Supplementary Materials for
"GeoExplorer: Active Geo-localization with Curiosity-Driven Exploration"

Li Mi, Manon Béchaz, Zeming Chen, Antoine Bosselut, Devis Tuia
EPFL, Switzerland
https://limirs.github.io/GeoExplorer/

The supplementary materials are organized as follows:

- **Dataset details** (Section S1).
- **Implementation details** (Section S2).
- **Comparison of AGL and related tasks** (Section S3).
- **Supplementary experiments**: parameter analysis, experiments on varying budget and larger grid size, evaluations using step-to-the-goal as metric, and supplementary results (Section S4).
- **Supplementary analysis**: path analysis, additional visualization samples and failure case analysis (Section S5).
- **Discussions**: limitations and future work (Section S6).

S1. Datasets Details

S1.1. Massachusetts Buildings (Masa) Dataset

Data Collection. The Massachusetts Buildings (Masa) dataset [17] consists of 1188 high resolution images of the Boston area. Building footprint annotations were obtained by rasterizing data from the OpenStreetMap project.

Dataset Composition. The dataset is split in 70% for training (832 images), 15% for testing and evaluation (178 for validation and 179 for testing) [19, 21]. Each image, or search area, is structured as a 5×5 grid of search cells, with 300×300 pixels per grid cell. During training, data augmentation is applied through top-right and left-right flipping, and each search area allows for 25 start positions with 24 possible goal locations, leading to approximately 2 million unique training trajectories. For testing and validation, only a fixed configuration is randomly selected per start-to-goal distance and per search area, ensuring 895 fixed test trajectories.

S1.2. MM-GAG Dataset

The MM-GAG dataset [21] was constructed to address the limitations of existing datasets for Active Geo-localization (AGL), which often lack precise coordinate annotations and meaningful goal representations across diverse modalities.

Multi-Modal Goal Representations. Unlike many existing datasets that focus solely on aerial-to-aerial or aerial-to-ground localization, MM-GAG introduces multi-modal goal representations:

- Aerial Imagery,
- Ground-Level Imagery,
- Natural Language Descriptions.

Data Collection. The MM-GAG dataset was built by collecting high-quality geo-tagged images from smartphone devices across diverse locations. Images have been filtered, resulting in 73 distinct search areas. Note that through the link provided in the original paper[1], we only find 65 search areas. To ensure fair comparison, we evaluate the proposed method and the pretrained baseline model provided in the original paper in Section S4. For each of the 73 ground-level images, high-resolution satellite image patches were retrieved at 0.6m per pixel resolution. From these patches, 5×5 search grids with 256×256 pixels per grid cell were constructed. To generate textual goal descriptions, each ground-level image was automatically captioned using LLaVA-7B [15]. The captioning prompt was carefully designed to ensure concise and relevant descriptions.

Dataset Composition. Trajectories are selected by randomly sampling start and goal locations within each of the 73 search areas, ensuring a diverse range of search scenarios. For each area, five {start, goal} pairs are chosen for every predefined distance category, resulting in a total of 365 evaluation trajectories per start-to-goal distance.

S1.3. xBD Dataset

The xBD dataset [13] is a large-scale aerial imagery dataset designed for disaster analysis. It contains images captured before (xBD-pre) and after (xBD-disaster) various natural disasters such as wildfires, floods, and earthquakes.

Data Collection. Imagery for the original xBD dataset was sourced from the Maxar/DigitalGlobe Open Data Program[2], which provides high-resolution satellite images for major crisis events. The dataset includes imagery from 19 natural disasters across $45,361.79$ km² of affected areas. A total of $22,068$ images were collected, covering $850,736$ human-annotated building polygons.

[1] https://huggingface.co/datasets/MVRL/MM-GAG/tree/main
[2] https://www.digitalglobe.com/

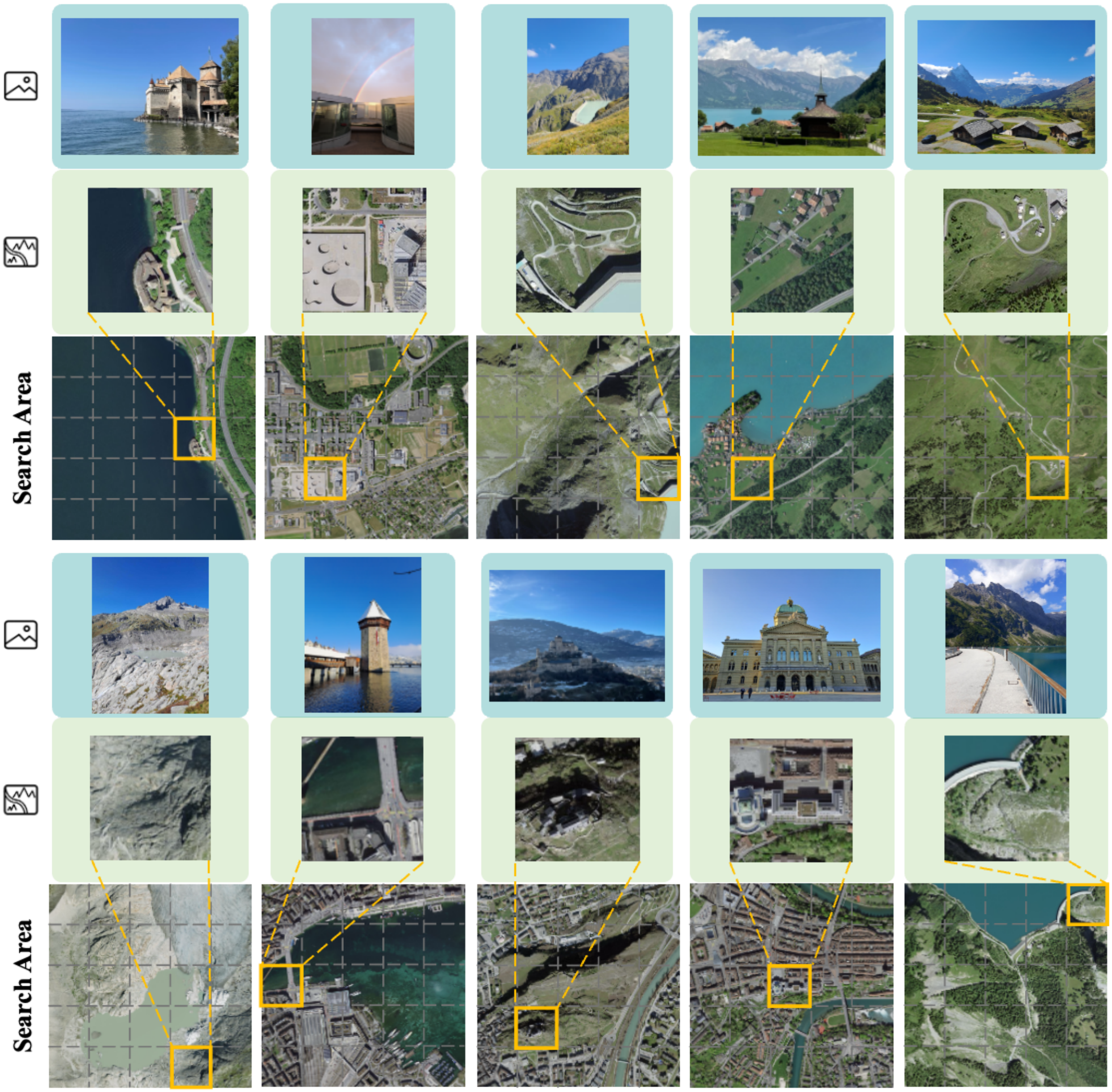


Figure S1. **Additional examples from the proposed SwissViewMonuments dataset** with unseen targets. From up to down: goal presented as ground-level images, goal presented in aerial view, and the search areas.

