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

From Satellite to Ground: Satellite Assisted Visual Localization with Cross-view Semantic Matching

Xiyue Guo[1], Haocheng Peng[1], Junjie Hu[2], Hujun Bao[1] and Guofeng Zhang[1†]
[1]State Key Lab of CAD&CG, Zhejiang University [2]Chinese University of Hong Kong, Shenzhen

***Abstract*— One of the key challenges of visual Simultaneous Localization and Mapping (SLAM) in large-scale environments is how to effectively use global localization to correct the cumulative errors from long-term tracking. This challenge presents itself in two main aspects: first, the difficulty for robots in revisiting previous locations to perform loop closure, and second, the considerable memory resources required to maintain point-cloud-based global maps. Recent solutions have resorted into neural networks, using satellite images as the references for ground-level localization. However, most of these methods merely provide cross-view patch-matching results, which leads to unfeasible in integration with the SLAM system. To address these issues, we present a semantic-based cross-view localization method. This approach combines semantic information with a reward and penalty mechanism, enabling us to obtain a global probability map and achieve precise 3-degree-of-freedom (3-DoF) localization. Based on that, we develop a SLAM system that capitalizes on satellite imagery for global localization. This strategy effectively bridges the gap between SLAM and real-world coordinates while also substantially reducing accumulated errors. Our experimental results demonstrate that our global localization method significantly outperforms existing satellite-based systems. Moreover, in scenarios where the robot struggles to find loop closures, employing our localization method improves the SLAM accuracy.**

I. INTRODUCTION

Global localization stands as a core component within visual Simultaneous Localization and Mapping (SLAM) and Structure from Motion (SfM) systems, containing important significance [1]–[3]. Its primary role revolves around solving the critical question of a robot's spatial awareness – "Where am I?" Specifically, it serves to rectify the accumulated errors arising from the visual tracking. Furthermore, it facilitates relocalization in situations where tracking is lost, while aligning the robot's position within a comprehensive, wide-ranging map coordinate system.

However, mainstream methods for vision-based global localization, relying on image retrieval and feature point matching, occupy a large amount of memory resources [4]–[9]. This limitation becomes particularly pronounced when tackling SLAM or SfM tasks on a grand scale environment, such as city-scale mapping. Furthermore, in real-world applications, there is no guarantee that the robot will encounter overlapping areas during the localization task, which makes these methods less generalizable.

To address these challenges, some approaches have incorporated satellite imagery and proposed cross-view localization techniques [10]–[14]. These approaches leverage satellite images to perform global localization of ground images. Compared with traditional methods, a single low-resolution satellite image necessitates much fewer memory resources, even in a city-scale task. This advantage greatly benefits robots to store the maps offline in advance when performing localization tasks. Nonetheless, most of these methods only simply match ground images with possible satellite patches; their localization accuracy greatly depends on the sampling density of the satellite image. Also, these methods can't get the robot's orientation information. Due to these limitations, the direct applicability of these methods to robot tasks remains unfeasible.

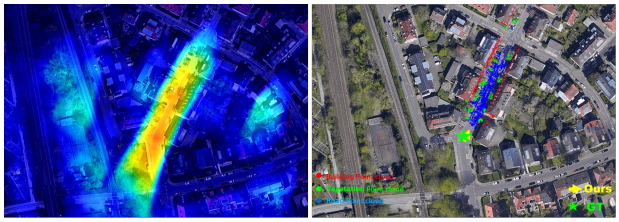


Fig. 1. A visualization of our cross-view localization result. The left image is the visualized probability map generated from the proposed semantic matching, while the right image is the result of pose estimation and point cloud projection. The Aerial image is captured from Google Map.

Seeking to enhance pose estimation accuracy, several methods have turned to template matching, which involves the comparison of bird's-eye view (BEV) sensor observations with 2D intensity maps [15], [16]. By fusing the matching results with raw GPS data, they can achieve extremely high precision. However, these approaches require expensive sensors such as LIDAR or RADAR, and necessitate a complex, costly supervision process.

In this context, our research builds on the existing template-matching framework to introduce a semantic-based cross-view localization strategy. Our approach relies on the integration of 3D point clouds with semantic data to generate the requisite BEV maps for template matching. Leveraging semantic information bears two primary advantages: it is easy and cost-efficient for extraction, and it exhibits a robust viewpoint invariance.

Our approach aims to replicate the behavior of humans consulting maps, allowing the robots to use satellite maps for localization by comparing ground-level semantic distributions with those outlined in satellite semantics. This method not only detects the optimal current location but also determines the correct orientation.

