2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

BEVRender: Vision-based Cross-view Vehicle Registration in Off-road GNSS-denied Environment

Lihong Jin, Wei Dong, Wenshan Wang, and Michael Kaess

***Abstract*— We introduce BEVRender, a novel learning-based approach for the localization of ground vehicles in Global Navigation Satellite System (GNSS)-denied off-road scenarios. These environments are typically challenging for conventional vision-based state estimation due to the lack of distinct visual landmarks and the instability of vehicle poses. To address this, BEVRender generates high-quality local bird's-eye-view (BEV) images of the local terrain. Subsequently, these images are aligned with a georeferenced aerial map through template matching to achieve accurate cross-view registration. Our approach overcomes the inherent limitations of visual inertial odometry systems and the substantial storage requirements of image-retrieval localization strategies, which are susceptible to drift and scalability issues, respectively. Extensive experimentation validates BEVRender's advancement over existing GNSS-denied visual localization methods, demonstrating notable enhancements in both localization accuracy and update frequency.**

I. Introduction

Global localization is a crucial component that supports smooth navigation of autonomous vehicles. It is typical to equip on-board localization systems with the Global Navigation Satellite System (GNSS) modules for consistent and reliable global poses. However, in reality, GNSS signals can be blocked due to natural or artificial barriers, causing temporal system failures, where vision-based localization (VBL) serves as an alternative in GNSS-denied localization. A variety of methods have been proposed for VBL in urban scenarios [1], yet off-road VBL for unmanned ground vehicle (UGV) is still challenging due to non-urban environments lacking stable and distinct visual features, such as roads and buildings. The varied and unpredictable terrain further complicates the task by inducing unstable vehicle poses, making it difficult to maintain consistent feature matching across frames.

In response to these challenges, our paper presents a novel learning-based method that synthesizes a local bird's eye view (BEV) image of the surrounding area by aggregating visual features from camera images. This approach integrates a modified BEVFormer [2] framework with a novel rendering head, employing template matching for precise cross-view registration between ground vehicles and aerial maps in GNSS-denied off-road environments.

The authors are with the Robotics Institute, Carnegie Mellon University, PA, USA {lihongj, weidong, wenshanw, kaess}@andrew.cmu.edu

This material is based upon work supported by the U.S. Army Research Office and the U.S. Army Futures Command under Contract No. W911NF-20-D-0002. The content of the information does not reflect the position or the policy of the government, and no official endorsement should be inferred.

We concentrate on 2D relocalization of unmanned ground vehicles (UGV) for non-urban settings bounded within defined areas. Equipped with trinocular RGB cameras and an Inertial Measurement Unit (IMU), the vehicle employs multi-view visual inertial odometry (VIO) for state estimation. Our aim is to achieve accurate 2D positioning relative to a geo-referenced aerial map, facilitating pose correction in the absence of GNSS signals, whether temporarily or persistently. A more detailed problem definition is in Sec. III.

Previous study [3] has explored the creation of orthographic view images by accumulating geometric features over consecutive frames, coupled with Normalized Cross-correlation (NCC) for relocalization in a GPS-denied situation. However, this approach is limited by the inherent drift of VIO systems, which can distort the accumulated geometric data, leading to inaccuracies in ground-to-air matching. Our paper introduces a learning-based strategy for generating BEV images, using a Vision Transformer (ViT) [4]-based network for feature encoding. This method shows improved performance in generating local BEV images and supporting vehicle localization with geo-referenced aerial maps.

Other research efforts [5], [6], [7] treat vision-based localization as an image retrieval problem, requiring substantial storage for on-board localization systems. On the contrary, our approach generates local BEV images for direct template matching. This significantly reduces the need for extensive data storage, relying instead on a geo-referenced map for real-time 2D localization. In summary, our contributions are threefold:

1) We propose a novel *learning*-based framework for ground vehicle localization that combines BEV image generation with *classical* template matching, eliminating the extensive dataset storage requirements of image-retrieval-based localization.
2) We integrate the deformable attention module in [8] with the BEVFormer network, enhancing feature encoding by using offset networks [8], followed by an efficient image rendering head as a feature decoder capable of producing detailed top-down views of the local terrain.
3) Through comprehensive experiments with real-world datasets, we demonstrate that our method exhibits superior localization accuracy and frequency compared to existing GNSS-denied visual localization techniques, and generalizes to unseen trajectories.