Dataset Composition. For the specific task of AGL, 800 bitemporal search areas (one image before and one image after the disaster) have been selected from the original dataset. Each search area corresponds to a 5 × 5 search grid with 300 × 300 pixels per grid cell. Trajectories are obtained by randomly sampling 5 pairs of start and goal location per start-to-goal distance, resulting in 4000 evaluation trajectories per start-to-goal distance.

S1.4. SwissView Dataset

Data Collection. The SwissView dataset consists in two complementary components: SwissView100 and SwissViewMonuments. For SwissView100, a total of 100 images were randomly sampled across the entire territory of Switzerland, sourced from Swisstopo's SWISSIMAGE 10 cm database[3]. The spatial distribution of the images is provided in Figure S2. The original images, with a spa-

[3] https://www.swisstopo.admin.ch/fr/orthophotos-swissimage-10-cm

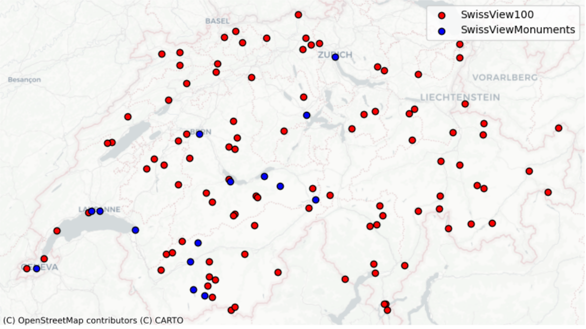


Figure S2. Geographic distribution of the images from the SwissView dataset. Red points indicate the locations of the 100 randomly sampled images from the SwissView100 subset, while blue points represent the images from the SwissViewMonuments subset.

tial resolution of 0.1 meters per pixel and dimensions of $10,000 \times 10,000$ pixels, were downsampled to a resolution of 0.6 meters per pixel, resulting in 1500×1500 pixels images. These downsampled images were subsequently partitioned into 5×5 patches, each measuring 300×300 pixels. For SwissViewMonuments, the procedure is identical, if only for the choice of images. For this part of the dataset, 15 specific areas of Switzerland have been carefully selected for their atypical constructions or landscapes. We consider targets from two categories: 1) *object uniqueness*: landmarks or localizable architectures; 2) *location and scene uniqueness*: unseen scene classes. The resulting dataset, for example, includes images from remarkable buildings such as cathedrals and castles, and rare landscapes like glaciers. Besides the aerial view of the search area, we also provide the corresponding ground-level images and its location associated to the aerial view. A few examples from the SwissViewMonuments dataset are shown in Figure S1.

Dataset Statistics. The location of the selected areas is also given in Figure S2. Samples from the SwissView100 dataset (in red) are distributed among the region and samples from the SwissViewMonuments (in blue) are chosen from cities, tourist attractions and nature reserves.

Dataset Composition. To generate trajectories for the SwissView100 subset, we followed a similar approach to other datasets by randomly generating 5 {start, goal} pairs for each trajectory and each start-to-goal distance, resulting in 500 trajectories per distance considered. In contrast, for the SwissViewMonuments subset, we provide 25 shifted aerial views for each aerial-ground pair (15×25 samples in total), fixing the goal at all positions in a 5×5 grid. For each specified start-to-goal distance $\mathcal{C} \in \{4, 5, 6, 7, 8\}$, we randomly select one starting point per sample that satisfies the given distance relative to the fixed goal. Samples with goal positions that do not permit any valid starting point at a given distance are excluded from the evaluation for that distance. This process results in $\{375, 360, 300, 180, 60\}$ configurations for distances $\mathcal{C}$ in $\{4, 5, 6, 7, 8\}$, respectively.

S2. Implementation Details

This section provides an overview of the implementation details, including the pretrained models used for text, ground-level, and aerial image encoding, as well as the causal transformer used for action-state modeling. Note that apart from the action-state dynamics modeling and curiosity-driven component of our model, the implementation and training parameters remain consistent with those outlined in the work of Sarkar *et al.* [21], which serves as the baseline for our study.

Text and Ground-level Image Encoders. To encode text descriptions and ground-level images of the goal, we use the pretrained encoders from the CLIP model [20], with the same pretrained weights used in [21], which are available on Hugging Face[4]. Specifically, the vision encoder is a Vision Transformer (ViT-b-32), and the text encoder is based on the BERT architecture, both of which are aligned in a shared multimodal embedding space through contrastive learning. These encoders remain frozen during training of the GeoExplorer model.

Aerial Image Encoder. The aerial images are processed with the Sat2Cap satellite encoder [8], which is fine-tuned to align its feature representations with the CLIP embedding space. The alignment is performed using contrastive learning with the InfoNCE loss [18], leveraging a large-scale dataset of paired aerial and ground-level images. Note that the CLIP image encoder remains frozen during this finetuning of the aerial image encoder. This alignment ensures that the features extracted from the aerial images share the same representation space as the features from the text descriptions and ground-level images. We use the same pretrained weights for Sat2Cap as the reference work [21], which can be found on Hugging Face[5].

Causal Transformer. For the Causal transformer used for sequential action and state prediction, we employ a pretrained Falcon-7B model [1]. The pretrained weights can be found on Hugging Face[6]. We follow the multi-modal projection layer introduced in GOMAA-Geo [21], to align the visual and language modalities into the latent space of

[4] https://huggingface.co/openai/clip-vit-base-patch32
[5] https://huggingface.co/MVRL/Sat2Cap
[6] https://huggingface.co/openai/clip-vit-base-patch16

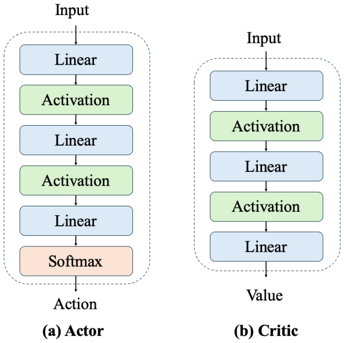


Figure S3. Model architecture of the actor-critic network (action prediction head).

the Falcon-7B model. Additionally, relative position encodings, measured with respect to the top-left position of the image, are incorporated into each state representation, allowing the model to encode spatial relationships between observed aerial images. Note that the primary distinction from the work of Sarkar et al. [21] is that, in our approach, both states and actions are predicted sequentially, rather than solely predicting actions. However, the overall structure remains unchanged. The Causal transformer is trained using a learning rate of $1e-4$, a batch size of 1, and the Adam optimizer over 300 epochs. We followed the settings in GOMAA-Geo [21] to ensure a fair comparison.

PPO. During the Curiosity-Driven Exploration (CE) phase, we introduce an action prediction head on top of the frozen pretrained Causal Transformer. Since the Causal Transformer alone does not inherently model decision policies, the addition of the action prediction head is crucial, as it allows the system to explicitly learn a mapping from the learned state representations to the concrete actions to take. The action prediction head is implemented using an actor-critic framework and optimized using the Proximal Policy Optimization (PPO) [22]. This framework consists of an actor network responsible for policy learning, $\pi_\theta(a|s_t)$ and a critic network $v_\psi(s_t)$ that evaluates the expected total reward from state s_t. As shown in Figure S3, the actor and critic networks are implemented as Multi-Layer Perceptrons (MLPs) with three hidden layers. Each hidden layer is followed by a `tanh` activation function. The final layer of the actor network includes a `softmax` activation to output a probability distribution over actions, ensuring valid action selection. The critic network outputs a single scalar value representing the estimated value function.