By using our novel reward-penalty mechanism, we can

†Corresponding author. Email: zhangguofeng@zju.edu.cn
This work was partially supported by the Key RD Program of Zhejiang Province (No. 2023C01039).

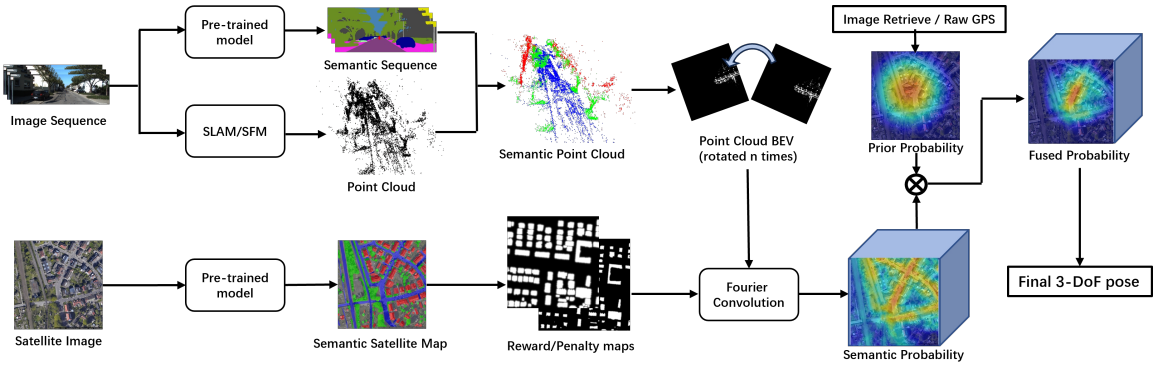


Fig. 2. The detailed framework of our cross-view localization. We start by using pre-trained networks to segment both the satellite and ground images. We blend labels from the ground image with point clouds created through SLAM/SfM to make a bird's-eye-view map. At the same time, we establish reward and penalty maps based on satellite labels. Following that, we align the bird's-eye-view map with the positive/negative maps. We then create a map that shows the likelihood of different poses based on matching scores. By combining this map with raw GPS data, we can determine the final 3-DoF pose.

further enhance the semantic matching accuracy. This mechanism scores the different poses according to the semantic projection quality, and then forms matching probabilistic maps from these scores. Fig. 1 shows an example of our cross-view localization result.

We test our approach within public datasets in the autonomous driving field [17], [18]. Apart from the standard localization evaluation, to ascertain the utility of our results within a SLAM system, we develop a cross-view SLAM system grounded in our localization methodology. Following the consistency strategy, we filter out the inlier localization results. Leveraging these results, we can compute the transformation relationship between SLAM and the real world, continuously refining the SLAM pose accordingly.

In summary, our contributions are threefold:

- We introduce a novel cross-view localization framework that leverages satellite and ground view semantic registration to generate a probability map and derive a pose, establishing a new state-of-the-art.
- We propose a novel reward-penalty strategy for semantic matching, offering a more precise evaluation of the semantic matching quality, consequently enhancing localization precision.
- We develop the visual-based cross-view SLAM system. The system can autonomously correct its pose using satellite images, even in the absence of loop closures.

II. RELATED WORK

A. Localization with Feature Matching

Traditional global localization methods typically consist of two main steps. The first step involves place recognition, where the current frame's coarse position on the global map is determined through the image retrieval method, finding keyframes with high feature similarity to the current frame. The second step entails pose estimation, utilizing feature matching techniques to align the current frame's feature points with 3D map points observed in associated frames [7], [8]. The 6-DOF localization information of the current frame on the map is then solved based on the geometric relationships between the matched points [4]–[6].

These methods have been proven to have great accuracy and robustness. However, they also suffer from several limitations. Firstly, they heavily rely on large-scale computer storage resources due to the inclusion of feature descriptor information in the global 3D point cloud. Secondly, they necessitate significant overlaps between the camera and map acquisition during historical frames to trigger localization. In real-world localization tasks, such overlaps are infrequent, and the storage strategy of 3D point clouds restricts the feasibility of densely acquiring maps. While certain neural network-based scene recognition and feature matching techniques may alleviate the requirement for extensive overlap in global localization [19]–[25], they fail to address the core issue fundamentally. These shortcomings become more pronounced in large-scale environments.

