

GeoRanker: Distance-Aware Ranking for Worldwide Image Geolocalization

Pengyue Jia^{1,2}, Seongheon Park², Song Gao², Xiangyu Zhao¹, Yixuan Li²,

¹Department of Data Science, City University of Hong Kong,

²Department of Computer Sciences, University of Wisconsin-Madison

jia.pengyue@my.cityu.edu.hk, sharonli@cs.wisc.edu

Abstract

Worldwide image geolocalization—the task of predicting GPS coordinates from images taken anywhere on Earth—poses a fundamental challenge due to the vast diversity in visual content across regions. While recent approaches adopt a two-stage pipeline of retrieving candidates and selecting the best match, they typically rely on simplistic similarity heuristics and point-wise supervision, failing to model spatial relationships among candidates. In this paper, we propose **GeoRanker**, a distance-aware ranking framework that leverages large vision-language models to jointly encode query–candidate interactions and predict geographic proximity. In addition, we introduce a *multi-order distance loss* that ranks both absolute and relative distances, enabling the model to reason over structured spatial relationships. To support this, we curate GeoRanking, the first dataset explicitly designed for geographic ranking tasks with multimodal candidate information. GeoRanker achieves state-of-the-art results on two well-established benchmarks (IM2GPS3K and YFCC4K), significantly outperforming current best methods.

1 Introduction

Worldwide geolocalization [1, 2] refers to the task of predicting the GPS coordinates of images captured anywhere on Earth. Unlike approaches constrained to specific cities or regions [3, 4, 5, 6, 7], global geolocalization poses significantly greater challenges due to the immense diversity in visual content, ranging from natural landscapes and climatic variations to architectural differences and cultural markers [8, 9, 10]. Despite these complexities, accurate global geolocalization holds broad practical relevance, with a wide range of applications including criminal investigations [11], navigation systems [12], and environmental monitoring [13].

Recent state-of-the-art approaches [8, 9, 10, 14] typically follow a two-stage pipeline: (1) retrieving and generating a set of candidates based on a global database, and (2) selecting the top match as the predicted geolocation. As shown in Figure 1, although the current SOTA model (G3 [14], which ranks candidates via GPS location–image embedding similarity) achieves 16.7% accuracy at 1km error threshold on IM2GPS3K, better candidates often exist among the top-k retrieved results. This suggests that one can retrieve reasonably high-quality candidates, yet the final prediction accuracy hinges on the second stage—the model’s ability to compare spatial relevance and select the most plausible candidate. Currently, the candidate selection is often limited by naïve heuristics such as cosine similarity [8, 14], which generally encode the query image and candidates *independently*, without modeling their mutual spatial relationships or allowing rich interactions. As a result, these methods frequently struggle to distinguish between

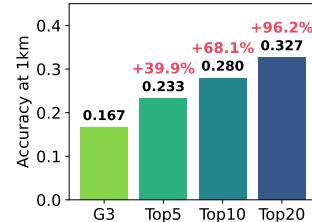


Figure 1: Accuracy at 1km error threshold for G3 (Current SOTA) vs. the best candidate within top-k retrieved results.

visually similar yet geographically distant scenes. Furthermore, existing training objectives primarily focus on point-wise similarity between individual images and locations [8, 9, 15], overlooking the rich spatial relationships among candidates—such as the spatial dependence (i.e., Tobler’s First Law of Geography [16]) and relative distances between them—which are crucial for geolocalization.

To address these limitations, we propose **GeoRanker**, a *distance-aware ranking framework* designed to model spatial relationships among candidate locations. Rather than relying on independent similarity scores, GeoRanker models the interaction between the query image and each candidate through a large vision-language model (LVLM), which captures rich spatial semantics via cross-modal alignment, and learns a scalar distance score that reflects their geographic proximity. Central to our approach is a multi-order distance optimization objective that ranks not only the absolute distances between the query and individual candidates (first-order supervision), but also the relative differences between candidate distances (second-order supervision). This formulation allows the model to learn both which candidate is closest and how much closer it is compared to others, capturing rich spatial structure that naïve heuristics overlook. Through this design, GeoRanker transforms the geolocalization task from one of isolated similarity matching to one of structured spatial reasoning.

To support this training paradigm, we construct GeoRanking, a new dataset that provides spatially diverse candidate sets for each query. Each candidate is annotated with GPS coordinates, textual descriptions (e.g., city, country), and image data. To the best of our knowledge, this is the first ranking dataset specifically designed for modeling spatial relationships among geographic entities. We believe this effort will significantly contribute to advancing research in related domains. We validate the effectiveness of GeoRanker through extensive experiments on two widely used benchmarks: IM2GPS3K [17] and YFCC4K [18]. GeoRanker achieves *state-of-the-art* performance across all geographic thresholds. For example, on IM2GPS3K, it improves street-level (1km) accuracy by **+12.9%** over the current best method [14], and on YFCC4K, it yields an **+37.3%** improvement at the same threshold. Our model also consistently outperforms existing approaches at coarser scales (25km, 200km, 750km, 2500km), highlighting its robustness across granularities. Ablation studies confirm that both components of our multi-order distance loss—first-order and second-order supervision—contribute to improved accuracy. We also conduct comprehensive ablations to understand the impact of various hyperparameter choices, leading to an improved understanding of our framework. Our key contributions are summarized as follows:

1. We introduce GeoRanker, a distance-aware ranking framework that models spatial relationships among candidate locations using a multi-order distance loss and large vision-language models.
2. We construct GeoRanking, the first dataset tailored for spatial ranking tasks, with rich multimodal annotations spanning GPS coordinates, textual descriptions, and image data—facilitating future research in related fields.
3. We achieve state-of-the-art performance on two well-established public geolocalization benchmarks, with substantial gains at fine-grained localization levels, and demonstrate the effectiveness of our approach through comprehensive ablations.

2 Related Work

Image Geolocalization. Worldwide geolocalization lies at the intersection of geography and computer vision, and is a core topic in GeoAI [19, 20, 21] and spatial data mining [22, 23]. Existing methods for worldwide geolocalization can be grouped into three main categories: classification-based, retrieval-based, and RAG-based approaches. (1) *Classification-based* methods [24, 15, 25, 26, 9] approach the task by partitioning the Earth’s surface into discrete Geo-grids and predicting the index of the grid that contains the image location. The final output is typically the center coordinate of the predicted grid. While these methods offer scalability, they may incur large errors when the true location lies far from the grid center, even if the grid prediction is correct. (2) *Retrieval-based* methods [27, 28, 29, 30, 31, 32] cast geolocalization as a similarity search problem. These methods either use a database of geotagged images [33, 27, 34, 35, 36] or a gallery of GPS points [8], returning the coordinates of the most similar entries to the input image as the prediction. However, these methods fail to capture the complex spatial relationships between the query image and candidate locations, making it difficult to reliably identify the most accurate match from the candidate pool. (3) *RAG-based* methods [10, 14] first retrieve a set of candidate locations similar to the query image from the database, then construct a prompt that integrates both the query and candidate information.

