2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

A Vision-Centric Approach for Static Map Element Annotation

Jiaxin Zhang[1], Shiyuan Chen[1], Haoran Yin[1], Ruohong Mei[1], Xuan Liu[2], Cong Yang[1†], Qian Zhang[3] and Wei Sui[1]

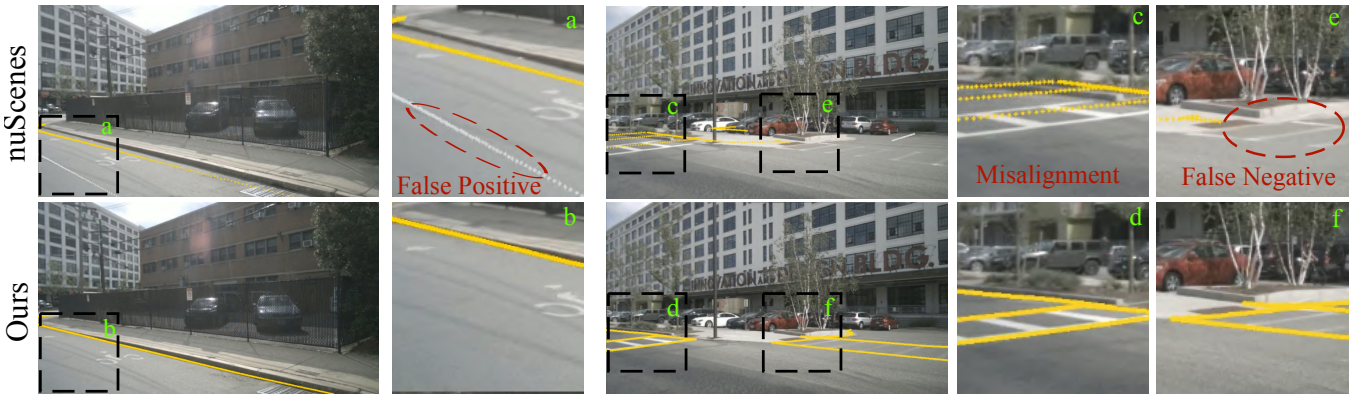


Fig. 1: Reprojection consistency and accuracy comparison. The top and bottom lines show HD map reprojection images and zoom-in details of nuScenes and our proposed method, respectively. The yellow dots represent road teeth, and the white dots represent lane dividers. The nuScenes HD Map has some inconsistent elements with respect to the actual road environments, including false positive (FP) and false negative (FN) road element annotation. For example, the image shows no lane marking between the bicycle and vehicle lane, but the HD Map indicates a lane divider (a,b). The image shows ped-crossing marking but no HD Map marking in the corresponding area (e, f). In contrast, the HD Map from our proposed method shows better reprojection accuracy (c, d) and consistency.

***Abstract*— The recent development of online static map element (a.k.a. HD Map) construction algorithms has raised a vast demand for data with ground truth annotations. However, available public datasets currently cannot provide high-quality training data regarding consistency and accuracy. To this end, we present CAMA: a vision-centric approach for Consistent and Accurate Map Annotation. Without LiDAR inputs, our proposed framework can still generate high-quality 3D annotations of static map elements. Specifically, the annotation can achieve high reprojection accuracy across all surrounding cameras and is spatial-temporal consistent across the whole sequence. We apply our proposed framework to the popular nuScenes dataset to provide efficient and highly accurate annotations. Compared with the original nuScenes static map element, models trained with annotations from CAMA achieve lower reprojection errors (e.g., 4.73 vs. 8.03 pixels).**

I. INTRODUCTION

The technical stack of perception algorithms for self-driving has recently been transforming from rule-based to data-driven methods [1]–[3]. Notably, the online high-definition map (HD Map) construction is becoming the mainstream of LiDAR-based and vision-centric bird-eye view (BEV) perception [4]. These methods train neural networks that take single or surrounding camera images as input and then estimate the surrounding environments directly in BEV and 3D space. The spatial transform from the perspective view to BEV is usually conducted in the neural network models explicitly [5]–[7] or implicitly [8]–[10]. These data-driven methods for perception algorithms drastically boost the advance of self-driving applications. It has several advantages, including less engineering efforts in corner cases debugging thanks to a data-driven closed-loop mechanism; better generalizability for various driving environments from the highway to city roads; better handling of occlusion and extreme illuminance conditions through temporal [11]–[14] and raw image inputs [15] in an end-to-end training manner.

Existing online HD map construction algorithms usually require high-quality and diverse labeled training data [16]. Accordingly, plenty of public datasets now provide annotation directly in 3D space. The available annotations can be roughly divided into HD-map-based and depth-reprojection-based [17]. For instance, the nuScenes dataset [18] provides HD Map alongside ego-poses. The HD-MapNet [19] is one of the pioneers that utilizes the nuScenes HD-Map as ground truth and trains a neural network to predict road elements directly in BEV space. Persformer [20] uses the depth-reprojection method to generate 3D lane annotation. Specifically, the LiDAR point clouds are projected into image space. Combined with 2D lane segmentation, 3D lane point clouds for each frame are generated through reprojection.

