2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Vision Global Localization with Semantic Segmentation and Interest Feature Points

Kai Li[1,§], Xudong Zhang[2,§], Kun LI[1,§,⋆] and Shuo Zhang[1]

***Abstract*— In this work, we present a vision-only global localization architecture for autonomous vehicle applications, and achieves centimeter-level accuracy and high robustness in various scenarios. We first apply pixel-wise segmentation to the front-view mono camera and extract the semantic features, e.g. pole-like objects, lane markings, and curbs, which are robust to illumination, viewing angles and seasonal changes. For the scenes without enough semantic information, we extract interest feature points on static backgrounds, such as ground surface and buildings, assisted by our semantic segmentation. We create the visual global map with semantic feature map layers extracted from LiDAR point-cloud semantic map and the point feature map layer built with a fixed-pose SFM. A lumped Levenberg-Marquardt optimization solver is then applied to minimize the cost from two types of observations. We further evaluate the accuracy and robustness of our method with road tests on Alibaba's autonomous delivery vehicles in multiple scenarios as well as a KAIST urban dataset.**

I. Introduction

Significant improvements in self-driving vehicles have been made in the last decades, which have been extended to various areas including robo-taxi, autonomous bus and delivery vehicles [12]. Map-based global localization is a fundamental functionality for autonomous driving vehicles with high-definition (HD) map based solution, i.e., decision and planning modules rely on the road elements from pre-built HD map. A robust and precise localization module serves as a prerequisite for a safe and healthy autonomous software architecture.

The solutions of visual global localization are mainly separated into two ramifications, i.e., tracking and matching with non-linear optimizer upon the usage of global map, and end-to-end learning network for place recognition. As a traditional approach, structure-based visual positioning can also be divided into two categories. The first category is to track the extracted features from current image frame with a pre-built feature map [10], [20], [26], which heavily depends on the repeatability and consistency of feature extraction under different viewing angle and illumination conditions. Recently, the corner features extracted by deep neural network (DNN) models [5], [23] have outperformed the traditional handcrafted methods such as speeding-up features (SURF) [21] or oriented FAST and rotated BRIEF (ORB) [22]. Tracking with the unsupervised learned features exhibits higher robustness to illumination change, and yet still fails to find correspondences in some extreme conditions or confused by repeated patterns. Instead of using textural information, some research works use semantic features to improve robustness. The research work [8] use lane markings as observations to match with global map, while [19], [32] keep tracking with curb and other road structural features for localization. More than that, with the development of pixel-level semantic segmentation, more semantic information can be stably extracted as landmark features, e.g. the traffic signs [16], and pole-like objects [17]. However, the semantic feature based approaches might not work properly when lacking of enough effective semantic information in the scene. In order to achieve more robustness, some research works manage to fuse the semantic segmentation with corner features. The research work [30] adopts semantic segmentation mask to filter the outliers of optical flow during the visual-odometry prediction. [4] combines the depth variance and semantic information to produce accurate key-points in Mono-SLAM system.

In recent years, learning-based localization methods have been proposed to solve pose estimation [7], [26]. They learn the features with stable appearance over time [18], and train CNNs to regress 2D-3D matches [2] or camera poses [29]. The work PoseNet [15] first proposed the end-to-end camera 6 degree-of-freedom (DOF) pose estimation, and is proven to be feasible in various occasions. MapNet [3] enables learning a data-driven map representation, exploits cheap and ubiquitous sensory inputs like visual odometry and GPS in addition to images and fuses them together for visual localization. The work [3] shows that the data-driven approaches perform poorly in pose estimation with respect to accuracy, compared to the traditional feature matching approaches.

To this end, we proposed a localization framework that combines both semantic feature observations for higher robustness as well as interest-point features for longer coverage. With the assistance of LiDAR point-cloud map, we are able to extract the precise ground truth semantic features in 3-D space. Fig. 1 shows the data process from the input image to optimized poses. In Sec. VI, we demonstrate that our visual algorithm attains state-of-the-art localization accuracy, high robustness and capability for generalization with multiple experiments.

The contributions to this work are twofold. Firstly, we combine the pixel-wise semantic segmentation as well as the key-point features in a coupled flavour that we use semantic

*This work was supported by the National Key R&D Program of China under Grant 2018YFB1600804 and ZheJiang Program in Innovation, Entrepreneurship and Leadership Team (2018R01017).

[1]Alibaba Group, UK Center, EFC, No.1122, Xiang Wang Street, Yuhang District, Hangzhou, China.

[2]Oppo Research Institute, Shanghai, China.

[§]Authors contributed equally to this work.

⋆Corresponding author. Contact: lk158400@alibaba-inc.com

masks to reject the features points located on dynamic obstacles, movable or changeable objects like vehicles or trees. Secondly, we integrate the LiDAR point-cloud map for its high precision as the input global reference. On the other hand, it provides a reference or ground truth, in some sense, for the feature map built by structure from motion (SFM).

