2019 IEEE Intelligent Transportation Systems Conference (ITSC)
Auckland, NZ, October 27-30, 2019

Reliable Monocular Ego-Motion Estimation System in Rainy Urban Environments

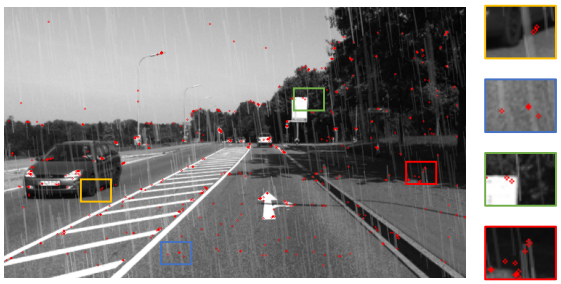
Huaiyang Huang, Yuxiang Sun and Ming Liu, *Senior Member*, *IEEE*

***Abstract*— Visual Simultaneous Localization and Mapping (SLAM) systems assume a static world. They usually fail under adverse weather conditions. In this paper, we propose a robust monocular SLAM system that is able to work under rainy conditions in urban environments reliably. To recover camera ego-motion from images with rain streaks, we apply a superpixel-based image content alignment method for the static background modelling. With coarse outputs estimated through averaging temporal matches, image details are recovered by a Convolutional Neural Network (CNN). Based on the statistic distribution of intensity variance between original and reconstructed image pairs, a robust and noise-sensitive weight function is explored for rejecting outliers when estimating camera poses. Quantitative evaluation results on the CARLA and synthetic KITTI datasets demonstrate the reliability of the proposed system and its superiority over the state-of-the-art approaches.**

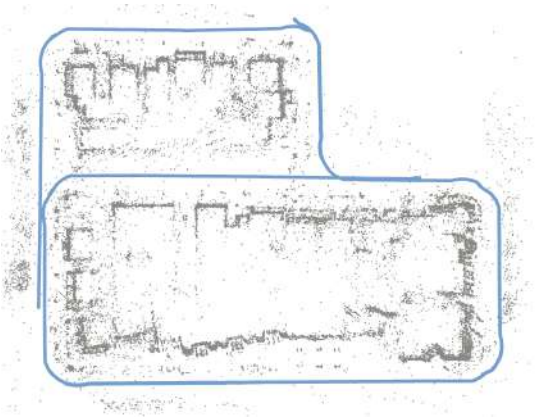
I. Introduction

Ego-motion estimation and 3D reconstruction are essential capabilities for autonomous driving. A robust visual state estimator provided by Simultaneous Localization and Mapping (SLAM) is exceptionally significant for autonomous navigation. It can serve as an alternative to Global Positioning System (GPS), especially under an urban environment with adverse natural conditions (e.g. heavy rainfall) [1]. Many approaches were proposed for monocular camera tracking and localization, while very limited works discuss how to robustly estimate ego-motion under adverse weather conditions, such as rainy environments, leaving it a still open problem.

As most of the state-of-the-art methods for monocular SLAM are proposed under the static world and moderate weather assumptions [2], rainy weather conditions raise incredible challenges, which lead these methods to failure. The difficulties for a visual SLAM system under such environments are various. Several typical cases are shown in Fig. 1. Firstly, rain streaks and moisture could generate artefacts over the background, so that reliable and repeatable features could not be extracted. Such artefacts would degrade the robustness of the general feature tracking module in a SLAM system. Secondly, as rain streaks could change the pixel intensity, some false features would be extracted, which would corrupt the consistency of the system and cause tracking failure. Thirdly, due to the visually noticeable distortion effect, feature mismatching could happen, so that the long-term robustness of the SLAM system would be degraded.



(a) Challenges for visual SLAM in rainy environment, including false-positive corners, mismatch and wrong descriptor.



(b) A sample result of the proposed system, blue line is trajectory estimation, gray points are sparse feature map.

Fig. 1: Typical challenges for visual SLAM under rainy conditions and sample result of our proposed system.

In this work, we propose a complete monocular SLAM system for reliable up-to-scale pose estimations in the rainy urban scenarios. The proposed system firstly aligns the image content at superpixel level. With temporal matching results, the static background is modelled by average weighted tensors in a sliding buffer. The coarse reconstructed images are then refined by a convolutional neural network. Finally, camera poses can be estimated reliably by weighting tracked features according to their intensity consistency. The contributions of this paper are listed as follows:

1) We propose a complete monocular ego-motion estimation system that could work reliably in rainy urban environments.

Huaiyang Huang is with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology. Yuxiang Sun and Ming Liu are with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong SAR, China (email: hhuangat@connect.ust.hk; eeyxsun@ust.hk, sun.yuxiang@outlook.com; eelium@ust.hk).

978-1-5386-7023-1/19/$31.00 ©2019 IEEE

2) We introduce a robust approach to integrate the image reconstruction with visual state estimation.
3) We carry out various experiments to demonstrate the effectiveness of our proposed method.

The remainder of this paper is structured as follows: In Section. II, related work about image derain and state estimation for challenging environments has been reviewed. In Section. III, an overview of our system is illustrated. In Section. IV and Section. V, we describe the derain method and the ego-motion estimation algorithm respectively. In Section. VI, experimental results and discussions are presented. Conclusions and future work are drawn in the last section.