B. Cross-view Image Retrieve

Recently, the localization between ground-level and overhead (satellite or UAV) imagery has garnered increasing attention. These methods demonstrate clear advantages in terms of storage resource usage compared to traditional point-cloud-based approaches. Specific studies, [1], [2] achieve localization between ground and UAV imagery utilizing graph matching techniques. [10]–[12] employed Siamese architectures and triplet loss to generate global features for both ground and satellite images. The ground image's location is obtained by querying the most similar satellite images. However, these methods solely focus on image retrieval tasks, providing only coarse location information without orientation. Furthermore, the localization accuracy heavily depends on the sampling density of the satellite image, as low sampling density can lead to significant drift between the retrieved location and the actual pose, while high sampling density entails considerable storage space.

C. Cross-view Pose Estimation

To address the need for accurate pose estimation, [26], [27] undertakes the task of projecting satellite features into the ground view, deriving a 3-DoF pose through an optimization module. However, this approach relies on appropriate initial input; the initial pose should closely approximate the correct one, or there's a risk of failure. Furthermore, the necessity for

ground truth pose supervision, which demands precise sensor data acquisition, renders this approach somewhat impractical for real-world robotic operations.

A solution to the local optimum predicament is presented in CCVPE [28], which employs two decoders to individually generate orientation and translation distributions. While [15], [16], [29] have also shown promise in tackling the local optimum issue through the utilization of template matching between BEV derived from sensor observations and 2D global maps. Although the template matching approaches offer a better resolution to the local optimum problem, these approaches need to undergo an intensive and expensive supervision process for their networks. Moreover, some of them [15], [16] still require the deployment of costly sensors.

In light of the above methods, our paper decides to base its framework on [15], [16], [30], given their diminished reliance on prior pose. However, we innovate by discarding supervised networks and LIDAR scans in favor of the semantic point cloud, which is generated from a short series of images to formulate the BEV maps for template matching.

III. Methodology

In this section, we elucidate our localization methodology. We initiate the process by inputting a sequence of frames along with their raw GPS data. Subsequently, we extract the corresponding satellite image and utilize distinct pre-trained segmentation networks for semantic segmentation of both ground and satellite images. The fusion of semantic labels from the ground view with SLAM/SfM-generated point clouds enables the generation of the semantic point cloud map in bird-eye-view. Simultaneously, we establish reward and penalty maps based on satellite semantic information.

Next, we proceed to match the BEV with the reward and penalty maps. Based on the scores projected from all poses, we subsequently generate a probability map. This map is then fused with GPS prior probability to derive the final 3-DoF pose. To better illustrate our overall framework, we provide a visual representation in Fig. 2.

Supplementing this, we further explain the integration of our proposed localization methodology on SLAM, aiming to provide the real-world coordinates for SLAM system, and refining its pose.

A. Satellite View Semantic Map Generation

Given the satellite image of size $W \times H$, we first segment it and generate each label's binary map by using a pre-trained model. We specifically choose three categories, namely road, building, and vegetation, as they are easily observable in both ground and satellite views. To efficiently measure the quality of different poses during subsequent matching processes, we adopt a probabilistic approach [31], and develop reward and penalty maps based on satellite semantic information.

To achieve a true semantic projection, we generate the reward map as follows:

$$MR_l[u_s, v_s] = e^{-\frac{1}{2\sigma} dis_l[u_s, v_s]}, \tag{1}$$

where the $MR_l[u_s, v_s]$ represents the pixel position of the label l's reward map, and $dis_l[u_s, v_s]$ represents the distance between the current pixel and the nearest region with label l. Additionally, σ is the Gaussian coefficient.

Furthermore, we create a penalty map to prevent wrong projections:

$$MP_l[u_s, v_s] = -e^{-\frac{1}{2\sigma} dis_{nl}[u_s, v_s]}, \tag{2}$$

where the MP_l denotes the penalty map of label l, and $dis_{nl}[u_s, v_s]$ signifies the distance between the current pixel and the nearest region not label l.

B. Ground View Semantic Map Generation

To generate the BEV point cloud map from ground-level frames, our method involves a series of steps. Firstly, we segment the ground-level images by using the same labels as those used in the satellite part. Simultaneously, we extract the 3D point clouds from SLAM/SfM, setting the camera coordinate of the initial frame as our ground coordinate. In the case of a monocular input sequence, we determine the scale of the point cloud by aligning the SLAM/SfM pose with raw GPS data. Following this, we fuse semantic labels with the point cloud, resulting in the creation of the semantic point cloud.

Next, we perform a projection of the point cloud from the ground coordinate into the pixel coordinate system of the bird's-eye-view. The projection equation is as follows:

$$[u, v] = [\frac{z}{\alpha} + \frac{w_b}{2}, \frac{x}{\alpha} + \frac{h_b}{2}], \tag{3}$$

where the $[x, z]$ is the point in ground coordinate (x, y, z), we assume the robot essentially navigates in a planar motion, and only x and z will be considered in the top-down view. The $[u, v]$ is the point in the BEV pixel coordinate. α represents the real distance in a meter of every pixel, and w_b and h_b are the width and height of the BEV image, respectively. This projection allows us to effectively represent the scene from a top-down perspective, which is instrumental in our localization process.