At each time step t (which specifies the time index in $[0, T]$), with the state representation s_t, the learning process of PPO can be described as:

(**1**) The actor chooses an action $\hat{a}_t \sim \pi_\theta(a|s_t)$, where $a \in \mathcal{A}$ is an action from avaliable action set.

(**2**) The agent executes the action in the environment and obtains the new state s_{t+1} and reward r_t^{CE}.

(**3**) The *value*, i.e. the total reward from state s_t is calculated by the critic $v_\psi(s_t)$.

(**4**) The *return*, i.e. the total reward from taking action $\hat{a}_t$ from state s_t is calculated:

$$\begin{aligned} q(s_t, \hat{a}_t) &= \sum_{t'=t}^{T} E_{\pi_\theta}\left[r_{t'}^{CE} \mid s_t, \hat{a}_t\right] \\ &= r_t^{\text{CE}} + \gamma r_{t+1}^{\text{CE}} + \ldots \\ &+ \gamma^{T-t+1} r_{T-1}^{\text{CE}} + \gamma^{T-t} v_\psi(s_T). \end{aligned} \tag{1}$$

(**5**) The advantage function A_t is calculated to estimate how much better (or worse) an action $\hat{a}_t$ is compared to the average performance expectation at state s_t:

$$A_t = q(s_t, \hat{a}_t) - v_\psi(s_t). \tag{2}$$

(**6**) The actor and critic are updated.

We update both the actor and critic networks using the PPO loss function, which consists of three main components:

- Actor loss (clipped surrogate objective), which compares the probabilities from the old and the updated policies and will constrain the policy change in a small range:

$$\mathcal{L}_{\text{Actor}} = \min\left\{\frac{\pi_\theta}{\pi_{\theta,\text{old}}} A_t,\ \text{clip}\left(1-\epsilon, 1+\epsilon, \frac{\pi_\theta}{\pi_{\theta,\text{old}}}\right) A_t\right\}. \tag{3}$$

- Critic loss, which measures how well the model predicts the expected reward:

$$\mathcal{L}_{\text{Critic}} = (v_\psi(s_t) - q(s_t, \hat{a}_t))^2. \tag{4}$$

- Entropy regularization, which encourages policy exploration by preventing premature convergence to suboptimal policies:

$$\mathcal{H}\left[\pi_\theta\right](s_t) = -\sum_a \pi_\theta(a|s_t) \log \pi_\theta(a|s_t). \tag{5}$$

The final loss function used for optimization is:

$$\mathcal{L}_{\text{PPO}} = \mathbb{E}\left[-\mathcal{L}_{\text{Actor}} + \omega \mathcal{L}_{\text{Critic}} + \rho \mathcal{H}\right], \tag{6}$$

where ω and ρ are hyperparameters controlling the balance between policy learning, value estimation, and exploration.

We choose the hyperparameters to be consistent with the baseline model [21]. The learning rate is set to $1e-4$ and the batch size is 1. The model is trained for 300 epochs using the Adam optimizer. The values for the hyperparameters α and β are set to 0.5 and 0.01, respectively. The clipping ratio ϵ is chosen to be 0.2 and the discount factor γ is set to 0.99 for all experiments. As in [21], we copy the parameters of π_θ onto $\pi_{\theta,\text{old}}$ every 4 epochs of policy training. For CE stage, patches are resized to 224×224 and normalized.

S3. Comparison of AGL and Related Tasks

The following section provides a comparison of four related tasks: Active Geo-Localization, Visual Geo-Localization, Cross-View Geo-Localization, and Visual Navigation. It highlights their characteristics and differences, as well as associated challenges.

Active Geo-Localization. Active Geo-Localization aims at locating a target by exploring an environment using a sequence of visual aerial observations [19]. This task is especially important in applications such as search-and-rescue [19, 21], where efficient exploration is crucial for success. Unlike Visual Geo-Localization [31], where localization relies solely on unique and static observations, AGL involves movement of an agent to refine position estimates and ultimately reach the goal. Generally, the goal can be specified with different modalities, such as images or text descriptions [19, 21]. Reinforcement learning is often used to define the agent's exploration strategy, guiding it towards the predefined target.

Visual Geo-Localization. The task of Visual Geo-Localization [31] is linked to the task of Active Geo-Localization in the sense that both aim at determining a location based on a given image. However, while Active Geo-Localization uses an agent to explore the environment to refine its position and reach the goal, visual geo-localization depends on single inputs, such as images or video frames, *without the need for an agent to move* [2, 26]. The input image's location is estimated by comparing the observed images to an existing database of geotagged images, often leveraging image-retrieval techniques [4, 5]. Visual geo-localization can operate in various settings, from small-scale areas like specific streets [2] to large urban environments [4], depending on the breadth of the dataset used. Common applications include mobile device localization [6] or autonomous vehicles using street-view data [9].

Cross-View Geo-Localization. The task of cross-view geo-localization aims at localizing a ground-level image by retrieving its corresponding geo-tagged aerial view [29, 29, 32, 33]. Especially, fine-grained cross-view geo-localization [11, 24, 25, 27, 30] requires estimating the 3 Degrees of Freedom (DoF) pose of a query ground-level image on an aerial image, which is similar to the AGL setting. Although related, cross-view geo-localization requires *full access* to the search area to perform matching, while AGL *only provides the agent with partial visibility to the search area from the outset* and performs observation only along the exploration trajectory.

Visual Navigation. Visual navigation [3, 14, 23] is similar to Visual Geo-Localization as both tasks involve an agent exploring its environment to reach a predefined goal. However, unlike Visual Geo-Localization, which functions in aerial, *i.e.*, bird-eye-view environments, visual navigation typically operates in a ground-level environment. Despite the similarities, Active Geo-Localization presents its unique challenges compared to Visual Navigation. One key difference is that, in Active Geo-Localization, *the goal may not be visible to the agent in advance or even presented in different modalities from the agent observation, which introduces a level of uncertainty and complexity*. Moreover, the environment may change abruptly between two actions, as the agent can quickly transition from one type of terrain to another (such as moving from an urban region to a wooded area) due to the larger spatial scope of observations at each step. In contrast, the environment in which the agent operates is more localized in visual navigation, which allows for more accurate location estimation and easier navigation.

Vision-Language Navigation. Vision-language navigation (VLN) [12, 28], especially Aerial VLN [10, 16], is linked to AGL as both tasks provide multimodal guidance to the agent to reach a goal in an environment. Unlike AGL, VLN performs the navigation in a continuous space in terms of both observation and action, which poses an additional challenge. However, AGL also has its unique challenges compared to VLN. Since VLN assumes that the instructor knows the goal's location, the natural language instructions are detailed throughout the navigation and typically correspond to the agent's actions (e.g.,"turn right"). In AGL, the only guidance provided is the goal information, making the setting more challenging due to the sparse reward and accumulated errors in a goal-reaching reinforcement learning context. Moreover, the mainstream methods of VLN are based on sequence modeling and prediction, which differs from the RL pipeline in AGL.

S4. Supplementary Experiments

S4.1. Parameter Analysis

The impact of loss weight α. We use loss weight α to balance the contribution of action modeling loss and state modeling loss. To evaluate the effectiveness of different ablations, we randomly generate the action-state trajectories with the optimal actions for each time step on the test set of the Masa dataset, following the steps described in Section 3.3 of the main paper. Then, we evaluate the models to predict optimal actions at each step of the trajectory. We use random seeds to ensure the same trajectories are tested for all the methods in a test, and we evaluate the models on 5 different tests. The action prediction accuracy for each test as well as the average prediction among 5 tests are reported

Table S1. **Parameter analysis of loss weight** α on the test set of the Masa dataset. The action prediction accuracy is reported for the DM stage. $\alpha = 0$ denotes the baseline with action modeling loss only.