However, there are several challenges with these annotation methods regarding accuracy and consistency. We argue

† Corresponding author: Cong Yang (cong.yang@suda.edu.cn). Affiliation: [1] Ecology and Innovation Center of Intelligent Driving, Soochow University, Suzhou, China. [2] School of Information Science and Technology, Northeast Normal University, Changchun, China. [3] Horizon Robotics, Beijing, China. The code and dataset are available at https://github.com/manymuch/CAMA. This work was supported in part by the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (22KJB520008); in part by the Research Fund of Horizon Robotics (H230666); and in part by the Jiangsu Policy Guidance Program, International Science and Technology Cooperation, The Belt and Road Initiative Innovative Cooperation Projects (BZ2021016).

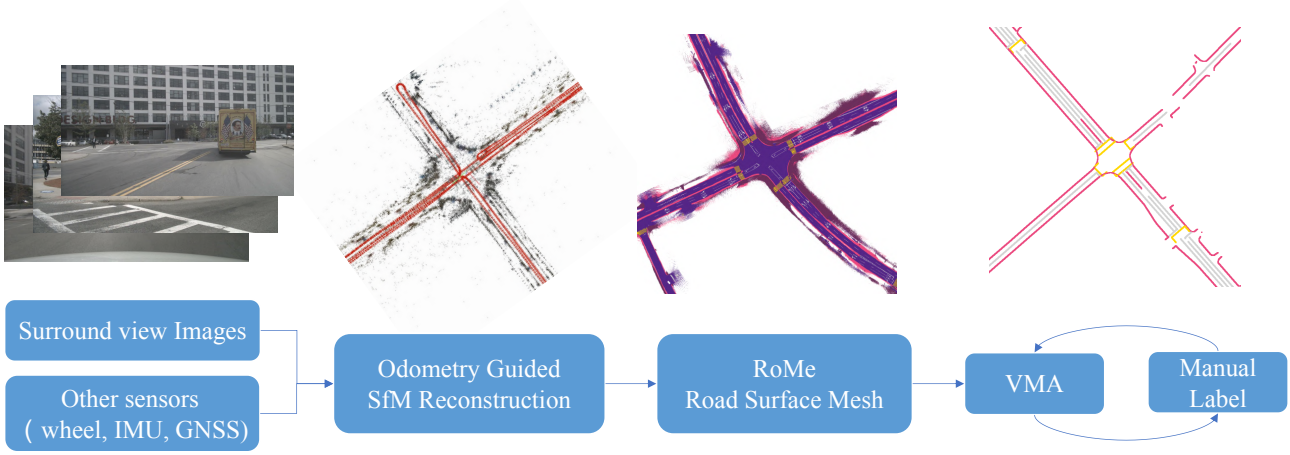


Fig. 2: Illustration of our proposed reconstruction and annotation pipeline. The surround images and auxiliary sensor data are fed into our proposed odometry-guided SfM to obtain highly accurate ego vehicle poses and sparse 3D points. A road surface mesh reconstruction called RoMe is applied to build dense 3D road surfaces with semantic labels. Finally, a vectorized map annotation (VMA) System is applied to produce a 3D HD map required by the perception algorithm as training data.

that annotating 3D road elements should royally reflect the actual world environments. In detail, we propose two aspects for analyzing: consistency and geometric accuracy. Fig. 1 (top line) shows an example that the default HD Map from the nuScenes dataset can not provide accurate information in these two aspects. The consistency emphasizes the correspondence between the 3D annotations and the 2D images. For example, in some areas, the lane divider between the vehicle and bicycle lane is provided in the HD Map. At the same time, the actual image shows no lane marking in the corresponding area. The geometric accuracy is reflected in the reprojection of the HD Map into original images. The reprojected road teeth (yellow dots) deviate from the actual road teeth in the image. The main reason is that the HD Map provided by the dataset does not have elevation information, and the ego-motion is not accurate with respect to the maps.

In light of these challenges, we propose **CAMA**: a vision-centric approach for **C**onsistent and **A**ccurate **M**ap **A**nnotation (see Fig. 2). Our proposed CAMA is distinguished in three aspects: (1) We propose using a whole 3D reconstruction pipeline to get accurate camera motion and a sparse point cloud mainly from surround images. Thus, it can be applied to even low-cost self-driving platforms without equipping LiDAR. (2) A road surface mesh reconstruction algorithm [21] is applied to reconstruct high-accuracy road surfaces. It can produce dense 3D road surfaces with both semantic and photometric information. (3) An auto map annotation tool [22] is applied to extract the vectorized lane representation from the road surface. Consequently, as shown in Fig. 1 (bottom), CAMA achieves high consistency and geometric accuracy compared with nuScenes default HD Map. Succinctly, our main contributions are as follows:

- We propose an efficient static element annotation framework, CAMA, for 3D road element annotation. The proposed CAMA can generate highly consistent and geometric accurate HD Map annotations. Through comprehensive experiments, we verified that such annotations can dramatically improve the accuracy and generalization of perception models in intelligent driving.
- To verify our proposed framework, we apply CAMA to the nuScene dataset and set up new HD Map annotations, namely nuScenes-CAMA. MapTR v2 [23] is used as a benchmark model. Extensive experiments show that models trained on nuScenes-CAMA can produce more consistent and accurate estimations of the static map elements compared with default HD Map.