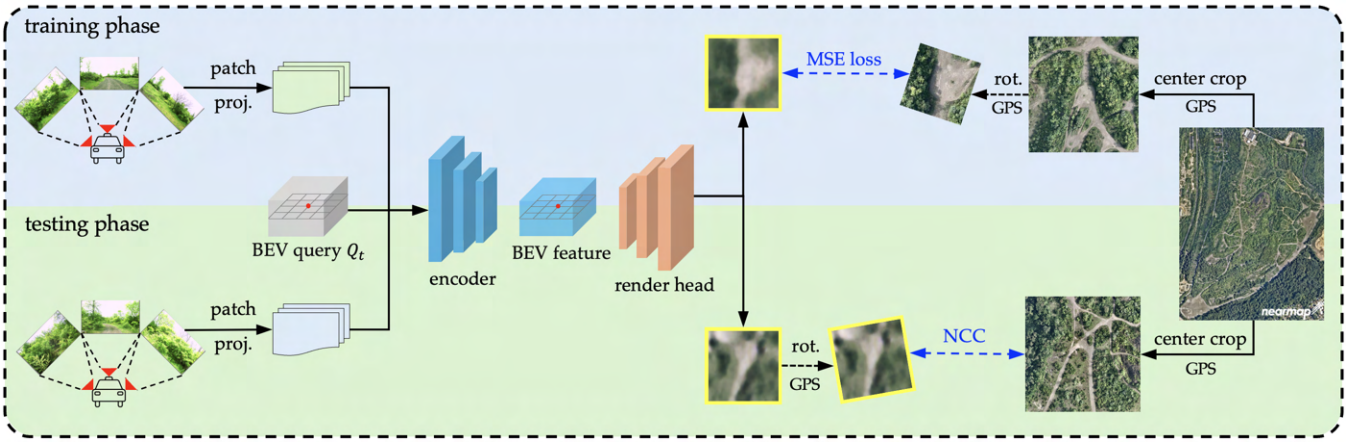

Fig. 1. A diagram of our system. The light blue background indicates the training phase and the light green background indicates the testing phase. During the training phase, camera images are patch projected and sent to the feature encoder (in blue) and rendering head (in orange) to generate BEV images (highlighted in yellow boxes). The aerial map image is rotated and cropped according to the GPS information provided, ensuring that the final label image accurately represents the BEV space surrounding the vehicle. During the testing phase, the rendered BEV image is rotated according to the azimuth angle provided by the GPS, and matched against a local search region surrounding the vehicle position.

II. Related Work

A. GNSS-denied Vehicle Localization

Vehicle localization in GNSS-denied environments can be broadly categorized into relative and absolute localization strategies. Relative localization aims to mitigate odometry drifts by fusing data from multiple onboard sensors with motion models, or by leveraging loop closures to correct drift relative to global frames [9]. Absolute localization, in contrast, involves constructing local maps from the vehicle's perspective and aligning them with a global georeferenced map to determine precise vehicle positions. Reference data for this process can vary, including High-definition (HD) maps [10], aerial satellite imagery, Digital Elevation Models (DEM) [11], [12], and OpenStreetMap (OSM) data [13]. While HD maps offer high accuracy, they are costly and data intensive. DEMs, primarily used for UAVs [11], cater to non-planar terrains and scale ambiguity, whereas OSM provides dense semantic and geometric details suitable for urban navigation. Aerial satellite maps present strong visual cues with detailed information for off-road localization.

Significant advancements have been made in aligning ground-level images with aerial imagery for localization. Viswanathan et al. [14] demonstrate effective ground-to-air image matching using satellite images by warping UGV panoramic images to a bird's eye view, comparing feature descriptors, and employing a particle filter for accurate localization. Based on this, recent work [3] focuses on generating an orthographic occupancy map by accumulation of local features and estimation of pose through NCC, and optimizing the prediction of global pose through a registration graph [15]. In contrast, our approach adopts a Vision Transformer (ViT)-based [4] learning network to generate BEV images for ground-to-air matching, emphasizing frame-by-frame registration accuracy and reducing reliance on global trajectory optimization.

B. Learning Vision-based Localization

The evolution of vision-based localization has seen it conceptualized as an image retrieval task [5], employing contrastive learning to enhance the matching of onboard camera and satellite images [6], [7]. Efforts to improve image alignment include warping satellite imagery by polar transformation to match ground perspectives [7], and constructing semantic neural maps from camera images [6]. Further innovations leverage CNNs for feature extraction and BEV representation, enabling precise localization through 3D structure inference and matching [13], [16], [17], [18].

The advent of foundation models offers promising directions for Visual Place Recognition (VPR), demonstrating the adaptability of pre-trained models (e.g., DINO [19], DINOv2 [20]) to diverse environments without fine-tuning [21]. Subsequent work [22] integrates dense visual feature extraction with advanced filtering and global-local pose estimation via Extended Kalman Filters (EKF) for refined localization accuracy. Our methodology aligns with these advancements, utilizing a streamlined ViT architecture for efficient and accurate BEV image rendering and localization, minimizing parameter overhead while maximizing performance.

In the realm of self-driving applications, BEV representations [23], [24] have been enriched by encoding temporal and spatial features, as demonstrated by BEVFormer [2], which leverages attention mechanisms [25], [8] for 3D object detection. Our work extends this concept by incorporating BEVFormer's feature propagation approach, ensuring our BEV representations integrate temporal information from successive frames. This strategy is complemented by recent explorations in temporal information encoding for BEV representation, highlighting the continuous evolution and application of these techniques in autonomous navigation [26], [27], [28], [29], [30].

