniques such as [13], as only a single pass computation is needed.

3. ***Robust pose optimization***. To minimize the pose estimation errors caused by inaccurate feature depths, the features are assigned weights based on their position information during pose optimization. In addition, we increase the robustness of Gauss-Newton optimization by controlling the impact of each individual feature in terms of residual error.

2 Proposed Visual Odometry

The number of features used per frame has a high impact on the complexity of VO system, while the quality and distribution of feature correspondences plays a key role in the pose accuracy. Hence, a balance between quality, distribution and quantity of features is essential to achieve high pose accuracy with low runtime. Outliers can contribute to low accuracy and high computation complexity. In addition, since few false feature matches with smaller reprojection errors are difficult to detect for removal, there is a need for a robust pose optimization to further reduce pose errors. Hence, we propose strategies for efficient feature extraction, accurate matching, adaptive outlier removal and robust pose optimization (Fig. 1).

2.1 Accurate and Efficient Feature Correspondence Setup

Feature correspondence setup includes feature extraction, stereo matching and sequential matching. Our proposed method aims to obtain a minimal number and high quality feature correspondence that can lead to low complexity and high accuracy.

***Distribution*:** The proposed VO relies on ORB features for its invariance to illumination, rotation and scale change, and fast extraction and matching speeds [18]. But we observed that direct feature extraction results in features getting clumped in certain texture-rich regions, as shown in Fig. 2 (left). Distribution of features over the image plays an important role in pose accuracy [13]. Large errors are incurred when the pose is estimated using unevenly distributed features, and is severe in poor contrast scenes. Further, if the number of close features is insufficient, the estimated pose is more erroneous (Fig. 2 (left)). To overcome this problem, we used adaptive thresholding combined with bucketing technique to obtain a set of uniformly spread features. The image is divided into buckets of size 30 sq. pix. For each bucket, the features are detected based on an initial minimum threshold. If enough features are collected in the bucket, the process terminates, otherwise the threshold is incremented and process repeats until a maximum threshold. It ensures that well-distributed high quality features are obtained as shown in Fig. 2 (right). Although it incurs a slight increase in computation, the overall runtime of the VO is notably reduced due to the lesser number of features required for accurate pose estimation.

***Matching*:** Generally, the extracted features undergo matching using the descriptor distance based metric. In sequential matching, the features from the previous left image frame are searched for matches by projecting them to the current left image using smooth motion model [13]. We also observed that many features in the scene are repetitive in nature (similar features appearing frequently), for e.g. the features corresponding to lane markings, side rails, etc. The SIFT [15] compares the ratio of descriptor distances of two best matches. But when such features appear frequently in the successive frames, there is high tendency of false matching if the matching criteria is purely based on descriptor distance. This is due to the high similarity within that class of features, and hence descriptor based metric is unable to select the correct matches. We developed a new similarity metric for sequential matching which utilizes the motion constraints from estimated previous relative pose $\{\hat{\boldsymbol{R}}_{n-1}, \hat{\boldsymbol{t}}_{n-1}\}$ and constant velocity model [7]. The proposed similarity metric combines descriptor and pixel distance as shown in Eq. 1. The function $dist(\hat{\boldsymbol{d}}_n^i, \boldsymbol{d}_{n-1}^i)$ represents the distance between ORB descriptors $\hat{\boldsymbol{d}}_n^i$ and $\boldsymbol{d}_{n-1}^i$ of image features $\hat{\boldsymbol{m}}_n^i$ and $\boldsymbol{m}_{n-1}^i$ respectively. Feature $\hat{\boldsymbol{m}}_n^i$ in frame I_n is the contender match for feature $\boldsymbol{m}_{n-1}^i$ in frame I_{n-1}, and k is the chosen contribution of pixel distance. Currently k is set 1 to give equal importance to both descriptor distance and pixel distance.

$$s_n^i = k \cdot \underbrace{\|\hat{\boldsymbol{m}}_n^i - \pi(\hat{\boldsymbol{R}}_{n-1} \cdot \pi^{-1}(\boldsymbol{m}_{n-1}^i) + \hat{\boldsymbol{t}}_{n-1})\|}_{\textbf{reprojection distance}} + \underbrace{dist(\hat{\boldsymbol{d}}_n^i, \boldsymbol{d}_{n-1}^i)}_{\textbf{descriptor distance}} \quad (1)$$