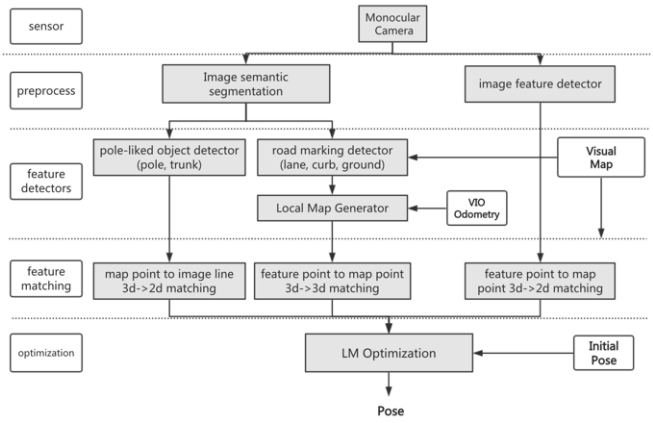


Fig. 1: Framework architecture for our visual localization module with input source as monocular image, visual-inertial odometry and global map. The main pipeline is serialized as four sub-blocks, i.e., image preprocessor, feature detector, feature matching and pose optimizer. The image preprocessor generates semantic segmentation and interest feature points with two DNN models. The feature detector extracts semantic features and interest-point descriptors, for matching with the corresponding layer in global map. The pose is iteratively solved by minimizing the lumped observation cost with a Levenberg-Marquardt (LM) optimizer.

II. Image Preprocessing

In order to obtain refined features to be residualized, raw images go through the preprocessor pipeline, where pixel-level information is extracted and quantized into vectors by two DNN models, 1) semantic segmentation to extract semantic regions and 2) key-point extraction with image point and descriptor for point features. Both architectures are restricted to be small and efficient for real-time processing on resource confined machines yet accurate and robust with high mean intersection over union (mIoU) index. Thus we design our network based on one of the most efficient semantic segmentation networks, BiSeNet [31] for the first task and adopt the unsupervised interest-point learner [9] for the second task. Fig. 2 shows the image preprocessing pipeline.

A. Semantic Segmentation

In the semantic segmentation task, we select the high efficient architecture, BiSeNet [31] as the backbone, which is remarkable for high-speed and real-time process. However, the original BiSeNet encounters low mIoU results for pole-like objects such as lamp posts or tree trunks. Such cases may be caused by the down-sample characteristic of convolutional

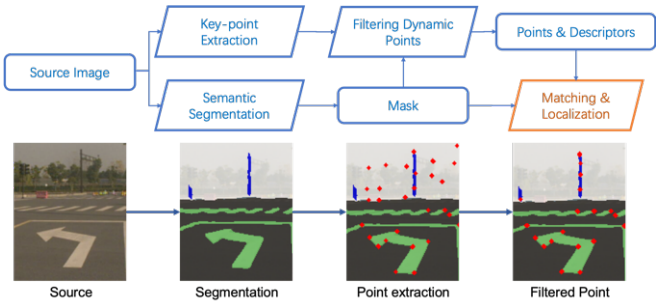


Fig. 2: Image preprocessing pipeline: (a) The raw input image with the size of 512×640; (b) Semantic segmentation network output with six categories, i.e., static elements, including "poles", "trunks", "curbs", "landmarks", "roads" and "buildings"; (c) Key-point extraction; (d) Points filtered with effective semantic masks.

layers in the last several layers of the network, as the pole-like objects occupy only several pixels in width and are hardly recognized by the network. In order to improve the performance, an upsampling structure is introduced to solve the pole-like object issues, shown in Fig. 3. Deconvolutional layers [11] outperforming other unpooling layers, is demonstrated to be effective in reconstructing fine-details of structures. Therefore, we apply three deconvolutional layers cascade with point-wise convolutional layer [13], which contains 1×1 kernel to achieve an enlarged feature map instead of classical interpolation. Performance of mIoU (Table. I) and visual results reveal that this scheme achieves higher semantic segmentation quality in edge details, especially beneficial for pole-like object categories.

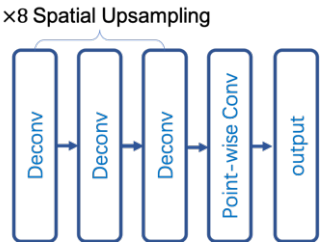


Fig. 3: Deconvolutional structure for up-sampling task to achieve refined details of edge features.

TABLE I: The mIoU performance comparison

No.	Categories	Our Method	BiSeNet [31]	Deeplab-V3
1	Poles	**68.6**	56.7	63.7
2	Trunks	**64.8**	55.2	62.1
3	Curbs	75.2	68.9	**77.8**
4	Landmarks	**82.7**	75.4	81.1
5	Roads	97.6	95.2	**98.0**
6	Buildings	91.7	90.7	**93.5**
7	Total	**80.1**	73.7	79.4

1) Key-Point Extraction: For scenes that lack effective semantic information, e.g. the open square, trails in the residential areas or inside tunnels, we have to consider the textural information for global localization even though it is affected by illumination and affine transformation.

