2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

HE-VPR: Height Estimation Enabled Aerial Visual Place Recognition Against Scale Variance

Mengfan He[1], Xingyu Shao[1], Chunyu Li[2], Chao Chen[1], Liangzheng Sun[3], Ziyang Meng[1], and Yuanqing Wu[4]

***Abstract*— In this work, we propose HE-VPR, a visual place recognition (VPR) framework that incorporates height estimation. Our system decouples height inference from place recognition, allowing both modules to share a frozen DINOv2 backbone. Two lightweight bypass adapter branches are integrated into our system. The first estimates the height partition of the query image via retrieval from a compact height database, and the second performs VPR within the corresponding height-specific sub-database. The adaptation design reduces training cost and significantly decreases the search space of the database. We also adopt a center-weighted masking strategy to further enhance the robustness against scale differences. Experiments on two self-collected challenging multi-altitude datasets demonstrate that HE-VPR achieves up to 6.1% Recall@1 improvement over state-of-the-art ViT-based baselines and reduces memory usage by up to 90%. These results indicate that HE-VPR offers a scalable and efficient solution for height-aware aerial VPR, enabling practical deployment in GNSS-denied environments. All the code and datasets for this work have been released on `https://github.com/hmf21/HE-VPR`.**

I. Introduction

Visual place recognition (VPR) in aerial platforms enables stable self-positioning in global navigation satellite system (GNSS)-denied environments. Such capabilities are critical for unmanned aerial vehicles (UAVs) facing severe accumulated dead-reckoning drift, necessitating absolute pose and scale re-initialization solely from vision. However, varying flight altitudes cause significant scale variations that severely degrade the accuracy of place retrieval. Since the visual footprint is strictly determined by the relative distance to the ground, we denote this scale-determining factor (i.e., relative altitude) as *height* throughout this paper to ensure terminology consistency. Since full-database retrieval covering all possible heights is infeasible for memory-constrained systems, estimating the flight height to explicitly narrow down the search space is essential. While monocular metric depth estimation (MMDE) approaches [1], [2], [3] can provide depth priors, such dense estimation solutions struggle on aerial platforms due to scarce annotations at large heights and weak local textures [4]. To address these challenges, we propose *HE-VPR*, an efficient framework leveraging a frozen DINOv2 backbone [5] equipped with two parallel branches of lightweight bypass adapters. Specifically, each Transformer block is paired with two separate adapters dedicated to height estimation and VPR, respectively.

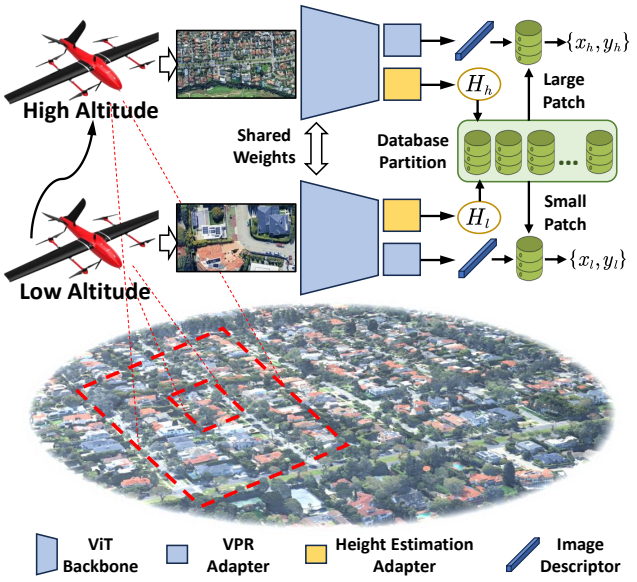


Fig. 1: Overview of the proposed HE-VPR, an aerial VPR pipeline with height estimation.

In the first stage, the height adapter branch coarsely estimates the query image's height via retrieval rather than direct regression, efficiently narrowing the search space to the correct sub-database for better generalization and lower memory consumption. In the second stage, the VPR adapter branch performs retrieval within the selected sub-database. To handle the remaining scale variations within the discrete height partition, we employ a center-weighted masking strategy. Since peripheral features are highly susceptible to truncation during height changes, the central region inherently maintains higher visual overlap. Therefore, this strategy amplifies the geometrically stable central pixels while

*This work was supported in part by the Tsinghua-Toyota Joint Research Fund, in part by the Beijing Natural Science Foundation (Grant No. L252095), and in part by the National Natural Science Foundation of China (Grant Nos. 62273195 and 62403269). *(Mengfan He and Xingyu Shao are co-first authors.) (Corresponding author: Ziyang Meng.)*

[1]Mengfan He, Xingyu Shao, Chao Chen, and Ziyang Meng are with the Department of Precision Instrument, Tsinghua University, Beijing 100084, China (e-mail: hmf21@mails.tsinghua.edu.cn; shao-xy21@mails.tsinghua.edu.cn; chen-c@mail.tsinghua.edu.cn; ziyangmeng@mail.tsinghua.edu.cn).

[2]Chunyu Li is with the School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China (e-mail: chunyuli@bit.edu.cn).

[3]Liangzheng Sun is with the School of Instrumentation Science and Opto-electronics Engineering, Beijing Information Science and Technology University, Beijing 100192, China (e-mail: 2023030031@bistu.edu.cn).