II. RELATED WORKS

Existing driving datasets make the most effort in data collection, calibration, and manual annotations. However, recent BEV perception algorithms raise the demand for high-accuracy 3D road surface element annotation, which is hard to obtain by manual annotation only. To fulfill the requests for 3D road surface ground truth and BEV perception algorithm training, a pipeline from data collection, calibration, scene reconstruction, and annotation must be built and considered together. To the best of our knowledge, existing publicly available datasets [17], [18], [20], [24]–[26] only meet a portion of such requirements.

HDMapNet [19] is one of the pioneers in predicting road elements using a neural network in BEV spaces. The BEV segmentation results from the networks are further processed into vectorized representation by post-processing. The authors proposed to use the HD Map provided by the nuScenes dataset for training. Following HDMapNet, MapTR [23], [27] further boosts the performance by improving the decoder and loss function modeling. To make a step towards end-to-end road structure understanding, LaneGAP [28], and TopoNet [24] regress the road topology structure directly. For a comprehensive development of the vision-centric BEV perception algorithm, we refer the readers to this survey [4]. All these methods use the same annotations provided by the nuScenes dataset. However, the provided HD Map lacks elevation information. As a result, the reprojection

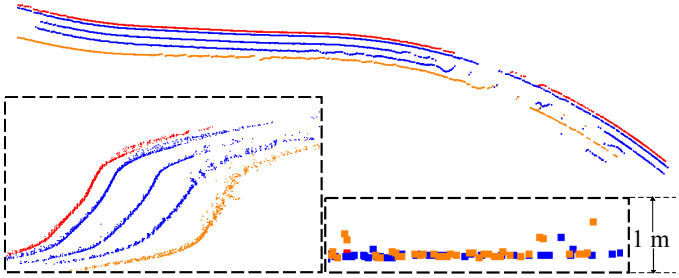


Fig. 3: Sparse 3D lane point clouds from the OpenLane V1 dataset. It uses LiDAR point clouds projection to generate 3D lane annotations. The red, blue, and yellow points refer to road teeth, lane dividers, and solid yellow lane marks, respectively. The side view shows that the elevation noises are around the meter level. The zoom-in view also shows that noises are distributed in all directions.

accuracy between the HD map and images is not guaranteed. Fig. 1 (top) shows the misalignment of the image and reprojected road edge. In practice, the reprojection accuracy is vital for BEV algorithm training. The flaws of the road surface annotation generation pipeline mainly cause such reprojection inconsistency. Notably, the HD map provided by the scenes dataset is annotated in 2D satellite images. The Global Navigation Satellite System (GNSS) and Real-Time Kinematic (RTK) positioning signals maintain the pose alignment between images and the map. Despite high RTK positioning accuracy, the actual camera pose accuracy is not guaranteed due to synchronization and calibration errors [29], [30]. Thus, the nuScenes HD Map cannot guarantee the quality of annotations in consistency and accuracy.

Without HD Map, some works employ LiDAR points for 3D road elements annotation. RSRD [31] deploys LiDAR and stereo cameras to reconstruct the road surface with high accuracy focusing on cracks, bumps and potholes, while our work emphasizes more on road structure reconstruction in large scale. OpenLane [20], [24] and Once-Lane3D [17] propose to combine 2D image segmentation and LiDAR points to generate 3D lane annotation. To do so, the LiDAR points are projected to the image plane to obtain sparse depth. Then, the 2D instance lane segmentation is back-projected to 3D space in camera coordinates from the sparse depth. A filtering algorithm removes the depth noises from inaccurate LiDAR points. Finally, the 3D lane annotations are back-projected into the 3D space using LiDAR points. The advantage is clear: the 2D lane segmentation guarantees geometric accuracy. However, such a method does not impose the spatial-temporal consistency of the 3D lane. The sparse depth could be noisy due to multiple factors, including calibration, synchronization, dynamic objects, and occlusion. Thus, the back-projected 3D lane may not align well with the real one. To verify it, Fig. 3 shows that the 3D lane annotations from LiDAR point projections usually suffer from noises and artefacts. The noises along the elevation direction are around the meter level.

Unlike the methods above, our proposed CAMA pipeline reconstructs the static map element with surround view images. Without the need for LiDAR and predefined HD Map, the annotation results are more consistent and accurate.

III. APPROACHES

Fig. 2 details the pipeline of our proposed CAMA. It mainly consists of scene reconstruction and road element vector annotation. The first part is fully automatic, while the second is addressed semi-automatically based on a human-in-the-loop fashion. Mainly, the offline map auto-annotation model [22] is first employed, followed by verification and modification by human annotators. Our pipeline is a vision-centric approach, as the inputs include surround view images and auxiliary sensors (wheel odometry, GNSS, and IMU). Since CAMA guarantees all 3D elements and their correspondence to 2D images, the 3D-2D correspondence and reprojection accuracy are also insured (even improved) without LiDAR.

