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

MOZARD: Multi-Modal Localization for Autonomous Vehicles in Urban Outdoor Environments

Lukas Schaupp[1], Patrick Pfreundschuh[3], Mathias Bürki[1,2], Cesar Cadena[1],
Roland Siegwart[1], and Juan Nieto[1]
[1]Autonomous Systems Lab, ETH Zürich, {firstname.lastname}@mavt.ethz.ch
[2]Sevensense Robotics AG, {firstname.lastname}@sevensense.ch [3]ETH Zurich, {firstname.lastname}@student.ethz.ch

***Abstract*— Visually poor scenarios are one of the main sources of failure in visual localization systems in outdoor environments. To address this challenge, we present MOZARD, a multi-modal localization system for urban outdoor environments using vision and LiDAR. By fusing key point based visual multi-session information with semantic data, an improved localization recall can be achieved across vastly different appearance conditions. In particular we focus on the use of curbstone information because of their broad distribution and reliability within urban environments. We present thorough experimental evaluations on several driving kilometers in challenging urban outdoor environments, analyze the recall and accuracy of our localization system and demonstrate in a case study possible failure cases of each subsystem. We demonstrate that MOZARD is able to bridge scenarios where our previous key point based visual approach, VIZARD, fails, hence yielding an increased recall performance, while a similar localization accuracy of $0.2m$ is achieved.**

I. Introduction

Due to increasing traffic in urban environments and changing customer demands, self-driving vehicles are one of the most discussed and promising technologies in the car and robotics industry. Still no system was presented yet, that allows to localize robustly under all light, weather and environmental conditions. However, precise localization is a vital feature for each autonomous driving task, since a wrong pose estimate may lead to accidents. Especially in urban environments, safety margins on the position of the car are small due to crowded traffic and other traffic participants (e.g. pedestrians, cyclists). Because of multi-path effects or satellite blockage, GPS sensors cannot be used reliably under those urban conditions. Thus, other sensors have to be used for localization. For this purpose, mainly LiDARs and cameras have been used in the last years [1]. Appearance changes in urban environments challenge visual localization approaches [2]. However, such driving scenarios contain persistent structures even under those appearance changes. Curbstones are one such feature [3]. Curbstones are used to protect pedestrians from cars and to separate the sidewalk from the street. As they delimit the street, they also offer information of the area where the car is allowed to be placed in. Detection of their position relative to the car can thus allow to localize inside the lane. In contrast to other geometrical shapes such as poles and road markings, curbstone measurements are found more frequently in urban environments and yield a reliable, continuous lateral constraint for pose refinement. Due to their shape and their contrasting color with respect to the pavement in many cases, they can be detected both in camera images as well as in LiDAR point clouds. Therefore, our pipeline named MOZARD extends our previous visual localization system - VIZARD [4] with the use of additional geometrical features from LiDAR data, for the self-driving cars used in the *UP-Drive* project[1]. In a thorough evaluation of our proposed localization system using our long-term outdoor dataset collection, we investigate key performance metrics such as localization accuracy and recall and demonstrate in a case study possible failure scenarios.

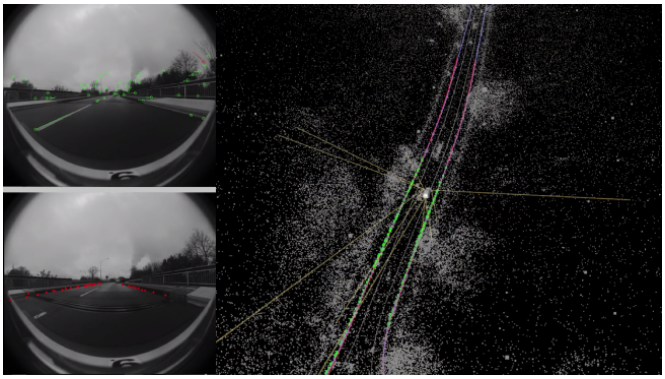

Fig. 1. We aim at accurately localizing the *UP-Drive* vehicle in a map of features extracted from vision and LiDAR data depicted on the right side. Our proposed algorithm can be separated into two distinct steps. We extract key point based features from camera and additional 3D geometrical curbstone information from a semantic vision-LiDAR pipeline. The features extracted from images of the surround-view camera system (top-left corner) are matched against $3D$ landmarks in the map while our raw curbstone measurements (bottom-left corner) are matched to their corresponding landmarks. Inlier matches, centered on the estimated $6DoF$ pose of the vehicle in the map, are illustrated as dark yellow lines on the right side. Purple indicates our pre-generated curbstone map data represented by splines. During runtime we downsample the nearest splines spatially to match with the current raw curbstone measurements indicated by the red and green color.