Similar to the satellite maps, our BEV maps are separated by labels. The values of each map depend on the projection of points with the same labels. To be specific, when a point with a certain label is projected, then the corresponding pixel of the certain label's map is accumulated.

C. Cross-view Semantic Matching

Upon obtaining the satellite and BEV maps, we proceed to wrap the BEV maps into the reward and penalty maps. Subsequently, we compute cross-correlations individually as follows:

$$Reward = \sum_l \sigma_l c(\pi(BEV_l, \mathbf{p}), MR_l), \tag{4}$$

$$Penalty = \sum_l \sigma'_l \sum_{l'} c(\pi(BEV_{l'}, \mathbf{p}), MP_l), \tag{5}$$

where $\mathbf{p}$ is the 3-DoF estimated pose includes (u, v, θ), while π represents the 2D projection function. c denotes

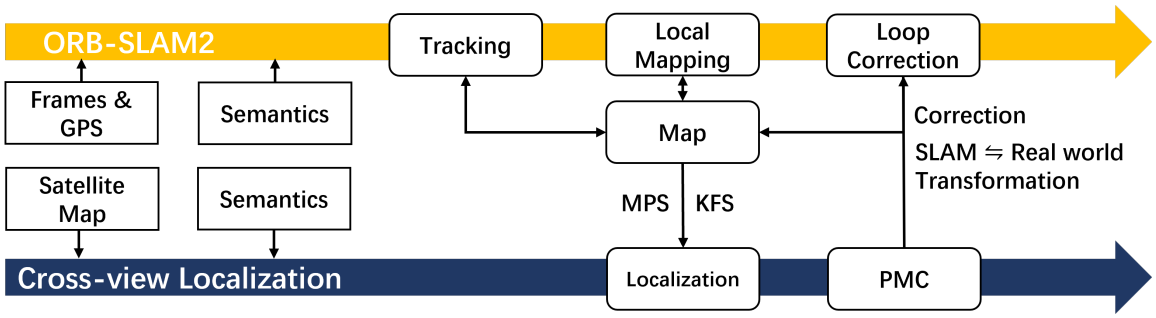


Fig. 3. Our SLAM framework. The cross-view localization module receives the keyframe pose and its observable map points from ORB-SLAM. It then generates and sends back the coordinate transformation and corrected keyframe pose.

the cross-correlations function, and BEV_l refers to the BEV map corresponding to label l. σ_l and σ'_l are the coefficients controlling the different labels' weighting for reward and penalty, respectively.

Basically, the reward function adds scores when projections align with regions of the same label, while the penalty function deducts scores when projections fall into areas with differing labels. Both reward and penalty have the size of $W \times H \times D$, where D is the sampling number of orientation. Moreover, in order to reduce the computational time, the correlation process is taken in Fourier domain.

Following this, using the concluding scores, we formulate the matching probability map:

$$P_{match} \propto (Reward + \gamma Penalty), \tag{6}$$

where γ denotes the coefficient of $Penalty$. In our case, we set it to 0.5.

This matching probability effectively measures the likelihood of a semantic match between the ground-level frames and the satellite image.

Subsequently, the matching probability is integrated with the GPS probability, which is approximated by a Gaussian distribution:

$$P_{GPS} \propto e^{-\frac{(u-gu)^2+(v-gv)^2}{2\sigma_g^2}}. \tag{7}$$

In this equation, $[gu,\ gv]$ is the raw GPS data in pixel coordinate, σ_g denotes the Gaussian coefficient. The final $W \times H \times D$ probability P_{final} is then generated as:

$$P_{final} = P_{GPS} P_{match}. \tag{8}$$

The final 3-DoF result refers to the pose with the highest probability, which means the index of the peak value in P_{final}.

D. SLAM Application

Based on our cross-view localization approach, we integrate it with ORB-SLAM2 [32] to develop a real-time cross-view SLAM system, as depicted in the system architecture shown in Fig. 3. During the operation of the SLAM, we continuously extract 3D feature points from each keyframe. After processing them into BEV maps, we perform periodic localization with the satellite images every few seconds.