II. Related Work

A. Image Sequence Rain Removal

Eliminating rain effect has long been a prevalent problem in the image processing community. For both single-shot image and continuous video sequence, numerous methods have been proposed to recover a clearer vision under this challenging environment.

For single-shot image rain removal or segmentation, Kang *et al.* [3]. firstly stated the single image derain problem and modelled rain streaks as classified atoms. Luo *et al.* [4] proposed to use discriminative sparse coding to better segment rain streaks from the original image. Li *et al.* [5] alternatively suggested learning the static background and dynamic rain streaks via Gaussian Mixture Model (GMM). Recently, deep learning based method [6], [7] became popular for image rain-removal problems. Zhang *et al.* [8] utilized a Generative Adversarial Network (GAN) to recover the background masked by rain streaks on raw images.

For continuous video-based rain removal, early methods [9], [10] usually modelled the background via image reconstruction to provide sufficient prior information. Chen *et al.* [11] managed to recover image details from a Convolutional Neural Network based on temporal image content alignment. Ren *et al.* [12] proposed to introduce low-rankness into rain detection.

B. State Estimation For Challenging Environments

Several approaches were initially designed [13], [14] to use a filter-based backend for pose optimization. Indirect-based methods such as [15], [16], were proposed to sparsify the system by keyframe selection and non-linear optimization. On the other hand, direct-based methods [17], [18], [19], seek solutions via optimizing photometric error directly on the image input. However, all these methods assume a static world, which is not suitable for specific tasks such as localization under the rainy urban scenario.

For state estimation under challenging urban scenario, Pascoe *et al.* [20], [21], [22] proposed to minimizing normalized information distance (NID) instead of photometric cost in direct visual odometry system, which achieved remarkable results under challenging illumination condition. Sun *et al.* [2], [23] provided a motion removal approach to handle foreground dynamic instances. Park *et al.* [24] evaluated the performance of direct visual odometry with changing illumination. However, they did not consider adverse natural weathers such as heavy rainfall and validated their algorithm under such environment. Another solution is to take advantage of a pre-built map and reduce the challenges by changing it into a localization problem. Porav *et al.* [25] introduced an adversarial network for camera localization under challenging conditions. Naseer *et al.* [26] proposed a framework for visual localization cross seasons. However, on the one hand, such localization methods based on a single-shot image are not fast and accurate enough to serve real-time vehicle control. On the other hand, considering a map-denied environment where no prior information is provided, visual odometry or SLAM could provide a more reliable relative pose estimation.

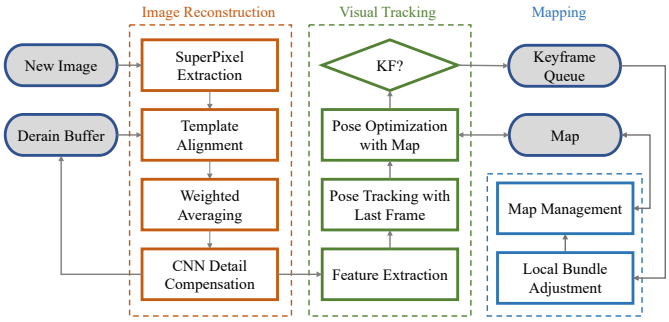


Fig. 2: System flowchart for the proposed system. The proposed system is consisted of two modules: rain removal and ego-motion estimation.

III. System Overview

The overall system can be divided into two main modules: the first is rain layer segmentation and background recovery for the incoming image; the second is tracking current camera pose and incrementally optimizing local structure. Both modules are based on recent state-of-the-art methods [11], [16]. We then adapt and integrate the two modules as a complete system for robust visual SLAM.

Fig. 2 illustrates the framework of the proposed system. With RGB images as input, we use a superpixel-based image content alignment method to occlude high-frequency noise (e.g. rain streaks). Then image details are recovered via a compensation network. Section. IV explains how the proposed system works in detail.

The camera ego-motion estimation module is consisted of two submodules, tracking and mapping. Firstly, we sample reliable features from extracted corners following the statistic distribution of intensity variance between the reconstructed image and the raw input. With a robust initialization strategy, camera poses are estimated coarsely by tracking the last frame, which is then optimized with the local map by minimizing reprojection error. Finally, a mapping module manages all the keyframes and landmarks, culling outliers and maintain the sparse feature map.