We see the following aspects as the main contributions of this paper:

- A semantic extension of our key point based localization pipeline based upon the extraction of curbstone information is presented, that allows to bridge sparse key point based feature scenarios in visual localization.
- In a thorough evaluation on the long-term dataset col-

[1]The **UP-Drive** project is a research endeavor funded by the European Commission, aiming at advancing research and development towards fully autonomous cars in urban environment. See www.up-drive.eu.

lection *UP-Drive*, we demonstrate a reliable localization performance across different appearance conditions in urban outdoor environments. We compare our results to our vision based localization pipeline and demonstrate significant performance increases.
- A computational performance analysis showing that our proposed algorithm exhibits real-time capabilities and better scalability.

II. Related Work

Since our localization system is a multi-modal semantic extension of our previous work we will concentrate the related work on frameworks that exploit semantic features using either one modality or fusing multiple modalities in different ways. Therefore these works can be subdivided into each of their specific sensor setup. In contrast to our previous work, VIZARD, which is solely based on a state estimation framework that uses visual key points fused with wheel odometry, MOZARD uses additional curbstone features as a constraint for state estimation. Further details on our prior visual key point based system can be found in [4].

Vision-only: A recent example of using semantic features for the purpose of localization and mapping is Lu et al. [5] using a monocular camera for road-mark detection whereas other studies use traffic signs [6], line segments from multi-view cameras [7] or poles [8], [9] for feature matching. On the detection of curbstones, traditional image based curbstone detection mostly uses the vanishing point and color distribution to detect the corresponding pixels [10], [11]. Recent work such from Enzweiler et al. [12] and Panev et al. [3] also demonstrate a learning based approach to detect curbs in images. In contrast to the vision based approaches, we concentrate on the multi-modal aspect as provided by Goga et al. [13].

LiDAR-only: LiDAR based methods use assumptions about the shape of the semantic features like curbs, poles and planes by evaluating the difference in elevation [14], [15], slope [14] or curvature [16], [17]. Authors such as Schaefer et al. [18] detect and extract 3D poles from the scenery which are then being used for map tracking. In regards to the usage of curbstones, most applications use geographical map data [19], [20] or road networks [21] as a reference to localize with detected curbstones. Unlike these works, our approach does not rely on external pre-generated data for curbstone map construction.

Vision and LiDAR: Recent work such from Kampker et al. [22] use a camera to extract pole like landmarks and a LIDAR for cylinder shapes for the task of self-localization. Kummerle et al. [23] demonstrate that basic geometric primitives can be extracted using vision and LiDAR to obtain road markings, poles and facades which can then be used for localization and mapping for the purpose of self-localization on various weather conditions. While these approaches use both modalities for mapping and localization separately, there has been recent research into cross-modality. Xiao et al. [24] uses a LiDAR to build an HD map and extract 3D semantic features from the map. Then a monocular camera is used with a deep learning based approach to match these semantic features with the ones from the camera. In contrast to the graph-based SLAM formulation used in our approach, the mentioned approaches are filter-based, with Extended Kalman Filters [25] or Monte Carlo Localization [20], [22], and they are evaluated at low speed and/or on short maps of a few hundred meters [24]. In addition they do not use raw curbstone measurements as a feature for localization and mapping [18]–[20], [23], [24]. Our work is evaluated on a long-term map with a length of over $5km$ using urban driving speed of around $50km/h$.

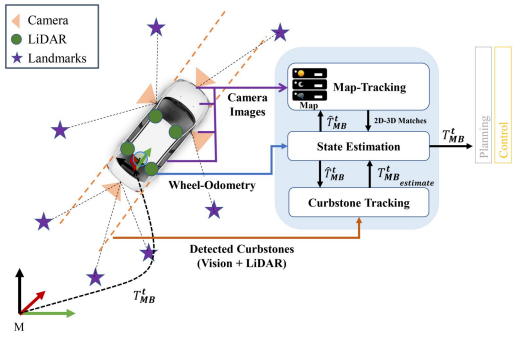

Fig. 2. The key point based map-tracking module extracts 2D features from current camera images, and matches them with 3D map landmarks locally in image space using a pose prior $\hat{T}^t_{MB}$ while our semantic map-tracking module matches point cloud based curbstone measurements between our map and immediate input. The state estimation module fuses the visual 2D-3D and geometrical 3D-3D matches with the current wheel-odometry measurement to obtain a current vehicle pose estimate T^t_{MB}.