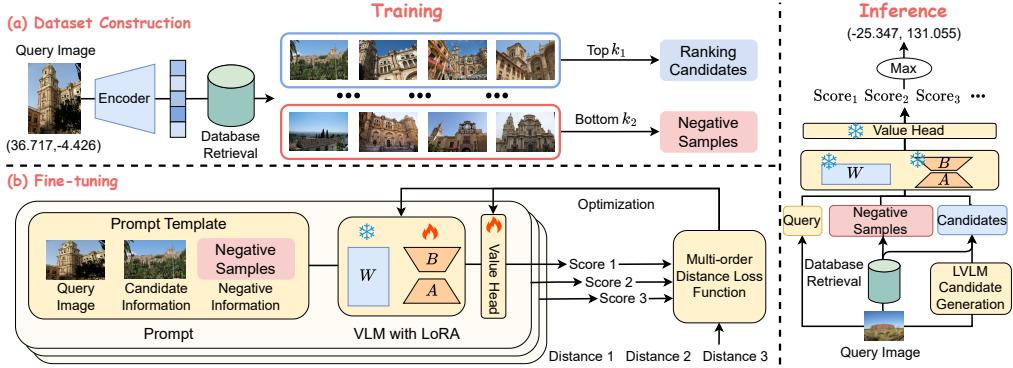


Figure 2: Overview of the Distance-aware Ranking framework–GeoRanker.

This prompt is passed to an LVLM to generate a plausible GPS location. In contrast to the above approaches, our proposed *Distance-Aware Ranking* framework, GeoRanker, focuses specifically on the candidate ranking stage. By explicitly modeling the complex spatial relationships between the query image and candidate geographic entities, an aspect overlooked by prior work, our approach offers more reliable candidate selection and leads to improved geolocalization performance at the global scale.

Learning to Rank. Learning-to-rank [37] (LTR) is a fundamental research direction in information retrieval [38] and recommender systems [39], primarily used to train ranking models that refine the order of retrieved candidates based on a given query [40]. Depending on their modeling and optimization strategies, LTR methods are typically categorized into three types: pointwise, pairwise, and listwise approaches. Pointwise methods [41, 42] take the ranking task as a regression or classification problem by assigning a relevance score or label to each query–candidate pair independently. This approach is simple and straightforward, yet overlooks the relative relationships among candidates. Pairwise methods [43, 44, 45] model the relative preferences between pairs of candidates for the same query. Pairwise methods encourage the model to assign higher scores to positive candidates while penalizing negative ones, learning the relative preferences between different candidates. Listwise methods [46, 47, 37, 48] can be seen as an extension of pairwise approaches, as they consider the entire list of candidates associated with a query and optimize a loss function that directly reflects the overall quality of the ranking. Our approach builds on the LTR foundation but adapts it to spatial ranking by explicitly modeling and optimizing distance-aware relationships between candidates.

3 Methodology

In this section, we introduce GeoRanker, a *Distance-aware Ranking* method for geolocalization. An overview of the framework is illustrated in Figure 2, which consists of two main phases: training and inference. The training phase begins with dataset construction, detailed in Section 3.1, where we describe how the GeoRanking dataset is built to support the training of GeoRanker. We then present the model architecture and optimization strategy of GeoRanker in Section 3.2 and Section 3.3. Finally, during inference, GeoRanker selects the most appropriate candidate as the prediction by scoring the spatial relationships between the query image and a set of candidates (Section 3.4).

3.1 GeoRanking Dataset Construction

Database. Following prior work [14], we adopt the MP16-Pro multimodal dataset [14] as our database and encode each sample into vectors. Each candidate entry includes GPS, textual descriptions (city, country, etc.), and image data. In a candidate database $\mathcal{C} = \{c_1, c_2, c_3, \dots, c_M\}$, each candidate c_m is encoded into a feature vector $\mathbf{v}_{c_m} = \text{concat}(\text{Encoder}_c^{\text{gps}}(c_m^{\text{gps}}), \text{Encoder}_c^{\text{text}}(c_m^{\text{text}}), \text{Encoder}_c^{\text{img}}(c_m^{\text{img}}))$, where c_m^{gps} , c_m^{text} , c_m^{img} represent the GPS, textual, and image modalities of the m -th candidate and $\text{Encoder}_c^{\text{gps}}$, $\text{Encoder}_c^{\text{text}}$, $\text{Encoder}_c^{\text{img}}$ are their corresponding modality-specific encoders.

Retrieval. Since the input query image q is a single modality (image), we design its representation to be compatible with the multimodal candidate vectors for similarity matching. The query image is encoded as: $\mathbf{v}_q = \text{concat}(f_{\text{img} \rightarrow \text{gps}}(\text{Encoder}^{\text{img}}(q)), f_{\text{img} \rightarrow \text{text}}(\text{Encoder}^{\text{img}}(q)), \text{Encoder}^{\text{img}}(q))$, where $f_{\text{img} \rightarrow \text{gps}}(\cdot)$ and $f_{\text{img} \rightarrow \text{text}}(\cdot)$ are adapter layers that project the visual features into the GPS and textual embedding spaces, respectively. These adapters, along with the GPS encoder, are trained using an InfoNCE loss [49] to align the query and candidate representations, as in G3 [14]. The remaining encoders are initialized with pretrained weights and kept frozen during training. To retrieve candidates for the query image, we compute the cosine similarity between \mathbf{v}_q and each candidate’s representation \mathbf{v}_{c_m} , and select the top- N candidates with the highest similarity scores to form the candidate set:

$$\mathcal{C}' = \{c'_1, c'_2, \dots, c'_N \mid \text{sim}(q, c'_1) \geq \text{sim}(q, c'_2) \geq \dots \geq \text{sim}(q, c'_N)\},$$

where the similarity function is defined as $\text{sim}(q, c_m) = (\mathbf{v}_q \cdot \mathbf{v}_{c_m}) / (\|\mathbf{v}_q\| \cdot \|\mathbf{v}_{c_m}\|)$. The query image, along with the retrieved candidates will be used for training the GeoRanker, which we described next.

3.2 GeoRanker

Figure 2 (b) shows the overview of GeoRanker. Existing methods [8, 14] typically model the query image and candidate geographic entities separately, embedding them into a shared representation space via independent encoders. The final prediction is based on similarity scores between these representations. However, such designs fail to capture the rich spatial interactions between the query and candidates, resulting in a decoupled modeling process that limits the accuracy of worldwide geolocation. To address this issue, we propose GeoRanker, a distance-aware ranking model designed to capture the spatial relationships between query–candidate pairs. Specifically, the query and candidates are assembled into a prompt following a predefined template. These inputs are then processed by an LVLM to model the complex interactions between the query and candidate. Finally, a linear value head maps the hidden states to a scalar score that reflects the geographic distance between the query image and the candidate location.

GeoRanking dataset and prompt construction. To support the ranking model training, we select the top- k_1 candidates from the candidate set \mathcal{C}' as ranking candidates, denoted by $\mathcal{C}_{\text{rc}} = \{c'_1, c'_2, \dots, c'_{k_1}\}$. We then take the last k_2 candidates in \mathcal{C}' to form the negative set \mathcal{C}_{neg} , which provides additional contextual diversity and helps the model understand the relative relevance of candidate locations. Thus, each sample in the GeoRanking dataset is represented as a triplet: $\{q, \mathcal{C}_{\text{rc}}, \mathcal{C}_{\text{neg}}\}$, where q is the query image, \mathcal{C}_{rc} contains the candidates to be ranked, and \mathcal{C}_{neg} provides hard negatives to enhance ranking discrimination. Each triplet is formatted using a structured prompt template:

{query image} How far is this place from latitude: {candidate latitude}, longitude: {candidate longitude}, {candidate textual descriptions}, {candidate image}? Negative examples: {negative information}.

The construction process can be formalized as: $\mathbf{x} = \text{Prompt}(q, \mathcal{C}_{\text{rc}}, \mathcal{C}_{\text{neg}}, p)$, where p denotes the prompt template. Note that, to reduce GPU memory consumption, we represent negative samples using only their textual GPS coordinates and textual descriptions.