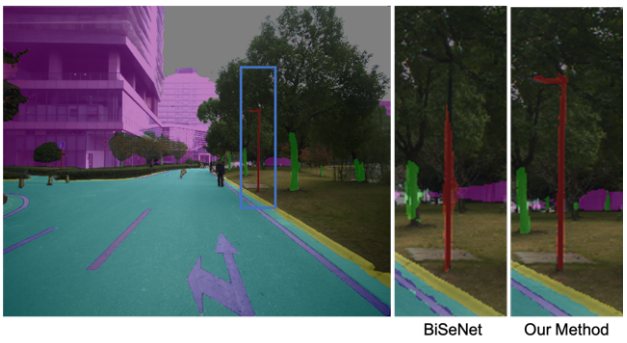


Fig. 4: Refinement of edge feature segmentation details: (a) Segmented image; (b) Segmentation detail of lamp post by BiSeNet; (c) Segmentation detail of our method.

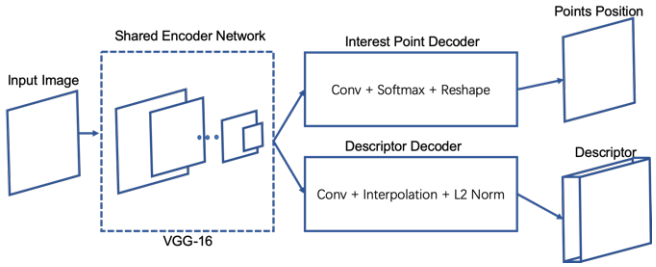


Fig. 5: Super-Point architecture: the whole data processing pipeline is based on the encoder-decoder framework.

Nowadays, more learning-based methods are replacing hand-crafted methods for their promising performance. Super-Point stands for one of the most efficient structures and a reasonable training pipeline.

In order to achieve better accuracy of the key-points, we enlarge the VGG network in Fig. 5 to store more textural information in the first step. In both decoders, reshape and interpolation operations are used to adjust the spatial dimension of output results.

III. Visual Features Extraction

Visual features used for optimization are further extracted from the image segmentation and the interest-point network output. The extraction methodology is as follows.

A. Ground Marker Features

Consider that a new image frame k, we can extract ground marking points[1] $\mathbf{p}_{\mathrm{I}}^{\mathrm{g}}$ with its identical semantic label, i.e., the lane markers (including dashed lines, solid lines, arrows, crosswalks and so on) and the curb points in the image frame coordinate $\{\mathscr{I}\}$. For the last estimated pose $\mathbf{T}_{\mathrm{C},k-1}^{\mathrm{W}} = \left[\mathbf{R}_{\mathrm{C},k-1}^{\mathrm{W}} \quad \mathbf{t}_{\mathrm{C},k-1}^{\mathrm{W}}\right]$, where $\mathbf{T}_{\mathrm{C}}^{\mathrm{W}}$ is the homogeneous transformation matrix from world frame $\{\mathscr{W}\}$ to camera frame $\{\mathscr{C}\}$, we are able to derive the current predicted pose

[1]The superscript stands for the semantic label and the subscript stands for the coordinate frame. We use lowercase to represent point 2-D point in image frame and uppercase to represent 3-D point.

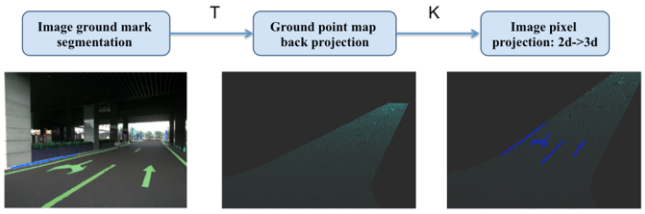


Fig. 6: The ground features (lane markings in green and curb lower edge in blue) extraction procedure: (a) Mask the semantic label in the raw image; (b) Obtain feature depth with map ground point projected into camera frame by predicted pose; (c) Re-project semantic features into 3-D point-cloud.

$\bar{\mathbf{T}}_{\mathrm{C},k}^{\mathrm{W}}$ with the delta motion from the VIO odometer,

$$\bar{\mathbf{R}}_{\mathrm{C},k}^{\mathrm{W}} = \mathbf{R}_{\mathrm{C},k-1}^{\mathrm{W}} \left(\mathbf{R}_{\mathrm{C},k-1}^{\mathrm{O}}\right)^{-1} \mathbf{R}_{\mathrm{C},k}^{\mathrm{O}} \tag{1}$$

$$\bar{\mathbf{t}}_{\mathrm{C},k}^{\mathrm{W}} = \bar{\mathbf{R}}_{\mathrm{C},k}^{\mathrm{W}} \left(\mathbf{R}_{\mathrm{C},k-1}^{\mathrm{O}}\right)^{-1} \left(\mathbf{t}_{\mathrm{C},k}^{\mathrm{O}} - \mathbf{t}_{\mathrm{C},k-1}^{\mathrm{O}}\right) + \mathbf{t}_{\mathrm{C},k-1}^{\mathrm{W}} \tag{2}$$