α	Test 1	Test 2	Test 3	Test 4	Test 5	Average
0	**0.6056**	0.1883	0.0838	0.6953	0.1429	0.3432
0.5	0.5162	0.1708	**0.1133**	0.7034	0.1429	0.3293
1	0.5777	**0.2602**	0.0407	**0.8299**	0.1429	**0.3703**
2	0.5687	0.2179	0.0064	0.8055	**0.1524**	0.3502

Table S2. **Parameter analysis of reward weight** β on the test set of the Masa dataset. $\beta = 0$ denotes the baseline with extrinsic reward only.

β	$\mathcal{C} = 4$	$\mathcal{C} = 5$	$\mathcal{C} = 6$	$\mathcal{C} = 7$	$\mathcal{C} = 8$
0	0.3978	0.4939	0.7609	0.8413	0.8648
0.25	**0.4324**	**0.5318**	**0.8156**	**0.9229**	0.9497
0.5	0.3597	0.4849	0.7687	0.9073	**0.9587**
1	0.3988	0.4983	0.7542	0.9028	0.9352

in Table S1. Among most of the tests except test 1, adding state modeling loss leads to a better action prediction performance, which confirms the fact that state and action transitions are inherently interconnected and dynamically influencing each other. When $\alpha = 1$, the model achieves best overall performance with an improvement of 0.0271 over the baseline with only the action modeling loss ($\alpha = 0$).

The impact of reward weight β**.** We also control the impact of intrinsic reward on the final reward by using reward weights β. Results in Table S2 suggest a good balance should be achieved between the goal-oriented extrinsic reward, which directs the agent to the goal and the curiosity-driven intrinsic reward, which encourages the agent to explore the environment. The empirical results show that when $\beta = 0.25$, the agent balances the guidance from extrinsic goal and intrinsic curiosity best.

S4.2. Results on the xBD dataset

We present the supplementary results on the xBD dataset in Table S3. The results show a small performance gap of GeoExplorer between the two subsets (0.0149 on average), indicating the generalization ability of the model.

S4.3. Results on the SwissView100 dataset

The proposed SwissView dataset has two subsets: SwissViewMonuments for unseen target generalization evaluation and SwissView100 for cross-domain transfer evaluation. We present the results from the former setting in the main paper and the latter one in this section. As stated before, the test setting for SwissView100 dataset is the same as the cross-domain transfer setting on the MM-GAG Aerial and xBD dataset: the model is only trained on the Masa dataset and the goal is presented in aerial view. As shown in Table S4, the model faces domain shifts across datasets: Compared with the performance on the Masa dataset, the performance of both methods decreases 0.0375 on average. However, GeoExplorer still outperforms the baseline, suggesting a better cross-domain transferability.

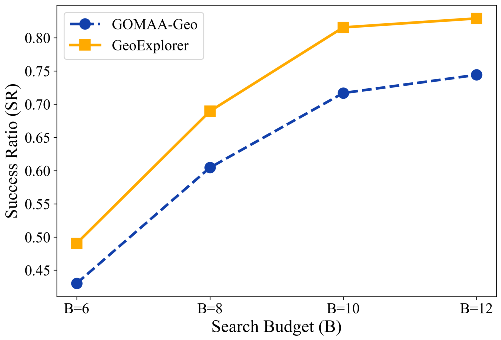


Figure S4. Comparison between GeoExplorer and the baseline model with **varying search budget** when $\mathcal{C} = 6$.

S4.4. Supplementary results on the MM-GAG dataset

As mentioned in Section S1, only 65 search areas are found through the link provided by the paper. To ensure fair comparison, we evaluate GeoExplorer and the pre-trained GOMAA-Geo on the same test configurations and report the results in Table S5. We also report the GOMAA-Geo performance from the original paper for reference. The results suggest similar observations with 65 and 73 search areas: GeoExplorer achieves performances comparable to the baseline on short paths, while significantly improving SR when the path is longer.

S4.5. Exploration with varying search budget

Search budget ($\mathcal{B}$) is an important factor in AGL. To give further insights on its impact on the exploration behaviour, we compare GeoExplorer and the baseline model GOMAA-Geo with varying $\mathcal{B}$, when $\mathcal{C} = 6$ in Figure S4. As expected, as $\mathcal{B}$ increases, the performance of both models improves, as the tolerance for mistakes is higher. More interestingly, the advantage of GeoExplorer is more obvious when $\mathcal{B}$ increases (0.0603 when $\mathcal{B} = 6$ while 0.0850 when $\mathcal{B} = 12$), as it allows more exploration during localization process.

S4.6. Exploration with larger grid size

We compare GeoExplorer and GOMAA-Geo [21] on a grid size of 10×10 with $\mathcal{C} = \{14 - 18\}$, providing *more variability for longer paths*. As shown in Table S6, GeoExplorer consistently shows improvements, especially for

Table S3. **Cross-domain transfer** on the **xBD-pre and xBD-disaster datasets**. Note that the models are only trained on the Masa dataset and the goal is always presented from the aerial view before the disaster for both datasets.

	Evaluation using xBD-pre Dataset					Evaluation using xBD-disaster Dataset				
Method	$\mathcal{C} = 4$	$\mathcal{C} = 5$	$\mathcal{C} = 6$	$\mathcal{C} = 7$	$\mathcal{C} = 8$	$\mathcal{C} = 4$	$\mathcal{C} = 5$	$\mathcal{C} = 6$	$\mathcal{C} = 7$	$\mathcal{C} = 8$
Random policy	0.1412	0.0584	0.0640	0.0247	0.0236	0.1412	0.0584	0.0640	0.0247	0.0236
PPO policy [22]	0.1237	0.1262	0.1425	0.1737	0.2075	0.1132	0.1146	0.1292	0.1665	0.1953
AiRLoc [19]	0.1191	0.1254	0.1436	0.1676	0.2021	0.1201	0.1298	0.1507	0.1631	0.1989
DiT [7]	0.1132	0.2341	0.3198	0.3664	0.3772	0.1012	0.2389	0.3067	0.3390	0.3543
GOMAA-Geo [21]	0.3825	0.4737	0.6808	0.7489	0.7125	**0.4002**	0.4632	0.6553	0.7391	0.6942
GeoExplorer	**0.3973**	**0.4990**	**0.7328**	**0.8390**	**0.8363**	0.3975	**0.5025**	**0.7185**	**0.8190**	**0.7923**

Table S4. **Cross-domain transfer evaluation** on the **SwissView100 subset of SwissView dataset**.

Method	$\mathcal{C} = 4$	$\mathcal{C} = 5$	$\mathcal{C} = 6$	$\mathcal{C} = 7$	$\mathcal{C} = 8$
GOMAA-Geo*	**0.4100**	0.5000	0.6580	0.7780	0.6880
GeoExplorer	0.4020	**0.5120**	**0.7660**	**0.9040**	**0.8800**

Table S5. **Supplementary results** on the **MM-GAG dataset** with 65 search areas. The goal is presented as an aerial image ("I"), a ground-level image ("G"), or a text ("T"). Results from the original paper are in gray and * denotes the results on the same configurations using pretrained model provided by the paper.