Model architecture. The constructed input \mathbf{x} is fed into LVLM to encode both visual and textual modalities and to capture the spatial interactions between the query and candidate. To enhance the model’s representation capacity while maintaining training efficiency, we insert LoRA (Low-Rank Adaptation) [50] modules into the intermediate layers of the LVLM backbone during training.

We use the hidden states corresponding to the final position token as the joint representation of the input. A lightweight value head, implemented as a single linear layer without bias, is then applied to map this representation to a scalar score. This score serves as the estimated geographic distance between the query image and the candidate location:

$$s = \mathbf{w}^\top \mathbf{h}_{\text{final}}, \quad \text{where } \mathbf{h}_{\text{final}} = \text{LVLM}(\mathbf{x})_{[-1]} \tag{1}$$

where $s \in \mathbb{R}$ denotes the final score, $\mathbf{w} \in \mathbb{R}^{\text{dim}}$ and $\mathbf{h}_{\text{final}} \in \mathbb{R}^{\text{dim}}$ are the weight matrix of the value head and the final position token’s representation, dim is the dimension of the last hidden states. In addition, $\text{LVLM}(\cdot)$ denotes the large vision-language model used to encode the input \mathbf{x} .

3.3 GeoRanker: Optimization with Multi-Order Distance Objective

Existing geolocalization training methods typically focus on point-wise image-to-location similarity, without modeling the spatial relationships among candidate locations. To address this limitation, we propose a multi-order distance optimization objective to train GeoRanker. Our objective incorporates both the first-order distances between the query and each candidate, and the second-order relationships, defined as the relative differences between first-order distances, to guide the model during training.

First-order distance loss. We optimize the first-order distance ranking using a partial Plackett-Luce (PL) loss [51, 52]. Given k_1 candidates with predicted scores $\{s_1, s_2, \dots, s_{k_1}\}$, we first sort the corresponding geodesic distances $\{d_1, d_2, \dots, d_{k_1}\}$ in ascending order to obtain an index permutation π such that $d_{\pi(1)} < d_{\pi(2)} < \dots < d_{\pi(k_1)}$. We then use the reordered scores $\{s_{\pi(1)}, s_{\pi(2)}, \dots, s_{\pi(k_1)}\}$ to compute the loss. Let $K^{(1)} \leq k_1$ be a hyperparameter controlling how many top-ranked candidates are included in the objective. For each sample, the partial Plackett-Luce loss is defined as:

$$\mathcal{L}_{\text{PL}}^{(1)} = -\frac{1}{K^{(1)}} \sum_{i=1}^{K^{(1)}} \log \frac{\exp(s_{\pi(i)})}{\sum_{j=i}^{k_1} \exp(s_{\pi(j)})} \quad (2)$$

Second-order distance loss. To capture the relative spatial differences among candidates, we introduce a second-order distance loss based on pairwise distance gaps. This objective supervises the *ranking of first-order distance differences*, encouraging the model to assign higher score differences to candidate pairs that are more distant in geolocation. Specifically, we first compute all pairwise first-order differences in distances and predicted scores:

$$\Delta d_{i,j} = d_{\pi(i)} - d_{\pi(j)}, \quad \Delta s_{i,j} = s_{\pi(i)} - s_{\pi(j)}, \quad \text{for } 1 \leq i < j \leq k_1 \quad (3)$$

This results in $P = \frac{k_1(k_1-1)}{2}$ pairs. We sort the distance differences $\Delta d_{i,j}$ in ascending order (so larger spatial gaps appear earlier), and apply the same permutation to the score differences $\Delta s_{i,j}$, resulting in an ordered sequence $\Delta s_{(1)}, \dots, \Delta s_{(P)}$.

Let $K^{(2)}$ be a hyperparameter that specifies the number of top-ranked pairs included in the loss. We define $K^{(2)} = \frac{\lfloor (k_1-1)+(k_1-K^{(1)}) \rfloor \times K^{(1)}}{2}$, which ensures that the second-order loss focuses on candidate pairs where at least one candidate is involved in the first-order loss computation. The second-order partial PL loss is then computed as:

$$\mathcal{L}_{\text{PL}}^{(2)} = -\frac{1}{K^{(2)}} \sum_{i=1}^{K^{(2)}} \log \frac{\exp(\Delta s_{(i)})}{\sum_{j=i}^P \exp(\Delta s_{(j)})} \quad (4)$$

This formulation encourages the model to preserve the ordering of spatial gaps in the score space, so that larger geographic differences lead to larger score gaps.

Joint optimization. We jointly optimize the model with both the first-order and second-order objectives. The total loss is defined as a weighted sum of the two components:

$$\mathcal{L}_{\text{total}} = \lambda \cdot \mathcal{L}_{\text{PL}}^{(1)} + (1 - \lambda) \cdot \mathcal{L}_{\text{PL}}^{(2)} \quad (5)$$

where λ is the weighting coefficient that balances the contribution of the first-order and second-order distance losses, respectively. We will ablate the impact of key hyperparameters in Section 4.3.

3.4 Inference

During inference, GeoRanker integrates both retrieved candidates from a database and generated candidates from a LVLM, following prior work [10, 14]. Given a query image q , we first retrieve a set of candidates \mathcal{C}_r and collect contextual negative samples \mathcal{C}_{neg} . Simultaneously, the query q is passed through an LVLM to generate a new set of candidates \mathcal{C}_g , referred to as generated candidates. The prompt for generating candidates is detailed in Appendix A. We then form query–candidate pairs by combining q with each candidate $c \in \mathcal{C}_r \cup \mathcal{C}_g$, and feed these inputs into GeoRanker to obtain a set of distance scores:

$$s_c = \text{GeoRanker}(q, c), \quad \forall c \in \mathcal{C}_r \cup \mathcal{C}_g \quad (6)$$

Finally, we select the candidate with the highest score and use its GPS coordinates as the prediction:

$$\hat{c} = \arg \max_{c \in \mathcal{C}_r \cup \mathcal{C}_g} s_c \quad (7)$$

It is worth noting that the generated candidates typically lack additional modalities such as textual descriptions and images. As a result, we use only their GPS coordinates during inference.

4 Experiments

4.1 Setup

Dataset and evaluation metrics. We use MP16-Pro from prior work [14] as our database in constructing the GeoRanking dataset. For evaluation, we follow previous work [8, 10, 14] and assess performance on two widely used public benchmarks IM2GPS3K [17] and YFCC4K [18]. The evaluation metric reports the percentage of predictions whose geodesic distance to the ground-truth coordinates falls within a set of thresholds: 1km, 25km, 200km, 750km, and 2500km.

GeoRanking dataset. In this work, we construct the first ranking dataset for modeling distances between geographic entities. Specifically, for each query image, we retrieve a set of candidates based on embedding similarity. Each candidate is associated with GPS coordinates, textual descriptions (e.g., city, country), and image data. These candidates serve as input for subsequent ranking models. In total, we construct **100k samples**, resulting in **2 million query–candidate pairs**. By releasing this dataset, we aim to support progress in geolocalization and related research areas such as GeoAI, information retrieval, and LVLM. Example entries are provided in the **Appendix B** for reference.