[4]Yuanqing Wu is with the School of Intelligent Systems Engineering, Sun Yat-sen University, Guangzhou 510275, China (e-mail: wuyuanqing@mail.sysu.edu.cn).

suppressing side features. Furthermore, unlike existing in-block adapter designs (e.g., CricaVPR [6]), our side-branch adapters avoid interfering with the backbone's features. This architecture restricts gradient flow exclusively within small side branches, significantly reducing training overhead. Together, these methods enable reliable and resource-efficient retrieval under dynamic height variations. In summary, our main contributions are as follows:

1) We introduce a two-stage pipeline that coarsely estimates the UAV's height partition and performs retrieval in the corresponding sub-database, decoupling scale variance to reduce search costs without compromising overall accuracy.
2) We integrate two independent parallel branches of bypass adapters across the shared ViT blocks. This design prevents feature interference between height estimation and descriptor extraction while retaining the backbone's generalization capability with minimal parameters.
3) We propose a center-weighted feature masking strategy for the VPR branch. By prioritizing central features less prone to FOV truncation, it mitigates residual scale variations and improves retrieval reliability within the selected height partition.

We validate the approach on two challenging multi-height UAV datasets. Results demonstrate that our *HE-VPR* system drastically reduces memory usage (by up to 90%) while maintaining comparable retrieval performance against full-database baselines. Ablations verify the benefits of the height estimation pipeline, advancing height-aware VPR toward practical deployment on aerial platforms.

II. Related Works

A. Aerial VPR with Height Variation

Early aerial VPR demonstrated the feasibility of deep-learning-based matching [7], [8], [9]. Subsequent research advanced UAV-to-satellite retrieval via sophisticated feature aggregation [10], [11], [12], [13] or foundation models [14]. Despite efforts in changing environments [15], [16], most methods assume stable scale relationships or fixed flight heights.

Unlike ground-based localization [17], [18], aerial VPR must manage severe height variations. Brute-force multi-scale retrieval is computationally prohibitive for UAVs. While explicit height estimation can narrow search spaces, multi-view approaches [19] require continuous sequences, and monocular metric depth estimation (MMDE) models [1], [2], [3] often fail at significant heights due to weak nadir textures [4].

Shao et al. [4] addressed these variations via a two-stage classification and retrieval pipeline. However, their reliance on disjoint heavy networks incurs significant computational overhead. Building on this paradigm, we reformulate height estimation as a lightweight retrieval task within a shared-backbone architecture. This design maintains task independence and enables efficient sub-database selection while drastically reducing the resource footprint compared to previous multi-model solutions.

B. Adapters for Vision Transformers

Leveraging pre-trained foundation models (e.g., DINOv2 [5]) benefits aerial VPR, but full fine-tuning is computationally prohibitive. Adapters provide a parameter-efficient alternative by updating only small bottleneck modules while freezing the backbone [20], significantly reducing trainable parameters and memory footprint. Recently, CricaVPR [6] and SelaVPR [21] introduced adapters to the VPR domain to enhance cross-image correlation and semantic localization.

Architecturally, adapters are categorized by placement. *(i) In-block adapters* are sequentially inserted within Transformer blocks, typically after multi-head attention (MHA) or feed-forward neural network (FFN) sublayers, as adopted by CricaVPR and SelaVPR. While yielding strong single-task adaptation, they force the main data stream to carry task-specific modifications, making them suboptimal for multi-task scenarios. *(ii) Bypass (side-branch) adapters* operate in parallel to the main backbone, merging outputs via residual connections. Crucially for *HE-VPR*, bypass adapters allow multiple task-specific branches (height estimation and VPR extraction) to share a frozen backbone without feature interference. This high modularity enables our two-stage height-aware retrieval pipeline without the overhead of deploying separate heavy networks.

III. Methodology

In this paper, we focus on the aerial VPR problem under significant height variance. To address this challenge, we propose *HE-VPR*, a two-stage retrieval pipeline that explicitly decouples height inference from place recognition. As illustrated in Fig. 3, our framework leverages a shared frozen foundation model equipped with two independent, parallel adapter branches: the height estimation branch and the VPR branch. For clarity and alignment with the illustrations in Fig. 3, the entire sequence of bypass adapters comprising these two paths is denoted as the *HE adapter branch* and the *VPR adapter branch*, respectively. The *HE adapter branch*

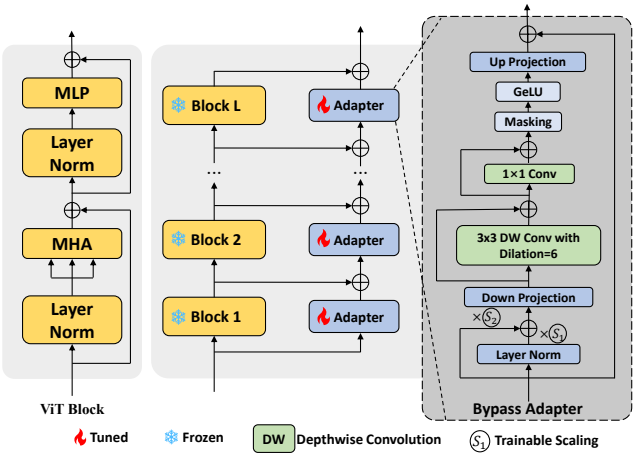


Fig. 2: Illustration of the adapter network in ViT.