Upon completion of cross-view localization, we convert the results from pixel coordinates to real-world coordinates, labeled as T_{wi}, using the geographical data from the satellite imagery. Simultaneously, we obtain the SLAM pose results $T_{si}(3D)$ for the respective keyframe and represent them as a 2D pose — including X, Z, and pitch variables, symbolized as $T_{si}(2D)$. By employing both coordinate systems' poses at the identical time point, we can establish the present coordinate transformation relationship, expressed as T_{ws}(i).

Following the PCM strategy [33], we create a dynamic graph that is constantly updated with new localization findings (T_{ws}(i)). The most substantial maximum clique of the graph is identified to eliminate outlier data. During the graph construction, we calculate a composite distance that encompasses both translation and orientation elements for every pair of localization results; this helps in determining the existence of an edge between them.

Once the PCM graph is initialized, we regard the most adjacent inliers $T_{ws}(k)$ in the initial maximum clique as the fixed transformation matrix between the real world and the SLAM coordinate systems. By using this transformation matrix, the real-time pose in the real-world coordinate can be deduced from the SLAM pose (post 2D projection).

On the other hand, whenever a new localization result is determined as an inlier, we can convert it into the pose in SLAM coordinate system $T'_{si}(2D)$. Since our global localization yields a 3-DoF pose, and the other 3-DoF aren't part of our correction, we incorporate this part data (y, roll, and yaw) from original SLAM pose into our result to generate a 6-DoF pose $T'_{si}(3D)$. Finally, we employ this pose result to rectify the original pose of SLAM.

IV. Experiment

A. Localization Experiment

1) Dataset and implementation: The KITTI dataset is widely used in the autonomous driving field. Since our approach mainly serves tasks like SLAM/SfM, we select the KITTI odometry subset for our experiments, specifically KITTI 00 - 10 (with the exception of 03, as the corresponding dataset could not be located). In addition to the ground images provided by KITTI, we also prepared satellite images that correspond to ground images [26].

For the satellite images, we use the UperNet framework [34], selecting dozens of images as the training set for semantic segmentation. For ground images, we employ a pre-trained model from SDCNet [35] on SemanticKITTI [36] for segmentation. Moreover, using COLMAP [37], we undertake

TABLE I

LOCALIZATION ACCURACY OF DIFFERENT METHODS ON KITTI DATASET

	Latitude %			Lontitude %			Translation %			Oritation%		
	1m	3m	5m	1m	3m	5m	1m	3m	5m	1°	3°	5°
±20m with ±10° prior												
LM [26]	29.15	62.32	72.51	8.42	15.65	26.16	1.39	9.06	18.37	17.28	48.42	69.73
CCVPE [28]	40.39	78.92	88.28	11.08	28.07	39.89	4.28	23.66	37.67	32.68	72.82	84.44
OrienterNet [30]	**41.97**	**79.59**	**89.75**	29.70	49.31	53.18	9.46	35.11	57.40	34.44	59.09	77.57
Ours-full	35.42	77.89	85.61	**45.50**	**87.23**	**93.95**	**14.53**	**68.72**	**82.45**	**67.40**	**89.44**	**93.91**
Ours-wo-r/p	33.15	72.62	81.42	40.71	81.99	90.85	12.74	61.28	76.99	55.73	84.40	91.20
±40m with ±10° prior												
LM [26]	9.98	27.13	37.08	2.49	7.49	12.43	0.51	2.68	5.66	9.92	29.70	49.76
CCVPE [28]	30.29	63.27	75.02	7.48	19.54	28.19	2.74	15.83	25.49	28.26	66.82	80.76
OrienterNet [30]	**34.34**	69.34	**83.77**	17.79	41.94	52.55	6.99	32.67	46.30	23.30	56.77	75.01
Ours-full	32.95	**73.70**	80.48	**44.05**	**84.35**	**90.44**	**13.58**	**65.19**	**77.61**	**66.27**	**78.57**	**92.22**
±20m without orientation prior												
LM [26]	5.04	15.23	25.45	4.91	14.81	24.88	0.20	1.74	5.01	0.58	1.66	2.62
CCVPE [28]	11.44	29.03	42.06	11.75	29.03	41.86	2.00	11.23	21.34	1.71	4.97	8.02
OrienterNet [30]	**34.47**	67.93	79.03	17.35	41.75	53.34	6.43	30.80	45.96	15.12	38.70	52.59
Ours-full	31.83	**71.25**	**82.87**	**45.71**	**84.74**	**91.57**	**14.97**	**61.67**	**78.80**	**52.95**	**84.14**	**88.68**

sparse reconstruction of six-second ground-level images, and with GPS data, we align the scale of the point cloud.