Implementation details. During training, we retrieve 20 candidates from the database for each query. The top-7 are used as retrieval candidates, while the bottom-5 serve as negative samples. The vision encoder and text encoder are pretrained models from CLIP [53]. The GPS encoder is initialized with weights from GeoCLIP [8] and then fine-tuned. We use Qwen2-VL-7b-Instruct¹ as the LVLM backbone in GeoRanker. For LoRA fine-tuning, we target the `q_proj`, `k_proj`, and `v_proj` modules, with a rank of 16, scaling factor of 32, and LoRA dropout of 0.05. GeoRanker is fine-tuned with AdamW [54] optimizer with a learning rate of 1e-4, a batch size of 4, and for 1 epoch. For joint optimization, we set the weighting coefficient $\lambda = 0.7$, and $K^{(1)} = 1$. All experiments are conducted using Pytorch on 4 NVIDIA L40S GPUs. During inference, following [10, 14], we use GPT4V² as the LVLM for candidate generation. We use $|C_r| = 12$ retrieved candidates and $|C_g| = 3$ generated candidates for IM2GPS3K and $|C_r| = 14$, $|C_g| = 5$ for YFCC4K. Additional details regarding the training environment and runtime are provided in **Appendix C**.

Baselines. To evaluate the effectiveness of our approach, we conduct comprehensive experiments and compare it against 11 baselines: [L]kNN, sigma=4 [1], PlaNet [15], CPlaNet [15], ISNs [55], Translocator [25], GeoDecoder [26], GeoCLIP [8], Img2Loc [10], PIGEON [9], G3 [14], including the state-of-the-art. A detailed description of each baseline is provided in **Appendix D**.

4.2 Main Results

As shown in Table 1, GeoRanker achieves state-of-the-art performance across all evaluation thresholds. For example, on IM2GPS3K, it improves the most challenging street-level accuracy by **12.9%** over the best baseline G3, and on YFCC4K, it achieves an **37.3%** relative gain at the same threshold. Among the baselines, GeoCLIP [8], Img2Loc [10], PIGEON [9], and G3 [14] exhibit relatively strong performance due to classification-based methods are limited by systemic biases from fixed candidate grids. Compared to these stronger baselines, our method GeoRanker achieves superior results by explicitly modeling the spatial relationship between each query–candidate pair using a multi-order distance optimization objective. This enables the model to accurately identify the geographically closest candidate as the prediction, further enhancing geolocalization accuracy. In summary, our approach achieves state-of-the-art performance across all datasets and metrics, demonstrating its effectiveness and superiority. Furthermore, **Appendix E** presents representative examples across various error thresholds to offer intuitive insights into the distribution of query images at different localization accuracies.

4.3 Ablation Study

To better understand the contribution of each component, we conduct ablation studies by systematically varying key modules of our approach. (1) **w/o** $\mathcal{L}_{PL}^{(2)}$. Our method without second-order distance loss in training. (2) **w/o** \mathcal{L}_{neg} . Our method without negative information in

¹<https://huggingface.co/Qwen/Qwen2-VL-7B-Instruct>

²<https://openai.com/>

Table 1: **Main results** on IM2GPS3K and YFCC4K. For all metrics, higher is better. The best-performing results are highlighted in **bold**, while the second-best results are underlined. Δ represents the relative improvement of our method over the best baseline.

Methods		IM2GPS3K					YFCC4K				
		Street 1km	City 25km	Region 200km	Country 750km	Continent 2500km	Street 1km	City 25km	Region 200km	Country 750km	Continent 2500km
[L]kNN, sigma=4 [1]	ICCV'17	7.2	19.4	26.9	38.9	55.9	2.3	5.7	11	23.5	42
PlaNet [24]	ECCV'16	8.5	24.8	34.3	48.4	64.6	5.6	14.3	22.2	36.4	55.8
CPlaNet [15]	ECCV'18	10.2	26.5	34.6	48.6	64.6	7.9	14.8	21.9	36.4	55.5
ISNs [55]	ECCV'18	10.5	28	36.6	49.7	66	6.5	16.2	23.8	37.4	55
Translocator [25]	ECCV'22	11.8	31.1	46.7	58.9	80.1	8.4	18.6	27	41.1	60.4
GeoDecoder [26]	ICCV'23	12.8	33.5	45.9	61	76.1	10.3	24.4	33.9	50	68.7
GeoCLIP [8]	NeurIPS'23	14.11	34.47	50.65	69.67	83.82	9.59	19.31	32.63	55	74.69
Img2Loc [10]	SIGIR'24	15.34	39.83	53.59	69.7	82.78	19.78	30.71	41.4	58.11	74.07
PIGEON [9]	CVPR'24	11.3	36.7	53.8	72.4	85.3	10.4	23.7	40.6	62.2	77.7
G3 [14]	NeurIPS'24	<u>16.65</u>	<u>40.94</u>	<u>55.56</u>	<u>71.24</u>	<u>84.68</u>	<u>23.99</u>	<u>35.89</u>	<u>46.98</u>	<u>64.26</u>	<u>78.15</u>
GeoRanker	Ours	18.79	45.05	61.49	76.31	89.29	32.94	43.54	54.32	69.79	82.45
Rel. Improvement	Δ	↑ 12.9%	↑ 10.0%	↑ 10.7%	↑ 5.4%	↑ 4.7%	↑ 37.3%	↑ 21.3%	↑ 15.6%	↑ 8.6%	↑ 5.5%

training and inference. (3) **w/o** c_m^{text} . Our method without textual descriptions of candidates in training and inference. (4) **w/o** c_m^{img} . Our method without image data of candidates in training and inference. (5) **w/o** C_g . Our method without generated candidates in inference.

Table 2 presents an ablation study on the IM2GPS3K dataset, and the results for YFCC4K are illustrated in Appendix F. From Table 2 we draw several key insights: (1) All components in our framework contribute positively to the final performance, demonstrating the effectiveness of our design. (2) Comparing our full model with the variant without second-order distance loss ($\mathcal{L}_{\text{PL}}^{(2)}$), we observe more substantial improvements at coarse-grained levels (e.g., country and continent). This highlights the benefit of modeling second-order spatial relationships among candidates, which enables finer-grained ranking and enhances geolocation accuracy. (3) Removing any of the modality-aware prompt components—such as negative candidates (C_{neg}), textual descriptions (c_m^{text}), or image data (c_m^{img})—leads to performance drops, confirming that incorporating multi-modal cues into the prompt is beneficial. Among these, visual information yields the most significant gain, underscoring the importance of image semantics. (4) Finally, the variant without generated candidates (C_g) underperforms our method, showing that generated candidates provide complementary value. This is especially important in scenarios where the retrieval database lacks relevant examples, and generation can introduce novel, informative candidates that enhance the overall candidate pool.

Table 2: Ablation study on IM2GPS3K.

Methods	Street 1km	City 25km	Region 200km	Country 750km	Continent 2500km
w/o $\mathcal{L}_{\text{PL}}^{(2)}$	18.48	44.61	60.96	75.61	88.28
w/o C_{neg}	17.35	44.51	60.82	<u>76.37</u>	88.28
w/o c_m^{text}	18.02	43.91	60.19	76.61	88.62
w/o c_m^{img}	15.58	41.77	59.15	75.40	88.35
w/o C_g	18.21	43.47	59.69	75.47	<u>88.75</u>
Ours	18.79	45.05	61.49	76.31	89.29

4.4 Hyperparameter Analysis

To better understand the impact of key hyperparameters, we conduct a systematic ablation study by varying each at a time while keeping others fixed. The hyperparameters considered include: (1) **Number of retrieval candidates in training** $|C_{rc}|$. (2) **Number of retrieval candidates during inference** $|C_r|$. (3) **Number of generated candidates during inference** $|C_g|$. (4) **Weighting coefficient** λ . (5) **Number of top elements involved in the first-order loss** ($K^{(1)}$). Unless otherwise specified, we use the following default values: $|C_{rc}| = 5$, $|C_r| = 10$, $|C_g| = 5$, $\lambda_1 = 0.7$, and

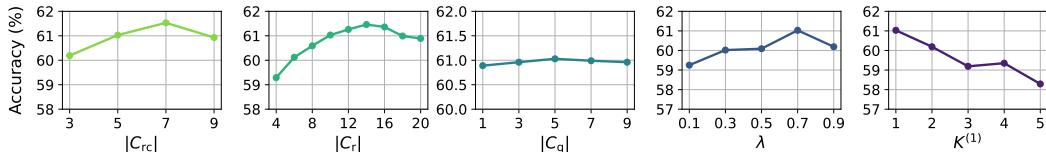


Figure 3: Hyperparameter analysis at the region level on IM2GPS3K. Trends observed at the region level are representative across different geographic levels. Results for all hyperparameters across all levels can be found in Appendix G.