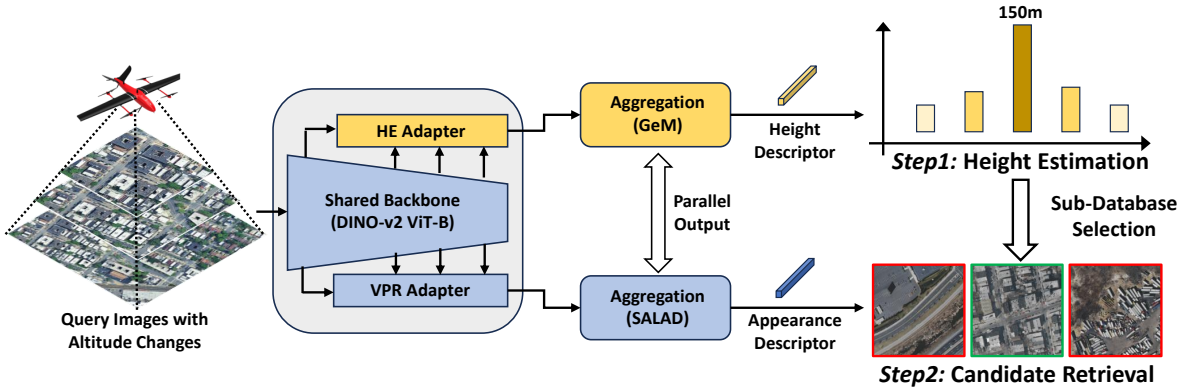


Fig. 3: **The proposed HE-VPR pipeline.** A height estimation branch is added for sub-database selection such that the proposed pipeline is robust to height variance. It requires only a single forward pass of the model for both parts.

extracts a coarse height descriptor to select the appropriate height-specific sub-database, while the *VPR adapter branch* extracts the appearance descriptor for the final place retrieval. The details of these components are formulated in the subsequent sections.

A. Preliminary

First, we provide a brief overview of the vision Transformer (ViT) and its adaptation mechanism for fine-tuning. The ViT slices a given image into N patches and projects them into D-dimensional embeddings, denoted as $x_p \in \mathbb{R}^{N\times D}$. A learnable class token $x_{\text{class}} \in \mathbb{R}^{1\times D}$ is concatenated with these embeddings, and the resulting sequence $x_0 = [x_{\text{class}}, x_p] \in \mathbb{R}^{(N+1)\times D}$ is fed into L cascaded Transformer blocks to generate a discriminative feature representation x_l (denoted as the output of the l-th Transformer block, where $l \in \{1, \ldots, L\}$). As shown in Fig. 2, each Transformer block consists of a MHA module, a multilayer perceptron (MLP), and layer normalization (LN).

To perform task-specific fine-tuning without updating the backbone, bypass adapter modules are placed in parallel with the Transformer blocks. Crucially, in our dual-branch architecture, each block l is paired with two separate adapters to process the height estimation and place recognition tasks independently. Let $t \in \{\text{he}, \text{vpr}\}$ denote the task index. The data flow within the adapter branch t at the l-th block is formulated as:

$$x_{l,t}^{(a)} = \text{Adapter}_t(x_{l-1} + x_{l-1,t}^{(a)}), \quad x_{0,t}^{(a)} = 0, \tag{1}$$

where $x_{l,t}^{(a)}$ and $x_{l-1,t}^{(a)}$ are the outputs of the task-specific adapter at the l-th and $(l-1)$-th blocks, respectively. In addition, x_{l-1} represents the frozen intermediate feature extracted from the shared main backbone.

B. Bypass Adapter

Following the design of Mona [22], we embed a bypass adapter for both the HE and VPR branches within each Transformer block. To ensure the computational efficiency of these bypass adapters, we employ only a single depth-wise convolution with a 3×3 kernel. We use dilated convolution to increase the kernel size, which ensures the adapter network has a larger receptive field while maintaining efficiency. Such a design also aligns with the large-scale feature information present in aerial imagery. Corresponding to the right subplot of Fig. 2, the overall calculation within the adapter can be formulated as follows:

$$x_{l,t}^{(a)} = x_0 + U\sigma\left(\mathcal{M}\left(f_{pw}\left(f_{dw}\left(D\left(s_1\|x_0\| + s_2 x_0\right)\right)\right)\right)\right), \tag{2}$$

where $x_0 = x_{l-1} + x_{l-1,t}^{(a)}$ is the input to the adapter, derived from the summation of the shared backbone feature x_{l-1} and the task-specific adapter feature $x_{l-1,t}^{(a)}$. D and U are the down-projection and up-projection linear layers, while σ denotes GeLU activation. In addition, f_{pw} and f_{dw} are the point-wise and depth-wise convolution operations containing shortcuts, respectively, and $\mathcal{M}$ is the masking strategy applied to the extracted features.

These bypass adapters are designed to operate exclusively within a parallel path, receiving outputs from preceding blocks without modifying the subsequent data flow in the main backbone. This architecture is a simple yet essential design, as it allows us to freeze the parameters of the pre-trained foundation model and exclusively train the adaptation networks. Furthermore, it enables the parallel branches to operate independently with isolated data streams. Leveraging this design, we construct two independent network branches dedicated to the HE and VPR tasks, respectively.

