International Journal of Computer Vision (2026) 134:183
https://doi.org/10.1007/s11263-026-02739-w

WildIng: A Wildlife Image Invariant Representation Model for Geographical Domain Shift

Julian D. Santamaria[1,2] · **Claudia Isaza**[1] · **Jhony H. Giraldo**[2]

Received: 21 March 2025 / Accepted: 1 January 2026

Abstract

Wildlife monitoring is crucial for studying biodiversity loss and climate change. Camera trap images provide a non-intrusive method for analyzing animal populations and identifying ecological patterns over time. However, manual analysis is time-consuming and resource-intensive. Deep learning, particularly foundation models, has been applied to automate wildlife identification, achieving strong performance when tested on data from the same geographical locations as their training sets. Yet, despite their promise, these models struggle to generalize to new geographical areas, leading to significant performance drops. For example, training an advanced vision-language model, such as CLIP with an adapter, on an African dataset achieves an accuracy of 84.77%. However, this performance drops significantly to 16.17% when the model is tested on an American dataset. This limitation partly arises because existing models rely predominantly on image-based representations, making them sensitive to geographical data distribution shifts, such as variation in background, lighting, and environmental conditions. To address this, we introduce WildIng, a **Wild**life image **In**variant representation model for **g**eographical domain shift. WildIng integrates text descriptions with image features, creating a more robust representation to geographical domain shifts. By leveraging textual descriptions, our approach captures consistent semantic information, such as detailed descriptions of the appearance of the species, improving generalization across different geographical locations. Experiments show that WildIng enhances the accuracy of foundation models such as BioCLIP by 30% under geographical domain shift conditions. We evaluate WildIng on two datasets collected from different regions, namely America and Africa. The code and models are publicly available at https://github.com/Julian075/CATALOG/tree/WildIng.

Keywords Wildlife monitoring · Camera trap images · Geographical domain shift · Foundation models

1 Introduction

Camera trap images are one of the most valuable data sources for wildlife monitoring, playing a crucial role in biodiversity conservation and climate change research (Reynolds et al., 2024; Gadot et al., 2024; Giraldo et al., 2019). These images provide a non-intrusive and scalable way to study animal populations, track endangered species, and understand ecological patterns over time (Pollock et al., 2025; Li et al., 2022; Santamaria et al., 2024). By capturing images in remote locations, camera traps allow researchers to collect extensive datasets without direct human intervention, making them an essential tool for ecological studies. Considering the vast volume of collected images, it is imperative to implement automatic techniques for the identification of animal species present within the images.

With the rise of large-scale deep learning models, researchers have started exploring the use of Foundation Models (FMs) in wildlife monitoring (Yang et al., 2025b; Gabeff et al., 2024; Fabian et al., 2023). FMs are trained on vast and diverse datasets, sometimes containing billions of data samples, allowing them to learn rich and transferable representations (Tang et al., 2025; Yang et al., 2025a; Wu et al., 2024). These models have demonstrated remarkable

Communicated by B Banerjee.

✉ Julian D. Santamaria
julian.santamaria@udea.edu.co

Claudia Isaza
victoria.isaza@udea.edu.co

Jhony H. Giraldo
jhony.giraldo@telecom-paris.fr

[1] SISTEMIC, Faculty of Engineering, Universidad de Antioquia-UdeA, Medellín, Colombia

[2] LTCI, Télécom Paris, Institut Polytechnique de Paris, Palaiseau, France

Published online: 09 March 2026

performance across various computer vision tasks, including image classification, object detection, and semantic segmentation, proving their flexibility and adaptability to different applications (Luo et al., 2024; Riz et al., 2024; Zang et al., 2024).

Recently, researchers have begun adapting FMs for camera trap image recognition. Instead of training models from scratch, current approaches aim to fine-tune (Yang et al., 2025b) or adjust pre-trained FMs to incorporate domain-specific knowledge (Fabian et al., 2023). Some methods introduce adapters, which allow models to specialize in camera trap images without losing their general knowledge (Pantazis et al., 2022). Other models apply *learning without forgetting* strategies, ensuring that models retain their broad capabilities while improving performance on wildlife images (Gabeff et al., 2024). Additionally, some approaches leverage external knowledge sources, such as internet databases, to refine the model's understanding of specific animal species and their attributes (Fabian et al., 2023). These strategies aim to bridge the gap between the general-purpose knowledge of FMs and the specialized needs of camera trap image recognition.