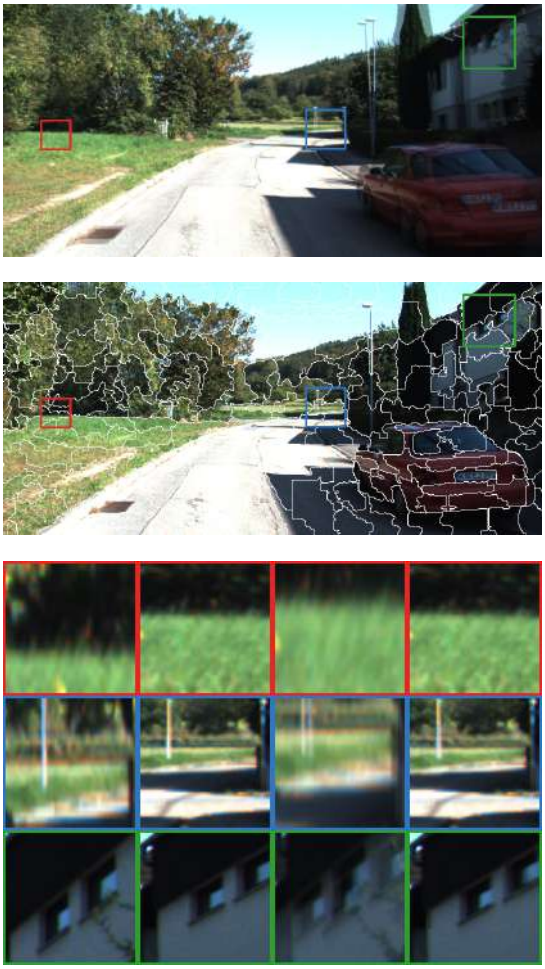


Fig. 3: Comparison of different alignment methods. Top: the alignment result from global homography transform; mid: superpixel segmentation for sample input; bottom: local detail comparison on three selected positions. While the first two columns are source and target patches, the last two columns are alignment results from homography-based method and superpixel-based method.

IV. Image Sequence Rain Removal

A. Temporal Content Alignment

For the rain removal module, our system first align the image content to take advantage of temporal visual information. There are two different approaches for image alignment: global-based approach and pixel-based approach.

Assuming partial planarity of the scene, the global approach estimates pixel-to-pixel transform between two frames via a set of matched features, which is the well-known homography transform [27]. Given a sequential image pair I_k, I_{k+1}, the global-based approach utilizes distinctive features [28], [29], [30] as sparse representation for a single image. Extracting feature points and computing their descriptors, temporal matching cross frames could be applied to this image pair, which generates two sets of matched correspondence set $\mathcal{X}_k$ and $\mathcal{X}_{k+1}$. With robust estimation method such as RANSAC [31], a global homography transform can be modelled from these two sparse feature sets, which transforms each pixel in the source frame to the target frame, denoted as:

$$\mathbf{x}_k^i = \mathbf{H}_{k,k+1}\mathbf{x}_{k+1}^i. \tag{1}$$

However, the planar assumption is not a good approximation and lead to misalignment across frames, as shown in Fig. 3. To overcome this issue, we adopt a superpixel-based solution [11] for temporal image content alignment. Superpixel is generally a set of image pixels that share photometric and geometric similarity. Here a sampling-based method [32] is used in the proposed system for superpixel segmentation.

Denoting the i-th superpixel extracted in frame k as $P_k^i \in \mathcal{P}$, a bounding box B_k^i is created for each superpixel P_k^i on extraction. B_k^i will then be extended as a wider window on neighbourhood frames in a temporal sliding buffer, denoted as W_k^i. To search the best matching patch in W_k^i, normalized cross-correlation (NCC) score is selected for accurate template matching, formulated as Eq. 2. As NCC is the optimal method statistically [33], it is very suitable to apply it under a rainy scene for robust superpixel matching.

$$E_{NCC}(P_k, \mathbf{d}) = \frac{\sum_{\mathbf{q}\in P_k} I(\mathbf{q}) I_R(\mathbf{q}-\mathbf{d})}{\sqrt{\sum_{\mathbf{q}\in P_\mathbf{k}} I(\mathbf{q})^2 \sum_{\mathbf{q}\in P_\mathbf{k}} I_R(\mathbf{q}-\mathbf{d})^2}}. \tag{2}$$

Therefore an optimal location of matching patch could be derived by maximizing NCC score between two patches as in Eq. 3, where $\hat{\mathbf{d}}$ is the optimal location estimated for patch matching.

$$\begin{aligned}\hat{\mathbf{d}} &= \arg\max_{\mathbf{d}\in W_k} E_{NCC}(P_k, \mathbf{d}) \\ &= \arg\max_{\mathbf{d}\in W_k} \frac{\sum_{\mathbf{q}\in P_k} I(\mathbf{q}) I_R(\mathbf{q}-\mathbf{d})}{\sqrt{\sum_{\mathbf{q}\in P_\mathbf{k}} I(\mathbf{q})^2 \sum_{\mathbf{q}\in P_\mathbf{k}} I_R(\mathbf{q}-\mathbf{d})^2}}.\end{aligned} \tag{3}$$

B. Rain detection and background reconstruction

Based on the matching result through a sliding window, we then generate a coarse reconstruction patch and rain streak segmentation mask by taking a weighted average of matched tensors. As formulated in Eq. 4, s_t is optimal matching NCC score for source template in the t-th frame in the sliding window.

$$I_{\text{mean}}^k(P_k^i) = \frac{1}{n}\sum_{t,t} \frac{I(P_t^i + \hat{\mathbf{d}}_t)\sum_j s_j}{s_t}. \tag{4}$$

The sample reconstructed images are shown in Fig. 3. Compared to a global-based alignment method, the proposed method better aligns images in details. However, certain blur and distortion on the boundary with high gradient can be noticed. Therefore, for the consideration of improving robustness and accuracy of state estimation, a CNN is utilized for better recover details of the current captured frame. The architecture of the compensation network is designed following [11] and illustrated in Fig. 4.