C. Height-Aware Aerial VPR

Traditional VPR methods designed for multi-height applications can only handle limited scale variations (as mentioned in Sec. II-A). When facing height variance in a large range, such as during takeoff and landing, existing VPR methods fail to maintain robustness. To address this, we propose *HE-VPR*, a novel paradigm for height-aware aerial VPR, as shown in Fig. 3. Following the standard VPR offline preparation process, we first construct a multi-level map database to accommodate flights at varying heights. Each height level corresponds to a specific sub-database. To establish a geometric relationship between the flight height and the map scale, the physical height ranges are mapped to the image's spatial footprint using camera intrinsics. This allows for a logical partitioning of the database:

$$\mathbf{D} = \underbrace{\{\mathbf{D}^1, \cdots, \overbrace{\mathbf{D}^l = \{\mathbf{d}_1^l, \mathbf{d}_1^l, \ldots, \mathbf{d}_n^l\}}^{n \text{ samples in } l\text{-th sub-database}}, \ldots, \mathbf{D}^L\}}_{L \text{ levels of subsets in the database}}, \quad (3)$$

where $\mathbf{D}^l$ and $\mathbf{d}_i^l$ represent the sub-databases and individual map samples, respectively. To ensure all flight heights are covered, the L levels of sub-databases are established based on the range of heights encountered. In a one-stage VPR pipeline, direct retrieval from the aggregated database $\mathbf{D}$ would result in significant computational and memory overhead, making edge-device implementation infeasible. Leveraging the previously independent adapter designs, we decompose this task into a two-step process: sub-database selection and retrieval, handled by two independent adapter branches, respectively.

For the height estimation branch, precise height values are not required, as height is only utilized for discrete sub-database selection. Therefore, we reformulate height estimation as a retrieval task, similar to VPR. We use a simple GeM [23] pooling layer to extract a height descriptor. A compact database for height retrieval is collected from map patches at different scales. The height information is obtained through a retrieval-based approach rather than direct regression. This enables more convenient generalization to new environments by changing the corresponding height database. Furthermore, since the database is expandable, we can leverage top-k candidates (e.g., top-5) to select multiple sub-databases, enhancing selection accuracy compared to direct depth estimation pipelines. The minimal overhead introduced by the compact height database allows us to maintain high retrieval efficiency, especially when compared to the case of a densely sampled full database.

In the VPR branch, we follow the classical feature extraction and aggregation pipeline to ensure retrieval performance. We use a modified Mona network [22] as the adapter and utilize SALAD [24] as the aggregation module. The extracted image descriptors are retrieved against the corresponding database, which is selected by the height estimation branch. Because only the simplest GeM layer is used for the height descriptor and the foundation model is shared by both branches, this approach only causes a slight increase in computational costs compared to a classical VPR approach. Two kinds of descriptors are extracted simultaneously and independently of each other, meaning they can be trained individually. Therefore, we can train these two branches with different datasets, the details of which will be given in Sec. IV-A.

D. Masked Feature Enhancement

Considering the height estimation stage, although we can obtain the correct sub-database selection, the discrete partitioning still causes a certain degree of scale difference between the query image and the referenced database image. As mentioned before, the design of one adapter branch is isolated and does not impact the other. Therefore, we can further design a masking strategy within the VPR adapter branch to provide better adaptability to these residual small-scale height changes, as shown in Fig. 2. Combined with the height estimation adapter branch, this design allows for accurate visual place recognition from coarse to fine, even under significant height variations.

At the VPR bypass, we add a masking mechanism based on the feature variance after the point-wise convolution, which can be represented by the following formulation:

$$\mathcal{M}_{i,j} = \exp\left(-\frac{(j-\frac{W}{2})^2 + (i-\frac{H}{2})^2}{2 \cdot (\frac{\max(H,W)}{2})^2} \cdot \text{Var}(x)\right), \quad (4)$$

where $\mathcal{M}_{i,j}$ denotes the feature mask value at feature position (i, j), and H, W are the height and width of the feature map. We calculate the feature variance of each channel, denoted as $\text{Var}(x)$. Subsequently, the features are multiplied element-wise with the mask $\mathcal{M}$ before being passed through the non-linear activation function σ.

As presented in Eq. 4, the feature mask is primarily determined by two factors: pixel position and the variance of the feature map. This design explicitly addresses the geometric realities of downward-facing cameras during height changes. Under significant height variations, the central region of the downward-facing camera's field of view remains relatively content-stable, merely undergoing scale changes. Conversely, the edge regions are highly susceptible to disappearing (truncation as height decreases) or reappearing. This formulation gives pixels closer to the image center higher activation values, mitigating the negative impact of unstable peripheral features. Furthermore, we assume that a higher feature variance indicates a higher flight height, where scale distortion effects are more pronounced. For such feature maps, we reduce the contribution of their edge pixels to the final result, focusing instead on the more stable central region. By using feature masking, we force the VPR adapter to focus on the central region of the feature map, making the retrieval process more robust to the remaining scale changes within the selected height partition.

IV. Experiments

A. Implementation Details

We use ViT-Base DINOv2 pre-trained model [5] as the foundation backbone and the two side branches are formulated as adapter branches as presented in Fig. 2. The resolution of the input image is 224×224 in both training and inference stages. The token dimension in the backbone is 768 and the feature dimension in the bypass adapters is 64. The VPR adapter branch and other evaluated methods are all retrained on the same dataset proposed by He et al. [25], which consists of satellite maps from multiple years. The HE adapter branch is trained on the same dataset as the VPR adapter branch by resizing the satellite images to various scales. As the two adapter branches both follow the metric learning pipeline, they share almost identical architectural configurations and parameter sizes. We train our models using the AdamW optimizer with the initial learning rate set as 0.00001 and the multi-similarity loss [26]. A training batch contains 32 classes with 2 images each. The models are trained for a sufficient number of epochs, and the weights

TABLE I: The statistics of GEStudio and MHFlight.

Dataset	Query Image	Height Database	VPR Database	Height Range (m)
GEStudio	1200	102	29768	100∼1200
MHFlight	1970	850	36400	200∼640

that achieve the highest performance on the validation set are selected. All the training experiments are deployed on an NVIDIA 4090 GPU with the same framework proposed in MixVPR [27] using PyTorch.