Table 3: Comparison with other ranking baselines.

Methods	IM2GPS3K				
	Street 1km	City 25km	Region 200km	Country 750km	Continent 2500km
Random	10.04	29.72	42.17	57.82	75.24
Top1	13.31	34.03	45.48	61.56	78.04
Prompting	<u>16.62</u>	<u>40.21</u>	<u>54.55</u>	<u>70.07</u>	<u>83.24</u>
Ours	18.79	45.05	61.49	76.31	89.29

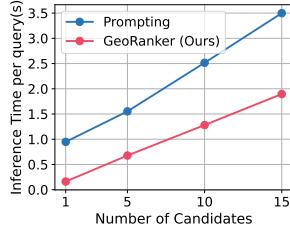


Figure 4: Time efficiency.

$K^{(1)} = 1$. Empirically, we find that the trends of each hyperparameter remain largely consistent across all levels, indicating the stability and robustness of our model under varying geolocation granularities. For clarity and brevity, we present results at the region level as a representative example in Figure 3, while full results for all levels are provided in Appendix G.

Impact of candidate scales in training and inference: (1) Increasing $|C_{rc}|$ initially improves performance, followed by a plateau. This suggests that increasing the number of candidates moderately raises task difficulty, which in turn provides more supervision signals and benefits model training. (2) The number of retrieval candidates used during inference $|C_r|$ also exhibits a rising-then-stabilizing trend. A larger pool of retrieval candidates increases the likelihood of including the correct candidate, and the consistent performance gain further demonstrates the effectiveness of our GeoRanker, which can robustly identify the most relevant one from a diverse set. (3) The model shows relatively flat performance when varying $|C_g|$, indicating that even a small number of generated candidates is sufficient to yield competitive performance.

Impact of hyperparameters in multi-order distance objective: (1) As the weighting factor λ increases, performance first improves and then declines. This highlights a trade-off between the first- and second-order objectives—overweighting the former can reduce the benefit of modeling relative spatial relationships. (2) $K^{(1)}$ shows a consistent downward trend. Larger values introduce more candidate combinations in the partial PL loss during training, which may deviate from the candidate distribution at inference and lead to train-test mismatch. This weakens the supervision signal and degrades performance.

4.5 Comparison with Other Ranking Baselines

To demonstrate the superiority of GeoRanker in ranking ability, we conduct comparative experiments with the following ranking baselines. (1) **Random**: Randomly sampling one candidate from $C_r \cup C_g$ as prediction. (2) **Top-1**: Using embedding similarity to rank candidates and select the top-1 as prediction. The embedding model is fine-tuned following G3 [14] with multi-modal information. (3) **Prompting**: The query image, candidate information, and negative samples are incorporated into the prompt, using LVLM to select the most appropriate candidate as the final prediction. We use Qwen2-VL-7b-Instruct for fair comparison. From Table 3, we can find that our approach (GeoRanker) outperforms all baselines, achieving the highest performance across all metrics. This is because GeoRanker leverages large vision-language models to jointly encode query-candidate interactions and learns fine-grained distance representation through multi-order distance loss during training, enabling it to effectively select accurate predictions from a pool of candidates.

4.6 Efficiency Analysis

Beyond accuracy, efficiency is critical for real-world deployment. We evaluate our approach along two dimensions: **time efficiency**, measuring inference latency, and **data efficiency**, assessing the effectiveness of data usage.

Time efficiency. Figure 4 compares the inference time of GeoRanker with the prompting-based method (introduced in Section 4.5) across varying numbers of candidate inputs. As expected, inference time increases for both methods as the number of candidates grows, due to additional scoring iterations required for GeoRanker and longer prompts for the prompting baseline. Notably, GeoRanker consistently achieves substantially lower inference latency compared to prompting. Within the 1-10 candidate size range, GeoRanker takes less than half the time required by prompting. It is also worth highlighting that GeoRanker naturally supports parallel computation over candidate

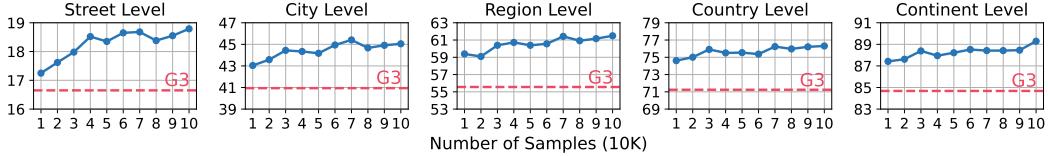


Figure 5: Data efficiency analysis.



Figure 7: Case study illustrating the effectiveness of GeoRanker in re-ranking candidates.

scoring, enabling substantial reductions in inference latency for large-scale deployment. In contrast, Prompting suffers from longer and sequential input construction, which limits such optimization.

Data efficiency. Figure 5 illustrates the performance of GeoRanker at different geographic scales when fine-tuned with varying amounts of data. The x-axis represents the number of samples (in units of 10K), and the y-axis shows the corresponding accuracy. From Figure 5, we observe the following: (1) GeoRanker exhibits a stable and consistent improvement in accuracy as the training data size increases across all geographic levels, demonstrating strong scalability and generalization capacity. (2) For comparison, we also plot the performance of the state-of-the-art method G3. Remarkably, GeoRanker surpasses G3 across all levels even when fine-tuned on just 10% samples, highlighting its data efficiency—the ability to achieve strong performance with limited supervision.

4.7 Impact of Backbone Scale

To investigate the impact of backbone model scale on performance, we conduct experiments using llava-onevision [56] (0.5B) and Qwen2-VL models [57] with 2B and 7B parameters. As shown in Figure 6, and results across all geographic levels in Appendix H, GeoRanker’s performance consistently improves as the backbone model size increases on both IM2GPS3K and YFCC4K. These results indicate that GeoRanker benefits from more powerful LVLM backbones and follows the scaling law, suggesting that its upper-bound performance can be further improved with larger models.

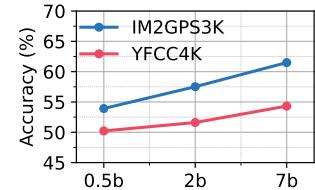


Figure 6: Impact of Backbone Scale on Region Level.

4.8 Case Study

To intuitively demonstrate the effectiveness of GeoRanker, we present a qualitative case study in Figure 7. As shown on the left, the top-5 candidates retrieved are not well ordered by geographic proximity; visually similar but geographically distant images (e.g., 870 KM away) appear at the top ranks. After reranking with GeoRanker, the candidates are successfully reordered by their true geographic distances, with the closest image (0.44 km away) ranked at the top and the farthest ones pushed lower in the list. This result highlights GeoRanker’s ability to model complex spatial relationships through query–candidate interactions, further improving the geolocalization accuracy.