III. Methodology

A schematic overview of MOZARD can be found in Figure 2. Since our work extends the VIZARD framework, we refer to the general methodology from Buerki et al. [4]. We assume that our visual localization pipeline already created a map by tracking and triangulating local 2D features extracted along a trajectory.

A. Curbstone Detection

For our curbstone detection we employ the work from Goga et al. [13]. Goga et al., fuse a vision-based segmentation CNN with LiDAR data. In a post-processing step they extract, refine and filter semantic curb ROIs to obtain new curb measurements. In the following we use their curbstone detection as input into our pipeline.

B. Map Extension

The curbstones are added to a map that was built using the VIZARD pipeline [4]. This map is called the base map in the following. Curbstone points detected in a specific LiDAR point cloud frame will be called curbstone observation. The detection pipeline finds curbstone point clouds in the vehicle frame $\mathcal{F}_B$. We find the closest vertex in time within the base map and allocate the curbstone point cloud. This is performed for all curbstone detections along the trajectory. From the base map, the respective transformations from the map coordinate frame $\mathcal{F}_M$ to the body frame at time t can be looked up. Using T^t_{MB} at each vertex in the base map

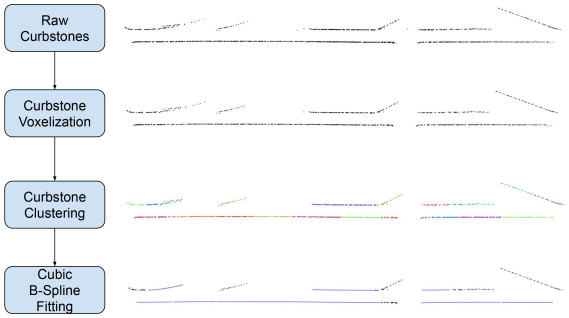

Fig. 3. Top view on a road with multiple entries and exits visualizing our curbstone parameterization pipeline. First we obtain raw curbstone measurements. After a voxelization processing step, we cluster the curbstones into segments. Finally, a cubic b-spline is fitted to each segment. In case the fitting algorithm fails, we store the raw points.

that contains a curbstone observation, a curbstone map in $\mathcal{F}_M$ can then be created.

C. Curbstone Map-Tracking

The curbstone tracking module is the core component of the curbstone localization pipeline. It performs an alignment between the map curbstones and the input curbstones to estimate the vehicle pose. A sanity check is performed to detect wrong alignments. If it is fulfilled, a pose constraint is added to the graph. The integration into the VIZARD system is shown in Figure 2. The single steps are explained in detail in the following section.

1) Reference Curbstone Retrieval: To retrieve the map curbstones, a prior estimate $\hat{T}^t_{MB}$ is used. In a fixed radius r_{lookup} around the estimated position $\hat{P}^t$, we then search for the closest vertex in Euclidean distance in the base map that contains curbstones. Furthermore, a criterion on the maximum yaw angle between the prior pose estimate and the base map vertex pose is used to prevent wrong associations.

2) Point Cloud Registration: Given the map and input curbstones a point cloud registration is performed. The NDT (Normal Distribution Transform) [26] implementation of the Point Cloud Library [27] is used as a registration algorithm. Additionally, outlier points are removed using a fixed ratio since some artifacts might only be included in one of both point clouds, due to occlusion or unsimilar detection. The point cloud registration estimates a transformation T_{align} that aligns the input cloud to the map cloud.

3) Sanity Check: In some regions, the input or map point cloud can consist of very few points. Matching in those scenarios can be ambiguous and lead to wrong associations. Thus, matching is only performed if both point clouds exceed a minimum amount of points. Since urban street scenarios change frequently, e.g due to constructions or parked cars, the input and map point cloud can diverge heavily. In those cases, point cloud registration might fail, ending up in wrong alignments and thereby wrong pose estimates. Therefore, a sanity check has to be performed, to detect wrong pose estimates. To do so, a matching score can be calculated, that can be used as an indicator if the alignment was successful. Magnusson et al. [26] proposed a matching score for NDT. It corresponds to the likelihood that the aligned input points lie on the reference scan. A more detailed explanation can be found in their work [26]. The alignment is considered valid, if the mean likelihood over all input points is higher than a threshold P_{min}.