where $\mathbf{R}_{\mathrm{C}}^{\mathrm{O}}$ and $\mathbf{t}_{\mathrm{C}}^{\mathrm{O}}$ stands for the rotation and translation from odometry coordinate frame $\{\mathscr{O}\}$ to $\{\mathscr{C}\}$. Based on the ground surface planar assumption, the inverse depth d_p of point $\mathbf{p}_{\mathrm{I}}^{\mathrm{g}}$ can be recovered with the adjacent 3-D points in the ground surface map points $\mathbf{P}_{\mathrm{m}}^{\mathrm{g}}$. The nearest neighbour of $\mathbf{p}_{\mathrm{I}}^{\mathrm{g}}$ from re-projected map points $\bar{\mathbf{p}}_{\mathrm{I}}^{\mathrm{g}}$ can be found by a kD-tree process,

$$\bar{\mathbf{p}}_{\mathrm{I}}^{\mathrm{g}} \simeq \pi\left[\left(\bar{\mathbf{T}}_{\mathrm{C},k-1}^{\mathrm{W}}\right)^{-1} \mathbf{P}_{\mathrm{m}}^{\mathrm{g}}\right] \tag{3}$$

where $\pi[\cdot]$ is the camera intrinsic model. Fig. 6 explains the procedures for ground marking feature extraction.

B. Ground Sampled Features

Ground points are sampled using the same methodology with ground lane markings. The sampled points are projected to camera frame to form constraints aimed at height, roll and pitch channel, especially in areas without effective ground marking features.

C. Pole-like Features

A pole-like object (e.g. lamp posts, tree trunks, etc.) provides bearing constraint for solving camera pose, and multiple constraints determine a unique 6-DOF pose. With the pixel-level pole-like segmentation output, we extract the pole-like features for residualization as the following procedures:

1) Process the pixel-level segmentation mask into a binary image and fill holes with common morphology operators.
2) In order to record the peak coordinates as the initial position of each pole, we obtain the histograms of the column response in the pole binary image and group the light poles based on each wave peak.
3) Based on the peak position, we search up pixels belonging to the light pole and perform a least squares fit to obtain the descriptor of the line model and end points.

D. Point Features

Details of interest-point feature extraction are explained in Section. II-A.1, and instead of selecting all the points extracted in a certain frame, we select those can be stably tracked among several consecutive frames and remove those on the dynamic objects with the aid of semantic segmentation (see Fig. 8).

IV. Global Map

Global map provides accurate global references to the online extracted features and in this work we use 3-D point-cloud map and global SFM to build the map.

1) Semantic Feature Map: With the high-precision point-cloud map built by INS system and multiple 3-D LiDAR devices, the semantic level point-cloud is further processed with the DNN model described in [28]. We extract the categories identical to the above-mentioned features types, i.e., ground surface, lane markings, curb points, pole-like objects and buildings. As described in Fig. 6, since the ground points are significant for depth estimation in the re-projection process, we further process the 3-D ground points for noise removal and ground hole filling based on the moving least squares (MLS) algorithm. The lane marking points and curb points can be directly acquired from the KpConv network[2] output.

The 3-D pole-like map features are extracted with the following steps:

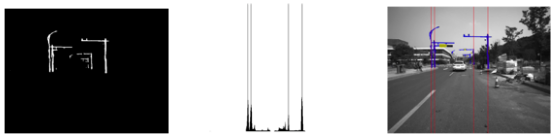
1) Get the instance-level pole-like objects points using point-cloud Euclidean clustering;
2) Fit a 7-parameter cylindrical model to each pole-like points cluster with *RANSAC*;
3) Merge multiple pole model into one if they are divided into segments vertically in the first step and redo the second step to get an updated model;
4) Save the pole with the end points and cylindrical parameters as its descriptors.

2) SFM Feature Map: We use the consecutive images of mapping dataset to reconstruct the whole 3-D point-cloud by the off-line SFM process [25]. To align the SFM point feature with the semantic feature map, we use the fixed pose approach (the poses are optimized from LiDAR SLAM backend) in the feature generation process. However, visual point behaving unstable to illumination and reconstruction error introduced in the reprojection and matching phase, the accuracy of reconstrcuted point-cloud can hardly compare to LiDAR SLAM mapping performance. We further remove the outlier of the SFM point-cloud with the assistance of kD-tree nearest neighbor search built by LiDAR point-cloud map. The visualization of both semantic feature layer and SFM point layer can be seen in Fig. 8.

V. Optimization Solver

For k-th frame, the states to be optimized is simply defined as $\mathbf{x}_k \doteq \begin{bmatrix}\boldsymbol{\theta}_k^{\mathrm{T}} & \mathbf{p}_k^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$ for a single-frame optimization and the problem is formed as a non-linear optimization to solve the pose iteratively. We divide the cost $\mathscr{C}_k$ into components according to different feature types.