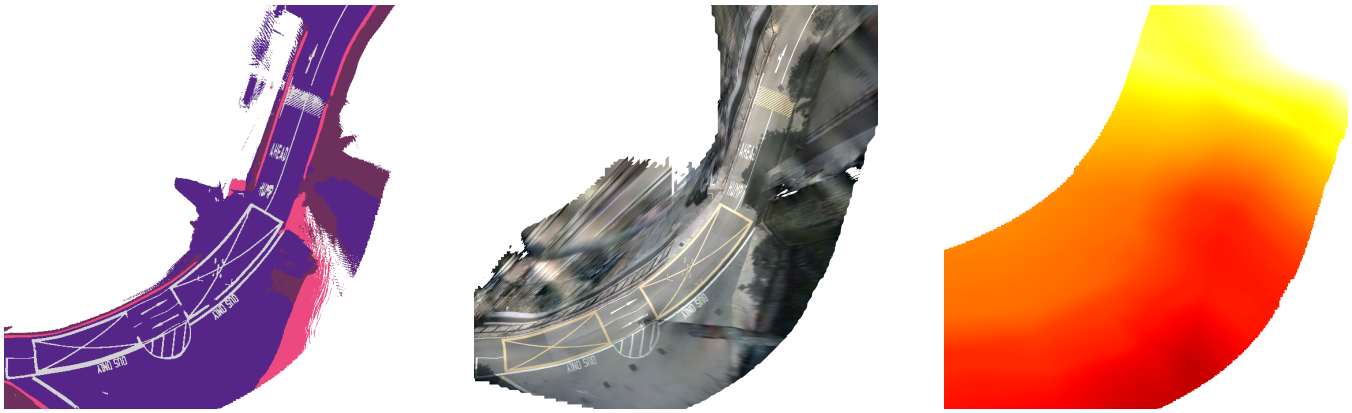
A. Scene Reconstruction

As presented in Fig. 2, our scene reconstruction comprises three parts (see): WIGO, SfM, and RoMe.

1) WIGO: We propose to use a Wheel-IMU-GNSS odometry (WIGO) algorithm to combine sensor reading with pose graph optimization. Following VIWO [32], we combine GNSS signals with pose graph optimization [33], [34] to ensure global localization and accurate scale. The IMU and wheel sensor provide relative pose constraints between consecutive frames. The WIGO algorithm gives a descent global 6-DoF (Degree of Freedom) pose with a real-world scale. The scale information mainly comes from the GNSS observations. The WIGO results are used as inputs of the Structure from Motion (SfM) pipeline illustrated below.

2) SfM: Inspired by COLMAP [35], [36], we introduce an efficient SfM method for whole scene reconstruction. To achieve higher accuracy and efficiency toward self-driving challenges, we optimize COLMAP from five aspects below:

- **Initialization**: The incremental SfM may suffer substantial computation burdens when dealing with large-scale reconstruction (e.g., 300m x 300m areas and thousands of images). In real-world driving scenarios, the pose prior to each image can be easily obtained from GPS and other sensors (e.g., IMU and wheel odometry). Motivated by this, we propose an Odometry Guided Initialization (OGI) for SfM. Specifically, the WIGO pose is transformed into the front camera coordinates with extrinsic. Given the initial poses, the incremental SfM can be replaced with the spatial-guided SfM. The images are initialized with poses from WIGO to conduct triangulation, followed by a series of iterative bundle adjustments (BA).
- **Matching**: The cost of feature matching grows exponentially in traditional exhaustive SfM. Sequential matching may alleviate this issue but bring new problems with matching missing for multiple driving clips. We propose homography-guided spatial pairs (HSP) for the specific driving scenario to balance matching recall and efficiency. Specifically, with the help of WIGO



(a) Semantic map. (b) Photometric map. (c) Elevation.

Fig. 4: Reconstructed HD Map of scene-0828 from nuScenes using our proposed method. (a) Semantic map in BEV, purple, pink, and white correspond to road surface, road teeth, and lane marking, respectively. (b) Photometric map in BEV. (3) Elevation visualization in hotmap, brighter indicates higher.

poses, the potential matched image pairs can be filtered by the visual cone overlap between images. Furthermore, as for a self-driving application, all the cameras have an approximate extrinsic to the ground plane. The visual overlap between images can be further filtered by applying a homography transform with respect to the ground plane to emphasize the importance of the road surface area.

- **Feature point**: To further boost the robustness of our pipeline in extreme illuminance and weather conditions, we train a feature point extraction network, SuperPoint [37], on our dataset and pay extra attention to the road surface.
- **BA**: We modify the bundle adjustment part to balance efficiency and accuracy. An iterative BA strategy is applied with a triangulation points filter. The inaccurate points can be removed from the SfM sparse model during the iterations.
- **Rigid prior**: We employ rigid BA to replace the ordinary BA process in COLMAP. The rationale is that multiple cameras on a vehicle are attached to the vehicle's body and can be regarded as mounting on a rigid body. Applying rigid BA directly not only improves the overall efficiency of the pipeline but also increases robustness.