The best match is chosen based on this minimum distance. The idea behind this formulation is: the repetitive feature which has best descriptor match with the

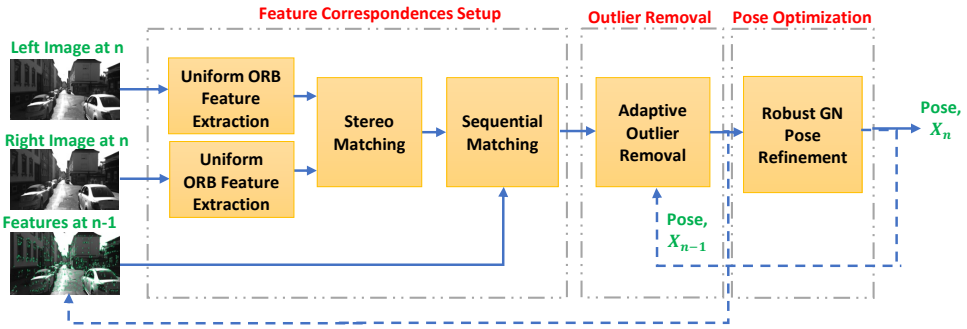


Figure 1: **VO framework** consists of feature correspondence setup, outlier removal and pose optimization. In feature correspondence setup, evenly distributed ORB features are extracted from incoming stereo images and are stereo-matched. Inliers from previous frame are then sequential-matched with currently extracted features, using proposed matching method. Outliers are removed using the proposed adaptive method. Finally, the pose is estimated using inliers, by employing controlled residual and depth-weighted Gauss-Newton optimization.

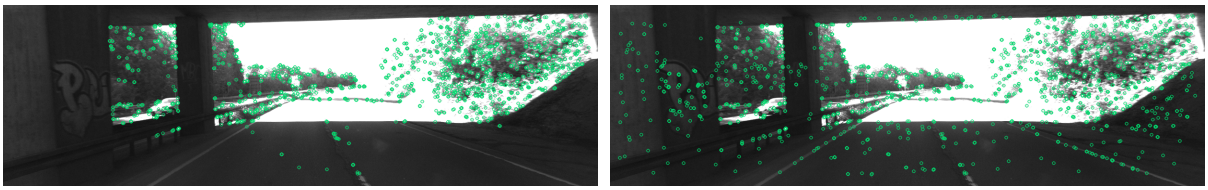


Figure 2: [left]- Features detected using constant threshold are unevenly distributed. [right]- More spread features using adaptive thresholding for features, while the number of features is the same in both of them

wrong feature will still be closer in pixel distance to the correct feature. This increases the robustness against false matches without incurring much additional computation cost. The reduction in false matches as a result of using the proposed metric leads to lesser computations in the later VO stages.

2.2 Non-Iterative Adaptive Outlier Removal

Usually, the outliers are considered to have significantly larger reprojection errors than the rest of the matches and therefore could have been removed by just based on a reprojection error threshold (RET). However, several outliers have reprojection errors close to the inliers and therefore selecting an optimal RET is not a straightforward task. To address this problem, existing stereo based methods usually employ 3-point RANSAC [13] for outlier removal. But, RANSAC based outlier removal is computationally expensive due to its iterative nature. In addition, the estimates are highly dependable on the choice of RANSAC parameters (iteration count and RET). RET depends on the scene. Moreover, it is not robust to outliers that are associated with a large moving object. If the maximum support criteria of RANSAC is used, then these points may not be detected as outliers. To resolve these issues, we propose a non-iterative adaptive outlier removal method, as shown in Fig. 3. Except initialization frame, all frames contain the previous pose information $\boldsymbol{X}_{n-1}$. Since VO assumes that the motion follows smooth camera trajectory [13], a rough estimate of the pose in the form of previous pose is always available. It can be observed from Fig. 3 that the initial prior pose is taken to be significantly different from the ideal pose (blue line), so we do not make any strong assumption about the prior pose and a rough estimate is sufficient for our method. Using the prior pose, reprojection errors of all the points are calculated and sorted (step 1). We assume that a minimum inlier ratio of r (30%) is always maintained. Then among 'N' available features, $(r \cdot N)^{th}$ minimum reprojection error is found and approximated using the ceil function to obtain RET,re_{th}.