[2]The model is trained with point intensity as an add-on point attribute.



(a) Binary (b) Response (c) Pole

Fig. 7: This shows the intermediate results of the pole extraction and representation. (a) Binary pole-like semantic image. (b) Wave peak of the light pole pixels' vertical histogram. (c) Fitted vertical line model.

A. Ground Marker Features

With the ground surface planar assumption, we are able to optimize these features with global semantic with 3-D point-cloud approaches. We use local map solution to accumulate the lane marking and curb points, since the optimization is not robust with the measurements only from the current frame, due to the limited field-of-view and the error in image segmentation, especially when the vehicle is performing aggressive motions. Then we use the predicted camera pose $\bar{\mathbf{x}}_k$ to project local map points to map frame as $\bar{\mathbf{P}}_k^{\mathrm{l}}$, search global map kD-tree and obtain the correspondence $\mathbf{P}_{\mathrm{m}}^{\mathrm{l}}$ and the cost of ground marking features is derived as:

$$\mathscr{C}_k^{\mathrm{l}} \doteq \sum \frac{1}{2} \| \bar{\mathbf{P}}_k^{\mathrm{l}} - \mathbf{P}_{\mathrm{m}}^{\mathrm{l}} \|^2 \tag{4}$$

B. Ground Sampled Features

Different from lane marking points, the residual selected for the sampled ground points is the distance d_{g} from point to plane and the corresponding cost term is:

$$\mathscr{C}_k^{\mathrm{g}} \doteq \sum \frac{1}{2} d_{\mathrm{g}}^2 \tag{5}$$

C. Pole-like Objects

Based on fundamental matrix $\mathbf{F}$, we are able to project the 3-D map pole-like objects point $\mathbf{P}_{\mathrm{m}}^{\mathrm{p}}$ to image frame with the predicted pose. We use three points $\mathbf{l}_{\mathrm{m}} = (\mathbf{p}_1^{\mathrm{T}}, \mathbf{p}_2^{\mathrm{T}}, \mathbf{p}_3^{\mathrm{T}})$ to describe the projected map pole feature in the image frame. To find a correct match, firstly performs nearest neighbor search among the extracted pole features set with a pre-defined distance threshold, and then reject the wrong match with line direction and length. Mark the correct match pair as $(\mathbf{l}_{\mathrm{k}}, \mathbf{l}_{\mathrm{m}})$, the residual is the sum of the distance from map line points to line $\mathbf{l}_{\mathrm{k}}$, accumulating the residuals we have,

$$\mathscr{C}_k^{\mathrm{p}} \doteq \sum \frac{1}{2} \left(d_{\mathrm{p},0}^2 + d_{\mathrm{p},1}^2 + d_{\mathrm{p},2}^2 \right) \tag{6}$$

D. Point Features

To find the correspondence of a certain feature point $\mathbf{p}_k^{\mathrm{f}}$ in the global map, the map feature points $\mathbf{P}_{\mathrm{m}}^{\mathrm{f}}$ are projected with predicted pose to the image frame. The initial features matching pairs are selected with descriptor distance, which may introduce mismatches with similar texture. Hence, we

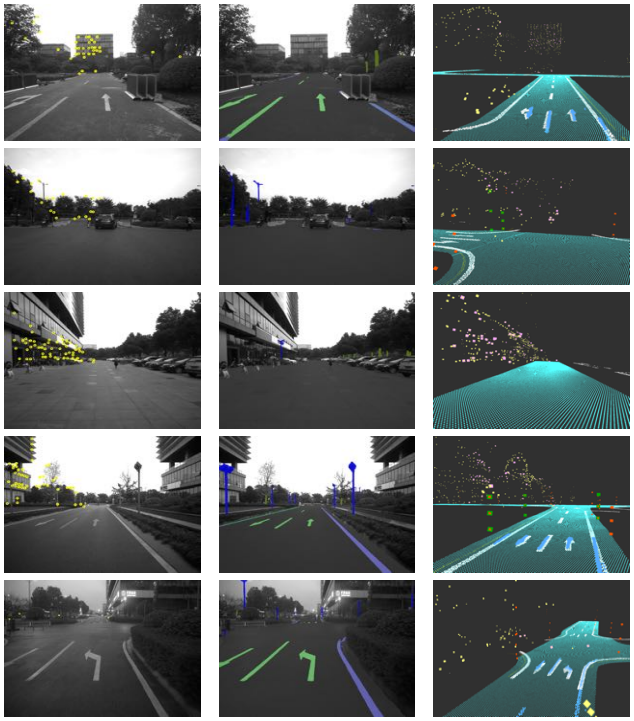
perform *RANSAC* to remove the outliers and the cost term is finalized with the inlier matches as,

$$\mathscr{C}_k^{\mathrm{f}} \doteq \sum \frac{1}{2} \|\bar{\mathbf{p}}_k^{\mathrm{f}} - \mathbf{p}'_{\mathrm{m}}\|^2 \tag{7}$$

We stack residuals and Jacobians of all the above terms and perform L-M optimization to solve the pose iteratively by Ceres solver [1]. To make the optimization numerically stable, we use Huber kernel as loss function and assign different weights to each feature type.