With the above optimizations, we achieve roughly five times efficiency boost and 20% robustness (success rate) improvements for self-driving datasets (see Section IV for details). The accurate 6-DoF poses and corresponding sparse 3D points generated by SfM models are used as input for RoMe to reconstruct road surface mesh (Section III-A.3). It should be noted that for the nuScenes dataset, we applied our pipeline by clip (or scene as the term by nuScenes) to get annotation results. The difference is that reconstructing by clips may result in a smaller reconstruction area. Although some scenes in the nuScenes dataset have geographical overlap, the proportion is small, so we neglect these factors.

3) RoMe: We extend our previous work RoMe [21] for road surface mesh reconstruction. The extensions can be roughly divided into three parts:

- **Surface points**: An off-the-shelf 2D segmentation network [38] is applied to get road and lane segmentation masks. Combined with the sparse SfM models, semantic sparse point clouds can be recovered. The sparse road surface point clouds are then extracted. To further increase the robustness when the SfM points are too sparse, we also sample additional points based on ego-pose and the extrinsic between the cameras to the ground. This approach improves the overall quality of the road surface initialization.
- **Elevation**: An elevation MLP (Multilayer perceptron) [39] is trained with the sparse road surface (3D point clouds) produced in the previous step.
- **Semantics**: The semantic labels and photometric features are trained by supervising the original images and their corresponding 2D segmentation results.

In practice, the elevation MLP could also be optimized and refined in the third part for better consistency among geometry and photometric features. Ultimately, the highly accurate 3D road surface mesh can be obtained. Note that this 3D road surface mesh and elevation maps could also be represented as 2D BEV images and an elevation image as shown in Fig. 4. Such representation is fed into the next stage: Map Annotation (Section III-B).

B. Map Annotation

We extend VMA (Vectorized Map Annotation System) [22] for initial map annotation. VMA is an offline map auto-annotation framework based on MapTR [27]. The input is the concatenated 2D BEV semantic photometric images, while the output is the vectorized representation of road surface elements (e.g., lane dividers, road teeth, crosswalks, etc.). We propose to use two-layer inputs since they can improve the VMA reasoning ability, especially when classifying the type of lane dividers. Note that this process is still in 2D BEV space. Once we obtain 2D representations, an elevation map is combined to lift the 2D vectors into actual 3D road surfaces and vectors.

Algorithm 1 Semantic Reprojection Error

Input:
All camera poses, T;
All camera intrinsics, K;
2D lane instance segmentation L;
HD Map 3D vectors, M;
Output:
Semantic Reprojection Error, SRE;
for frame i in range(# of images) **do**
transform HD Map from world to camera frame ;
Map in each frame $M^i \leftarrow transform(T_i, M)$;
Crop $M^i \leftarrow crop(M_i, x_{max}, x_{min}, y_{max}, y_{min})$;
Project HD Map $M^i \leftarrow project(K, M_i)$;
Match M^i and L_i with Hungray algorithm;
Matched pairs $P \leftarrow \{M_{ij}, L_{ij}\}$;
for instance j range(# of P) **do**
$L_{ij} \leftarrow skeletonize(L_{ij})$;
for pixels S_k in L_{ij} **do**
$d_k \leftarrow points\ to\ curve\ distance(S_k, M_{ij})$;
end for
$Error\{M_{ij}, L_{ij}\} \leftarrow mean(d_k)$;
end for
$Error_i \leftarrow mean(Error\{M_{ij}, L_{ij}\})$;
end for
$SRE \leftarrow mean(Error_i)$;

TABLE I: Quantitative comparison between the original nuScenes HD Map and our proposed nuScenes-CAMA HD Map. We evaluate the Semantic Reprojection Error (SRE), Precision, and Recall.

	SRE ↓	Precision ↑	Recall ↑	F_1 ↑
nuScenes	8.03	0.59	0.51	0.54
CAMA (Ours)	**4.73**	**0.87**	**0.54**	**0.66**

In practice, the VMA results are used as priors to accelerate the manual annotation process. While the labeled data is accumulated, the manual works are converted from labeling to verification and minor modification. Still, the entire reconstruction and annotation workflow runs automatically.

IV. EXPERIMENTS

The nuScenes dataset is one of the most actively used datasets for online HD Map construction [18]. It has been widely used to validate recently proposed BEV perception algorithms [19], [23], [27], [40]. Other datasets like OpenLaneV2 [20] are also built upon the origin nuScenes dataset. Thus, we validate our proposed framework on the nuScenes dataset to generate optimized annotations.

A. Dataset samples

The nuScenes dataset contains 1000 scenes, each containing 15 seconds of driving data. There are 28130, 6019, and 6008 samples for training, validation, and testing. Our proposed annotation is a subset of the nuScenes dataset, with 14707, 3583, and 3794 samples for training, validation, and testing. Since our method reconstructs each scene independently, the area around each scene's first and last frames can not be fully reconstructed due to a lack of observations from neighboring frames. The head and tail frames are dropped according to the driving distance to guarantee the coverage of our annotations on training and testing sample ranges. To fairly compare our proposed annotation with the original nuScenes dataset, we subsample the nuScenes dataset according to our annotation frames accordingly. We named them nuScenes-sub and nuScenes-CAMA in the experiments.