$$re_{th} = \lceil (r \cdot N)^{th} \text{ minima of } \mathbf{re}_n \rceil \quad (2)$$

Finally, all points with reprojection error lesser than this RET are set as inliers. Thus, these inliers are always based on the previous pose and are not affected by random outliers and outliers associated with moving objects. The proposed method is computationally more efficient than existing iterative outlier removal methods as only a single pass computation is needed.

2.3 Robust Pose Estimation

To improve the robustness against false matches with smaller reprojection errors, a new saturated residual-error method is proposed. We also exploit the fact that far features are more prone to depth inaccuracies and propose depth-weighted scheme.

***Saturated Residual*:** False matches with relatively large reprojection errors are more harmful as they negatively impact the pose optimization. The adaptive outlier removal method described earlier removes outliers and false matches with large reprojection errors. But there is possibility that false matches

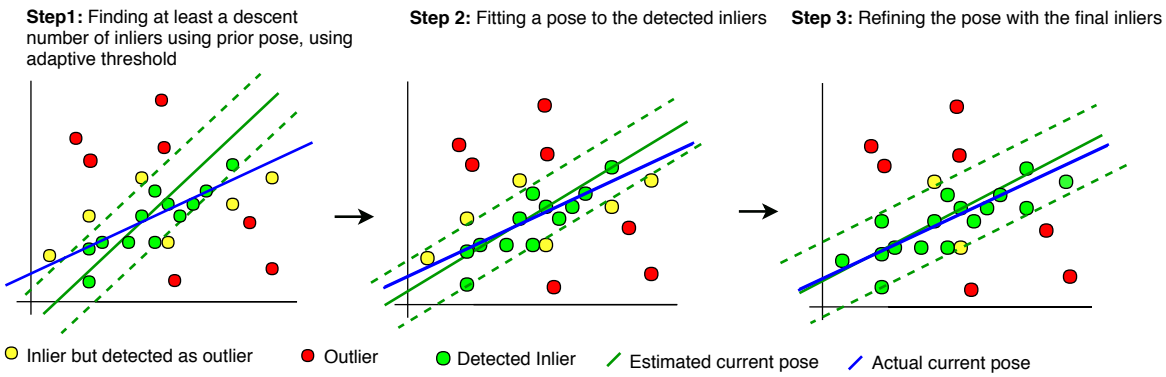
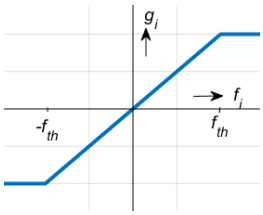


Figure 3: **Adaptive outlier removal concept.** Step 1 computes reprojection errors of all feature correspondences using the estimated prior poseX_{n-1}, and finds the inlier threshold adaptively using Eq.2. Step 2 uses this threshold to get the primary set of inliers, which are finally used by Step 3 to refine the pose. The red dots in the plots represent outliers and the rest of the points are inliers.



$$g_i = \begin{cases} f_{th} & if\ f_i > f_{th}, \\ -f_{th} & if\ f_i < -f_{th}, \\ f_i & otherwise \end{cases}$$

Figure 4: **Limiting the residual errors, f_i.** The residual errors above f_{th} and below $-f_{th}$, are limited to the respective thresholds

which are close to the inlier threshold are considered as inliers. To control the impact of any single feature point to the pose optimization, its contribution in the form of residual error f_i is limited as defined in Fig. 4. The residual errors f_i are replaced by g_i, before applying weighting. This increases the robustness of pose estimation to false matches. f_{th} is the upper limit of residual error that is allowed and is set upon studying the distribution of all residual errors experimentally. The weights are assigned to matches as discussed next.