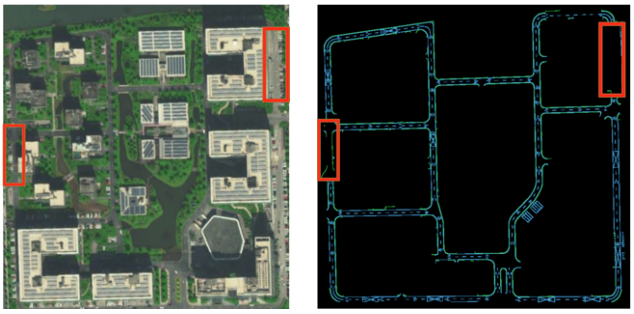
VI. Experimental Results

This section presents experiments of our proposed visual localization approach on the practical autonomous vehicle dataset. The testing platform is our in-house designed vehicle for autonomous delivery services. The camera selected is an industrial image sensor MV-CA032-10GM from HIKRobot, which outputs 1280×1024 resolution image at a rate of 10 Hz. The sensor suite also includes a low-cost MTi-3 MEMS IMU and a chassis build-in wheel encoder for visual-inertial odometry purpose. The whole software system runs on an Intel 8700 CPU and an Nvidia 2080 GPU module.



(a) Features (b) Semantic (c) Map

Fig. 8: Illustrations of different scenarios on the industrial park dataset. Three columns represent the point features matched with SFM map layer, semantic element extraction and matching with global map. The first row indicates the scene with point features and lane marking semantic elements; the second and the third rows show the scenes lack of effective semantic features, with no or only one lamp post; the fourth shows the case that both key-point features and semantic features exist and the last row presents the performance at dusk.



(a) Satellite map (b) Semantic map

Fig. 9: The industrial park area selected for experiment 1, with the red boxes indicating the zones without enough semantic information.

A. Road Test Experiments

To validate the performance and robustness of our proposed algorithm, we select various scenes for practical tests, which cover the cases with enough semantic information, i.e. the public road with lane markings, curbs or poles, and the cases without semantic information, e.g. narrow road inside residential areas or industrial parks. To evaluate the effectiveness and significance of our *semantic+feature points* approach, we have conducted several tests that enable only one or several of the proposed approaches.

1) Experiment 1: This experiment is designed to test the accuracy and robustness of each proposed optimization method. We select an industrial park area (see Fig.9), where most of the scenes have enough semantic information. We have designed four combinations of function modules as follows:

- Lane-Curb (LC) module: enables lane and curb optimizer only to test the lane markings based approach;
- Pole-Curb-Ground (PCG) module: enables pole-like objects, curbs and ground sampled points;
- Feature-Only (FO) module: enables feature point only to test key-point feature based approach;
- Fusion: enables all feature types to show the superiority of our proposed method.

The experiment is also carried out to validate the visual pose robustness in both day and night condition as a standard benchmark[3](see Fig.8).

2) Experiment 2: This experiment is conducted to test the effectiveness of each module in various scenarios, e.g., the industrial park scene, the university campuses, the public roads, etc. We have accumulated around 52 km dataset and covered 6 different areas to obtain statistical results. We also tried out the above-mentioned four approaches on these datasets, and in some circumstances, relying on a certain type of feature would fail to localize due to lack of corresponding features.

[3]A visual localization benchmarking site: https://www.visuallocalization.net/benchmark/.

TABLE II: Performance comparison of our proposed optimization approaches under day and night conditions

dataset	day	night
m	.25/.50/5.0	.25/.50/5.0
deg	2/5/10	2/5/10
LC	15.79/69.06/90.15	4.1/45.3/52.1
PCG	12.55/18.06/50.44	8.3/12.8/34.2
FO	52.26/87.41/98.12	0.3/6.2/10.4
Fusion	**69.77/96.55/98.34**	**10.2/50.2/55.7**

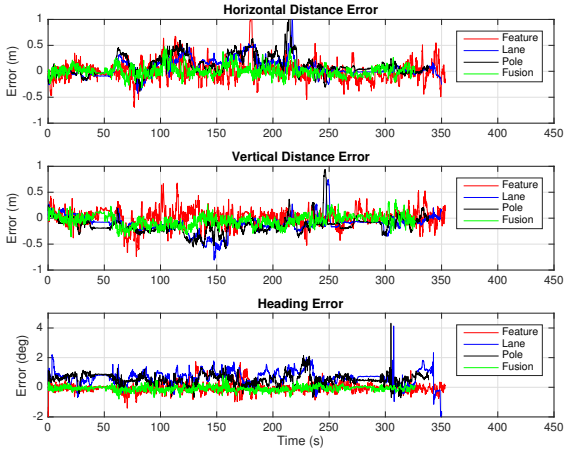


Fig. 10: Error statistics of vertical and horizontal direction in the vehicle frame, and heading angle respectively. The methods details are explained in experiment 1.