Goal	Method	$\mathcal{C} = 4$	$\mathcal{C} = 5$	$\mathcal{C} = 6$	$\mathcal{C} = 7$	$\mathcal{C} = 8$
I	GOMAA-Geo [21]	0.4085	0.5064	0.6638	0.7362	0.7021
	GOMAA-Geo*	0.4246	0.4769	0.7385	0.7662	0.6369
	GeoExplorer	**0.4338**	**0.5415**	**0.7631**	**0.8369**	**0.8277**
G	GOMAA-Geo [21]	0.4383	0.5150	0.6808	0.7489	0.6893
	GOMAA-Geo*	**0.4585**	0.4554	0.6646	0.7169	0.6708
	GeoExplorer	0.4308	**0.5138**	**0.7200**	**0.8246**	**0.7815**
T	GOMAA-Geo [21]	0.4000	0.4978	0.6766	0.7702	0.6595
	GOMAA-Geo*	0.4277	**0.5015**	0.6523	0.7538	0.6677
	GeoExplorer	**0.4431**	0.4892	**0.7200**	**0.8062**	**0.7631**

Table S6. Comparison between GeoExplorer and the baseline model with **larger grid size** on the Masa dataset. Note that all the methods are trained on the 10×10 grid size. ** corresponds to results obtained from the re-trained model using official code [21].

Method	$\mathcal{C} = 14$	$\mathcal{C} = 15$	$\mathcal{C} = 16$	$\mathcal{C} = 17$	$\mathcal{C} = 18$
GOMAA-Geo**	0.2603	0.2704	0.2916	0.2413	0.2201
GeoExplorer	**0.2883**	**0.3117**	**0.3352**	**0.3073**	**0.3151**

longer paths, which aligns with results tested on 5×5. Note that all the models are retrained for the grid size of 10×10.

S4.7. Step-to-the-goal (SG) Evaluation

Besides the commonly used metric success ratio (SR) in AGL, we also provide step-to-the-goal (SG) to evaluate the Manhattan distance between the path-end and goal locations on the SwissView dataset. Results in the Table S7 indicate that GeoExplorer *improves success rate and brings the agent much closer to the goal*.

Table S7. **Step-to-the-goal evaluation of unseen objects generalization ability** on the **SwissViewMonuments dataset**. * corresponds to results obtained from the pretrained model [21].

	Method	$\mathcal{C} = 4$	$\mathcal{C} = 5$	$\mathcal{C} = 6$	$\mathcal{C} = 7$	$\mathcal{C} = 8$
I	GOMAA-Geo*	2.16	1.91	1.18	0.93	0.77
	GeoExplorer	**2.08**	**1.56**	**0.65**	**0.28**	**0.30**
G	GOMAA-Geo*	2.19	1.81	1.07	0.69	0.77
	GeoExplorer	**2.14**	**1.51**	**0.64**	**0.29**	**0.70**

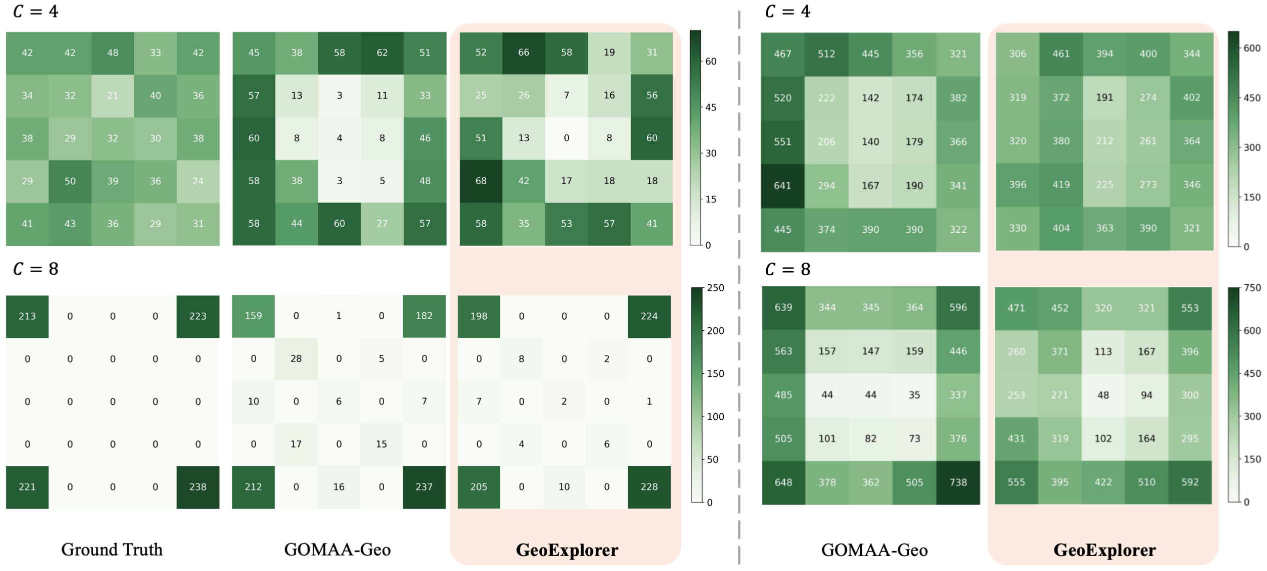
S5. Suppelmentary Analysis

S5.1. Path statistics

We provide an in-depth analysis of the exploration ability of the model by tracking the visited patches of the baseline model and GeoExplorer. In Figure S5 (a), we count the end location of 895 paths from the Masa test set for the ground truth (goal location), GOMAA-Geo and GeoExplorer. The results confirm that 1) for $\mathcal{C} = 4$ the goal locations are more evenly distributed in the search area while the configurations are limited for $\mathcal{C} = 8$. This could explain why we usually have higher performance when $\mathcal{C} = 8$. As the trajectory grows, the agent may infer that the goal is likely located at the corner, based on the training data distribution. 2) The path end distribution of the baseline model suggests a tendency to visit edge patches (with 84.36% of paths ended on the edge patches when $\mathcal{C} = 4$), while GeoExplorer improves the exploration of inside patches. The visited patch distribution shown in Figure S5 (b) confirms this observation: when $\mathcal{C} = 4$, only 20.08% of the visited patches are inside for GOMAA-Geo and GeoExplorer increases this ratio as 30.79%. This finding indicates that Geoexplorer improves the performance on the AGL benchmarks by a better exploration ability of the environment.

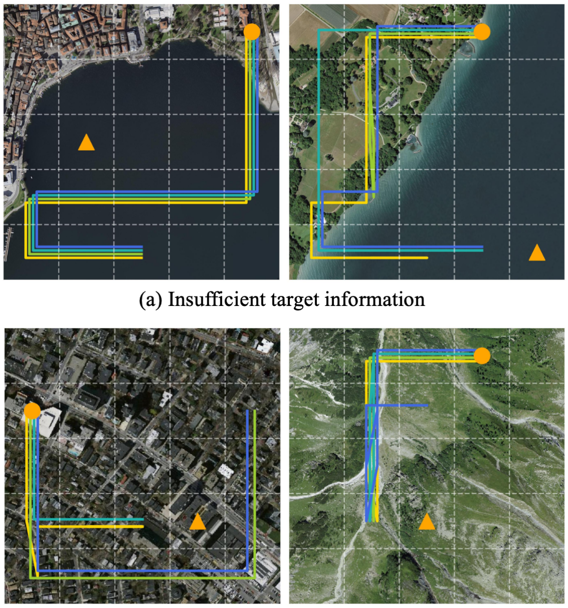
S5.2. Failure case analysis

We check and analyze the failure cases of GeoExplorer. Examples shown in Figure S6 imply two main reasons of failure: (a) Insufficient target information and (b) Homogenized search area. In Figure S6 (a), the goals are located in the water and are very similar to the surrounding patches,



(a) Path end **(b) Path visited**

Figure S5. **GeoExplorer tends to explore more patches in the search areas, especially the inside patches.** (a) *Statistics of the path end.* We count the end location of the 895 paths in the Masa dataset test set for ground truth (goal location), GOMAA-Geo and GeoExplorer when $\mathcal{C} = 4$ and $\mathcal{C} = 8$. (b) *Statistics of the path visited.* We count all the visited patches of 895 paths in the Masa dataset test set for GOMAA-Geo and GeoExplorer when $\mathcal{C} = 4$ and $\mathcal{C} = 8$.