***Depth Based Weighting*:** The depth of a feature is computed using stereo camera baseline [13]. The uncertainty in depth increases with the depth of the feature. ORB-SLAM2 [16] indicated that far points contribute weakly to scale and translation information. Hence, it neglects all far features (depth >18m) for pose optimization, unless the close-feature count is significantly low. However, directly removing far features disrupts the uniformity of feature distribution and incurs more drift as a result. Unlike SLAM, it is a major concern in VO where there is no way to correct the drifts. In the proposed pose optimization, we exploit depth information to provide an indication of the inaccuracies in features. However, instead of applying hard thresholding and removing far features [16], we apply weights to features based on its depth (d_n^i) such that the farther features are associated with smaller weights on logarithmic scale (Eq. 3). To compensate for the inadequate calibration [14], the features (u_n^i, v_n^i) which are near the image-center $((c_u, c_v)$ are given higher weights as Viso2 [13]. We define the combined weights independently along two image axes (w_i^x, w_i^y) as:

$$w_i^x = \frac{1}{\ln(d_n^i) \cdot (\left|\frac{u_n^i - c_u}{c_u}\right| + 0.05)}$$
$$w_i^y = \frac{1}{\ln(d_n^i) \cdot (\left|\frac{v_n^i - c_v}{c_v}\right| + 0.05)} \quad (3)$$

3 Experiments and Discussions

We used popular outdoor dataset KITTI [12], which is highly challenging due to varying motion and diverse scenes. It has low frame rate (10fps). Larger frame rates would lead to smoother inter-frame motion that will improve pose estimation accuracy in our method. The proposed VO is built on LibViso2. Only 1700 ORB features are extracted per frame, minimum-inlier-ratio r is 0.3 and f_{th} is set to 10 based on KITTI. The evaluation [2] gives average relative % translation and rotation errors. Runtime of only the ***open-source***[2] systems are evaluated on same platform averaged over 10 runs. The testing statistics are extracted from KITTI website [12] by submitting estimated poses. The existing VO and SLAM with high-accuracy or lowest-runtime have been selected for comparison. SOFT VO [6] has the highest accuracy in the KITTI site[12] , but its runtime (0.1 secs) does not meet real-time requirements. Viso2 [13], which is one of the fastest VO methods is also compared. Among all the SLAM systems, ORB-SLAM2 [16] is the most accurate and fastest and is therefore chosen. In addition, as SLAM has an extra advantage of using loop-closure and BA to correct the drifts, the VO version of ORB-SLAM2 i.e. ORB-VO is also included in evaluations on training dataset.

[2] Open Sources ORB-SLAM2, Viso2 and ORB-VO

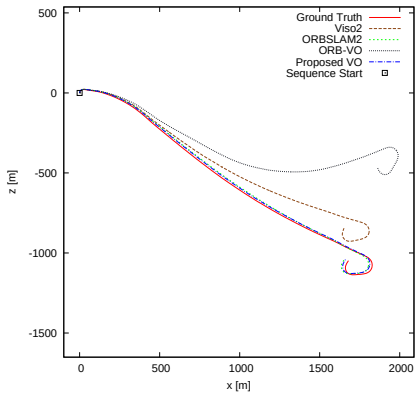


Figure 5: Plot compares the estimated trajectories with groundtruth. A challenging KITTI sequence 01 is shown.

Ablation Studies: To investigate the effect of each of the proposed strategy, detailed ablation study is conducted on training sequences. The experiments were performed on three configurations (B,C,D) and compared with the full proposed VO (A) (Table 1).

*1.**Efficient Feature Correspondence Setup** (EFCS):* In configuration B, the EFCS functionality is excluded and replaced with the basic feature extraction in Viso2, while other functionality of the proposed VO system are retained. Table 1 shows that in the absence of the proposed EFCS, the pose errors increased notably from 0.96% (A) to 2.19% (D). The proposed EFCS incurs only slightly longer runtime.

*2.**Proposed Outlier Removal** (POR):* Unlike the conventional RANSAC which requires 200 iterations to obtain maximum set of inliers, the proposed POR computes the inlier set using the motion-prior in a single pass. This leads to significant reduction in computation (Table 1), i.e. runtime reduced from 53 ms (C) to 44 ms (A). In addition, the proposed scheme uses adaptive thresholding for outlier removal that is independent of the scene, which resulted in improved translation error i.e. from 1.00% (C) to 0.96% (A).