Despite their impressive performance with in-domain geographical data (data coming from the same geographical locations), these FM-based approaches often struggle when tested on out-of-domain geographical data (training and test data come from different geographical locations) (Hogeweg et al., 2024; Norman et al., 2023; Tuia et al., 2022). This limitation is particularly problematic for camera trap applications, where the geographical locations differ substantially from those seen during the training phase (Schneider et al., 2020; Beery et al., 2018; Gomez-Villa et al., 2017).

We observe that incorporating text into the input representation for camera trap images helps extract stronger features, alleviating the geographical domain shift issue. In contrast, current models depend only on visual features, which are highly sensitive to changes in data distribution (Fang et al., 2025; Yu et al., 2023). Furthermore, many of these models are built on CLIP (Radford et al., 2021), which has shown a tendency to lose its generalization ability (Wang & Kang, 2025; Li et al., 2025a) and become more susceptible to spurious correlations (Wang et al., 2024; Kempf et al., 2025) when the model is fine-tuned. As a result, CLIP-based models (e.g., WildCLIP (Gabeff et al., 2024) and BioCLIP (Stevens et al., 2024)) that rely solely on visual features often struggle to recognize images correctly in new geographical locations. Figure 1 provides an example of these observations, showing how such geographical variations cause the model's learned features to fail in generalizing effectively, leading to misclassifications (Liang et al., 2023; Wald et al., 2021).

In this paper, we introduce the **Wild**life image **In**variant representation model for **g**eographical domain shift (WildIng). Our approach introduces a simple yet effective new

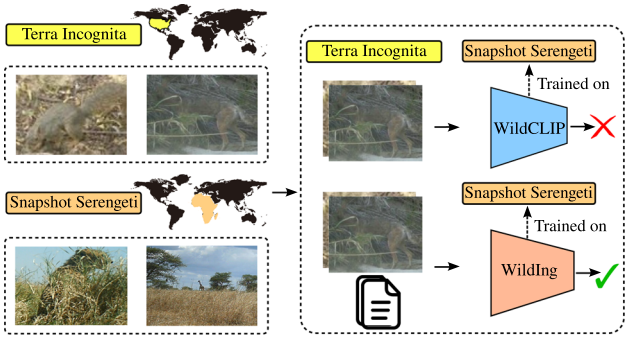

Fig. 1 Comparison of WildIng and WildCLIP (Gabeff et al., 2024) under geographical domain shift. Both models are trained on the Snapshot Serengeti dataset from Africa (Swanson et al., 2015) and evaluated on the Terra Incognita dataset from the United States (Beery et al., 2018). WildIng demonstrates superior performance

representation for wildlife monitoring data to address geographical domain shifts. This representation consists of using visual features in addition to text descriptions about the input images. This approach allows the model to capture geographical domain-invariant features by leveraging text descriptions, which remain consistent across different geographical regions. WildIng consists of three main components: a text encoder, which includes a Large Language Model (LLM); an image encoder; and an image-text encoder, which incorporates a Vision-Language Model (VLM) and a Multi-Layer Perceptron (MLP). The MLP is used to address the domain shift between encoders, caused by the different feature spaces introduced by the VLM and LLM components (Duan et al., 2022). An overview of the architecture is provided in Section 3.2 and illustrated in Figure 2.

To evaluate WildIng, we train our model on one dataset and test it on another dataset from a different geographical region. This setup allows us to analyze how well the model adapts to new environments where differences in background, lighting, and species composition create geographical domain shifts. Our results show that WildIng either outperforms or achieves competitive performance compared to general-purpose and domain-specific FMs in camera trap image recognition, particularly when the training and testing distributions differ due to geographical variations.