5 Conclusion

In this paper, we propose GeoRanker, a distance-aware ranking framework built upon LVLM. To enhance training, we introduce a novel multi-order distance loss that captures both absolute distances and relative spatial relationships among candidate locations. To support this framework, we construct GeoRanking, the first dataset specifically designed for spatial ranking tasks. Extensive experiments on IM2GPS3K and YFCC4K demonstrate the effectiveness of GeoRanker over baselines.

Acknowledgements

The authors would like to thank Leitian Tao for the valuable feedback on the work.

References

- [1] Nam Vo, Nathan Jacobs, and James Hays. Revisiting im2gps in the deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 2621–2630, 2017.
- [2] Daniel Wilson, Xiaohan Zhang, Waqas Sultani, and Safwan Wshah. Visual and object geo-localization: A comprehensive survey. *arXiv preprint arXiv:2112.15202*, 2021.
- [3] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE international conference on computer vision*, pages 3456–3465, 2017.
- [4] Bingyi Cao, Andre Araujo, and Jack Sim. Unifying deep local and global features for image search. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 726–743. Springer, 2020.
- [5] Fuwen Tan, Jiangbo Yuan, and Vicente Ordonez. Instance-level image retrieval using reranking transformers. In *proceedings of the IEEE/CVF international conference on computer vision*, pages 12105–12115, 2021.
- [6] Seongwon Lee, Hongje Seong, Suhyeon Lee, and Euntai Kim. Correlation verification for image retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5374–5384, 2022.
- [7] Shihao Shao, Kaifeng Chen, Arjun Karpur, Qinghua Cui, André Araujo, and Bingyi Cao. Global features are all you need for image retrieval and reranking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11036–11046, 2023.
- [8] Vicente Vivanco Cepeda, Gaurav Kumar Nayak, and Mubarak Shah. Geoclip: Clip-inspired alignment between locations and images for effective worldwide geo-localization. *Advances in Neural Information Processing Systems*, 36:8690–8701, 2023.
- [9] Lukas Haas, Michal Skreta, Silas Alberti, and Chelsea Finn. Pigeon: Predicting image geolocations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12893–12902, 2024.
- [10] Zhongliang Zhou, Jielu Zhang, Zihan Guan, Mengxuan Hu, Ni Lao, Lan Mu, Sheng Li, and Gengchen Mai. Img2loc: Revisiting image geolocation using multi-modality foundation models and image-based retrieval-augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2749–2754, 2024.
- [11] Yi Liu, Junchen Ding, Gelei Deng, Yuekang Li, Tianwei Zhang, Weisong Sun, Yaowen Zheng, Jingquan Ge, and Yang Liu. Image-based geolocation using large vision-language models. *arXiv preprint arXiv:2408.09474*, 2024.
- [12] Neel Jay, Hieu Minh Nguyen, Trung Dung Hoang, and Jacob Haimes. Evaluating precise geolocation inference capabilities of vision language models. *arXiv preprint arXiv:2502.14412*, 2025.
- [13] Zhiqiang Wang, Dejia Xu, Rana Muhammad Shahroz Khan, Yanbin Lin, Zhiwen Fan, and Xingquan Zhu. Llmgeo: Benchmarking large language models on image geolocation in-the-wild. *arXiv preprint arXiv:2405.20363*, 2024.
- [14] Pengyue Jia, Yiding Liu, Xiaopeng Li, Xiangyu Zhao, Yuhao Wang, Yantong Du, Xiao Han, Xuetao Wei, Shuaiqiang Wang, and Dawei Yin. G3: an effective and adaptive framework for worldwide geolocalization using large multi-modality models. *Advances in Neural Information Processing Systems*, 37:53198–53221, 2024.

- [15] Paul Hongsuck Seo, Tobias Weyand, Jack Sim, and Bohyung Han. Cplanet: Enhancing image geolocation by combinatorial partitioning of maps. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 536–551, 2018.
- [16] Harvey J Miller. Tobler’s first law and spatial analysis. *Annals of the association of American geographers*, 94(2):284–289, 2004.
- [17] James Hays and Alexei A Efros. Im2gps: estimating geographic information from a single image. In *2008 ieee conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008.
- [18] Bart Thomée, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.
- [19] Gengchen Mai, Weiming Huang, Jin Sun, Suhang Song, Deepak Mishra, Ninghao Liu, Song Gao, Tianming Liu, Gao Cong, Yingjie Hu, et al. On the opportunities and challenges of foundation models for geoai (vision paper). *ACM Transactions on Spatial Algorithms and Systems*, 10(2):1–46, 2024.
- [20] Krzysztof Janowicz, Song Gao, Grant McKenzie, Yingjie Hu, and Budhendra Bhaduri. Geoai: spatially explicit artificial intelligence techniques for geographic knowledge discovery and beyond, 2020.
- [21] Ling Li, Yu Ye, Bingchuan Jiang, and Wei Zeng. Georeasoner: Geo-localization with reasoning in street views using a large vision-language model. In *Forty-first International Conference on Machine Learning*, 2024.
- [22] Senzhang Wang, Jiannong Cao, and S Yu Philip. Deep learning for spatio-temporal data mining: A survey. *IEEE transactions on knowledge and data engineering*, 34(8):3681–3700, 2020.
- [23] Yuxuan Liang, Haomin Wen, Yutong Xia, Ming Jin, Bin Yang, Flora Salim, Qingsong Wen, Shirui Pan, and Gao Cong. Foundation models for spatio-temporal data science: A tutorial and survey. *arXiv preprint arXiv:2503.13502*, 2025.
- [24] Tobias Weyand, Ilya Kostrikov, and James Philbin. Planet-photo geolocation with convolutional neural networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VIII 14*, pages 37–55. Springer, 2016.
- [25] Shruman Pramanick, Ewa M Nowara, Joshua Gleason, Carlos D Castillo, and Rama Chellappa. Where in the world is this image? transformer-based geo-localization in the wild. In *European Conference on Computer Vision*, pages 196–215. Springer, 2022.
- [26] Brandon Clark, Alec Kerrigan, Parth Parag Kulkarni, Vicente Vivanco Cepeda, and Mubarak Shah. Where we are and what we’re looking at: Query based worldwide image geo-localization using hierarchies and scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23182–23190, 2023.
- [27] Sijie Zhu, Mubarak Shah, and Chen Chen. Transgeo: Transformer is all you need for cross-view image geo-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1162–1171, 2022.
- [28] Jinliang Lin, Zhedong Zheng, Zhun Zhong, Zhiming Luo, Shaozi Li, Yi Yang, and Nicu Sebe. Joint representation learning and keypoint detection for cross-view geo-localization. *IEEE Transactions on Image Processing*, 31:3780–3792, 2022.
- [29] Xiaohan Zhang, Xingyu Li, Waqas Sultani, Yi Zhou, and Safwan Wshah. Cross-view geo-localization via learning disentangled geometric layout correspondence. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 3480–3488, 2023.
- [30] Scott Workman, Richard Souvenir, and Nathan Jacobs. Wide-area image geolocalization with aerial reference imagery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3961–3969, 2015.