*3.**Robust Pose Optimization** (RPO):* RPO includes calculation of weights and kernel function values for all the inliers. This increases the computations marginally but contribute to significant reduction in pose errors. As a result of employing RPO, translation errors reduces from 1.21% to 0.96% and rotation errors reduces from 0.0044 deg/m to 0.0030 deg/m.

The ablation studies show that each of the proposed EFCS, POR and RPO contribute to increasing pose estimation accuracy. Although EFCS and RPO requires slightly higher computation time, the overall runtime of the proposed VO system is significantly reduced due to the computationally efficient POR method.

Accuracy Evaluation: Similar to other reported works, we first perform accuracy evaluations on the ***training dataset***[2]. The comparison of our proposed VO with ORB-SLAM2, Viso2 and ORB-VO on training dataset is qualitatively shown in Fig. 5, using one representative sequence. The proposed VO traces the groundtruth remarkably well, because it utilizes better-distributed features, a mix of close and far features, along with efficient outlier removal. In few sequences containing loops, ORB-SLAM2 showed slightly better accuracy than proposed VO; because it takes advantage of BA and loop closure. ORB-VO performs worse, as it does not rely on BA and loop closure. Based on the results of ORB-VO and ORB-SLAM2, we expect that the proposed VO will achieve significant increase in pose accuracy if it is integrated with BA and loop closure. Viso2 deviates significantly; being unable to handle outliers effectively. As given in Table 2, on training dataset our proposed method achieves 0.96% translation errors which are close to ORB-SLAM2 (0.81%). The proposed VO significantly outperforms ORB-VO (3.41%) and Viso2 (2.64%). It is noteworthy that the rotation errors of proposed VO (0.0030deg/m) is lower than ORB-SLAM2 (0.0039deg/m). In addition, our proposed method (1.30%, 0.50% & 1.02%) shows improvement over ORB-SLAM2 (1.39%, 0.51% &1.05%) on trajectories 01, 06 and 08 respectively. The ***testing poses***[3] of the proposed VO were also submitted for online evaluation and results are shown in Table 2. Average translation error of 1.24% was reported for the proposed VO, which is close to the ORB-SLAM2 (1.15%). These results clearly demonstrate that even without using BA and global pose optimization, our method can perform comparable to the state-of-the-art systems in terms of pose accuracy.

Table 1: Ablation studies of the proposed VO on training dataset.

Method	**Trans. error** (%)	**Rot. error** (deg/m)	**Runtime** (secs) on 1-core @3.5GHz
A. *Proposed VO*	0.96	0.0030	0.044
B. *Without EFCS*	2.19	0.0150	0.036
C. *Without POR*	1.00	0.0037	0.053
D. *Without RPO*	1.21	0.0044	0.041

Runtime evaluation: The average runtime of ORB-SLAM2, ORB-VO, Viso2 and proposed VO is shown in Table 3. For comparisons, we obtained the runtime of ORB-SLAM2 using both single and two cores for pose estimation. It is worth mentioning that ORB-SLAM2 uses 2 more cores for back-end (BA and loop closure), where back-end runtime is not included in our comparisons. It is evident that the average runtime of the proposed VO system (0.044 secs) is 50% lower than the ORB-SLAM2 (0.088 secs) using single core for pose estimation. If we compare the timings on two cores, where feature extraction of left and right frames is performed in two parallel threads, our method achieves 0.034 secs (29 fps) compared to 0.065 secs for

[3] The testing results are extracted from KITTI website [2] and those results available online are discussed (ORB-SLAM2, Viso2, SOFT).

Table 2: Accuracy Evaluation on KITTI training and testing datasets.

Method	Training dataset		Testing dataset	
	Trans. error (%)	Rot. error (deg/m)	Trans. error (%)	Rot. error (deg/m)
Viso2	2.64	0.0201	2.44	0.0114
SOFT	NA**	NA**	0.88	0.0022
ORB-VO	3.41	0.0266	NA**	NA**
ORB-SLAM2	0.81	0.0039	1.15	0.0027
Proposed VO	0.96	0.0030	1.24	0.0027

**NA denotes that results in that configuration are not made available to us. For training Viso2, ORB-SLAM2 and ORB-VO, and for testing Viso2, ORB-SLAM2 and SOFT are available.