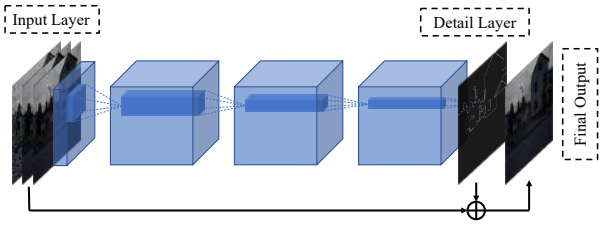


Fig. 4: CNN architecture for detail recovery. The output image is generated from the averaging layer and detail layer.

As demonstrated in Fig. 4, a three-layer feature is fed to the input of the neural network, which is connected to four convolutional layers with kernel sizes of 11, 5, 3, 1 respectively. A rectified linear unit is applied to each layer's output. With the compensation image I_{comp} generated by the network, final reconstructed image $\hat{I}$ and mask for rain streak are derived as Eq. 5:

$$\hat{I} = I_{\text{mean}} + I_{\text{comp}}, \quad \hat{M}_{rain} = \hat{I} - I_{\text{raw}}. \tag{5}$$

V. Pose Estimation

In this section, we derived a statistic distribution for the intensity variance between reconstructed image and raw input, explained in Section. V-A. In addition, Section. V-B and Section. V-C explain how we introduce this property into initialization and pose estimation.

A. Intensity Consistency Factor

To obtain a more robust pose estimation result, we propose to utilize features in reconstructed images in a probabilistic manner, instead of simply removing features according to masks generated from rain removal module.

A typical histogram of intensity variance between the reconstructed image and raw image with rain streaks is shown in Fig. 5. As general SLAM system assumes a Gaussian distribution of error, some robust kernel functions such as Huber function [34] are more commonly used. However, it is noticeable that normal distribution does not well describe the distribution of intensity variance, which represents the probability of a pixel belonging to rain layer. Two distribution forms that are more sensitive to outliers are compared. One is Gamma distribution [35] and another is t-distribution [36]. Here we select t-distribution as the probabilistic representation of pixel intensity variance, which better covers the data with large variance and low occurrence. Therefore it is more suitable to model the images reconstructed by rain removal module. For monocular initialization, pixels are sampled assuming distribution $\mathbf{x} \sim P(\mathbf{p})$. Additionally, a weight function could be derived as:

$$w(r) = \frac{\log p(r)}{\partial r}\frac{1}{r} = \frac{\upsilon + 1}{\upsilon + (\frac{r}{\sigma})^2}, \tag{6}$$

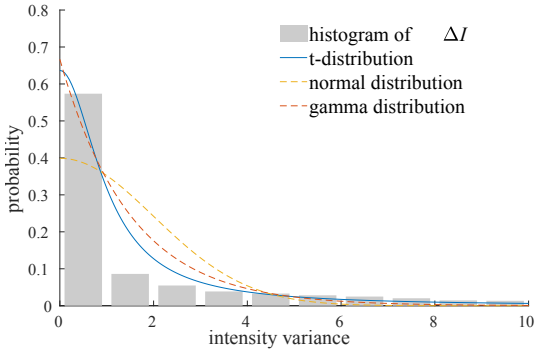


Fig. 5: Comparison of different distribution forms for image intensity variance.

Algorithm 1: Pose Initialization Algorithm

Data: $P_{\text{feat}}, P(\mathbf{p})$.
$P_{\text{homo}} = P_{\text{feat}}, P_{\text{fund}} = P_{\text{feat}}$
for $i \leftarrow 1$ ***to*** *max_iter* **do**
  $X_{\text{homo}}, X_{\text{fund}} =$ sample_features$(P_{\text{homo}}, P_{\text{fund}}, P(\mathbf{p}))$
  $s_{\mathbf{H}}, \mathbf{H} =$ compute_homography(X_{homo})
  $s_{\mathbf{F}}, \mathbf{F} =$ compute_fundamental(X_{fund})
  record_best_score$(s_{\mathbf{F}}, s_{\mathbf{F}_\text{best}}, s_{\mathbf{H}}, s_{\mathbf{H}_\text{best}})$
  $P_{\text{feat}} =$ remove_outliers$(P_{\text{feat}},)$
end
$r =$ compute_score$(s_{\mathbf{F}_\text{best}}, s_{\mathbf{H}_\text{best}})$
$M =$ select_best_model$(r, \mathbf{H}_{\text{best}}, \mathbf{F}_{\text{best}})$
if *check_disparity_and_epipolar*(P_{feat}, M) **then**
  $T_{\text{init}} =$ recover_pose(M)
  $P_{\text{map}} =$ triangulation$(P_{\text{feat}}, T_{\text{init}})$
  global_optimization$(T_{\text{init}}, P_{\text{map}})$
end
else
  return $False$
end
return $T_{\text{init}}, P_{\text{map}}$

where $r = \Delta I(\mathbf{p}) = \hat{I}(\mathbf{p}) - I_{\text{raw}}(\mathbf{p})$ represents for intensity variance between the reconstructed image and raw image. υ is the degree of t-distribution and σ is the variance pre-estimated from the training set.