- [31] Liu Liu and Hongdong Li. Lending orientation to neural networks for cross-view geo-localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5624–5633, 2019.
- [32] Sijie Zhu, Taojiannan Yang, and Chen 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*, pages 3640–3649, 2021.
- [33] Hongji Yang, Xiufan Lu, and Yingying Zhu. Cross-view geo-localization with layer-to-layer transformer. *Advances in Neural Information Processing Systems*, 34:29009–29020, 2021.
- [34] Yicong Tian, Chen Chen, and Mubarak Shah. Cross-view image matching for geo-localization in urban environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3616, 2017.
- [35] Yujiao Shi, Xin Yu, Dylan Campbell, and Hongdong 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*, pages 4064–4072, 2020.
- [36] Sijie Zhu, Linjie Yang, Chen Chen, Mubarak Shah, Xiaohui Shen, and Heng Wang. R2former: Unified retrieval and reranking transformer for place recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19370–19380, 2023.
- [37] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136, 2007.
- [38] Tie-Yan Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.
- [39] Alexandros Karatzoglou, Linas Baltrunas, and Yue Shi. Learning to rank for recommender systems. In *Proceedings of the 7th ACM Conference on Recommender Systems*, pages 493–494, 2013.
- [40] Md Ahsanul Kabir, Mohammad Al Hasan, Aritra Mandal, Daniel Tunkelang, and Zhe Wu. A survey on e-commerce learning to rank. *arXiv preprint arXiv:2412.03581*, 2024.
- [41] Anthony Bell, Prathyusha Senthil Kumar, and Daniel Miranda. The title says it all: a title term weighting strategy for ecommerce ranking. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 2233–2241, 2018.
- [42] Siamak Zamani Dadaneh, Shahin Boluki, Mingyuan Zhou, and Xiaoning Qian. Arsm gradient estimator for supervised learning to rank. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3157–3161. IEEE, 2020.
- [43] Yukihiko Tagami, Shingo Ono, Koji Yamamoto, Koji Tsukamoto, and Akira Tajima. Ctr prediction for contextual advertising: Learning-to-rank approach. In *Proceedings of the Seventh International Workshop on Data Mining for Online Advertising*, pages 1–8, 2013.
- [44] Mattia Cerrato, Marius Köppel, Alexander Segner, Roberto Esposito, and Stefan Kramer. Fair pairwise learning to rank. In *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 729–738. IEEE, 2020.
- [45] Yiling Jia, Huazheng Wang, Stephen Guo, and Hongning Wang. Pairrank: Online pairwise learning to rank by divide-and-conquer. In *Proceedings of the web conference 2021*, pages 146–157, 2021.
- [46] Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81, 2010.
- [47] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, pages 1192–1199, 2008.

- [48] Jun Xu and Hang Li. Adarank: a boosting algorithm for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 391–398, 2007.
- [49] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [50] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.
- [51] Robin L Plackett. The analysis of permutations. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 24(2):193–202, 1975.
- [52] R Duncan Luce et al. *Individual choice behavior*, volume 4. Wiley New York, 1959.
- [53] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [54] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [55] Eric Muller-Budack, Kader Pustu-Iren, and Ralph Ewerth. Geolocation estimation of photos using a hierarchical model and scene classification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 563–579, 2018.
- [56] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.
- [57] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Technical Appendix

Table of Contents

A Prompts	15
B GeoRanking Data Entries	15
C More Information on Training and Inference	16
D Baseline Method Details	16
E Query Images with Different Error Thresholds	17
F Complete experimental results on ablation study	18
G Hyperparameter Analysis with All Geographic Levels	18
H Complete Experimental Results on Backbone Model Scale	19
I Limitations	19

Query Image	Candidates				
 (48.306, 11.907)	 (48.311, 11.918) St. Paul, Landkreis Erding, Germany	 (51.941, 13.897) Lübben (Spreewald), Dahme- Spreewald, Germany	 (48.306, 11.907) St. Paul, Landkreis Erding, Germany		
 (9.751, 118.725)	 (12.444, -83.432) Nicaragua	 (8,740, 76.712) Varkala, Kerala, India	 (10.589, 124.313) San Francisco, Cebu, Philippines		
 (37.782, -122.409)	 (40.763, -73.973) New York, New York, United States	 (40.759, -73.984) New York, New York, United States	 (37.789, -122.401) San Francisco, California, United States		
 (52.597, -117.804)	 (44.574, -110.518) Wyoming, United States	 (63.576, 14.589) Jämtland County, Sweden	 (52.396, -116.076) Clearwater County, Alberta, Canada		
 (45.929, 8.662)	 (46.009, 9.282) Lecco, Lombardy, Italy	 (46.011, 9.282) Lecco, Lombardy, Italy	 (45.995, 9.263) Como, Lombardy, Italy		

Figure 8: Examples of GeoRanking Data Entries.

A Prompts

Prompting for generating candidates \mathcal{C}_g . Following previous work [14], we use the following prompt template for generating candidates:

{query image} Suppose you are an expert in geolocation. You have the ability to give two number GPS coordinates given an image. Please give me the location of the given image. Your answer should be in the following JSON format without any other information: {"latitude": float,"longitude": float}.

B GeoRanking Data Entries

Figure 8 illustrates example entries from the GeoRanking dataset. Specifically, each query image is associated with 20 candidates, and each candidate contains GPS coordinates, textual descriptions, and image data. In total, GeoRanking includes 100K samples and 2 million query–candidate pairs. To the best of our knowledge, GeoRanking is the first dataset specifically designed for modeling distance-aware ranking between geographic entities. We release the dataset publicly and hope it will foster future research in areas such as GeoAI, information retrieval, and vision-language modeling.

Table 4: More Details on Training and Inference.

Parameter	Setting
GPU	NVIDIA L40S * 4
Training Time	16 hours / epoch
Total params	8,298,256,896
Trainable params	6,881,280 (0.083%)
Dataset Samples	100K
Batch Size	4
Batch Size per Device	1
Training GPU Memory Consumption	30 GB / GPU
VLM Backbone	Huggingface Qwen2-VL-7b-Instruct
Deepspeed	Stage 2

C More Information on Training and Inference

In this section, we provide additional details regarding the training and inference setup. Table 4 summarizes the key hyperparameters used during these phases. Most experiments were conducted on four NVIDIA L40S GPUs. We also performed tests on two NVIDIA H200 GPUs, where training took approximately 7.5 hours per epoch with a batch size of 4, consuming around 90 GB of GPU memory per device with the gradient checkpointing off.

D Baseline Method Details

In this section, we will give introductions to the baselines:

- **[L]kNN**, $\sigma = 4$ [1]. kNN first retrieves the top- k nearest neighbor images and aggregates their coordinates to form the final prediction. As the k decreases, the aggregation process becomes more focused. When k equals 1, the method turns to the NN.
- **PlaNet** [24]. PlaNet is the first work to formulate the worldwide geolocation task as a classification problem. It partitions the Earth’s surface into a large number of geographical cells and trains a convolutional neural network to predict the correct cell for each image. Unlike previous approaches that primarily rely on landmark recognition or approximate matching with global image descriptors, PlaNet effectively integrates multiple visible cues within the image to enhance localization accuracy.
- **CPlaNet** [15]. CPlaNet follows PlaNet and propose combinatorial partitioning, which generates fine-grained output class by intersecting larger partitions.
- **ISNs** [55]. ISNs enhance the input image information by extracting additional scene context features, such as indoor, natural, or urban environments, alongside the original image content. By incorporating these richer contextual cues, ISNs achieves improved localization performance.
- **Translocator** [25]. Translocator designs a dual-branch transformer framework that simultaneously ingests the original image and its semantic segmentation map. This architecture enables the extraction of fine-grained spatial cues and the construction of more robust feature representations for geolocation.
- **GeoDecoder** [26]. GeoDecoder identifies that earlier methods insufficiently leverage hierarchical spatial information. It addresses this by proposing a cross-attention mechanism that explicitly captures relationships across heterogeneous features, enhancing the model’s ability to interpret complex location-dependent features.
- **GeoCLIP** [8]. GeoCLIP extends the CLIP architecture by introducing a GPS encoder, aligning geographic coordinates with image and GPS embeddings. This enhancement enables more effective modeling of worldwide geolocation tasks by incorporating spatial information directly into the learned feature space.
- **Img2Loc** [10]. Img2Loc advances geolocation by integrating a RAG pipeline. It first retrieves visually similar candidates, then formulates a prompt incorporating these candidates’ coordinates, guiding a vision-language model to generate a final prediction.