In this work, we build upon and improve our preliminary study (Santamaria et al., 2025). To achieve this, we introduce modifications to the model architecture and perform additional experiments. More specifically, we replace the backbone of our model by changing the combination of CLIP (Contrastive Language-Image Pre-Training) (Radford et al., 2021) and BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) with Long-CLIP (Zhang et al., 2024a). Additionally, we modify the class representation, using only the information provided by LLMs. We also evaluate WildIng's robustness to multiple random

initializations to analyze the effect of the introduced changes on its performance. Furthermore, we add new baselines for comparison. Finally, we perform additional ablation studies and sensitivity analyses to provide a deeper understanding of the contribution of each component in our approach.

In summary, our main contributions are:

- We introduce a novel WildIng model to represent wildlife monitoring data, enhancing the extraction of geographical domain-invariant features.
- When tested on datasets that differ from its training data, WildIng outperforms previous FMs in recognizing animal species from camera trap images.
- We conduct a series of ablation studies to validate the effectiveness of each component in our model.

2 Related Work

2.1 Foundation Models

In recent years, FMs have emerged as a new approach that achieves remarkable performance across a wide range of tasks without requiring task-specific training. These models leverage large-scale pre-training to learn high-level representations, leading to significant advancements in the field of machine learning (Awais et al., 2025; Huang et al., 2024; Jiang et al., 2023; Touvron et al., 2023). A key advancement in this field was CLIP (Radford et al., 2021), which introduced a new learning approach by aligning visual features with text descriptions. CLIP significantly improved generalization across different tasks. Later models, such as Long-CLIP (Zhang et al., 2024a), extend the sequence length for better contextual understanding. Furthermore, CLIP-Adapter (Gao et al., 2024), which refines CLIP's learned representations using lightweight adaptation layers, continues to improve the alignment between visual features and text descriptions. More recently, LLMs and VLMs have demonstrated strong capabilities in processing and generating both textual and visual content (Zhang et al., 2024; Abdin et al., 2024; Li et al., 2023). Examples include GPT-4 (Achiam et al., 2023) and LLaVA (Liu et al., 2024), which leverage large-scale datasets to improve language and vision understanding across various applications.

2.2 Foundation Models for Biology

FMs have been adapted to address domain-specific challenges, particularly in biological research, where data is often complex and specialized. Most of the adaptations of FM in biology are related to processing text, extracting biological information (Jung et al., 2024; Lam et al., 2024), and modeling biological structures (Garau-Luis et al., 2025; Jumper et al., 2021). Beyond language processing and structural modeling, FMs have also been applied to vision-based biological tasks. One example is BioCLIP (Stevens et al., 2024), which extends the principles of CLIP (Radford et al., 2021) to biological data, enabling the classification of diverse categories such as plants, animals, and fungi. Unlike general-purpose vision-language models, BioCLIP integrates structured biological knowledge and leverages taxonomic information, improving performance in fine-grained classification tasks (Stevens et al., 2024).

2.3 Foundation Models for Camera Trap Images

The adaptation of FMs has extended beyond general and biological applications to camera trap image recognition, where they play a crucial role in wildlife monitoring and conservation. One such model is WildCLIP, which leverages CLIP's ability to align visual features with text descriptions to accurately classify animal species in camera trap images (Gabeff et al., 2024). Similarly, WildMatch introduces a zero-shot classification framework by generating detailed visual descriptions of camera trap images and matching them to an external knowledge base for species identification (Fabian et al., 2023). Another approach, Eco-VLM, enhances models that align visual features with text descriptions for ecological applications by fine-tuning on wildlife-specific datasets and applying text augmentation techniques (Yang et al., 2025b). In contrast to models that align visual features with text descriptions, a more traditional deep learning approach was proposed by Gadot et al. (2024), who explored large-scale training for EfficientNetV2-M (Tan & Le, 2021), a CNN-based architecture.

While previous methods have significantly improved camera trap image recognition in geographically in-domain evaluation, they still struggle when applied to different geographical regions and unseen species (Zhu et al., 2024; Simões et al., 2023; Gadot et al., 2024). To address this limitation, our proposal introduces a more robust representation for wildlife monitoring data. It leverages detailed descriptions from a VLM to incorporate semantic invariant features, which are then used together with image features. Furthermore, the inclusion of more detailed class descriptions generated by the LLM improves the alignment between the input representation and the corresponding class. This approach improves the input representation, enhancing robustness to geographical domain shifts.