(a) Insufficient target information

(b) Homogenized search area

Figure S6. **Failure case analysis.** (a) Insufficient target information. The target provides limited information for the localization. (b) Homogenized search area. Patches in the search area are similar and could mislead the exploration.

providing limited localization information. Along the generated path, several similar patches have been visited, but not the target patch. In Figure S6 (b), all patches are similar in the search area (e.g., similar building patches in an urban area or mountain patches), which could probably mislead the exploration process.

S5.3. Additional path visualization examples

To further support our findings, we provide additional visualization examples from the SwissViewMonuments dataset (Figure S7), the Masa dataset (Figure S8) and the SwissView100 dataset (Figure S9). These figures illustrate key aspects of GeoExplore's performance, in particular in terms of adaptation to new environments and targets and better exploration strategies, and confirm the quantitative results presented before. In particular, the three figures show that **(1)** GeoExplorer achieves higher success rates (SR) and generalizes better in novel environments (Figure S9, SwissView100 examples) and when faced with unusual and unseen goals (Figure S7, SwissViewMonuments examples). **(2)** GeoExplorer produces also more diverse paths, while the baseline model (GOMAA-Geo) tends to follow edge patches, often navigating towards corners before heading to the goal. This aligns with the statistical observations made in the main paper, which highlighted that the goals were more frequently located on the edges and corners and may lead GOMAA-Geo to overfit in these areas. **(3)** Visualization samples also indicate the robustness of GeoExplorer's exploration. The right panels of Figures S8 illustrate how a slight change in the goal location affects the exploration process for both the Masa dataset. GeoExplorer seems to be more robust and adapt its exploration, demonstrating diverse and flexible path selection. In contrast, the baseline model tends to follow similar paths regardless of these small changes. The right panels of Figure S9 show another controllable configuration on the SwissView100 dataset, where we reverse start and goal locations between the upper and lower examples, which demonstrates GeoExplorer's exploration is more robust to this reversion. Moreover, the images in the two figures are sourced from different platforms, have varying resolutions, and depict different locations. This highlights how the proposed dataset enhances data diversity for the task.

S5.4. Additional intrinsic reward visualization examples

To provide further insights to the intrinsic reward, we provide additional samples from the SwissViewMonuments dataset in Figure S10. The patches with higher intrinsic rewards are usually unique patches in the search area (*e.g.*, the first example of the right column: a green land in an urban region) or the surprising sample along the path (e.g., the first example of the left column: moving from an urban patch to river). The findings indicate the intrinsic rewards are content-aware and improve the model's exploration ability with dense and goal-agnostic guidance.

S6. Discussion

As an emerging research topic, the task configuration of AGL still includes some limitations: (1) Continuous state and action space. Currently, AGL considers a grid-like environment space for states and actions. For example, different states have no overlaps and actions are chosen from a discrete space. This setting could be further improved as a continuous space for both states and actions to meet the requirements of real-world search-and-rescue operations. (2) Real-world development. When developing the models on an UAV agent, there are some other challenges, for example, the noisy ego-pose of the agent and the observation deformation. Those challenges should also be considered for further real-world development of AGL tasks.

As for the methodology, the proposed Curiosity-Driven Exploration has shown impressive exploration ability for the AGL task, and would encourage future work to further understand the exploration pattern of an agent. For example, an in-depth study and analysis of how intrinsic reward affects extrinsic reward and a comprehensive analysis of how to combine those two motivations would further provide insights to not only AGL but also other goal-reaching reinforcement learning tasks.

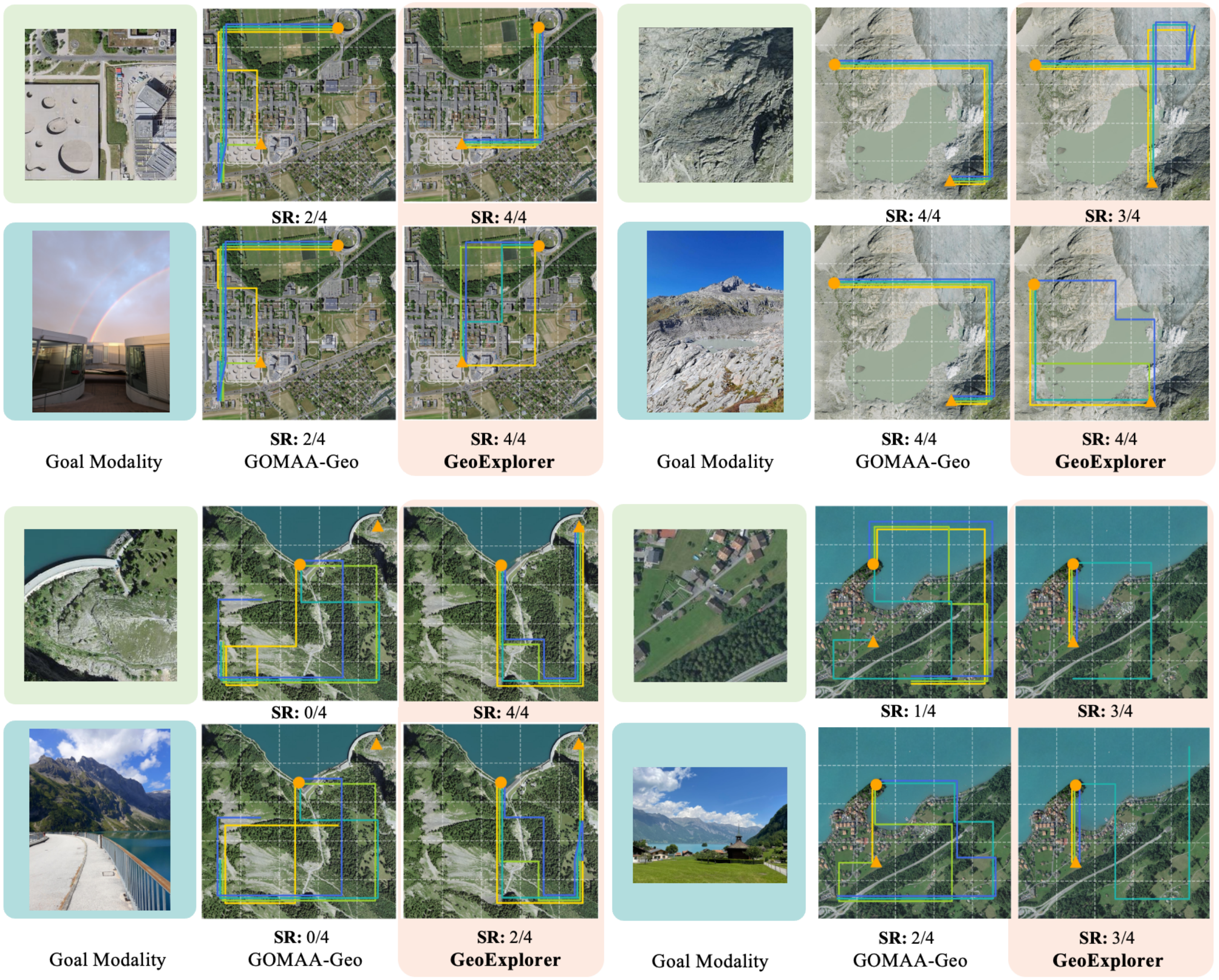


Figure S7. **Path visualization from the SwissViewMonuments dataset** with goals presented in aerial and ground views.