We benchmark our approach against three source-available works for comparative analysis [26] [28] [30]. Since these works are rooted in supervised learning, we train them using the 2011-09-26 subset of KITTI raw data. In the comparative experiments, following these works' setting, we randomly sample the initial translation and orientation at ±20 meters and ±10 degrees of GT. In addition, to analyze the robustness of each method to different prior settings, we introduce additional tests that either double the initial translation shift or eliminate the orientation prior. We also test our method without the reward-penalty approach, to evaluate the effect of this strategy. For performance measurement, we report the success rate for the translation within 1, 3, and 5 meters and orientation within 1, 3, and 5 degrees. To assist analysis, we also present the success rate in latitude (perpendicular) and longitude (parallel) directions.

Moreover, beyond the comparative experiment on KITTI dataset, we also extract a test set from the CityScapes dataset to evaluate the generalization capabilities of various methods except OrienterNet, it is not available for CityScapes. During the evaluation, we use models (including the semantic segmentation model) trained on the KITTI and assess their performance on the CityScapes test set. The localization is executed on an RTX 3090 Ti GPU, with the convolution runtime approximately 40ms.

2) Results on KITTI Dataset: The results of our precision comparison experiment are displayed in Table I. These results demonstrate that our approach significantly outperforms the baseline methods in both translation and orientation aspects. When a standard prior (± 20m and ± 10°) is incorporated in the tests, the success rates of achieving a localization error less than 5m for LM, CCVPE, OrienterNet, and our method are 18.67%, 37.67%, 57.40% and 82.54%, respectively. Moreover, the success rates of obtaining an angular error less than 5° are recorded as 69.73%, 84.44%, 77.57%, and 93.91% for each respective method.

Besides, the comparison of results under various prior settings indicates the superior robustness of our method.

TABLE II

LOCALIZATION ACCURACY ON CITYSCAPE DATASET

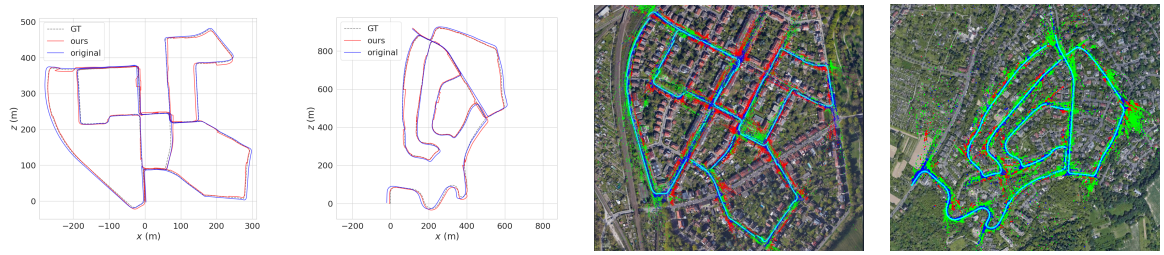
	Translation %			
	1m	3m	5m	8m
LM [26]	0	1.33	4.67	14.33
CCVPE [28]	**1.16**	11.50	21.00	30.50
Ours	1.11	**15.19**	**39.81**	**73.52**

Upon increasing the translation shift to ±40m, our method retains a high success rate, with a translation error less than 5m standing at 77.61%. In comparison, the LM , CCVPE, and OrienterNet show a considerable decline, with success rates dropping to 5.66%, 25.49%, and 46.30%, respectively. Furthermore, when the orientation prior is removed, our system continues to have high accuracy with a success rate of 78.8%, while LM, CCVPE, and OrienterNet falter to 5.01%, 21.31%, and 45.96%, respectively.

Several factors contribute to this enhanced accuracy: First, our method integrates sequences as the ground-level input, allowing for an enriched cross-view match with more data. Second, current segmentation models can precisely delineate semantic labels for large objects on satellite images. Simultaneously, the point cloud reconstructed using camera projection relationships accurately depicts the semantic spatial distribution observed on the ground. The combination of both ensures precise localization results. Third, our reward-penalty strategy is adept at allocating higher probabilities for correct match results, as shown in Table I, this strategy can significantly improve the accuracy.

Additionally, when we analyze the accuracy of latitude and longitude estimations, we find that the baseline methods, especially OrienterNet, are very accurate in estimating latitude. However, their accuracy in estimating longitude is significantly poorer. On the other hand, our method, leveraging semantic matching, offers not only localization but also aligns with the road, thereby yielding excellent results in the longitude direction as well.