B. Quantitative Validation

As introduced in Section I, the nuScenes dataset provides HD Maps for all scenes, while the annotation does not contain elevation. Reprojecting an HD Map into the image space using the ego-motion and calibration, as presented in Fig. 1, we can clearly observe the misalignments between the lane vectors and image space. For quantitative analysis, we propose a metric denoted semantic reprojection error (SRE) to analyze better and compare the reprojection accuracy. The detailed steps are illustrated in Algorithm 1: **Step 1**: Reproject the 3D annotation vector elements into each 2D image. **Step 2**: Extract all the instances in image space using an off-the-shelf 2D lane instance segmentation [41] and fit polylines for each instance. **Step 3**: Match the projected elements from Step 1 and the extracted elements from Step 2 using the Hungarian algorithm [42]. **Step 4**: Calculate the mean pixel distance for each matched element.

Table I compares SRE, precision, recall, and F_1 Score between the original nuScenes-sub dataset and our proposed nuScenes-CAMA dataset. We also calculate the precision and recall of the lane divider. Our proposed method achieves 41% lower SRE, indicating that our reconstructed HD Map and ego-pose have better consistency and higher accuracy. The Precision and Recall are improved 0.28 and 0.03 respectively because our proposed method reconstructs the map based on images, inherently ensuring the correspondence between reconstructed maps and actual image observations.

C. Qualitative Validation

We observe that the HD Map provided by nuScenes does not always reflect the actual environments captured by cameras. For example, most HD Map annotations show the lane divider between the ego-vehicle and bicycle lanes. However, observe carefully the camera image in Fig. 1 (a, b): there is no lane marking on the ground in the relevant area. In practice, such inconsistency between the image and HD map annotation increases the training difficulty, particularly the convergence of perception models. An intuitive understanding: there is no lane marking in the image, and the model is likelier to "memorize" the HD Map according to the nearby environment instead of "reasoning" the local map based on the observations. Consequently, such inconsistent training data will weaken the potential generalizability of the perception model. Differently, our CAMA can produce accurate (Fig. 1 d) and consistent (Fig. 1 f) lane marking, including road teeth, lane divider, and ped-crossing.

TABLE II: Comparison of different MapTR model evaluation results on the nuScenes validation dataset. The training and validation use the same kind of annotation for each experiment. We also evaluate the reprojection accuracy (SRE) and consistency (precision, recall and F_1 score) for the prediction results of different models.

Exp. #	annotation	w/ elevation	AP_{ped}	$AP_{divider}$	$AP_{boundary}$	mAP	SRE ↓	Precision ↑	Recall ↑	F_1 ↑
# 1	nuScenes		50.8	45.8	55.4	50.7	8.43	0.51	0.37	0.42
# 2	CAMA		47.2	41.2	36.4	41.6	6.68	0.72	0.52	0.60
# 3	CAMA	✓	46.8	38.7	37.3	40.9	5.81	0.93	0.49	0.64

(a) MapTR (trained with nuScenes) prediction

(b) MapTR (trained with CAMA) prediction

Fig. 5: Comparison of the model prediction reprojection. (a) The result reprojection of MapTR model trained with the nuScenes dataset. The red circle shows a false positive prediction, as there is no lane marking on the ground. (b) The result reprojection of MapTR model trained with the CAMA annotation. The red circle shows better reprojection accuracy compared to (a).

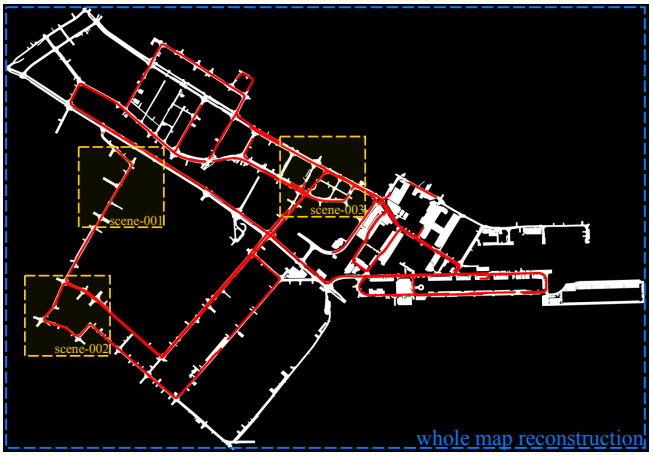


Fig. 6: One of the nuScenes map (boston-seaport) visualization. Currently, the CAMA pipeline is conducted by each scene (yellow boxes). In the future, we will release the annotation by each map as presented in the blue dashed box for the whole image. In this way, all the annotations are consistent across different scenes.