3) Experiment 3: To further evaluate the performance, we conducted an experiment using the KAIST [6] Urban Dataset [14] sequence urban39 with a total distance of 11.06 km. We chose three scenes, i.e. urban, highway, and suburban to compare our algorithm with CL+PA and PC semantic. The ground truth is KAIST vehicle baseline data and the evaluation method is the translational and rotational root mean squared error (RMSE).

TABLE III: RMSE statistics of KAIST urban 39 dataset.

dataset Trans (m) Rot (deg)	Suburban	Urban	Highway
CL+PA [17]	0.604 0.882	0.580 1.080	1.806 0.935
PC Semantic [27]	1.798 0.464	0.893 0.914	2.494 0.907
Ours	**0.573** **0.510**	**0.54** **0.68**	**1.964** **0.853**

B. Performance Analysis

1) Pose accuracy and robustness: Pose accuracy is obtained by comparison of the visual global pose estimation and the ground truth, which is recorded by a high-precision INS system. Fig. 10 shows the pose error of vertical and horizontal position error in the vehicle frame as well as the heading error. It can be revealed that the fused optimization outperforms all the other approaches with the highest precision. Fig. 11 shows the correlation of the matched feature number and the pose accuracy, where we can find the heading error is correlated with the matched pole count. The horizontal position error in the vehicle frame is affected by the lane matching performance, since lane markings or curbs can hardly provide vertical constraints on straight roads.

TABLE IV: Localization accuracy comparison with 4 optimization methods for multiple datasets.

dataset	Park	Campus	Public Road
m	.25/.50/5.0	.25/.50/5.0	.25/.50/5.0
deg	2/5/10	2/5/10	2/5/10
LC	15.79/69.06/90.15	14.2/55.3/85.1	18.02/75.43/95.3
PCG	12.55/18.06/50.44	8.3/12.8/54.2	23.5/35.2/45.5
FO	52.26/87.41/98.5	49.9/77.2/96.9	34.4/65.2/83.6
Fusion	**69.77/93.55/99.3**	**60.9/83.2/95.7**	**69.77/90.12/92.3**

For pose robustness, the last row in Fig. 8 shows the test scene at around 6 pm and the first figure reveals that few feature points are matched with the SFM map layer due to the dark light condition, while the semantic approach present high reliability to illumination change. Table. II presents the fusion method is more robust than other methods.

2) Pose effectiveness: We follow the standard visual localization evaluation method proposed in [24], where we use three categories, i.e. (0.25m, 2°), (0.5m, 5°), (5m, 10°), to represent the localization accuracy. Indeed, for autonomous driving function, the localization error above 0.5m is unacceptable. Table. IV presents the statistical results on the 52 km road tests in various scenes, where we can see the semantic based methods fail to localize without effective semantic information, e.g. the university campuses where inside the building area there are no lane markings or pole-like objects. The feature based method performs poorly in the texture-less scenario, e.g. on the public road where the view is blocked or features extracted are too far away and only lane marking or curb can be used.

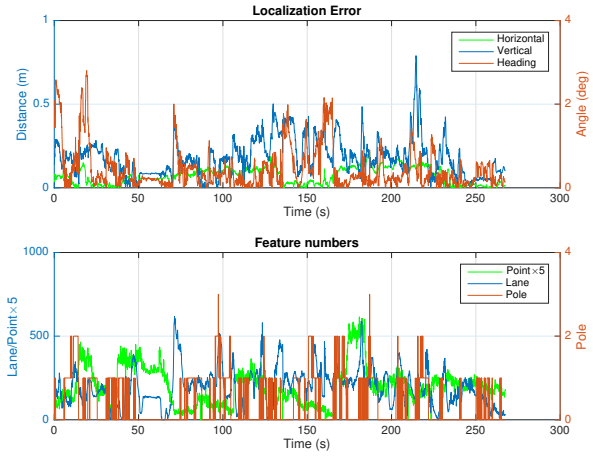


Fig. 11: The relationship between pose error and feature count with our Fusion method. The lower chart shows the number of each type of features where we put 5× key-point feature count for visualization.

3) Runtime performance: Our module is able to provide real-time visual pose estimation with 10Hz image input when deploying on the real vehicle platform. The whole extraction, matching and optimization process consumes 70% of a CPU thread and around 1GB GPU memory. The computational delay for each module is: semantic segmentation around 15ms, feature point extraction less than 10ms in average and optimization around 20ms with 5 maximum iterations.