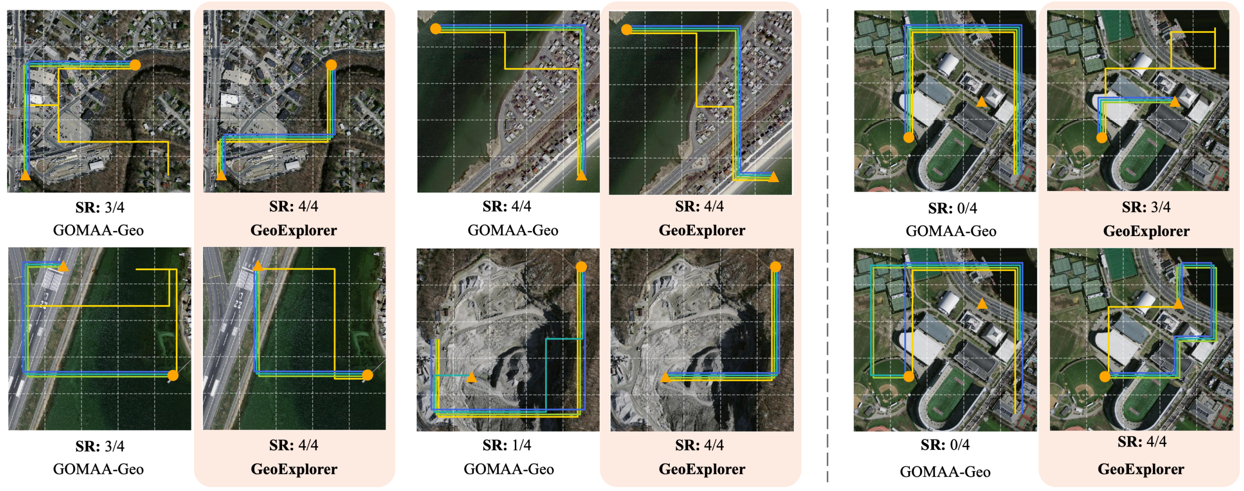


Figure S8. **Path visualization from the Masa dataset test set** with goals presented in aerial view. On the right, we show examples with more controllable {start, goal} configuration: the location of the goal patches changed slightly and the start patch remains the same.

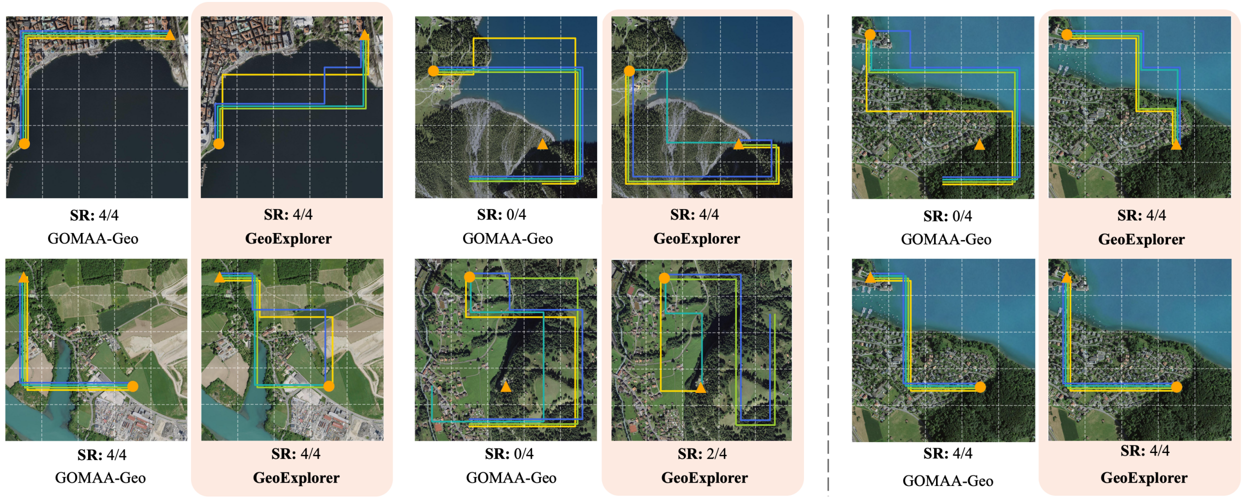


Figure S9. **Path visualization from the SwissView100 dataset** with goals presented in aerial view. On the right, we show examples with more controllable {start, goal} configurations: with reversed start and goal locations.

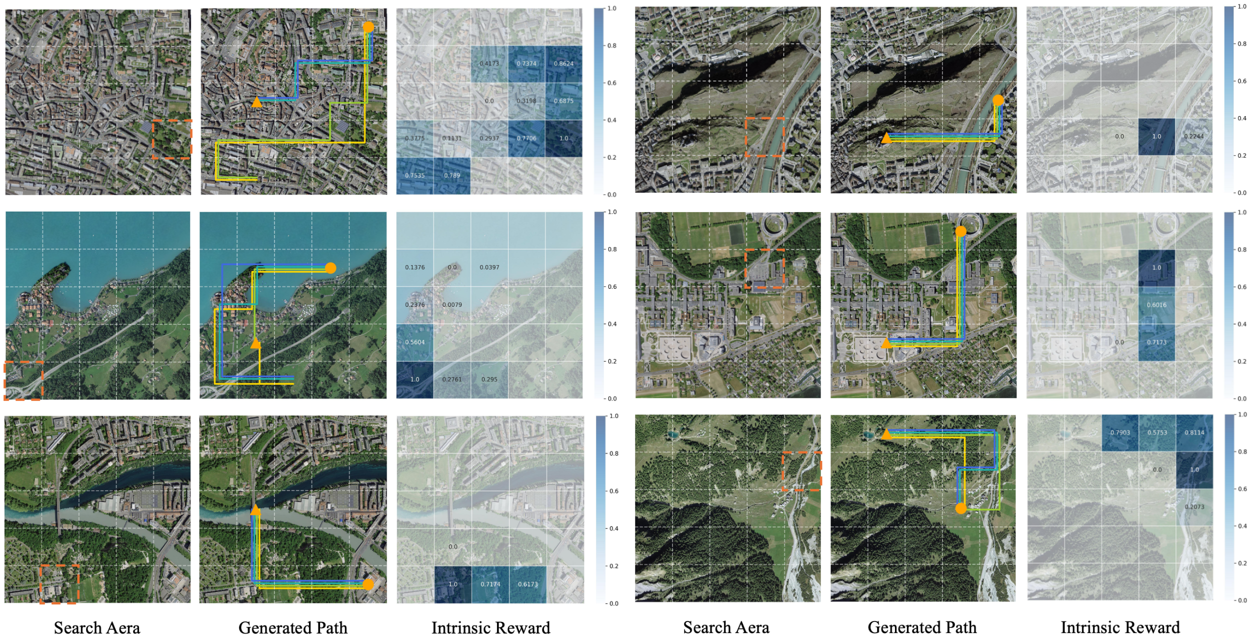


Figure S10. **Intrinsic reward visualization with images from the SwissViewMonuments dataset**. For each sample, from left to right: the search area, path visualization and intrinsic reward per patch. The patch with the highest intrinsic reward is highlighted with an orange rectangle in the search area.