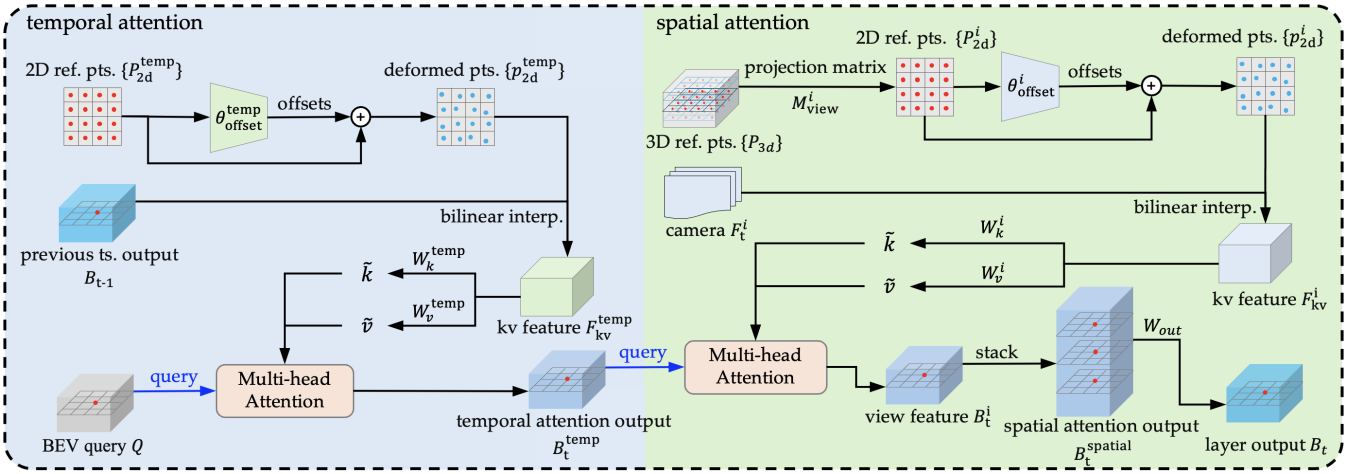

Fig. 2. Encoder layer architecture. An encoder layer is composed of temporal and spatial attention. In temporal attention, a set of 2D reference points with a spatial dimension of $l \times w$ is sampled and deformed. Next, bilinear sampling is performed to extract tokens for multi-head attention (MHA) [25] given deformed reference points from previous timestamp BEV feature B_{t-1}. The MHA output from temporal attention serves as a query for the subsequent spatial attention module. In spatial attention, we sample one point per 3D grid cell in the BEV space as reference points and project them to the three camera image frames with extrinsic and intrinsic parameters to obtain 2D reference points for each image view. Similarly to temporal attention, the 2D reference points are deformed and used for bilinear sampling, but from camera feature. A more detailed description can be found in Sec. III-A.

III. Methodology

Our system contains three main components: a feature encoder to map the visual features from the camera to the top-down view, a rendering head to decode the features and generate top-down BEV images, and an image registration component for cross-view localization. An overview of our system is shown in Fig. 1.

We consider a scenario where a vehicle, equipped with trinocular cameras and an IMU, is traversing flat natural terrain. A pre-stored aerial map of the area aids in localization. The vehicle's pose is predicted by the VIO system in a local coordinate frame as follows:

$$\mathbf{X}_t = \Big[x_t, y_t, \theta_t\Big] \in \mathbf{SE}(2). \tag{1}$$

We assume that the prediction for the azimuth angle θ_t from VIO is accurate, but the position estimates (x_t and y_t) may drift over time. Our system aggregates information from consecutive frames to construct a top-down representation of the environment for map registration.

We define a 3D BEV space centered on the vehicle with a length of L, a width of W and a height of H. The space is divided into $l \times w \times h$ grid cells, so that each cell represents a cubic size of $\frac{L}{l} \times \frac{W}{w} \times \frac{H}{h}$ in the real world. The BEV query is a 3D trainable embedding with a dimension of $l \times w \times h$ representing the BEV space and serving as the query for deformable attention modules in the encoder. All intermediate BEV features in the network also follow the same spatial dimension. The specific range and dimension chosen for our experiment are described in Sec. IV-A.

Our system seeks to find the optimal pose prediction that minimizes the difference between camera feature representation and local aerial image:

$$\mathbf{X}^* = \arg\min_{\mathbf{X}} \Phi\Big(I_{\text{bev}}^{'}(\mathbf{X}), I_{\text{map}}(\mathbf{X})\Big), \tag{2}$$

where Φ is a function to find $\mathbf{X}^*$ to achieve minimum distance between two representations in feature space, and is provided by template matching module in our system. I_{map} is a subset of the aerial map with respect to a vehicle pose.