VII. Conclusions

In conclusion, a vision-only global localization approach using both semantic features and key-point features matched with LiDAR semantic point-cloud and SFM reconstructed feature map, is proposed for autonomous vehicle applications. We have carried out several experiments with our in-house developed autonomous vehicle platforms in various challenging cases, including urban public road, industrial parks, university campuses and etc. According to the road test results, our module outperforms the other approaches using only semantic features or key-point features on the aspects of precision and robustness. To compare with the state-of-the-art visual localization methods, we test our algorithm on KAIST Urban 39 dataset and achieve satisfactory results. In practice, our method has been deployed on Alibaba's autonomous vehicles for delivery services with a low-cost sensor suite in disparate challenging scenes with high accuracy and robustness.

References

[1] S. Agarwal, K. Mierle, et al. Ceres solver. 2012.

[2] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother. Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6684–6692, 2017.

[3] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz. Geometry-aware learning of maps for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2616–2625, 2018.

[4] N. Brasch, A. Bozic, J. Lallemand, and F. Tombari. Semantic monocular slam for highly dynamic environments. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 393–400. IEEE, 2018.

[5] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua. Brief: Computing a local binary descriptor very fast. *IEEE transactions on pattern analysis and machine intelligence*, 34(7):1281–1298, 2011.

[6] Y. Choi, N. Kim, S. Hwang, K. Park, J. S. Yoon, K. An, and I. S. Kweon. Kaist multi-spectral day/night data set for autonomous and assisted driving. *IEEE Transactions on Intelligent Transportation Systems*, 19(3):934–948, 2018.

[7] R. Clark, S. Wang, A. Markham, N. Trigoni, and H. Wen. Vidloc: A deep spatio-temporal model for 6-dof video-clip relocalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6856–6864, 2017.

[8] L. Deng, M. Yang, B. Hu, T. Li, H. Li, and C. Wang. Semantic segmentation-based lane-level localization using around view monitoring system. *IEEE Sensors Journal*, 19(21):10077–10086, 2019.

[9] D. DeTone, T. Malisiewicz, and A. Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 224–236, 2018.

[10] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8092–8101, 2019.

[11] J. Fu, J. Liu, Y. Wang, J. Zhou, C. Wang, and H. Lu. Stacked deconvolutional network for semantic segmentation. *IEEE Transactions on Image Processing*, 2019.

[12] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012.

[13] B.-S. Hua, M.-K. Tran, and S.-K. Yeung. Pointwise convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 984–993, 2018.

[14] J. Jeong, Y. Cho, Y.-S. Shin, H. Roh, and A. Kim. Complex urban dataset with multi-level sensors from highly diverse urban environments. *The International Journal of Robotics Research*, page 0278364919843996, 2019.

[15] A. Kendall, M. Grimes, and R. Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pages 2938–2946, 2015.

[16] H. Li, F. Nashashibi, and G. Toulminet. Localization for intelligent vehicle by fusing mono-camera, low-cost gps and map data. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 1657–1662. IEEE, 2010.

[17] Z. Liao, J. Shi, X. Qi, X. Zhang, W. Wang, Y. He, R. Wei, and X. Liu. Coarse-to-fine visual localization using semantic compact map. *arXiv preprint arXiv:1910.04936*, 2019.

[18] T. Naseer, G. L. Oliveira, T. Brox, and W. Burgard. Semantics-aware visual localization under challenging perceptual conditions. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2614–2620. IEEE, 2017.

[19] S. Nedevschi, V. Popescu, R. Danescu, T. Marita, and F. Oniga. Accurate ego-vehicle global localization at intersections through alignment of visual data with digital map. *IEEE transactions on intelligent transportation systems*, 14(2):673–687, 2012.

[20] O. Pink. Visual map matching and localization using a global feature map. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–7. IEEE, 2008.

[21] K. Qiu, T. Liu, and S. Shen. Model-based global localization for aerial robots using edge alignment. *IEEE Robotics and Automation Letters*, 2(3):1256–1263, 2017.

[22] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, volume 11, page 2. Citeseer, 2011.

[23] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12716–12725, 2019.

[24] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8601–8610, 2018.

[25] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[26] J. L. Schönberger, M. Pollefeys, A. Geiger, and T. Sattler. Semantic visual localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6896–6906, 2018.

[27] E. Stenborg, C. Toft, and L. Hammarstrand. Long-term visual localization using semantically segmented images. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6484–6490. IEEE, 2018.

[28] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6411–6420, 2019.

[29] F. Walch, C. Hazirbas, L. Leal-Taixe, T. Sattler, S. Hilsenbeck, and D. Cremers. Image-based localization using lstms for structured feature correlation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 627–637, 2017.

[30] C. Yu, Z. Liu, X.-J. Liu, F. Xie, Y. Yang, Q. Wei, and Q. Fei. Ds-slam: A semantic visual slam towards dynamic environments. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1168–1174. IEEE, 2018.

[31] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 325–341, 2018.

[32] Y. Yu, H. Zhao, F. Davoine, J. Cui, and H. Zha. Monocular visual localization using road structural features. In *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pages 693–699. IEEE, 2014.