3) Results on CityScapes Dataset: Since the CityScapes dataset doesn't provide accurate orientation information, we only compute the translation error without any orientation prior in the generalization test. Additionally, due to the low



(a) The trajectory of KITI 00 (b) The trajectory of KITTI 02 (c) Mapping result of KITTI 00 (d) Mapping result of KITTI 02

Fig. 4. (a) and (b) are trajectories of SLAM results of KITTI 00 and 02, respectively. Each figure has three trajectories: ground truth (GT), uncorrected trajectory, and our trajectory. (c) and (d) are mapping results on satellite images of KITTI 00 and 02.

accuracy of the GPS data from CityScapes and its slower update rate compared to image refresh rate, we introduce an additional eight-meter precision tier to better reflect the localization success rate of various methods.

The results of the generalization experiment are shown in Table II. Clearly, even across datasets, our method demonstrates significantly higher accuracy than other approaches. This can be attributed to the superior generalization capabilities of semantic models compared to models trained on GT poses. From a cost perspective, supervised learning methods dependent on GT pose require high-precision instruments to obtain GT pose for training. In contrast, our approach necessitates training a semantic segmentation model, but obtaining semantic labels is much easier and more cost-effective. Considering these factors, our method proves to be more versatile for robotics tasks.

B. SLAM Experiment

To assess the performance of our cross-view SLAM system, we conduct SLAM tests on two long-distance sequences: KITTI 00, 02. To simulate scenarios where loop closure for the robot is not possible, we turned off the loop closure module in ORB-SLAM. Throughout our experiment, we use stereo images and their corresponding raw GPS data as input and leveraged our integrated SLAM system to output the localization and mapping results with latitude and longitude information.

During the evaluation, we not only compare the final results of our system with the uncorrected results from ORB-SLAM, but also compare them with the results that is corrected by two baseline methods in previous section: LM [26] and CCVPE [28]. In this comparison, we substitute the cross-view localization module with these two methods, keeping other parts like the PCM module and localization correction module unchanged. In this section, we mainly show the absolute pose error (APE) of the estimation results from SLAM compared to the ground truth, the comparison graph between the generated trajectories and the ground truth trajectories, as well as the mapping results of our SLAM system on satellite images.

1) Experimental Result: The APE results for SLAM are presented in the Table III. Firstly, by contrasting the results from different cross-view localization methods, thanks to our significantly higher localization accuracy, only our approach brings positive corrections to SLAM's localization results.

TABLE III
ATE OF DIFFERENT METHODS IN SLAM TASK (METER)

	KITTI00	KITTI02
Original	4.51	9.39
LM [26]	7.83	18.60
CCVPE [28]	6.84	18.71
Ours	**4.30**	**8.05**

Next, comparing our method with uncorrected results, our method shows its capability of accuracy improvement. However, due to the already high accuracy of ORB-SLAM, such improvement is not significant. Analysis of the trajectory plots in Fig. 4 (a) and (b), our method shows its capability to reduce cumulative errors. Nonetheless, it also introduces additional noise, which can disrupt localization.

2) Dataset and implementation: Apart from pose corrections, Fig. 4 (c) and (d) illustrate the results of mapping two sequences onto satellite imagery using our method. The mapping results indicate that our cross-view SLAM is capable of providing real-time localization and mapping in the real-world coordinate system. When combined with the notable storage benefits of satellite imagery over feature point maps (for instance, sequences 00 and 02 with over 4000 frames require more than 2GB memory for global map storage, whereas a satellite map demands just around 10MB), we believe our approach not only provide SLAM systems with a broader range of practical application scenarios but also offers vast potential for future research in robot systems.

V. CONCLUSIONS

In this paper, we have presented a method that uses satellite maps to assist in robot visual localization. By aligning and comparing the semantic distribution from ground and satellite perspectives, we can optimally estimate a robot's pose. Building on this concept, we develop a cross-view localization approach and integrate it into a SLAM system. Our experiment demonstrated that our approach is not only highly accurate for the localization task but also effectively minimizes cumulative errors from visual measurements in the SLAM system. The primary limitation of our method is its reliance on semantic and geometric information, which may render it less effective in unstructured environments. Despite this, our approach shows broader application possibilities for robots and open up extensive avenues for future research.

REFERENCES

[1] A. Gawel, C. Del Don, R. Siegwart, J. Nieto, and C. Cadena, "X-View: Graph-based semantic multi-view localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1687–1694, 2018.

[2] X. Guo, J. Hu, J. Chen, F. Deng, and T. L. Lam, "Semantic histogram based graph matching for real-time multi-robot global localization in large scale environment," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8349–8356, 2021.

[3] Z. Ye, C. Bao, X. Liu, H. Bao, Z. Cui, and G. Zhang, "Crossview mapping with graph-based geolocalization on city-scale street maps," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 7980–7987.