B. General Pose Estimation

When a frame is processed by the rain layer segmentation module, it will be received by the pose estimation module. As we mentioned before, image noise introduced by rain streaks and aggressive averaging gradually corrupted geometry model for initialization, therefore we extend raw initialization module proposed in [16] to be more robust and fast to recover initial scene structure. Our initialization algorithm is illustrated in the pseudo-code Algorithm. 1.

For a more robust initialization, pixels are sampled according to the distribution mentioned in Section. V-A. Given an

Fig. 6: Sample results of derain module. The first and third rows are images captured in rainy weather; the second and forth rows are reconstructed results. Images in the first two columns are from CARLA dataset, while the right two columns are from synthetic KITTI dataset.

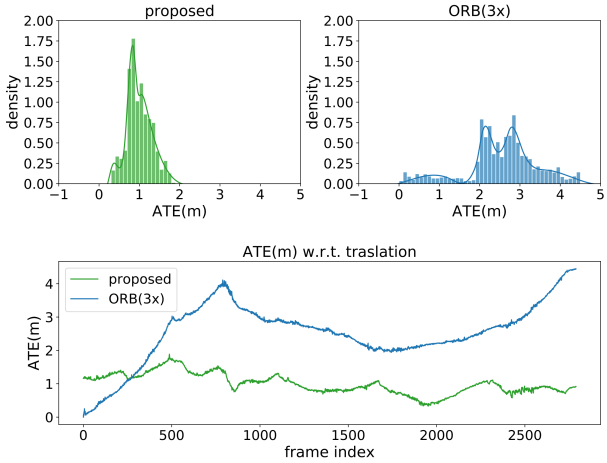


Fig. 7: ATE comparison on Carla01.

image captured in rain, pixels such as endpoint of rain streaks are very likely to be detected as corners. While rain removal module provides a reflective rain mask, contributions from these features will be reduced as they are covered by the low probability area and less likely to be sampled. Then we concurrently estimate the homography model and fundamental model in a RANSAC scheme. For each iteration step, outliers are filtered out based on Chi-Squared Test assuming one-pixel variance. During the loop, we keep a record of the best model respectively, and finally, an initialization model M is selected based on the score of homography and fundamental matrix estimation. We follow [16] to use a combined score to determine which model is better, formulated as:

$$r = \frac{s_{\mathbf{H}}}{s_{\mathbf{H}} + s_{\mathbf{F}}}, \tag{7}$$

where $s_{\mathbf{H}}$ and $s_{\mathbf{F}}$ are the scores for homography and fundamental matrix, respectively. If no model could be successfully estimated, the system will be reset and wait for the next frame. Otherwise, the initial motion and structure will be recovered from the inliers that have enough stereo disparity and satisfy epipolar constraint.

C. Camera Tracking and Mapping

For estimating current camera pose, matches are searched within a window in the neighbourhood frame. Pose estimations by minimizing reprojection error are performed with the last frame and with a queue of past keyframes. The error function could be derived as following:

$$E = \sum_i \frac{1}{2} w_i \|\mathbf{p}'_i - \pi_k(T_{k,j}\pi_j^{-1}(\mathbf{p}_i))\|_2^2, \tag{8}$$

where $T_{k,j}$ is the relative transform between k-th and j-th frames, respectively. π_k is the projection function of k-th frame. Unlike common SLAM systems that use robust kernel such as Huber to reduce the significance of outliers on the optimization result, we derived our weight function from a t-distribution as explained in Section. V-A. Therefore, minimizing reprojection error, optimal pose estimation could be solved by iterative least square:

$$T_{k,j} = \arg\min_{T_{k,j}} \frac{1}{2} \sum_i \frac{\upsilon + 1}{\upsilon + (\Delta I(\mathbf{p})/\sigma)^2} \cdot \|\mathbf{p}'_i - \pi_k(T_{k,j}\pi_j^{-1}(\mathbf{p}_i))\|_2^2. \tag{9}$$

After pose optimization, the frame will be delivered to the mapping module if it satisfies keyframe criteria. The mapping module will perform a global bundle adjustment to optimize structure and motion simultaneously. With the information of final optimization error and covisiblity relationship, redundant keyframes and landmarks or outliers are removed from the map.

VI. Experimental Results and Discussions

To evaluate the performance of the proposed system, we carry out experiments on synthetic image sequences and simulation respectively. We use CARLA [37] as the simulation platform, which is able to generate realistic image data under various weather conditions with groundtruth camera poses. In addition, we validate our method on KITTI odometry dataset with synthesized rain effect. For qualitative evaluation, we provide raw images with weather conditions varies in degree of rain and time of the day, along with reconstruction results generated by rain removal module, shown in Fig. 6.