References

[1] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, et al. The Falcon series of open language models. *arXiv preprint arXiv:2311.16867*, 2023. 3

[2] Guillaume Astruc, Nicolas Dufour, Ioannis Siglidis, Constantin Aronssohn, Nacim Bouia, Stephanie Fu, Romain Loiseau, Van Nguyen Nguyen, Charles Raude, Elliot Vincent, et al. OpenStreetView-5M: The many roads to global visual geolocation. In *CVPR*, pages 21967–21977, 2024. 5

[3] Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *CVPR*, pages 15791–15801, 2025. 5

[4] Gabriele Berton, Carlo Masone, and Barbara Caputo. Rethinking visual geo-localization for large-scale applications. In *CVPR*, pages 4878–4888, 2022. 5

[5] Gabriele Berton, Riccardo Mereu, Gabriele Trivigno, Carlo Masone, Gabriela Csurka, Torsten Sattler, and Barbara Caputo. Deep visual geo-localization benchmark. In *CVPR*, pages 5396–5407, 2022. 5

[6] David M. Chen, Georges Baatz, Kevin Köser, Sam S. Tsai, Ramakrishna Vedantham, Timo Pylvänäinen, Kimmo Roimela, Xin Chen, Jeff Bach, Marc Pollefeys, Bernd Girod, and Radek Grzeszczuk. City-scale landmark identification on mobile devices. In *CVPR*, pages 737–744, 2011. 5

[7] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. In *NeurIPS*, pages 15084–15097, 2021. 7

[8] Aayush Dhakal, Adeel Ahmad, Subash Khanal, Srikumar Sastry, Hannah Kerner, and Nathan Jacobs. Sat2Cap: Mapping fine-grained textual descriptions from satellite images. In *CVPRW*, pages 533–542, 2024. 3

[9] Dzung Doan, Yasir Latif, Tat-Jun Chin, Yu Liu, Thanh-Toan Do, and Ian Reid. Scalable place recognition under appearance change for autonomous driving. In *ICCV*, 2019. 5

[10] Yue Fan, Winson Chen, Tongzhou Jiang, Chun Zhou, Yi Zhang, and Xin Eric Wang. Aerial vision-and-dialog navigation. In *ACL Findings 2023*, pages 3043–3061, 2022. 5

[11] Florian Fervers, Sebastian Bullinger, Christoph Bodensteiner, Michael Arens, and Rainer Stiefelhagen. Uncertainty-aware vision-based metric cross-view geolocalization. In *CVPR*, pages 21621–21631, 2023. 5

[12] Jing Gu, Eliana Stefani, Qi Wu, Jesse Thomason, and Xin Eric Wang. Vision-and-language navigation: A survey of tasks, methods, and future directions. In *ACL*, pages 7606–7623, 2022. 5

[13] Ritwik Gupta, Bryce Goodman, Nirav Patel, Ricky Hosfelt, Sandra Sajeev, Eric Heim, Jigar Doshi, Keane Lucas, Howie Choset, and Matthew Gaston. Creating xBD: A dataset for assessing building damage from satellite imagery. In *CVPRW*, pages 10–17, 2019. 1

[14] Mukul Khanna, Ram Ramrakhya, Gunjan Chhablani, Sriram Yenamandra, Theophile Gervet, Matthew Chang, Zsolt Kira, Devendra Singh Chaplot, Dhruv Batra, and Roozbeh Mottaghi. Goat-bench: A benchmark for multi-modal lifelong navigation. In *CVPR*, pages 16373–16383, 2024. 5

[15] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 36:34892–34916, 2023. 1

[16] Shubo Liu, Hongsheng Zhang, Yuankai Qi, Peng Wang, Yanning Zhang, and Qi Wu. AerialVLN: Vision-and-language navigation for uavs. In *ICCV*, pages 15384–15394, 2023. 5

[17] Volodymyr Mnih. *Machine learning for aerial image labeling*. PhD thesis, University of Toronto (Canada), 2013. 1

[18] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 3

[19] Aleksis Pirinen, Anton Samuelsson, John Backsund, and Kalle Åström. Aerial view localization with reinforcement learning: Towards emulating search-and-rescue. *Swedish Artificial Intelligence Society*, pages 28–37, 2023. 1, 5, 7

[20] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI blog*, 2018. 3

[21] Anindya Sarkar, Srikumar Sastry, Aleksis Pirinen, Chongjie Zhang, Nathan Jacobs, and Yevgeniy Vorobeychik. GOMAA-Geo: Goal modality agnostic active geo-localization. In *NeurIPS*, pages 104934–104964, 2024. 1, 3, 4, 5, 6, 7

[22] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 4, 7

[23] Dhruv Shah, Ajay Sridhar, Nitish Dashora, Kyle Stachowicz, Kevin Black, Noriaki Hirose, and Sergey Levine. ViNT: A foundation model for visual navigation. In *CoRL*, 2023. 5

[24] Yujiao Shi, Fei Wu, Akhil Perincherry, Ankit Vora, and Hongdong Li. Boosting 3-dof ground-to-satellite camera localization accuracy via geometry-guided cross-view transformer. In *ICCV*, pages 21516–21526, 2023. 5

[25] Tavis Shore, Oscar Mendez, and Simon Hadfield. PEnG: Pose-enhanced geo-localisation. *IEEE Robotics and Automation Letters*, 10(4):3835–3842, 2025. 5

[26] Vicente Vivanco Cepeda, Gaurav Kumar Nayak, and Mubarak Shah. GeoCLIP: Clip-inspired alignment between locations and images for effective worldwide geo-localization. In *NeurIPS*, pages 8690–8701, 2023. 5

[27] Xiaolong Wang, Runsen Xu, Zhuofan Cui, Zeyu Wan, and Yu Zhang. Fine-grained cross-view geo-localization using a correlation-aware homography estimator. In *NeurIPS*, pages 5301–5319, 2023. 5

[28] Xiangyu Wang, Donglin Yang, Ziqin Wang, Hohin Kwan, Jinyu Chen, Wenjun Wu, Hongsheng Li, Yue Liao, and Si Liu. Towards realistic uav vision-language navigation: Platform, benchmark, and methodology. In *ICLR*, 2025. 5

[29] Scott Workman, Richard Souvenir, and Nathan Jacobs. Wide-area image geolocalization with aerial reference imagery. In *ICCV*, pages 3961–3969, 2015. 5

[30] Zimin Xia and Alexandre Alahi. FG^2: Fine-grained cross-view localization by fine-grained feature matching. In *CVPR*, pages 6362–6372, 2025. 5

[31] Amir R Zamir, Asaad Hakeem, Luc Van Gool, Mubarak Shah, and Richard Szeliski. Introduction to large-scale visual geo-localization. In *Large-Scale Visual Geo-Localization*, pages 1–18. Springer, 2016. 5

[32] Xiaohan Zhang, Xingyu Li, Waqas Sultani, Chen Chen, and Safwan Wshah. GeoDTR+: Toward generic cross-view geolocalization via geometric disentanglement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12): 10419–10433, 2024. 5

[33] Sijie Zhu, Taojiannan Yang, and Chen Chen. VIGOR: Cross-view image geo-localization beyond one-to-one retrieval. In *CVPR*, pages 3640–3649, 2021. 5