The image rendering head Ψ_{render} defines a mapping from the encoded image feature F_{feat} to top-down BEV image $I_{\text{bev}}^{'}$:

$$I_{\text{bev}}^{'}(\mathbf{X}) = \Psi_{\text{render}}\Big(F_{\text{feat}}(\mathbf{X})\Big). \tag{3}$$

A. Feature Encoding with BEVFormer

Adopting BEVFormer's framework [2], we propagate consecutive frame features to capture temporal information. Within a temporal window of T seconds, n frames ($3 \times n$ images in a trinocular setup) are sampled. A detailed setting can be found in Sec. IV-A. Starting with the earliest frame, camera images I_t^{cam} are processed through patch projection, which is a convolutional layer in our implementation, to obtain camera feature F_t^{cam} and sent to the encoder together with the BEV query Q and previous BEV feature B_{t-1} to obtain the encoded BEV feature for current timestamp B_t. The encoding process consists of two stages: a temporal attention stage that takes in query Q and previous timestamp BEV feature B_{t-1} for deformable attention:

$$B_t^{\text{temp}} = \text{DeformableAttn}\Big(B_{t-1}, Q\Big), \tag{4}$$

followed by a spatial attention stage that takes in temporal output and camera feature F_t for deformable attention:

$$B_t = B_t^{\text{spatial}} = \text{DeformableAttn}\Big(F_t^{\text{cam}}, B_t^{\text{temp}}\Big). \tag{5}$$

B_t is then projected to the subsequent frame vehicle pose as $B_t^{'}$ according to the movement of the vehicle provided by GPS. The projection is performed by affine transformations in $\mathbf{SE}(2)$ followed by a bilinear interpolation:

$$\Delta\mathbf{X} = \mathbf{X}_t - \mathbf{X}_{t-1} = \Big[\Delta x, \Delta y, \Delta\theta\Big], \tag{6}$$

$$\begin{bmatrix} x_t \\ y_t \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\Delta\theta & -\sin\Delta\theta & \Delta x \\ \sin\Delta\theta & \cos\Delta\theta & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ 1 \end{bmatrix}, \quad (7)$$

$$B_t^{'}(x_t, y_t) = \text{BilinearInterp}(B_t(x_t, y_t)). \quad (8)$$

Subsequently, $B_t^{'}$ serves as a query to the encoder to query the next timestamp camera feature F_{t+1} to obtain B_{t+1}. The propagation continues in the temporal window until we obtain the latest timestamp feature. A diagram of temporal propagation is shown in Fig. 3. It should be noted that B_{t-1} is the same as query Q for the first frame in the temporal window:

$$B_{t-1} = Q \quad \text{if} \quad t = 0. \quad (9)$$

Unlike BEVFormer, our encoder simplifies to a single layer, totaling 1.44 million parameters, while supporting effective feature learning for downstream localization tasks.

The architecture of the encoding layer is shown in Fig. 2, and the ablation study of the number of layers can be found in Table V.

B. Deformable Attention Vision Transformer

In contrast to BEVFormer that employs Deformable DETR [31], our approach utilizes the deformable attention [8], which uses offset networks to calculate adjustments to each reference point. The offsets are processed by an additional convolution layer θ_{offset}, as shown in Fig. 2, and its output modifies the original reference point to generate deformed reference points. For spatial attention, offsets θ^i_{offset} are added to the reference points unique to each camera view i, acting as adjustments to the pixel locations of reference points. Consequently, we employ three distinct convolution layers dedicated to learning offsets as an adaptation to the trinocular system setting. The final output of the spatial attention layer is a stacking of features from three camera views, undergoing another convolutional layer to maintain the same spatial dimension as the BEV query and BEV features.

The output of deformable attention heads is formulated as

$$z^{(m)} = \sigma\Big(\frac{q^{(m)}k^{(m)\top}}{\sqrt{d}} + \phi(B;R)\Big)v^{(m)}, \quad (10)$$

where q, k, v constructs the standard transformer attention [25] with softmax activation σ and scale normalization $\sqrt{d}$, enhanced by relative positional bias [32] in $\phi(B;R)$. A more detailed description of deformable attention formulation can be found in [8].

C. BEV Image Rendering Head

The BEV image rendering head is designed to translate encoded features into interpretable top-down views of the vehicle's surroundings. It is a straightforward convolutional neural network (CNN) architecture that takes as input the encoded BEV features with dimensions of $d \times l \times w$, where d is the model embedding dimension. Through a series of convolutional and upsampling layers, the BEV features are processed to generate a colored image of certain size, which serves as a top-down visual representation of the BEV space around the vehicle. The rendering head ensures that the resulting BEV image retains critical spatial information required for ground-to-aerial vehicle localization in GNSS-denied environments. The detailed structure of the rendering head is illustrated in Table I.