A. CARLA simulation

For comparison, we use ORB SLAM on the image sequence captured in the rainy environments. While it fails on all the raw sequences with the same setting as the proposed system, we change the setting of ORB SLAM and extract 3 times more features on each frame compared to the raw setting, which is denoted as ORB SLAM(3x). This is to ensure a proper initialization and more robust feature tracking for ORB SLAM under rainy weather, which

TABLE I: Comparison of absolute trajectory error (ATE) (m) on CARLA Dataset. × means tracking failure. ORB SLAM(3×) is results from ORB SLAM with 3 times more features, with details in Section. VI-A. Statistic shows our system outperforms ORB SLAM on all the sequences on accuracy and stability.

Seq	Description	Proposed				ORB SLAM				ORB SLAM(3×)			
		RMSE	*Mean*	*Median*	*S.D.*	*RMSE*	*Mean*	*Median*	*S.D.*	*RMSE*	*Mean*	*Median*	*S.D.*
00	heavy rain noon	0.3446	0.2966	0.3233	0.1754		×			2.3256	0.6511	0.9905	2.1041
01	heavy rain noon	1.3681	1.1239	1.0332	0.7801		×			2.6875	2.5325	2.6253	0.8993
02	heavy rain noon	10.095	8.2209	5.8073	5.8590		×					×	
03	heavy rain noon	1.9850	1.8288	1.6692	0.7719		×			13.977	10.982	7.9535	8.6470
04	mid rain noon	2.0217	1.7848	1.6961	0.9497		×			3.2847	2.9619	2.7062	1.4200
05	mid rain sunset	0.3899	0.3282	0.2941	0.2106		×			3.9880	3.3797	3.8821	2.1170
06	mid rain sunset	1.9790	1.8259	1.6965	0.7632		×					×	
07	heavy rain sunset	3.0568	2.5309	2.0747	1.7142		×			8.0926	6.9565	8.7163	4.1349

TABLE II: Comparison of Translation relative pose error (RPE) (m/s) on CARLA Dataset. The comparison demonstrates that our system have less drift than ORB SLAM.

Seq	Description	Proposed				ORB SLAM				ORB SLAM(3×)			
		RMSE	*Mean*	*Median*	*S.D.*	*RMSE*	*Mean*	*Median*	*S.D.*	*RMSE*	*Mean*	*Median*	*S.D.*
00	heavy rain noon	0.0618	0.0285	0.0212	0.0548		×			0.0935	0.0463	0.0246	0.0812
01	heavy rain noon	0.0342	0.0271	0.0221	0.0209		×			0.0470	0.0383	0.0315	0.0273
02	heavy rain noon	0.0418	0.0331	0.0274	0.0255		×					×	
03	heavy rain noon	0.0421	0.0277	0.0194	0.0317		×			0.4606	0.1299	0.0459	0.4419
04	mid rain noon	0.0487	0.0320	0.0239	0.0367		×			0.1086	0.0585	0.0358	0.0915
05	mid rain sunset	0.0526	0.0307	0.0235	0.0427		×			0.0588	0.0459	0.0362	0.0367
06	mid rain sunset	0.0487	0.0283	0.0229	0.0397		×					×	
07	heavy rain sunset	0.0322	0.0268	0.0230	0.0178		×			0.0493	0.0395	0.0324	0.0295

TABLE III: Comparison of Rotation RPE (deg/s) on CARLA Dataset. The comparison demonstrate that our system is more consistent in rotation.

Seq	Description	Proposed				ORB SLAM				ORB SLAM(3×)			
		RMSE	*Mean*	*Median*	*S.D.*	*RMSE*	*Mean*	*Median*	*S.D.*	*RMSE*	*Mean*	*Median*	*S.D.*
00	heavy rain noon	0.3192	0.1813	0.1033	0.2627		×			2.3256	0.6511	0.9905	2.1041
01	heavy rain noon	0.1268	0.1061	0.0917	0.0695		×			0.1926	0.1599	0.1351	0.1074
02	heavy rain noon	0.1175	0.1009	0.0886	0.0602		×					×	
03	heavy rain noon	0.0916	0.0758	0.0686	0.0514		×			1.6866	0.3770	0.1239	1.6440
04	mid rain noon	0.1660	0.1189	0.0985	0.1159		×			0.3290	0.1954	0.1425	0.2647
05	mid rain sunset	0.1430	0.1066	0.0877	0.0953		×			0.2225	0.1696	0.1284	0.1441
06	mid rain sunset	0.1932	0.1175	0.0967	0.1534		×					×	
07	heavy rain sunset	0.1256	0.1050	0.0902	0.0689		×			0.1895	0.1547	0.1275	0.1095

in turn succeeds in some sequences. The metrics absolute trajectory error (ATE) and relative pose error (RPE) are used for quantitative evaluation. ATE measures the accuracy of trajectory estimation of the whole system, while RPE better demonstrates the drift of a SLAM system.