We evaluate the VPR performance with the commonly used Recall@N (R@N) metric, which is the percentage of queries that have the correct result among the N retrieved images. As the evaluation datasets contain various image scales, the distance threshold for correct place retrieval also varies. Therefore, we present the performance at different positive thresholds to evaluate the correct retrieval across various scales. Given that the HE adapter branch follows the retrieval pipeline, we also provide the evaluation with R@N. In order to evaluate the algorithm efficiency, we report the memory usage and the number of parameters. All the evaluations are conducted on an NVIDIA 4060 GPU.

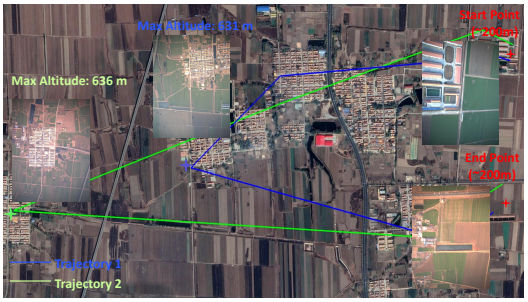
B. Evaluation Datasets

Since the proposed height estimation module is capable of accommodating height variance of several kilometers, the evaluation datasets should ideally contain a sufficiently wide range of height changes. Several commonly-used aerial VPR datasets such as University-1652 [28], SUES-200 [29], DenseUAV [30] only contain data with a height difference no more than two hundred meters and are therefore not enough for our experiments. Therefore, we provide two self-collected datasets with larger height variance that are more suitable for the evaluation of HE-VPR and the details are shown as follows:

1) GEStudio is a simulated dataset of drone imagery collected from Google Earth Studio, which can provide animated imagery content. We sample 1200 images from 100 locations in New York, U.S., with heights ranging from 100 to 1200 m. The images in this dataset feature significant variations in scale from near-ground to height over 1000 m, and Fig. 4a illustrates the sampling process at one of the selected locations. *2) MHFlight* (Multi-Height Flight) is a real-world dataset collected from drone flights. We collect this dataset in a rural village, with varying flight heights between 200 m and 600 m. In contrast to GEStudio dataset, which primarily features urban scenes, MHFlight dataset mostly covers agricultural areas. Another difference is that the heights of MHFlight dataset vary continuously, whereas the height in the previous dataset changes discretely. MHFlight dataset comprises 1970 images captured from two distinct flight trajectories, as shown in Fig. 4b.

These two datasets are both utilized in the evaluations of the HE and VPR adapter branches, with detailed statistics provided in Tab. I. Notably, as both of these datasets are unseen during training, they provide a direct validation of the model ability for zero-shot generalization. All of these query images are paired with their corresponding satellite maps, which are also collected from the Google Earth. Following the multi-level map database strategy mentioned before, we create a map database composed of map tiles of various sizes, with each group representing a 50-meter interval. The height information in the two evaluation datasets is converted into image widths using the intrinsic camera parameters, and then the images can match with a sub-database according to the width.

(a) GEStudio Dataset: urban area with buildings and roads.



(b) MHFlight Dataset: rural area with farmland and ponds.

Fig. 4: Overviews of two evaluation datasets.

TABLE II: Evaluation for the height estimation. We present the performance with height thresholds of 50m and 100m to define a correct retrieval. We compare with two MMDE methods, namely UniDepth V2 and Depth Anything (DA) v2. The best result is highlighted in bold.

Method	R@1	R@5	R@10	E_{avg}
	threshold at 50 m and 100 m (%/%)			(m)
GEStudio dataset				
UniDepth V2	1.25/8.50	N/A	N/A	470.83
DA V2	5.58/15.00	N/A	N/A	550.03
HE-VPR	**63.08/91.75**	**92.33/99.92**	**98.25/100.00**	**43.63**
MHFlight dataset				
UniDepth V2	0.00/1.37	N/A	N/A	213.24
DA V2	0.00/3.86	N/A	N/A	310.55
HE-VPR	**34.01/60.96**	**76.95/91.83**	**89.85/96.80**	**93.50**

C. HE Evaluation

In this section, we report the performance of the proposed height estimation adapter branch on the aforementioned two datasets and the comparison with two state-of-the-art MMDE methods. The range for a correct retrieval in height estimation is set to 50 meters and 100 meters. We present the results using standard retrieval metrics: R@1, R@5 and

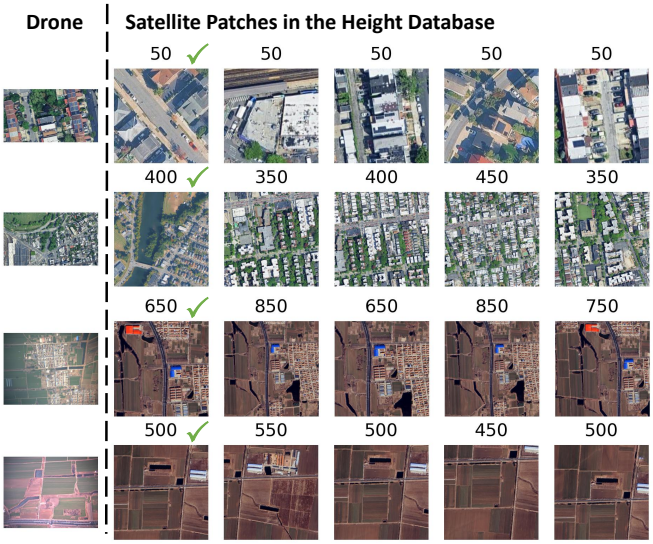


Fig. 5: Qualitative results for height estimation. The top two rows are from GEStudio dataset and the bottom two rows are from MHFlight dataset. The height label for top-5 database image is annotated at the top and all the samples are correctly retrieved at the top-1 candidate.