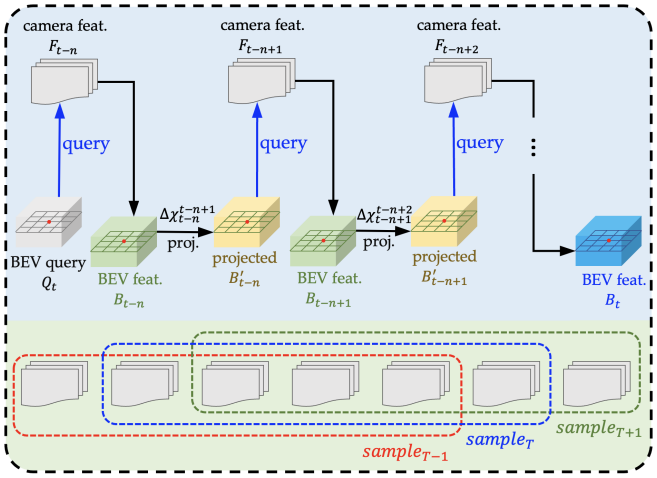

Fig. 3. Temporal feature propagation and dataset organization. For each timestamp, we sample n frames from past T seconds, composing a training sample of $n+1$ camera frames together with current timestamp frame. Staring with the earliest timestamp in the window, BEV query Q is used to query camera feature F to obtain BEV feature B, which is subsequently projected to next timestamp vehicle position given GPS outputs, to obtain new feature B'. Propagation continues until the latest frame is processed. A detailed description on projection can be found in Sec. III-A.

TABLE I
BEV RENDERING HEAD ARCHITECTURE

block	layer
Decoder block 0	Conv2d + BN + ReLU
Decoder block 1	(Conv2d + BN)×4 + ReLU
Decoder block 2	(Conv2d + BN)×4 + ReLU
Decoder block 3	(Conv2d + BN)×4 + ReLU
Upsample block 0	Upsample + (Conv2d + BN)×2 + ReLU
Upsample block 1	Upsample + (Conv2d + BN)×2 + ReLU
Upsample block 2	Upsample + (Conv2d + BN)×2 + ReLU
Upsample block 3	Upsample + (Conv2d + BN)×2 + Sigmoid

IV. EXPERIMENTS

A. Experiment Setting

Since the satellite image[1] has a resolution of 0.229 meters per pixel, we define the length and width of the BEV space as 25.648 meters centered on vehicle position, equivalent to a size of 112×112 pixels on the aerial map. We also define the height of the BEV space as 2 meters. The space is divided into $28 \times 28 \times 5$ 3D grid cells, so that each cell represents a voxel of $0.916 \times 0.916 \times 0.4\, m^3$ in the real world. We utilize a temporal window of 5 seconds and randomly sample 5 frames in the window to compose a training sample.

[1]The satellite image used in this paper is provided by Nearmap.

Seq 1 *Seq 2* *Seq 3* *Seq 4*

Fig. 4. Qualitative comparison of our method and Litman [3]. *Top row*: the rendering and registration result of our method, where the BEV images are highlighted in yellow boxes, the red dots indicate the NCC predictions from our system, and the blue dots indicate the GPS ground truth position. Our approach produces coherent rendering to the aerial image. *Bottom row*: predictions from Litman [3]. Similarly, the red and blue dots indicate the predictions and ground truth, while the yellow boxes indicate the generated occupancy image overlaid on the groundtruth. Only semi-dense rendering are available for Litman [3] (see the saturated white and green points around red dots), resulting in compromised registration accuracy.

TABLE II

STATISTICS OF GNSS-DENIED REAL-WORLD DATASET

	# images	traj. length (m)	coverage (m^2)
Seq 1	1634	1059.42	349.34 × 159.70
Seq 2	1563	1067.08	349.34 × 159.67
Seq 3	1427	1415.72	353.07 × 164.65
Seq 4	1210	1228.61	350.99 × 161.92
Seq 5	1707	1179.64	462.53 × 359.25
Seq 6	838	495.64	340.13 × 239.08
Seq 7	815	439.67	410.86 × 74.74
Seq 8	1395	1425.88	368.51 × 166.06
aerial map	-	-	1278.20 × 1646.46

We conduct two main experiments, one to compare against state-of-the-art VBL methods in GNSS-denied setting [3], [7], where we use 4 sequences and split them into 80% training, 20% testing data; and another to show our model's ability to generalize across different scenes given limited training data, where we use 2 sequences for training and 4 sequences for testing. The trajectory plots for sequences used in the cross-sequence testing experiment are shown in Fig. 5.

Training is distributed on 8 NVIDIA A100 GPUs for a total of 2500 epochs and with a learning rate of $4e^{-5}$. The configuration of the testing computer is described in Sec. IV-C in system runtime.

During the testing phase, we crop and rotate the aerial map based on the GPS ground truth position as the center of the image with a size of 874×874 pixels, which corresponds to a real-world coverage of approximately 200×200 square meters. This search region is sufficient to accommodate VIO drift for more than 10 minutes without registration. For cross-sequence testing, we loosen the assumption of drifting range and use a search region of 100×100 square meters. Our camera system captures 3 frames per second and predicts registration consistent with camera frame; therefore, sufficient for preventing from failing with the 100×100 square meter search range.

We use NCC for template matching. NCC identifies the best match within the search area, maximizing similarity between the generated BEV image and the aerial map, thus predicting the vehicle's position relative to the aerial map. We observe failure cases where rendered BEV images are of moderate visual quality, whereas NCC fails in prediction. An example of failure cases is shown in Fig. 6.