Pose estimation accuracy and robustness are evaluated and shown in Tab. III. ORB SLAM without changing setting fails for all the sequences, while the proposed system could provide the most accurate and robust result. Additionally, with more features extracted to achieve robustness, ORB SLAM(3×) still fails on two sequences. This is mainly because the noise introduced by heavy rain to the image is not neglectable and therefore affects the accuracy and robustness, especially the initialization procedure, of general feature-based SLAM method, which assumes a static world. On the contrary, the proposed system not only uses a derain module in the front-end, but also better culls outliers with

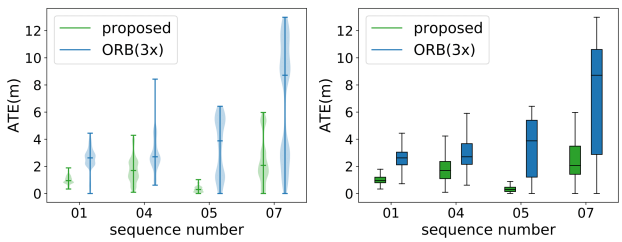


Fig. 8: Accuracy variance comparison on CARLA01, CARLA-04, CARLA05 and CARLA07.

a noise-sensitive weight function. Besides the accuracy and robustness, trajectories estimated by the proposed system encountered the least drift according to translation and rotation RPE, which shows the consistency of our methods under

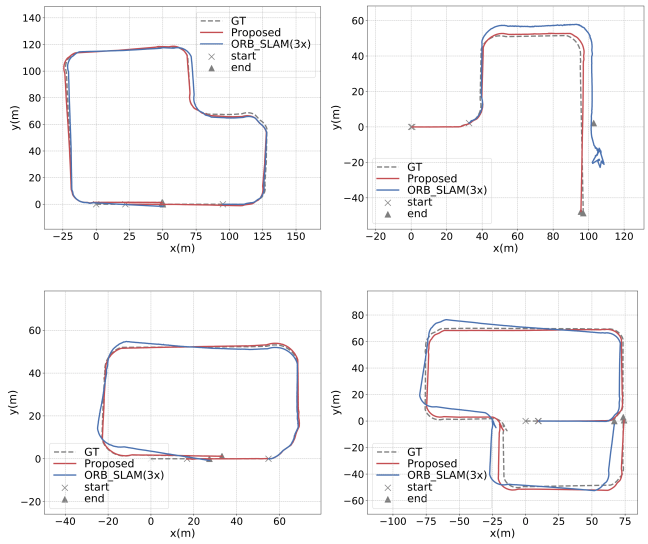


Fig. 9: Comparison of trajectories on CARLA01, CARLA03, CARLA05, CARLA07 respectively.

rainy weather.

Fig. 9 compares the estimated trajectories against ORB SLAM(3×). Besides the fact that the proposed system generates a more accurate trajectory, some common failure situations for general SLAM system can be concluded. As in all the displayed trajectories, it is noticeable that even the number of features is increased, ORB SLAM(3×) is not able to correctly initialize the system within a small amount of frames. On the contrary, a robust sampling strategy based on photometric distribution helps our system initialized fast within several frames. In sequence CARLA03, a false positive loop-closure detection causes end part of the trajectory deviating from groundtruth, which is also a side effect of visual SLAM brought by heavy rain fall. In contrast, our system reconstructs the background of the images, thus it is less likely to encounter wrong loop detection.

To demonstrate the long-term consistency and accuracy of the proposed system under adverse rainy condition. Fig. 7 shows the ATE of each frame and histogram of ATEs for both proposed and ORB SLAM(3×) on the CARLA01 sequence. For most of the frames, our system achieves better estimation results. In addition, less scale drift is encountered by the proposed system compared with ORB SLAM(3×). Therefore a better trajectory estimation is maintained. Variation from errors implicate the proposed system is capable of providing a more consistent estimation, even under adverse rainy weather and with a normal number of features. The quantitative evaluation results are illustrated in Fig. 8, where the proposed system keeps low variances and meanwhile accurate estimations compared to ORB SLAM(3×). This is mainly because visual slam frameworks such as ORB SLAM lack effective methods to distinguish outliers.

B. Synthetic KITTI Dataset

The synthetic dataset is based on KITTI odometry [38] RGB image sequence, where raw images are not captured under adverse weather conditions. We synthesized rain effect with commercial editing software *Adobe After Effect* [39], which was then added to the raw images. With several effect parameters such as raindrop size, wind strength, opacity and rain direction adjusted, different realistic rainy conditions were created for thorough evaluation. Sample synthetic images, with rain detection and image reconstruction results, are shown in Fig. 6.

TABLE IV: Comparison of RMSE (m) of translational ATE on synthetic KITTI. × means tracking or initialization failure. Here we use the results of ORB SLAM on raw images for reference (the last column) to demonstrate the stability of our system.

Seq	***Environment***	***Proposed***	***ORB SLAM***	***ORB SLAM(ref)***
00	heavy	7.14	×	6.68
01	heavy	×	×	×
02	mid+oblique	24.33	×	21.75
03	mid	3.68	×	1.59
04	heavy	2.58	×	1.79
05	heavy+wind	9.79	×	8.23
06	mid	14.71	×	14.68
07	mid+oblique	5.14	×	3.36
08	mid+wind	46.30	×	46.58
09	mid+oblique	8.52	×	7.62
10	light	9.37	×	8.68

To further demonstrate the accuracy of the proposed system, we compare the trajectory estimated from ORB SLAM on raw image sequences, with results from ours on images with rain effect. As shown in Tab. IV, our system could provide competitive pose estimations even against results estimated from , which indicates the stable performance of our system.