Error \leq	Images		
1KM			
25KM			
200KM			
750KM			
2500KM			

Figure 9: Example query images fall in different error thresholds.

- **PIGEON [9]**. PIGEON introduces an innovative framework that combines semantic geocell partitioning, multi-task contrastive pretraining, and a novel loss function. By clustering candidate locations semantically and refining predictions through targeted retrieval, PIGEON significantly boosts localization accuracy.
- **G3 [14]**. G3 proposes a three-stage framework comprising Geo-alignment, Geo-diversification, and Geo-verification. Geo-alignment aligns GPS coordinates, textual descriptions, and visual data into a unified multi-modal representation to strengthen retrieval capabilities. Subsequently, Geo-diversification and Geo-verification are integrated within a RAG framework to robustly generate and select candidate geolocations.

E Query Images with Different Error Thresholds

Figure 9 presents example query images under different error thresholds (1km, 25km, 200km, 750km, and 2500km). We observe that images with errors within 1km often contain distinctive location cues, such as landmark buildings, which facilitate accurate geolocalization. This is partly because retrieval candidates are more likely to retrieve visually similar images from the database due to the popularity of such locations. Additionally, generated candidates tend to produce more reliable predictions in these cases, as the locations are well-represented in the world knowledge embedded

Table 5: Complete ablation study on IM2GPS3K and YFCC4K.

Methods	IM2GPS3K					YFCC4K				
	Street 1km	City 25km	Region 200km	Country 750km	Continent 2500km	Street 1km	City 25km	Region 200km	Country 750km	Continent 2500km
w/o $\mathcal{L}_{PL}^{(2)}$	18.48	44.61	60.96	75.61	88.28	31.97	43.12	53.53	69.03	81.19
w/o C_{neg}	17.35	44.51	60.82	76.37	88.28	31.57	43.06	53.62	69.09	81.67
w/o c_m^{text}	18.02	43.91	60.19	76.61	88.62	31.70	43.06	54.03	69.42	82.07
w/o c_m^{img}	15.58	41.77	59.15	75.40	88.35	15.81	27.86	41.31	61.39	77.66
w/o \mathcal{C}_g	18.21	43.47	59.69	75.47	88.75	32.60	43.03	53.43	69.77	82.71
Ours	18.79	45.05	61.49	76.31	89.29	32.94	43.54	54.32	69.79	82.45

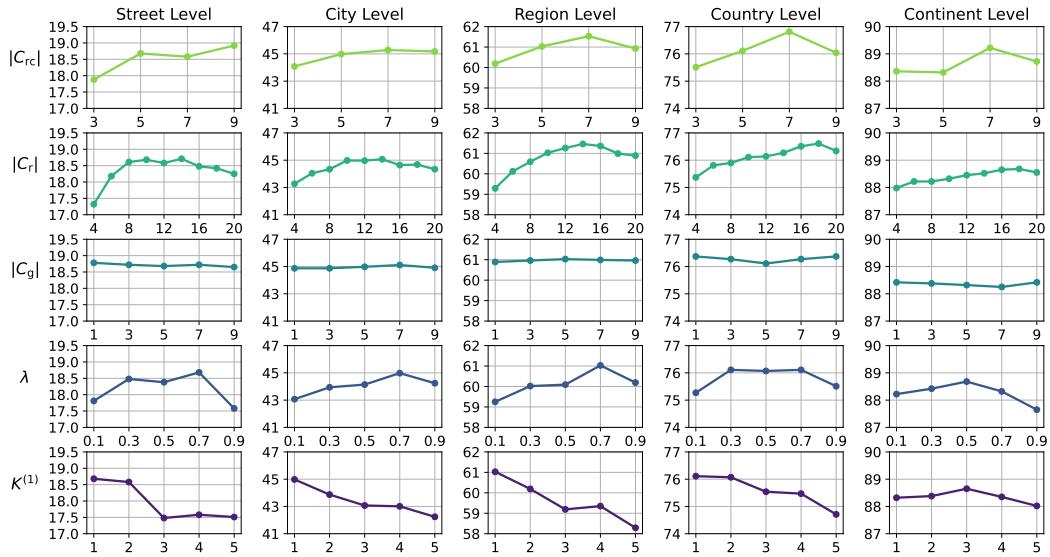


Figure 10: Hyperparameter analysis with all geographic levels on IM2GPS3K.

in large vision-language models. In contrast, query images with large geolocation errors (e.g., 2500km) typically lack informative visual cues—such as images depicting open oceans or vast grasslands—making it extremely challenging to infer their true locations. In such cases, neither retrieval nor generation is likely to yield useful candidates.

F Complete experimental results on ablation study

Table 5 presents the complete ablation results on both IM2GPS3K and YFCC4K. Consistent with the findings discussed in the main text, we observe that each component in our framework contributes positively to overall performance. Moreover, different types of contextual information incorporated into the prompt—such as visual cues, textual descriptions, and negative examples—all help improve both model training and inference. Finally, generated candidates are shown to complement retrieval-based candidates effectively. This is particularly beneficial for rare or long-tail query images, where retrieval candidates alone may fail to provide sufficient clues for accurate geolocation.

G Hyperparameter Analysis with All Geographic Levels

Figure 10 shows the impact of different hyperparameters on GeoRanker across all geographic levels. As observed, the trends of each hyperparameter remain largely consistent across levels, highlighting the stability and robustness of our model under varying localization granularities.

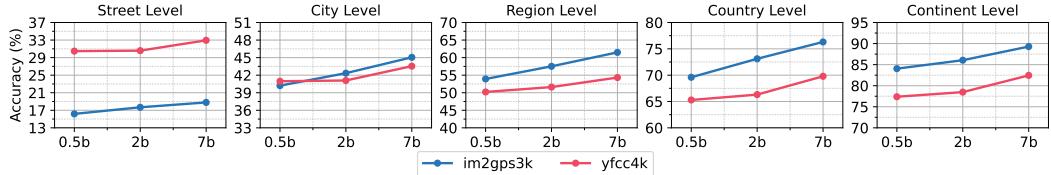


Figure 11: Impact of Backbone Scale across All Levels.

H Complete Experimental Results on Backbone Model Scale

Figure 11 shows the effect of backbone model size across all geographic levels. Consistent performance improvements are observed on both IM2GPS3K and YFCC4K datasets as the backbone scales from 0.5B to 7B parameters, further confirming GeoRanker’s scalability and compatibility with stronger LVLM.

I Limitations

Our method achieves notable improvements in geolocalization accuracy over existing baselines. In addition, it demonstrates superior time efficiency compared to LVLM prompting methods, and its data efficiency allows strong performance even with relatively limited supervision. However, compared to direct embedding-based retrieval approaches, GeoRanker introduces an additional ranking stage, which leads to increased computational overhead during inference. One solution is to analyze the retrieval results: if the top-k candidates are already geographically concentrated, the ranking step can be skipped without significant loss in accuracy, thereby reducing the overall inference time. In addition, GeoRanker supports parallel scoring of candidates during large-scale deployment, which can significantly improve runtime and computational efficiency.