B. Dataset Organization

We collect our real-world data set in the Pittsburgh area, with a VIO system on board. Detailed information on the sequences can be found in Table II. For each training sample, we use the information of timestamp, trinocular RGB images, and GPS ground truth including x, y, and azimuth angle in the UTM coordinate system for training. The preprocessing for cropping the aerial map can be found in Fig. 1.

C. Quantitative Comparison

We compare our method with GPS denied registration via occupancy mapping proposed in [3], and GeoDTR proposed in [7]. The comparison result is shown in Table III.

TABLE III

QUANTITATIVE COMPARISONS ON OUR REAL-WORLD DATASET

approach	*Seq 1*			*Seq 2*			*Seq 3*			*Seq 4*		
	mean ↓	std ↓	match (%) ↑	mean ↓	std ↓	match (%) ↑	mean ↓	std ↓	match (%) ↑	mean ↓	std ↓	match (%) ↑
Litman [3]	24.35	13.50	21.62 (Rmk.1)	34.45	21.59	12.12	26.27	13.44	11.46	61.04	55.80	8.89
GeoDTR [7] (top 1)	82.72	25.52	0.00 (Rmk.2)	90.27	29.92	1.28	84.40	27.33	0.36	86.53	27.60	0.00
GeoDTR [7] (top 5 avg.)	67.35	24.22	0.94	74.06	30.43	1.60	71.33	28.91	1.07	66.34	29.35	1.67
Ours	19.33	26.09	63.44	22.40	27.92	60.90	20.60	24.96	58.93	21.18	25.49	57.50

1. The darker shading indicates the best results, and the lighter shading indicates the second-best results.
2. The mean and std are calculated for the APE for predicted positions, see the registration accuracy of Sec. IV-C for more details.
3. The search region is set to 200×200 square meters.

Since GeoDTR is an image-retrieval-based method and relies on cultivating the corresponding information between camera inputs and polar transformation of aerial map images, it is required to preserve a database of candidate polar transformed images for real-world vehicle localization. We randomly sample 5000 particles within the search region at each timestamp and apply polar transformation according to the particle location on the map together with the azimuth angle of GPS ground truth. After obtaining the candidate polar images, for each timestamp, we pass in the camera images and polar images to the model, and calculate the distance between camera descriptors and polar descriptors, we choose candidate with closest descriptor distance as the top 1 prediction, and its corresponding real-world location as top 1 location, and we average the top 5 predicted locations as top 5 prediction.

Registration accuracy To evaluate the accuracy of vehicle registration, we calculate the mean and standard deviation (STD) of absolute position error (APE) between predicted position and the ground truth vehicle location given by on-board GPS.

Registration frequency In the real-world localization scenario, the update frequency is another important factor that determines the stability of registration system. We report the matching frequency by counting the total successful matches when the APE is within a threshold of 10 meters (the range we deem tolerable for our VIO system) and calculate the match rate as total successful matches divided by total camera frames for a sequence:

$$\mathbf{x}^i = (x_i, y_i), \tag{11}$$

$$\mathbf{d}^i_{\text{Euclidean}} = \|\mathbf{x}^i_{\text{gt}} - \mathbf{x}^i_{\text{pred}}\|_2, \tag{12}$$

$$\mathbf{p}_{\text{match}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1} \cdot (\mathbf{d}^i_{\text{Euclidean}} < \mathbf{d}_{\text{threshold}}), \tag{13}$$

where N is the number of images for a sequence per camera module.

Remark 1 (Testing with Litman [3])**:** *It should be noted that the method proposed in [3] accumulates geometric features on a certain number of consecutive camera frames (50 by default), leading to a limited number of registration try-outs throughout a sequence. For comparison sake, we calculate the match rate as the total number of successful matches divided by the total number of occupancy maps synthesized in a sequence.*

Remark 2 (Testing with GeoDTR)**:** *It takes up to 21 hours to sample polar images for 5000 particles for 320 testing samples; hence, we cannot further increase the density of particles. To apply image-retrieval-based method for on-board localization, it is required to have a pre-stored dataset, specifically in our case, of polar images sampled from all candidate positions on local aerial map enumerating all possible rotations, which is prohibitively expensive storage for on-board system in real-world localization.*

System runtime Testing is performed on a machine equipped with an AMD Ryzen 9 5900X 12-Core processor and a NVIDIA GeForce RTX 4090. The total time to localize 280 testing samples is 33.32 seconds, equivalent to 0.12 seconds to localize per camera frame. The camera frame rate for our system is 3 per second; therefore, our system is able to support online localization in real-world scenario.

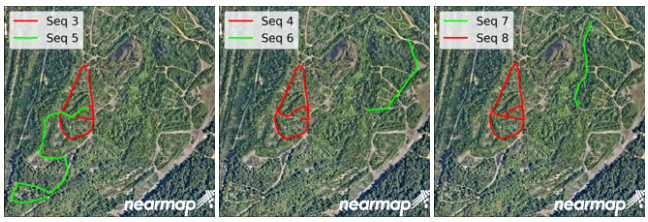

Fig. 5. Trajectory plot for cross-sequence testing. Sequence 3 and 8 are used in training, sequence 4 to 7 are used in testing.