Table 3: Runtime evaluation on KITTI datasets.

Method	Runtime (secs)	Platform
Viso2	0.050	1 core @3.5GHz
SOFT	0.100	2 cores @2.5GHz
ORB-VO	0.065	2 core @3.5GHz
	0.087	1 core @3.5GHz
ORB-SLAM2	0.065	2 cores @3.5GHz
	0.088	1 core @3.5GHz
Proposed VO	0.034	2 cores @3.5GHz
	0.044	1 core @3.5GHz

ORB-SLAM2, i.e. our method achieves 47% lower runtime. This improvement is made possible due to the proposed computationally efficient methods in feature correspondence setup and adaptive outlier removal.

4 Conclusions

In this work, we proposed accurate and efficient feature correspondences setup for VO, which results in a lesser number of features (but of high quality) for pose estimation. The accuracy of the correspondences is enhanced by using the proposed similarity metric which takes into account both pixel distance and descriptor distance. In order to rapidly remove the outliers, an adaptive and deterministic scheme has been proposed that relies on the estimated pose of the previous frame. The proposed outlier removal scheme is non-iterative, and hence it can be accomplished in significantly lower computation time compared to its counterparts, e.g. RANSAC. We also propose strategies to increase the robustness of pose optimization by introducing a depth-weighted Gauss-Newton optimization and controlled residual approach. The timing and accuracy results show that the proposed VO achieves better trade-off than existing state-of-the-art systems.

References

[1] Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78.

[2] *Site*. www.cvlibs.net/datasets/kitti/eval_odometry.php.

[3] H. Badino, A. Yamamoto, et al. Visual odometry by multi-frame feature integration. In *2013 IEEE International Conference on Computer Vision Workshops*, pages 222–229, Dec 2013.

[4] M. Buczko and V. Willert. Flow-decoupled normalized reprojection error for visual odometry. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1161–1167, Nov 2016.

[5] C. Cadena, L. Carlone, et al. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *Trans. Rob.*, 32(6):1309–1332, Dec. 2016.

[6] I. Cvisic and I. Petrovic. Stereo odometry based on careful feature selection and tracking. In *2015 European Conference on Mobile Robots (ECMR)*, 2015.

[7] A. J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *null*, page 1403. IEEE, 2003.

[8] J. Engel, J. Stckler, et al. Large-scale direct slam with stereo cameras. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1935–1942, Sept 2015.

[9] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.

[10] F. Fraundorfer and D. Scaramuzza. Visual odometry : Part ii: Matching, robustness, optimization, and applications. *IEEE Robotics Automation Magazine*, 19(2):78–90, June 2012.

[11] R. G. Garcia-Garcia, M. A. Sotelo, et al. 3d visual odometry for gps navigation assistance. In *2007 IEEE Intelligent Vehicles Symposium*, 2007.

[12] A. Geiger, P. Lenz, et al. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[13] A. Geiger, J. Ziegler, et al. Stereoscan: Dense 3d reconstruction in real-time. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 963–968, June 2011.

[14] I. Kreo and S. egvi. Improving the egomotion estimation by correcting the calibration bias. In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications - Vol. 3: VISAPP, (VISIGRAPP 2015)*, 2015.

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004.

[16] R. Mur-Artal and J. D. Tards. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 2017.

[17] D. Nistér. Preemptive ransac for live structure and motion estimation. *Machine Vision and Applications*, 16(5):321–329, Dec 2005.

[18] E. Rublee, V. Rabaud, et al. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, Nov 2011.

[19] D. Scaramuzza and F. Fraundorfer. Visual odometry [tutorial]. *IEEE Robotics Automation Magazine*, 18(4):80–92, Dec 2011.

[20] R. Wang, M. Schworer, et al. Stereo dso: Large-scale direct sparse visual odometry with stereo cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3903–3911, 2017.

[21] J. Ziegler, P. Bender, et al. Making bertha drive -an autonomous journey on a historic route. *IEEE Intelligent Transportation Systems Magazine*, 2014.