R@10. However, since MMDE methods directly estimate a single height value, they do not produce multiple ranked candidates. Therefore, we report only the R@1 metric for these methods and use N/A for the R@5 and R@10 metrics in the results. And we also compare the average estimation error (denoted as E_{avg}) for the evaluated methods. The evaluation results are shown in Tab. II.

Since traditional MMDE methods fail to consider the downwards perspective of UAV imagery, their depth estimation performance is almost entirely unacceptable on the two evaluation datasets. Our method, which obtains image height information in a retrieval manner, demonstrates stronger performance. The height estimation performs significantly well on GEStudio dataset, with the majority of images being correctly categorized. In contrast, the performance on the MHFlight dataset is relatively lower, where the farmland imagery is prevalent. However, given that the flight heights in this dataset are all above 200 m, a top-1 classification accuracy of 60.96% with a 100 m threshold is acceptable. Regarding the average height estimation error E_{avg}, we use the label of the top-1 candidate as the estimation result, and the errors are 43.63 m and 93.50 m respectively. Given that the height database is partitioned with a 50-m interval, this level of model performance is considered acceptable. Additionally, the role of height estimation differs from that of place recognition since top-5 or even top-10 candidates can be used to select sub-database as described in Sec. III-C. Therefore, such height estimation performance proves to be entirely sufficient for VPR in the following stage.

We also present the top-5 database images retrieved by HE-VPR for four query examples from two evaluated datasets, as shown in Fig. 5. As demonstrated by the examples, the proposed height estimation module is capable of recognizing the height changes of the captured samples and retrieve database images with similar heights. It also indicates that the HE adapter branch can be used for samples at both very high and relatively low heights. As shown in Fig. 5, the proposed HE adapter branch uses a height database that can be tailored to different datasets, which helps to ensure accurate height estimation. This offers a key advantage over MMDE methods, as our module can leverage knowledge from the database, rather than relying on direct height regression. By applying a suitable height database, the proposed HE adapter achieves strong generalization across a wide range of scenarios. To validate this approach, we conduct an experiment on the effect of using different height databases, with the results presented in Tab. III. Although the HE adapter branch does not strictly require a certain database, there is a notable decrease using a different database compared to the performance achieved by the database with similar appearance. Moreover, using a larger height database does not lead to better performance. This suggests that, unlike the VPR task which benefits from densely sampled database, height estimation can be negatively impacted by a large number of database samples. This is because the number of height labels is significantly smaller than in traditional VPR tasks, often no more than one hundred. Consequently, establishing an appropriately sized database is critical for this task.

D. VPR Evaluation

In this section, we compare the proposed scale-invariant VPR adapter branch on the proposed two datasets with a wide range of SOTA VPR methods. We select methods from both CNN-based architectures (GeM [23], Cosplace [31], MixVPR [27]) and ViT-based architectures (CricaVPR [6], SALAD [24]). To guarantee a fair comparison, all the methods are retrained on the same dataset and use appropriate inference parameters. In this part of the evaluation, we do not include the results of the HE adapter branch and only evaluate the VPR adapter branch based on the full database. Therefore, the retrieval database contains all the sub-databases for different height levels. As the two branches are isolated, we can independently evaluate the performance

TABLE III: Height estimation performance with different databases. In this evaluation, **G** and **M** denote denote databases sharing similar scene domains (e.g., urban and rural environments) with the GEStudio and MHFlight datasets, respectively. And the superscript * indicates a database that is larger than the original database.

Dataset	Database	R@1	R@5	R@10
GEStudio	**M**	36.50/54.25	73.33/91.08	91.58/98.33
	G	**63.08/91.75**	**92.33/99.92**	**98.25/100.00**
	G*	61.58/91.33	90.25/99.75	95.58/99.92
MHFlight	**G**	29.04/49.75	52.08/67.26	66.29/72.13
	M	**34.01/60.96**	**76.95/91.83**	**89.85/96.80**
	M*	30.00/50.36	70.66/88.58	82.44/95.13

TABLE IV: VPR comparison of different methods. We only use the VPR adapter branch for evaluation. Here we also present the performance under different thresholds with 100 m and 200 m. The best results are shown in bold.

Method	Backbone	Dim	Param. (MB)	GEStudio dataset			MHFlight dataset		
				R@1	R@5	R@10	R@1	R@5	R@10
GeM [23]	ResNet50	1024	8.5	20.08/24.00	35.75/42.33	43.08/51.67	37.31/63.81	61.88/75.69	69.34/78.88
CosPlace [31]	ResNet50	2048	27.7	47.92/51.50	61.42/66.50	66.17/71.83	55.89/83.35	67.61/84.62	72.99/84.87
MixVPR [27]	ResNet50	4096	10.9	50.75/53.92	61.08/65.67	66.00/70.92	**58.63**/84.47	71.83/84.82	77.26/85.03
CricaVPR [6]	DINOv2-B	10752	106.8	61.17/65.58	73.25/76.75	76.42/79.67	57.21/83.20	70.00/84.37	76.19/84.47
SALAD [24]	DINOv2-B	8448	88.0	63.42/67.00	73.17/76.67	76.33/79.75	53.15/83.45	67.16/85.13	72.49/85.23
HE-VPR	DINOv2-B	8448	90.6	**69.50/71.25**	**76.42/78.17**	**79.08/81.50**	57.61/**85.48**	**72.49/86.04**	**77.82/86.35**

of the proposed VPR model. In this part of the experiment, we select 100 m and 200 m as the positive thresholds. In addition, we evaluate the number of parameters for each method, and the results are presented in Tab. IV.