4) State Estimation: We use the dual representation of an Extended Kalman Filter [28], [29], to estimate the state of our localization system. The pose estimate resulting from our curbstone localization is calculated as:

$$T^t_{MB_{estimate}} = \hat{T}^t_{MB} * T_{align}$$

If the sanity check is successful, $T^t_{MB_{estimate}}$ is added as an additional factor to the cost function presented in our previous work using a fixed covariance. For our experiments, the covariance was determined empirically. If the sanity check fails, the cost factor is not added to the cost function. To fuse the information of our camera, odometry and curbstone localization estimates, the MAP pose is estimated through minimization of the cost function. We use the GTSAM framework [30] and the Levenberg-Marquardt algorithm for iterative minimization. This enables a fall-back scenario where one of the sensor could potentially fail, while our pipeline would still yield a robust estimate.

D. Curbstone Parameterization

Since curbstone maps can scale quickly given the multitude of possibly redundant observations, a memory overhead is induced. To reduce this memory footprint, we perform a curbstone parameterization. Since the map contains several artifacts like intersections or roundabouts, a curve parameterization was preferred over a polyline. Curbstones are not continuous throughout the whole map, as they often end at intersections. Thus, it naturally makes sense to split the map into single connected regions. In a first step, the raw curbstone point cloud is subsampled. A clustering is then performed on the subsampled points, to find connected segments of a maximum length. The length-to-width ratio of each segment is then calculated. If a certain threshold is fulfilled, a Cubic B-Spline is fitted to the segment. By doing so, only the control points of the spline have to be saved, instead of all raw curbstone points. If the threshold is not fulfilled, the raw points are saved. The steps are explained in detail in the following and shown in Figure 3.

1) Subsampling: The high point density of the raw point cloud can result in high runtimes of the clustering as well as in overfitting of the spline to noise in the points. Thus, a spatial subsampling using a voxel grid with a leaf size of $30cm$ is performed. A point cloud of the means of the points inside each voxel is then used for clustering.

2) Clustering: The clustering is performed in a two-step fashion. First, a Euclidean clustering using a tolerance of more than $2m$ is performed to find large segments. Since the curvature can vary along long segments, fitting a single spline to it can be problematic, as different levels of detail are needed along the segment. An example is a curb going around a corner: While low curvature is desired in the straight sections, high curvature is needed in the area of the

corner to properly describe the curb. Thus, the coarse cluster is split into smaller sub-clusters with a maximum expansion of $20m$ before validating each sub-cluster by using an SVD Decomposition.

3) Cubic B-Spline Fitting: Spline fitting usually refers to fitting a spline that goes through each single input point. However, due to the noisy nature of the curbstone segment, a best fit given a fixed amount of control points is preferred in this case instead of fitting every single point. To achieve this, the approach proposed by Wang et al. [31] to fit an open cubic B-Spline is used. The number of control points is calculated proportionally to the approximate length of the segment, using $0.25\ points/m$, but a minimum of 4. For segments with a large width (indicating an intersection, road curve or round-about) a fixed amount of 20 points is used to allow for a proper representation. To validate our fitted spline, we define our goodness score GS as follows:

$$GS = \frac{\#Spline\ Inliers}{\#Spline\ Points} * \frac{\#Point\ Inliers}{\#Points}$$

whereas $SplineInliers$ is the number of sampled spline points close to a raw point and $PointInliers$ the number of raw points close to points sampled from a spline. Naively using all sub-segment points for the spline fitting can lead to an overfitting of the curve. Thus, the best set of points is found in a RANSAC-like manner. In each iteration, one third of the sub-segment points is sampled randomly. The spline is then fitted to the sampled points. Eventually, the spline with the highest score is chosen.

4) Spline Sampling: To be able to perform matching with the input cloud, points are spatially uniformly sampled from the splines during runtime. Those sampled points are then used as the map point cloud.

IV. Evaluation

In the following section, the performance of the proposed pipeline is evaluated and compared against the VIZARD pipeline as a benchmark. Long-term experiments in an urban scenario are performed on varying weather and appearance conditions. A special focus is set on how curbstone map tracking influences localization accuracy and recall. Example cases are presented, where localization gaps in the VIZARD pipeline could be bridged using curbstone localization. The sensor set-up of the *UP-Drive* vehicle and the datasets used in the experiments are described in the next section.