D. Application

Highly accurate and spatial-temporal consistent HD map annotations are vital for BEV perception algorithm training. To verify the effectiveness of our annotations, we choose MapTRv2 as our baseline model. We removed the auxiliary dense prediction loss in MapTRv2 to reduce the influence of irrelevent supervision signals. We conduct three experiments: 1) MapTRv2 trained on nuScenes-sub dataset; this is set as the baseline. 2) MapTRv2 trained on CAMA but without elevation information. 3) MapTR trained on the CAMA annotation with elevation information. All the experiments are trained 24 epochs with ResNet-50 backbone [43]. Table II details model prediction accuracy and the reprojection metrics, including SRE and F_1 score. We can clearly find that models trained with CAMA annotation predict better reprojection accuracy and consistency. The SRE improves from 8.43 to 5.81 by 31 %, and the F_1 score improves from 0.42 to 0.64. We also visually compare the predictions in Fig. 5. Due to the inconsistent annotation in the nuScenes dataset, the trained MapTR makes false positive predictions (red circle in Fig. 5 a). With our proposed CAMA annotation, the model predictions align better with the image in reprojection (red circle in Fig. 5 b).

V. CONCLUSION

We present CAMA: a vision-centric approach for consistent and accurate static map element annotations. We investigate the critical factors for online HD Map construction and argue that the annotation quality in reprojection accuracy and spatial-temporal consistency is vital for perception algorithm training. Based on this insight, we propose a new baseline for the BEV perception algorithm. Currently, the annotation is a subset of the original nuScenes dataset because we only generate CAMA results for each scene. Fig. 6 draws the relation between the scenes (yellow boxes) and map (blue box) of the nuScenes dataset. In future works, we will release the CAMA results by map (blue box).

We choose a decoupled method for reconstructing the physical and logical layers. Initially, it reconstructs the road surface in mesh. Subsequently, it extracts a vectorized representation of lanes in a data-driven method. In contrast, methods like RoadMap [44] advocate reconstructing both layers simultaneously. The logical layers are highly complex, with numerous long-tail corner cases, which, we argue, can be more effectively addressed through data-driven approaches. Considering the swift progress of foundation models within computer vision, we anticipate the future development of fully integrated end-to-end reconstruction techniques.

REFERENCES

[1] Q. Rao and J. Frtunikj, "Deep learning for self-driving cars: Chances and challenges," in *International Workshop on Software Engineering for AI in Autonomous Systems*, 2018, pp. 35–38.

[2] A. Gupta, A. Anpalagan, L. Guan, and A. S. Khwaja, "Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues," *Array*, vol. 10, p. 100057, 2021.

[3] J. Zhou and J. Beyerer, "Corner cases in data-driven automated driving: Definitions, properties and solutions," in *IEEE Intelligent Vehicles Symposium*, 2023, pp. 1–8.

[4] Y. Ma, T. Wang, X. Bai, H. Yang, Y. Hou, Y. Wang, Y. Qiao, R. Yang, D. Manocha, and X. Zhu, "Vision-centric bev perception: A survey," *arXiv preprint arXiv:2208.02797*, 2022.

[5] N. Garnett, R. Cohen, T. Pe'er, R. Lahav, and D. Levi, "3d-lanenet: end-to-end 3d multiple lane detection," in *IEEE International Conference on Computer Vision*, 2019, pp. 2921–2930.

[6] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *European Conference on Computer Vision*, 2020, pp. 194–210.

[7] L. Reiher, B. Lampe, and L. Eckstein, "A sim2real deep learning approach for the transformation of images from multiple vehicle-mounted cameras to a semantically segmented image in bird's eye view," in *International Conference on Intelligent Transportation Systems*, 2020, pp. 1–7.

[8] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *European conference on computer vision*. Springer, 2022, pp. 1–18.

[9] Y. Liu, T. Wang, X. Zhang, and J. Sun, "Petr: Position embedding transformation for multi-view 3d object detection," in *European Conference on Computer Vision*, 2022, pp. 531–548.

[10] S. Chen, T. Cheng, X. Wang, W. Meng, Q. Zhang, and W. Liu, "Efficient and robust 2d-to-bev representation learning via geometry-guided kernel transformer," *arXiv preprint arXiv:2206.04584*, 2022.

[11] J. Huang and G. Huang, "Bevdet4d: Exploit temporal cues in multi-camera 3d object detection," *arXiv:2203.17054*, 2022.

[12] Y. Li, H. Bao, Z. Ge, J. Yang, J. Sun, and Z. Li, "Bevstereo: Enhancing depth estimation in multi-view 3d object detection with temporal stereo," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 2, 2023, pp. 1486–1494.

[13] J. Huang and G. Huang, "Bevpoolv2: A cutting-edge implementation of bevdet toward deployment," *arXiv:2211.17111*, 2022.

[14] T. Wang, Q. Lian, C. Zhu, X. Zhu, and W. Zhang, "Mv-fcos3d++: Multi-view camera-only 4d object detection with pretrained monocular backbones," *arXiv preprint arXiv:2207.12716*, 2022.

[15] S. Diamond, V. Sitzmann, F. Julca-Aguilar, S. Boyd, G. Wetzstein, and F. Heide, "Dirty pixels: Towards end-to-end image processing and perception," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 3, pp. 1–15, 2021.