TABLE IV

CROSS-SEQUENCE TESTING FOR MODEL GENERALIZATION

sequence	mean ↓	std ↓	match(%) ↑
Seq 4	11.24	6.64	45.38
Seq 5	13.77	6.74	31.16
Seq 6	12.72	6.38	36.63
Seq 7	16.30	6.92	21.81

D. Qualitative Comparison

Visualizations of the rendering and registration result can be found in Fig. 4. The image rendering head processes the encoded BEV feature with a spatial dimension of $64 \times 28 \times 28$ through a set of convolutional layers and 4 upsample layers, as shown in Table I. The final BEV image is an RGB image with a size of 224×224 pixels, representing an area of $51.296 \times 51.296\ m^2$. The occupancy map reconstructed from [3] aggregates geometric features from 50 consecutive frames, of which the coverage may vary for each prediction.

E. Model Generalization

To test the generalizability of the proposed system, we perform cross-sequence tests. Specifically, training with se-

quences 3 and 8 while testing with sequences 4-7. We report search regions of $100 \times 100\, m^2$ in Table IV.

F. Ablation Study

In this section we explore the influence of choosing different hyperparameters and BEV space resolutions on the final registration result. Since the aerial map resolution is 0.229 meters, we experiment with the BEV grid resolutions of 0.458 meters and 0.916 meters, corresponding to 2 pixels and 4 pixels on the map, respectively. We also experiment with an increased number of layers and report the results in Table V. Considering the result from ablation study, we choose the resolution of the BEV grid as 0.916 meters, and the number of encoder layers as 1 for Table III and Table IV.

TABLE V
ABLATION STUDY ON SEQ 4
EFFECTS OF ARCHITECTURE CHOICE AND HYPER PARAMETERS

# layers	grid reso. (m)	# params	mean ↓	std ↓	match(%) ↑
1	0.458	1.71M	27.47	27.83	48.75
2	0.458	2.09M	27.75	27.32	45.42
1	0.916	1.44M	21.17	25.49	57.50
2	0.916	1.72M	36.40	25.66	20.42

V. Conclusion and Future Work

We present a learning-based system to generate local BEV images combined with NCC for ground vehicle localization in GNSS-denied off-road environments. Our system incorporates the deformable attention module with BEVFormer for a multi-view camera sensor setting, followed by a novel rendering head to generate high-precision BEV images to enable downstream localization task.

To enhance our ground vehicle localization system for operation across different seasons, future research will focus on improving the network's ability to learn and generalize features from varied seasonal landscapes. This is essential for deploying our system in real-world scenarios where environmental conditions fluctuate significantly over the year. Additionally, we aim to advance the fidelity of BEV image generation by incorporating techniques such as the diffusion module, inspired by the diffusion transformer [33]. This enhancement is expected to refine the detail and precision of the BEV images, thus enriching contextual data for more accurate vehicle localization.

Further improvements will also explore the integration of temporal features to accumulate historical data more effectively, addressing current limitations caused by projection adjustments and vehicle pose changes. Moreover, explorations can be made on removing dependence on GPS information for training by leveraging local state estimates from VIO. Furthermore, a sophisticated approach to incorporate data from previous frames could significantly improve rendering quality and system performance. In addition, a transition from classic template matching to learnable template matching for vehicle positioning is anticipated to overcome the limitation of NCC's uniform pixel weighting, as shown in Fig. 6, and to enable the system to prioritize strategically significant areas, potentially elevating the accuracy of vehicle registration in challenging environments.

Fig. 6. Examples of failure cases due to the uniform weighting of NCC.

Acknowledgement

The authors thank Kaicheng Yu for the fruitful discussion and valuable suggestions, and staff members of the National Robotics Engineering Center (NREC) for helping with data collection.

References

[1] Y. Alkendi, L. Seneviratne, and Y. Zweiri, "State of the art in vision-based localization techniques for autonomous navigation systems," *IEEE Access*, vol. 9, pp. 76 847–76 874, 2021.

[2] 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 Conf. on Computer Vision (ECCV)*. Springer, 2022, pp. 1–18.

[3] Y. Litman, D. McGann, E. Dexheimer, and M. Kaess, "Gps-denied global visual-inertial ground vehicle state estimation via image registration," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2022, pp. 8178–8184.

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *Intl. Conf. on Learning Representations (ICLR)*, 2021.

[5] J. Wolf, W. Burgard, and H. Burkhardt, "Robust vision-based localization by combining an image-retrieval system with monte carlo localization," *IEEE Trans. Robotics*, vol. 21, no. 2, pp. 208–216, 2005.

[6] P.-E. Sarlin, E. Trulls, M. Pollefeys, J. Hosang, and S. Lynen, "SNAP: Self-Supervised Neural Maps for Visual Positioning and Semantic Understanding," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[7] X. Zhang, X. Li, W. Sultani, Y. Zhou, and S. Wshah, "Cross-view geo-localization via learning disentangled geometric layout correspondence," in *AAAI Conf. on Artificial Intelligence*, vol. 37, no. 3, 2023, pp. 3480–3488.