A. The UP-Drive Platform

For the collection of the datasets, the *UP-Drive* vehicle was used. Its sensor setup consists of four fish-eye cameras, resulting in a surround view of the car. Gray-scale images with a resolution of 640 x 400 pixels are recorded at $30Hz$. Five Velodyne LiDARs are mounted on top of the car. Curbstones are obtained from the approach as described by Goga et al. [13]. Additionally, a low-cost IMU and wheel tick encoders are used to provide odometry measurements. A consumer-grade GPS sensor is used to gain an initial position estimate and near-by map poses are used to generate an initial orientation estimate.

B. UP-Drive Dataset Collection

The *UP-Drive* dataset collection was recorded between December 2017 and November 2019 in Wolfsburg, Germany, at the Volkswagen factory and its surrounding area and aggregate a total driving distance of multiple 100 kilometers. The environment is urban, with common artifacts such as busy streets, buses, zebra crossings and pedestrians. Since the data was collected over several months, seasonal appearance changes as well as multiple weather and day-time conditions are present. For this work, our dataset selection is dependent on the availability of curbstone measurements, which result from the curbstone detection pipeline from Goga et al. [13]. Since the curbstone detection module was only enabled in some of our datasets, our evaluation dataset collection consists of 5 sessions which totals to 10 drives from August 2019 to November 2019. Each session consists of a total length of around 7.4 km, is gathered on the same trajectory and contains two partially overlapping routes in opposite directions. The recorded sessions consist of an equal amount of sunny and cloudy/rainy conditions. Recordings in rainy conditions are categorized as cloudy, since there is little difference in performance on rainy datasets as opposed to in dry conditions.

C. Metrics

1) Localization Recall: The fraction of the total travelled distance in which a successful localization was achieved is calculated as the localization recall $r[\%]$. While using only the visual pipeline, a localization attempt at time t is accepted as successful if a minimum of 10 inlier landmark observations is present after pose optimization. When using the combined pipeline, a localization is counted as successful, if the condition above is fulfilled or if a viable curbstone alignment (see section III-C) could be performed.

2) Localization Accuracy: As no ground-truth for the described dataset exists, the poses estimated by an *RTK GPS* sensor are used instead as a reference. *RTK GPS* altitude estimates are not reliable, thus the error in z can not be calculated reliably. Therefore, we focus on the planar $\mathbf{p^e_{xy}}$ and lateral translation error $\mathbf{p^e_y}$ as well as on the orientation error $\theta^{\mathbf{e}}_{\mathbf{xyz}}$.

D. Localization Accuracy and Recall

In order to fully rely on MOZARD to control the car in the UP-Drive project, a high localization recall with an accuracy below $0.5m$ is paramount, as only short driving segments with no localization may be bridged with wheel-odometry before the car may deviate from its designated lane. Curbstones are not available for the whole trajectory, but for around 89% of the distance of the map. We compare localization recall and accuracy of our localization system to our prior work on visual localization - VIZARD [4]. Note, however, that our prior work relied on the use of cameras for localization, as in contrast to the former, the latter is now able to use LiDAR and vision. To demonstrate that curbstones provide useful additional information, we construct and expand a map iteratively using multiple datasets. Our first