In summary, the proposed VPR adapter branch achieves outstanding performance, especially on the GEStudio dataset. MHFlight dataset presents a significant challenge as it is largely comprised of low-texture areas. Therefore, most methods fail to demonstrate strong performance. Nevertheless, our approach is still notable, achieving the best performance in the overall metrics. Also, the VPR adapter branch outperforms the original DINOv2-based SALAD model across all the evaluated datasets. This demonstrates that the adapter structure effectively improves the performance of VPR. In terms of the number of parameters, our method adds 2.6 MB of parameters by introducing inter-block adapters. The higher performance of MixVPR on the MHFlight dataset is partly due to its use of a higher image resolution (320×320), which is larger than the 224×224 resolution typically used by DINO-based methods. Finally, the proposed modular design also allows for seamless integration of future advanced methods, enabling us to update individual components based on different applications without affecting the adapter branches within the system.

TABLE V: Comparison between the standalone VPR adapter branch and the proposed HE-VPR system.

Method	GEStudio dataset			MHFlight dataset		
	R@1	R@5	R@10	R@1	R@5	R@10
Adapter(full)	69.50	76.42	**79.08**	**57.61**	72.49	**77.82**
HE-VPR(1)	57.25	66.50	70.00	49.14	68.07	74.06
HE-VPR(5)	69.92	76.17	78.67	56.80	**72.94**	77.72
HE-VPR(10)	**70.42**	**76.83**	79.00	57.41	71.93	77.46

TABLE VI: Comparison for memory usage and performance ratio of database.

Method	Full	HE-VPR(1)	HE-VPR(5)	HE-VPR(10)
GEStudio dataset				
Memory Usage (%)	100	**12.50**	38.71	58.51
Performance Ratio (%)	100	$86.11_{-13.89}$	$99.89_{-0.11}$	$\mathbf{100.56}_{+0.56}$
MHFlight dataset				
Memory Usage (%)	100	**7.14**	27.18	42.66
Performance Ratio (%)	100	$91.99_{-8.01}$	$\mathbf{99.78}_{-0.22}$	$99.46_{-0.54}$

E. HE-VPR Evaluation

We finally evaluate the overall HE-VPR system that integrates the height estimation adapter branch with the VPR adapter branch, with results summarized in Tab. V. By incorporating height estimation, retrieval is restricted to a relevant sub-database rather than the entire gallery. For comparison, we also evaluate the VPR adapter branch alone using full-database retrieval, denoted as *Adapter(full)*. As described in Sec. III-C, we further exploit not only the top-1 height estimate but also the top-5 and top-10 candidates, denoted as *HE-VPR(1)*, *HE-VPR(5)*, and *HE-VPR(10)*, respectively. All results are reported at a 100 m positive threshold for clarity. Without height estimation, a multi-level sub-database covering different height ranges is required; otherwise, accuracy drops sharply, as shown in the first row of Tab. V. Although a full-scale database contains all height information, it does not guarantee optimal performance in all scenarios. Height estimation enhances retrieval accuracy, but relying solely on the top-1 estimate can lead to performance loss due to estimation uncertainty. Selecting multiple top-k candidates for sub-database retrieval mitigates this issue and is a key advantage of the proposed HE adapter branch over direct regression approaches.

Our method leverages only a small subset of the database for retrieval, substantially reducing memory consumption. To quantify this, we report the memory footprint and retrieval performance of various methods in Tab. VI. The baseline method, denoted as *Full*, uses the VPR adapter to process the entire database and is assigned 100% memory usage. Since the number of sub-databases selected by the HE adapter branch varies per query, all reported values are averaged over the dataset. We also present the relative performance of each method compared to the baseline, where *Full Performance* is defined as the sum of R@1, R@5, and R@10 metrics from full-database retrieval. The *Performance Ratio* is computed as the ratio between the performance of our method and the baseline, with gains and losses indicated by green and red subscripts, respectively. Results on the GEStudio dataset show that as height estimation improves, the required database size decreases. Using only

top-1 candidates, our method achieves approximately 90% of the baseline performance with just 10% of the memory. For optimal performance, selecting top-5 or top-10 candidates further boosts accuracy while still requiring significantly less memory — less than half of what the baseline consumes. These findings highlight the efficiency advantage of HE-VPR in both retrieval quality and resource usage.

V. Conclusion

In this paper, we present HE-VPR, an efficient system designed for UAVs operating under dynamic and varying heights. By deploying two independent parallel bypass adaptation branches, our framework effectively decouples height estimation and place recognition into a two-stage pipeline while sharing a single frozen foundation backbone. This modular architecture successfully mitigates severe scale variations through retrieval-based sub-database selection and center-weighted feature masking, significantly reducing memory consumption and search space without sacrificing retrieval accuracy. While the current reliance on discrete height partitions presents a limitation when dealing with continuous height variations, the proposed parameter-efficient design ensures flexibility and provides a robust foundation for future continuous height-adaptive VPR research on aerial platforms.

References

[1] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao, "Depth Anything V2," in *NeurIPS*, 2024.

[2] L. Piccinelli, C. Sakaridis, Y.-H. Yang, M. Segu, S. Li, W. Abbeloos, and L. Van Gool, "UniDepthV2: Universal monocular metric depth estimation made simpler," *arXiv e-prints*, Feb. 2025.