[8] Z. Xia, X. Pan, S. Song, L. E. Li, and G. Huang, "Vision transformer with deformable attention," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 4794–4803.

[9] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1271–1278.

[10] F. Poggenhans, J.-H. Pauls, J. Janosovits, S. Orf, M. Naumann, F. Kuhnt, and M. Mayr, "Lanelet2: A high-definition map framework for the future of automated driving," in *IEEE Intl. Conf. on intelligent transportation systems (ITSC)*. IEEE, 2018, pp. 1672–1679.

[11] X. Wan, Y. Shao, S. Zhang, and S. Li, "Terrain aided planetary uav localization based on geo-referencing," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 60, pp. 1–18, 2022.

[12] G. Kuppudurai, K.-y. Hwang, H.-G. Park, and Y. Kim, "Localization of airborne platform using digital elevation model with adaptive weighting inspired by information theory," *IEEE Sensors Journal*, vol. 18, no. 18, pp. 7585–7592, 2018.

[13] P.-E. Sarlin, D. DeTone, T.-Y. Yang, A. Avetisyan, J. Straub, T. Malisiewicz, S. R. Bulo, R. Newcombe, P. Kontschieder, and V. Balntas, "OrienterNet: Visual Localization in 2D Public Maps with Neural Matching," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR)*, 2023.

[14] A. Viswanathan, B. R. Pires, and D. Huber, "Vision based robot localization by ground to satellite matching in gps-denied situations," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2014, pp. 192–198.

[15] F. Dellaert and M. Kaess, "Factor graphs for robot perception," *Foundations and Trends in Robotics (FNT)*, vol. 6, no. 1-2, pp. 1–139, 2017.

[16] A. B. Camiletto, A. Bochicchio, A. Liniger, D. Dai, and A. Gawel, "U-bev: Height-aware bird's-eye-view segmentation and neural map-based relocalization," 2023.

[17] F. Fervers, S. Bullinger, C. Bodensteiner, M. Arens, and R. Stiefelhagen, "C-bev: Contrastive bird's eye view training for cross-view image retrieval and 3-dof pose estimation," 2023.

[18] Z. Zhang, M. Xu, W. Zhou, T. Peng, L. Li, and S. Poslad, "Bev-locator: An end-to-end visual semantic localization network using multi-view images," 2022.

[19] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proc. IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, October 2021, pp. 9650–9660.

[20] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision," 2024.

[21] N. Keetha, A. Mishra, J. Karhade, K. M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg, "Anyloc: Towards universal visual place recognition," 2023.

[22] Y. He, I. Cisneros, N. Keetha, J. Patrikar, Z. Ye, I. Higgins, Y. Hu, P. Kapoor, and S. Scherer, "Foundloc: Vision-based onboard aerial localization in the wild," *arXiv preprint arXiv:2310.16299*, 2023.

[23] Y. B. Can, A. Liniger, D. P. Paudel, and L. Van Gool, "Structured bird's-eye-view traffic scene understanding from onboard images," in *Proc. IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, October 2021, pp. 15 661–15 670.

[24] H. Li, C. Sima, J. Dai, W. Wang, L. Lu, H. Wang, J. Zeng, Z. Li, J. Yang, H. Deng, H. Tian, E. Xie, J. Xie, L. Chen, T. Li, Y. Li, Y. Gao, X. Jia, S. Liu, J. Shi, D. Lin, and Y. Qiao, "Delving into the devils of bird's-eye-view perception: A review, evaluation and recipe," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 46, no. 4, pp. 2151–2170, 2024.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[26] C. Hu, H. Zheng, K. Li, J. Xu, W. Mao, M. Luo, L. Wang, M. Chen, Q. Peng, K. Liu, Y. Zhao, P. Hao, M. Liu, and K. Yu, "Fusionformer: A multi-sensory fusion in bird's-eye-view and temporal consistent transformer for 3d object detection," 2023.

[27] A. Saha, O. Mendez, C. Russell, and R. Bowden, "Enabling spatio-temporal aggregation in birds-eye-view vehicle estimation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5133–5139.

[28] Z. Qin, J. Chen, C. Chen, X. Chen, and X. Li, "Unifusion: Unified multi-view fusion transformer for spatial-temporal representation in bird's-eye-view," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023, pp. 8690–8699.

[29] H. Cai, Z. Zhang, Z. Zhou, Z. Li, W. Ding, and J. Zhao, "Bevfusion4d: Learning lidar-camera fusion under bird's-eye-view via cross-modality guidance and temporal aggregation," 2023.

[30] A. K. Akan and F. Güney, "Stretchbev: Stretching future instance prediction spatially and temporally," in *Computer Vision – ECCV 2022*, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds. Cham: Springer Nature Switzerland, 2022, pp. 444–460.

[31] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.

[32] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[33] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *Proc. IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2023, pp. 4195–4205.