[16] B. Zhou and P. Krähenbühl, "Cross-view transformers for real-time map-view semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 760–13 769.

[17] F. Yan, M. Nie, X. Cai, J. Han, H. Xu, Z. Yang, C. Ye, Y. Fu, M. B. Mi, and L. Zhang, "Once-3dlanes: Building monocular 3d lane detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 143–17 152.

[18] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 621–11 631.

[19] Q. Li, Y. Wang, Y. Wang, and H. Zhao, "Hdmapnet: An online hd map construction and evaluation framework," in *International Conference on Robotics and Automation*, 2022, pp. 4628–4634.

[20] L. Chen, C. Sima, Y. Li, Z. Zheng, J. Xu, X. Geng, H. Li, C. He, J. Shi, Y. Qiao *et al.*, "Persformer: 3d lane detection via perspective transformer and the openlane benchmark," in *European Conference on Computer Vision*. Springer, 2022, pp. 550–567.

[21] R. Mei, W. Sui, J. Zhang, Q. Zhang, T. Peng, and C. Yang, "Rome: Towards large scale road surface reconstruction via mesh representation," *arXiv:2306.11368*, 2023.

[22] S. Chen, Y. Zhang, B. Liao, J. Xie, T. Cheng, W. Sui, Q. Zhang, C. Huang, W. Liu, and X. Wang, "Vma: Divide-and-conquer vectorized map annotation system for large-scale driving scene," *arXiv preprint arXiv:2304.09807*, 2023.

[23] B. Liao, S. Chen, Y. Zhang, B. Jiang, Q. Zhang, W. Liu, C. Huang, and X. Wang, "Maptrv2: An end-to-end framework for online vectorized hd map construction," *arXiv preprint arXiv:2308.05736*, 2023.

[24] T. Li, L. Chen, X. Geng, H. Wang, Y. Li, Z. Liu, S. Jiang, Y. Wang, H. Xu, C. Xu *et al.*, "Topology reasoning for driving scenes," *arXiv preprint arXiv:2304.05277*, 2023.

[25] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan *et al.*, "Argoverse: 3d tracking and forecasting with rich maps," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8748–8757.

[26] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes *et al.*, "Argoverse 2: Next generation datasets for self-driving perception and forecasting," *arXiv preprint arXiv:2301.00493*, 2023.

[27] B. Liao, S. Chen, X. Wang, T. Cheng, Q. Zhang, W. Liu, and C. Huang, "Maptr: Structured modeling and learning for online vectorized hd map construction," *arXiv preprint arXiv:2208.14437*, 2022.

[28] B. Liao, S. Chen, B. Jiang, T. Cheng, Q. Zhang, W. Liu, C. Huang, and X. Wang, "Lane graph as path: Continuity-preserving path-wise modeling for online lane graph construction," *arXiv preprint arXiv:2303.08815*, 2023.

[29] J. Zhang, W. Sui, X. Wang, W. Meng, H. Zhu, and Q. Zhang, "Deep online correction for monocular visual odometry," in *IEEE International Conference on Robotics and Automation*. IEEE, 2021, pp. 14 396–14 402.

[30] J. Zhang, W. Sui, Q. Zhang, T. Chen, and C. Yang, "Towards accurate ground plane normal estimation from ego-motion," *Sensors*, vol. 22, no. 23, p. 9375, 2022.

[31] T. Zhao, C. Xu, M. Ding, M. Tomizuka, W. Zhan, and Y. Wei, "Rsrd: A road surface reconstruction dataset and benchmark for safe and comfortable autonomous driving," *arXiv preprint arXiv:2310.02262*, 2023.

[32] W. Lee, K. Eckenhoff, Y. Yang, P. Geneva, and G. Huang, "Visual-inertial-wheel odometry with online calibration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2020, pp. 4559–4566.

[33] F. Dellaert and G. Contributors, "borglab/gtsam," May 2022. [Online]. Available: https://github.com/borglab/gtsam)

[34] F. Dellaert and M. Kaess, *Factor Graphs for Robot Perception*. Foundations and Trends in Robotics, Vol. 6, 2017. [Online]. Available: http://www.cs.cmu.edu/~kaess/pub/Dellaert17fnt.pdf

[35] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4104–4113.

[36] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision*, 2016, pp. 501–518.

[37] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 224–236.

[38] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1290–1299.

[39] M. W. Gardner and S. Dorling, "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," *Atmospheric environment*, vol. 32, no. 14-15, pp. 2627–2636, 1998.

[40] Y. Liu, T. Yuan, Y. Wang, Y. Wang, and H. Zhao, "Vectormapnet: End-to-end vectorized hd map learning," in *International Conference on Machine Learning*. PMLR, 2023, pp. 22 352–22 369.

[41] J. Jain, J. Li, M. T. Chiu, A. Hassani, N. Orlov, and H. Shi, "Oneformer: One transformer to rule universal image segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2023, pp. 2989–2998.

[42] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[44] T. Qin, Y. Zheng, T. Chen, Y. Chen, and Q. Su, "A light-weight semantic map for visual localization towards autonomous driving," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 11 248–11 254.