[3] S. Farooq Bhat, R. Birkl, D. Wofk, P. Wonka, and M. Müller, "ZoeDepth: Zero-shot transfer by combining relative and metric depth," *arXiv e-prints*, Feb. 2023.

[4] X. Shao, M. He, C. Li, L. Sun, and Z. Meng, "Altitude-Aware Visual Place Recognition in Top-Down View," *arXiv e-prints*, Feb. 2026.

[5] 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," *Transactions on Machine Learning Research*, 2024.

[6] F. Lu, X. Lan, L. Zhang, D. Jiang, Y. Wang, and C. Yuan, "CricaVPR: Cross-image correlation-aware representation learning for visual place recognition," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 16 772–16 782.

[7] K. Amer, M. Samy, R. ElHakim, M. Shaker, and M. ElHelw, "Convolutional neural network-based deep urban signatures with application to drone localization," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2017, pp. 2138–2145.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv e-prints*, Sep. 2014.

[9] B. Patel, T. D. Barfoot, and A. P. Schoellig, "Visual localization with google earth images for robust global pose estimation of UAVs," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 6491–6497.

[10] X. Meng, W. Guo, K. Zhou, T. Sun, L. Deng, S. Yu, and Y. Feng, "AirGeoNet: A map-guided visual geo-localization approach for aerial vehicles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–14, 2024.

[11] C. Liu, S. Peng, S. Li, H. Qiu, Y. Xia, Z. Li, and L. Zhao, "A novel EAGLe framework for robust UAV-view geo-localization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 63, pp. 1–17, 2025.

[12] B. Sun, G. Liu, and Y. Yuan, "F3-Net: Multiview scene matching for drone-based geo-localization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–11, 2023.

[13] X. Guo, H. Peng, J. Hu, H. Bao, and G. Zhang, "From satellite to ground: Satellite assisted visual localization with cross-view semantic matching," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 3977–3983.

[14] N. Keetha, A. Mishra, J. Karhade, K. M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg, "AnyLoc: Towards universal visual place recognition," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1286–1293, 2024.

[15] M.-M. Gurgu, J. P. Queralta, and T. Westerlund, "Vision-based GNSS-free localization for UAVs in the wild," in *2022 7th International Conference on Mechanical Engineering and Robotics Research (ICMERR)*, 2022, pp. 7–12.

[16] T. Lømo, J. Torresen, M. Kolberg, and R. Maffei, "Multi map visual localization for unmanned aerial vehicles," *IEEE Robotics and Automation Letters*, vol. 10, no. 2, pp. 1353–1360, 2025.

[17] C. Toft, W. Maddern, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, T. Pajdla, F. Kahl, and T. Sattler, "Long-term visual localization revisited," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 4, pp. 2074–2088, 2022.

[18] S. Lynen, B. Zeisl, D. Aiger, M. Bosse, J. Hesch, M. Pollefeys, R. Siegwart, and T. Sattler, "Large-scale, real-time visual-inertial localization revisited," *Int. J. Rob. Res.*, vol. 39, no. 9, pp. 1061–1084, Aug. 2020.

[19] M. Khurshid, M. Shahzad, H. A. Khattak, M. I. Malik, and M. M. Fraz, "Vision-based 3-D localization of UAV using deep image matching," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 12 020–12 030, 2024.

[20] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for NLP," *arXiv e-prints*, Feb. 2019.

[21] F. Lu, L. Zhang, X. Lan, S. Dong, Y. Wang, and C. Yuan, "Towards seamless adaptation of pre-trained models for visual place recognition," in *The Twelfth International Conference on Learning Representations*, 2024.

[22] D. Yin, L. Hu, B. Li, Y. Zhang, and X. Yang, "5% $>$ 100%: Breaking performance shackles of full fine-tuning on visual recognition tasks," in *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 20 071–20 081.

[23] F. Radenović, G. Tolias, and O. Chum, "Fine-tuning CNN image retrieval with no human annotation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1655–1668, 2019.

[24] S. Izquierdo and J. Civera, "Optimal transport aggregation for visual place recognition," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 17 658–17 668.

[25] M. He, J. Liu, P. Gu, and Z. Meng, "Leveraging map retrieval and alignment for robust UAV visual geo-localization," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–13, 2024.

[26] A. Ali-bey, B. Chaib-draa, and P. Giguère, "GSV-Cities: Toward appropriate supervised visual place recognition," *Neurocomputing*, vol. 513, pp. 194–203, 2022.

[27] A. Ali-Bey, B. Chaib-Draa, and P. Giguére, "MixVPR: Feature mixing for visual place recognition," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 2997–3006.

[28] Z. Zheng, Y. Wei, and Y. Yang, "University-1652: A multi-view multi-source benchmark for drone-based geo-localization," in *Proceedings of the 28th ACM International Conference on Multimedia*, ser. MM '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1395–1403.

[29] R. Zhu, L. Yin, M. Yang, F. Wu, Y. Yang, and W. Hu, "SUES-200: A multi-height multi-scene cross-view image benchmark across drone and satellite," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 9, pp. 4825–4839, 2023.

[30] M. Dai, E. Zheng, Z. Feng, L. Qi, J. Zhuang, and W. Yang, "Vision-based UAV self-positioning in low-altitude urban environments," *IEEE Transactions on Image Processing*, vol. 33, pp. 493–508, 2024.

[31] G. Berton, C. Masone, and B. Caputo, "Rethinking visual geo-localization for large-scale applications," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 4868–4878.