	$\mathbf{r}_{mt}[\%]$				$\bar{\mathbf{p}}^{\mathbf{e}}_{\mathbf{xy}}$, $\bar{\mathbf{p}}^{\mathbf{e}}_{\mathbf{y}}$				$\bar{\theta}^{\mathbf{e}}_{\mathbf{xyz}}$			
Evaluated on Session:	10-08	10-25	11-08	11-20	10-08	10-25	11-08	11-20	10-08	10-25	11-08	11-20
Sessions Contained in Map:												
MOZARD-Map:												
(08-21)	100.0	99.94	99.06	99.82	0.08 [0.34],0.04 [0.21]	0.07 [0.26], 0.03 [0.13]	0.09 [0.37], 0.04 [0.2]	0.13 [0.37], 0.05 [0.2]	0.64 [0.75]	0.74 [0.76]	1.04 [1.33]	1.09 [1.4]
(08-21; 10-08)	-	100.0	100.0	100.0	-	0.06 [0.15], 0.03 [0.08]	0.07 [0.24], 0.03 [0.13]	0.1 [0.29], 0.04 [0.14]	-	0.66 [0.65]	1.15 [1.4]	1.2 [1.4]
(08-21; 10-08; 10-25)	-	-	100.0	100.0	-	-	0.07 [0.22], 0.03[0.11]	0.09 [0.27], 0.03 [0.12]	-	-	1.21 [1.43]	1.23 [1.42]
(08-21; 10-08; 10-25; 11-08)	-	-	-	100.0	-	-	-	0.07 [0.18], 0.02 [0.1]	-	-	-	1.13 [1.38]
VIZARD-Map:												
(08-21)	100.0	98.2	97.94	91.76	0.08 [0.28], 0.04 [0.16]	0.07 [0.26], 0.03 [0.13]	0.09 [0.29], 0.04 [0.17]	0.13 [0.37], 0.05 [0.21]	0.64 [0.76]	0.74 [0.76]	1.1 [0.16]	1.07 [1.28]
(08-21; 10-08)	-	100.0	99.9	97.89	-	0.06 [0.13], 0.02 [0.07]	0.07 [0.24], 0.03 [0.13]	0.1 [0.29], 0.04 [0.14]	-	0.55 [0.67]	1.15 [1.4]	1.19 [1.37]
(08-21; 10-08; 10-25)	-	-	100.0	99.18	-	-	0.07 [0.22], 0.03 [0.11]	0.09 [0.25], 0.03 [0.13]	-	-	1.21 [1.43]	1.23 [1.41]
(08-21; 10-08; 10-25; 11-08)	-	-	-	99.7	-	-	-	0.07 [0.18], 0.02 [0.1]	-	-	-	1.13 [1.38]

TABLE I

THE LOCALIZATION PERFORMANCE ON THE *UP-Drive* DATASET, SHOWING LOCALIZATION RECALL, AND THE MEDIAN PLANAR $\bar{\mathbf{p}}^{\mathbf{e}}_{\mathbf{xy}}$, LATERAL $\bar{\mathbf{p}}^{\mathbf{e}}_{\mathbf{y}}$ AND ORIENTATION ($\bar{\theta}^{\mathbf{e}}_{\mathbf{xyz}}$) ACCURACY. THE 90 PERCENTILE IS SHOWN IN SQUARE BRACKETS. NUMBERING IN ROUND BRACKETS DEFINES THE TIMESTAMP OF THE SESSIONS USED FOR MAPPING. E.G. (08-21) REPRESENTS AUGUST, 21ST.

map is constructed from two datasets (one session) from August 2019. We then evaluate this map against multiple sessions from different months and add these sessions to our (multi-session) map in a iterative fashion. We present the resulting key evaluation metrics (localization recall $\mathbf{r}_{mt}[\%]$ and localization accuracy) in Table I over all sessions. By including this comparison, we aim at highlighting the gain in localization recall attainable by using MOZARD while keeping a consistent median translation and orientation error. As shown in Table I, MOZARD is able to attain close to a 100% recall performance on all 4 sessions on the *UP-Drive* dataset, while VIZARD performance increase correlates with the addition of sessions to its base map due to the change in visual appearance. We further note that in both cases the planar median localization accuracy are below $15cm$, while the median lateral error is below $10cm$. The median orientation errors are on average less than 1 degree. For MOZARD the 90th percentile shows an increase which is likely to be due to the higher uncertainty in precision of curbstone measurements.

E. Runtime

On our live car platform Goga et al. [13] demonstrated that their curbstone detection pipeline deployed on 2 Nvidia GTX 1080 takes around $20ms$ for the CNN image segmentation to complete on all 4 cameras. An additional $32ms$ are needed for the fusion of 5 LiDARs to run on an Intel i7-3770K CPU. Our curbstone alignment module takes an average of approximately $25ms$, while the map tracking module (with vision) can take from $27ms$ with a single session map up to $48ms$ on our largest multi-session map (see Table I) and has been evaluated on an Intel Xeon E3-1505M CPU. This would allow MOZARD to run with around $10Hz$ on a single machine on a single session map. Table II summarizes our findings.

F. Case Study

We provide further insights into our pipeline by showing specific failure examples for each component. Sample images of a section where the visual localization fails on the evaluated datasets are depicted in Figure 4. Due to occlusion and the absence of surrounding building structures, barely any stable visual cues are found in this section, preventing the visual localization system from matching a sufficient amount

Module	Average Runtime [ms]
Curbstone Detection	52
Curbstone Tracking	25
Map Tracking (VIZARD)	27-48
Total	104-125

TABLE II

RUNTIME OF EACH COMPONENT OF MOZARD . CURBSTONE DETECTION AND CURBSTONE TRACKING WITH AVERAGE RUNTIME OVER ALL EVALUATED DATASETS ON A SINGLE SESSION MAP, WHILE MAP TRACKING SHOWS AVERAGE RUNTIME FOR RUNNING ON A SINGLE SESSION MAP AND ON THE LARGEST MULTI-SESSION MAP.