[4] 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*, vol. 34, no. 7, pp. 1281–1298, 2011.

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

[6] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[7] M. Cummins and P. Newman, "FAB-MAP: Probabilistic localization and mapping in the space of appearance," *The International journal of robotics research*, vol. 27, no. 6, pp. 647–665, 2008.

[8] D. Gálvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[9] X. Guo, J. Hu, H. Bao, and G. Zhang, "Descriptor distillation for efficient multi-robot slam," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 6210–6216.

[10] S. Hu, M. Feng, R. M. Nguyen, and G. H. Lee, "CVM-Net: Cross-view matching network for image-based ground-to-aerial geo-localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7258–7267.

[11] Y. Shi, X. Yu, L. Liu, T. Zhang, and H. Li, "Optimal feature transport for cross-view image geo-localization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 11 990–11 997.

[12] Y. Shi, X. Yu, D. Campbell, and H. Li, "Where am i looking at? joint location and orientation estimation by cross-view matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4064–4072.

[13] S. Zhu, T. Yang, and C. Chen, "VIGOR: Cross-view image geo-localization beyond one-to-one retrieval," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3640–3649.

[14] Y. Shi, X. Yu, D. Campbell, and H. Li, "Where am i looking at? joint location and orientation estimation by cross-view matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4064–4072.

[15] I. A. Barsan, S. Wang, A. Pokrovsky, and R. Urtasun, "Learning to localize using a lidar intensity map," in *Conference on Robot Learning*. PMLR, 2018, pp. 605–616.

[16] D. Barnes, R. Weston, and I. Posner, "Masking by moving: Learning distraction-free radar odometry from pose information," in *Conference on Robot Learning*. PMLR, 2020, pp. 303–316.

[17] 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*. IEEE, 2012, pp. 3354–3361.

[18] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[19] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5297–5307.

[20] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, "SuperGLUE: A stickier benchmark for general-purpose language understanding systems," *Advances in neural information processing systems*, vol. 32, 2019.

[21] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas, "Working hard to know your neighbor's margins: Local descriptor learning loss," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[22] Y. Tian, B. Fan, and F. Wu, "L2-Net: Deep learning of discriminative patch descriptor in euclidean space," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 661–669.

[23] Y. Tian, X. Yu, B. Fan, F. Wu, H. Heijnen, and V. Balntas, "SOSNet: Second order similarity regularization for local descriptor learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 016–11 025.

[24] S. Garg, N. Suenderhauf, and M. Milford, "Lost? appearance-invariant place recognition for opposite viewpoints using visual semantics," *Robotics: Science and Systems XIV*, 2018.

[25] J. Ni, Y. Li, Z. Huang, H. Li, H. Bao, Z. Cui, and G. Zhang, "PATS: Patch area transportation with subdivision for local feature matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 776–17 786.

[26] Y. Shi and H. Li, "Beyond cross-view image retrieval: Highly accurate vehicle localization using satellite image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 010–17 020.

[27] S. Wang, Y. Zhang, A. Vora, A. Perincherry, and H. Li, "Satellite image based cross-view localization for autonomous vehicle," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3592–3599.

[28] Z. Xia, O. Booij, and J. F. Kooij, "Convolutional cross-view pose estimation," *arXiv preprint arXiv:2303.05915*, 2023.

[29] F. Fervers, S. Bullinger, C. Bodensteiner, M. Arens, and R. Stiefelhagen, "Uncertainty-aware vision-based metric cross-view geolocalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 21 621–21 631.

[30] P.-E. Sarlin, D. DeTone, T.-Y. Yang, A. Avetisyan, J. Straub, T. Malisiewicz, S. R. Bulò, R. Newcombe, P. Kontschieder, and V. Balntas, "OrienterNet: Visual localization in 2d public maps with neural matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 21 632–21 642.

[31] K.-N. Lianos, J. L. Schonberger, M. Pollefeys, and T. Sattler, "VSO: Visual semantic odometry," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 234–250.

[32] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[33] J. G. Mangelson, D. Dominic, R. M. Eustice, and R. Vasudevan, "Pairwise consistent measurement set maximization for robust multi-robot map merging," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 2916–2923.

[34] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun, "Unified perceptual parsing for scene understanding," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 418–434.

[35] F. A. Reda, G. Liu, K. J. Shih, R. Kirby, J. Barker, D. Tarjan, A. Tao, and B. Catanzaro, "SDC-Net: Video prediction using spatially-displaced convolution," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 718–733.

[36] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "SemanticKITTI: A dataset for semantic scene understanding of lidar sequences," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9297–9307.

[37] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.