VII. Conclusions

In this paper, we demonstrated the challenges for visual SLAM under rainy urban scenario and propose a novel system for robust and accurate pose estimation to tackle these problems. We adapted both rain removal and visual state estimation modules and integrate both as a complete system. With quantitative evaluation, our system is examined to provide robust and accurate trajectory estimations under the scenario where state-of-the-art methods would usually fail.

Although our system shows an outstanding performance on the datasets, some limitations leave us some space for improvement. For instance, feature tracking module could provide useful anchors for template matching, which might improve current alignment result. Additionally, introducing inexpensive inertial measurement unit cloud produce a more smooth prior and robust pose tracking.

REFERENCES

[1] Y. Sun, W. Zuo, and M. Liu, "Rtfnet: Rgb-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2576–2583, 2019.

[2] Y. Sun, M. Liu, and M. Q.-H. Meng, "Improving RGB-D SLAM in dynamic environments: A motion removal approach," *Robotics and Autonomous Systems*, vol. 89, pp. 110–122, 2017.

[3] L.-W. Kang, C.-W. Lin, and Y.-H. Fu, "Automatic single-image-based rain streaks removal via image decomposition," *IEEE Transactions on Image Processing*, vol. 21, no. 4, p. 1742, 2012.

[4] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3397–3405.

[5] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2736–2744.

[6] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley, "Removing rain from single images via a deep detail network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1715–1723.

[7] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, "Deep joint rain detection and removal from a single image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1357–1366.

[8] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," *arXiv preprint arXiv:1701.05957*, 2017.

[9] J. Bossu, N. Hautière, and J.-P. Tarel, "Rain or snow detection in image sequences through use of a histogram of orientation of streaks," *International journal of computer vision*, vol. 93, no. 3, pp. 348–367, 2011.

[10] P. C. Barnum, S. Narasimhan, and T. Kanade, "Analysis of rain and snow in frequency space," *International journal of computer vision*, vol. 86, no. 2-3, p. 256, 2010.

[11] J. Chen, C.-H. Tan, J. Hou, L.-P. Chau, and H. Li, "Robust video content alignment and compensation for rain removal in a cnn framework," *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[12] W. Ren, J. Tian, Z. Han, A. Chan, and Y. Tang, "Video desnowing and deraining based on matrix decomposition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4210–4219.

[13] 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.

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

[15] 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*. IEEE, 2007, pp. 225–234.

[16] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[17] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 15–22.

[18] 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.

[19] C. Kerl, J. Sturm, and D. Cremers, "Dense visual slam for rgb-d cameras," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. Citeseer, 2013, pp. 2100–2106.

[20] G. Pascoe, W. Maddern, M. Tanner, P. Piniés, and P. Newman, "Nid-slam: Robust monocular slam using normalised information distance," in *Conference on Computer Vision and Pattern Recognition*, 2017.

[21] G. Pascoe, W. P. Maddern, and P. Newman, "Robust direct visual localisation using normalised information distance." in *BMVC*, 2015, pp. 70–1.

[22] G. Pascoe, W. Maddern, and P. Newman, "Direct visual localisation and calibration for road vehicles in changing city environments," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 9–16.

[23] Y. Sun, M. Liu, and M. Q.-H. Meng, "Motion removal for reliable RGB-D SLAM in dynamic environments," *Robotics and Autonomous Systems*, vol. 108, pp. 115 – 128, 2018.

[24] S. Park, T. Schöps, and M. Pollefeys, "Illumination change robustness in direct visual slam," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4523–4530.

[25] H. Porav, W. Maddern, and P. Newman, "Adversarial training for adverse conditions: Robust metric localisation using appearance transfer," *arXiv preprint arXiv:1803.03341*, 2018.

[26] T. Naseer, L. Spinello, W. Burgard, and C. Stachniss, "Robust visual robot localization across seasons using network flows." in *AAAI*, 2014, pp. 2564–2570.

[27] H. Longuet-Higgins, "The reconstruction of a plane surface from two perspective projections." vol. 227, no. 1249, pp. 399–410, 1986.

[28] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[29] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*. Springer, 2006, pp. 404–417.

[30] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Computer Vision (ICCV), 2011 IEEE international conference on*. IEEE, 2011, pp. 2564–2571.

[31] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[32] M. Van den Bergh, X. Boix, G. Roig, B. de Capitani, and L. Van Gool, "Seeds: Superpixels extracted via energy-driven sampling," in *European conference on computer vision*. Springer, 2012, pp. 13–26.

[33] H. Hirschmuller and D. Scharstein, "Evaluation of stereo matching costs on images with radiometric differences," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 9, pp. 1582–1599, 2009.

[34] P. J. Huber *et al.*, "Robust estimation of a location parameter," *The annals of mathematical statistics*, vol. 35, no. 1, pp. 73–101, 1964.

[35] T. K. Marks, A. Howard, M. Bajracharya, G. W. Cottrell, and L. Matthies, "Gamma-slam: Using stereo vision and variance grid maps for slam in unstructured environments," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 3717–3724.

[36] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for rgb-d cameras," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3748–3754.

[37] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[38] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[39] "Adobe after effects software," http://www.adobe.com/AfterEffects.