Fig. 4. On the left, a sample image of a trajectory segment that fails to localize due to occlusion. A lack of key points renders it unfeasible to match a sufficient number of map landmarks. On the middle, the projected curbstone information is depicted in the camera frame in red - enabling a continued localization although visual localization failed. On the right, a sample image is depicted where our curbstone and vision pipeline fail. In this case curbstones are actually detected but alignment fails due to our constraints.

of landmarks from the map. This example demonstrates the current limitations of VIZARD, while our MOZARD pipeline is able to handle these sparse key point based scenarios. Unfortunately there are also scenarios where a lack of key points and curbstones exists or our curbstone alignment fails - hence conditions where both pipeline are likely to fail as depicted in the right image of Figure 4. A further extension of our current framework to other geometric shapes such as poles, road markings could provide additional useful information that would allow us to further increase our localization performance. Note that we used a single session map for the evaluation of this case study and VIZARD is able to bridge some of these scenarios if enough datasets are provided during the mapping process.

V. CONCLUSIONS

We presented MOZARD, a geometric extension to our visual localization system for urban outdoor environments. Through our evaluation on 8 datasets, including several kilometers of real-world driving conditions, we demonstrated the benefits of using curbstone information for localization

and mapping. Our datasets used in the experiments contain challenging appearance conditions such as seasonal changes, wet road surfaces and sun reflections. A comparison with our prior work demonstrated that we can achieve a higher recall performance while using less datasets during the mapping process, as the pipeline would fail due to sparse key point scenarios. Our run-time analysis shows that our approach demonstrates real-time capabilities. Although the curbstone detection stack of MOZARD takes in average more computing time than VIZARD, it is to note that an object segmentation/detection algorithm on a self-driving car has to be deployed for environmental perception independent of whether a localization takes place or not. Even taking in account the total computational time, our approach still runs at $10Hz$ while needing up to four times less data while achieving the same localization performance. We also showed specific cases where both of our pipelines would fail due to occlusions and/or curbstone misalignment giving suggestions for future work such as the extension of our approach to poles and road markings. Our findings showed that by extending a key point based visual localization approach with geometric features - curbstones in our case, an improvement in robustness with consistent high accuracy in localization is obtained.

ACKNOWLEDGMENT

This project has received funding from the EU H2020 research project under grant agreement No 688652 and from the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number 15.0284.

REFERENCES

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Towards the robust-perception age," 2016.

[2] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, 2016.

[3] S. Panev, F. Vicente, F. De la Torre, and V. Prinet, "Road curb detection and localization with monocular forward-view vehicle camera," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3568–3584, Sep. 2019.

[4] M. Bürki, L. Schaupp, M. Dymczyk, R. Dubé, C. Cadena, R. Siegwart, and J. Nieto, "Vizard: Reliable visual localization for autonomous vehicles in urban outdoor environments," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 1124–1130.

[5] Y. Lu, J. Huang, Y.-T. Chen, and B. Heisele, "Monocular localization in urban environments using road markings," 06 2017, pp. 468–474.

[6] L. D'Orazio, N. Conci, and F. Stoffella, "Exploitation of road signalling for localization refinement of autonomous vehicles," 07 2018, pp. 1–6.

[7] K. Hara and H. Saito, "Vehicle localization based on the detection of line segments from multi-camera images," vol. 27, pp. 617–626, 01 2015.

[8] L. Weng, M. Yang, L. Guo, B. Wang, and C. Wang, "Pole-based real-time localization for autonomous driving in congested urban scenarios," in *2018 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, Aug 2018, pp. 96–101.

[9] R. Spangenberg, D. Goehring, and R. Rojas, "Pole-based localization for autonomous vehicles in urban scenarios," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 2161–2166.

[10] N. John, B. Anusha, and K. Kutty, "A Reliable Method for Detecting Road Regions from a Single Image Based on Color Distribution and Vanishing Point Location," *Procedia Computer Science*, vol. 58, pp. 2–9, jan 2015.

[11] S. Hosseinyalamdary and M. Peter, "Lane level localization : using images and hd maps to mitgate the lateral error," in *Proceedings of ISPRS Hannover Workshop : HRIGI 17 – CMRT 17 – ISA 17 – EuroCOW 17, 6–9 June 2017, Hannover, Germany*, ser. ISPRS Archives, C. Heipke, Ed., vol. XLII-1/W1. International Society for Photogrammetry and Remote Sensing (ISPRS), 2017, pp. 129–134.

[12] M. Enzweiler, P. Greiner, C. Knoppel, and U. Franke, "Towards multi-cue urban curb recognition," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, jun 2013, pp. 902–907.

[13] S. E. C. Goga and S. Nedevschi, "Fusing semantic labeled camera images and 3D LiDAR data for the detection of urban curbs," in *Proceedings - 2018 IEEE 14th International Conference on Intelligent Computer Communication and Processing, ICCP 2018*, 2018.

[14] Z. Liu, J. Wang, and D. Liu, "A new curb detection method for unmanned ground vehicles using 2d sequential laser data," 2013.

[15] A. Miraliakbari, M. Hahn, and S. Sok, "Automatic extraction of road surface and curbstone edges from mobile laser scanning data," 2015.

[16] C. F. Lopez, D. F. Llorca, C. Stiller, and M. Á. Sotelo, "Curvature-based curb detection method in urban environments using stereo and laser," *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 579–584, 2015.

[17] C. F. Lopez, R. Izquierdo, D. F. Llorca, and M. Á. Sotelo, "Road curb and lanes detection for autonomous driving on urban scenarios," *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1964–1969, 2014.

[18] A. Schaefer, D. Buscher, J. Vertens, L. Luft, and W. Burgard, "Long-term urban vehicle localization using pole landmarks extracted from 3-d lidar scans," 09 2019, pp. 1–7.

[19] P. Bonnifait, M. Jabbour, and V. Cherfaoui, "Autonomous navigation in urban areas using GIS-managed information," *International Journal of Vehicle Autonomous Systems*, vol. 6, no. 1/2, p. 83, 2008.

[20] B. Qin, Z. J. Chong, T. Bandyopadhyay, M. H. Ang, E. Frazzoli, and D. Rus, "Curb-Intersection Feature Based Monte Carlo Localization on Urban Roads," Tech. Rep.

[21] J. Stueckler, H. Schulz, and S. Behnke, "In-lane localization in road networks using curbs detected in omnidirectional height images," in *Proceedings of Robotik 2008*, 2008.

[22] A. Kampker, J. Hatzenbuehler, L. Klein, M. Sefati, K. Kreisköther, and D. Gert, *Concept Study for Vehicle Self-Localization Using Neural Networks for Detection of Pole-Like Landmarks: Proceedings of the 15th International Conference IAS-15*, 01 2019, pp. 689–705.

[23] J. Kummerle, M. Sons, F. Poggenhans, T. Kuhner, M. Lauer, and C. Stiller, "Accurate and efficient self-localization on roads using basic geometric primitives," 05 2019, pp. 5965–5971.

[24] Z. Xiao, K. Jiang, S. Xie, T. Wen, C. Yu, and D. Yang, "Monocular vehicle self-localization method based on compact semantic map," 05 2018.

[25] H. Lee, J. Park, and W. Chung, "Localization of Outdoor Mobile Robots Using Curb Features in Urban Road Environments," *Mathematical Problems in Engineering*, vol. 2014, pp. 1–12, apr 2014.

[26] M. Magnusson, "The Three-Dimensional Normal-Distributions Transform —an Efficient Representation for Registration, Surface Analysis, and Loop Detection," 2009.

[27] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[28] M. Burri, M. Bloesch, D. Schindler, I. Gilitschenski, Z. Taylor, and R. Siegwart, "Generalized information filtering for mav parameter estimation," in *IROS*, 2016.

[29] H. Strasdat, J. M. Montiel, and A. J. Davison, "Visual slam: why filter?" *IVC*, 2012.

[30] F. Dellaert, "Factor graphs and gtsam: A hands-on introduction," Georgia Institute of Technology, Tech. Rep., 2012.

[31] W. Wang, H. Pottmann, and Y. Liu, "Fitting b-spline curves to point clouds by curvature-based squared distance minimization," *ACM Transactions on Graphics*, vol. 25, pp. 214–238, 05 2006.