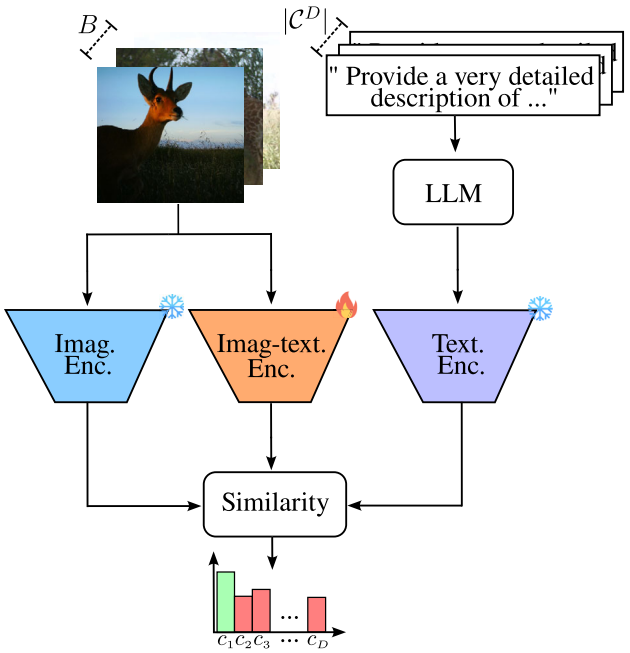

Fig. 2 Overview of WildIng. The model integrates image, text, and image-text encoders along with an LLM. By leveraging text descriptions and image features, it extracts invariant features, improving robustness against geographical domain shifts

3 WildIng

3.1 Problem Definition

The objective of this paper is to train a model in an annotated dataset of camera trap images from a specific geographical location, denoted as $\mathcal{D}$, which consists of N_d image-label pairs, $\mathcal{D} = \{(\mathbf{x}_i^D, \mathbf{y}_i^D)\}_{i=1}^{N_d}$, with a set of classes $\mathcal{C}^D$. Therefore, we evaluate the model's performance on a different camera trap image dataset from another geographical location, denoted as $\mathcal{S}$, which represents a distinct geographical domain, containing N_s image-label pairs, $\mathcal{S} = \{(\mathbf{x}_i^S, \mathbf{y}_i^S)\}_{i=1}^{N_s}$, with a set of classes $\mathcal{C}^S$. The set of classes of both datasets may or may not overlap, meaning that $\mathcal{C}^D \cap \mathcal{C}^S$ may or may not be empty. Both datasets are derived from the natural world, but their image distributions differ due to being collected from different geographical regions, as illustrated in Figure 1. Our goal is to train a deep learning model using only the training dataset $\mathcal{D}$ and deploy it on the testing dataset $\mathcal{S}$. This setting is highly practical in camera trap image research because the data used for testing usually comes from a different geographical domain than the training data.

3.2 Overview of the Approach

The architecture of our proposed model, WildIng, is illustrated in Figure 2. It consists of three main components: i) text encoder, ii) image encoder, and iii) image-text encoder. In the text component, WildIng uses an LLM to extract class-specific knowledge for each category in our dictionary of classes, $\mathcal{C}^D$. Then, the LLM-generated descriptions are processed by the text encoder. The resulting text embeddings are used to compute class-specific centroids for each class in $\mathcal{C}^D$ (Section 3.3). This process produces a single embedding of dimension F. For the image component, the model uses the image encoder to extract embeddings from a mini-batch of B images (Section 3.4). In the image-text encoder, WildIng employs a VLM coupled with a text encoder and an MLP to compute image-text embeddings from the mini-batch of images (Section 3.5). Text, image, and image-text embeddings are matched using a similarity mechanism (Section 3.6). Finally, we utilize the output of the similarity mechanism to compute a contrastive loss, which is used to train our model (Section 3.7). Most modules in WildIng are frozen (❄) apart from the image-text encoder (🔥).

3.3 Text Encoder

To generate textual descriptions for each category in our dataset, $\mathcal{C}^D$, we use an LLM that provides detailed information about the animals without requiring expert input. The prompt used to extract these descriptions from the LLM is provided in Appendix Appendix A. The LLM generates M_c short descriptions for each class $c \in \mathcal{C}^D$. WildIng assumes that this approach introduces more diverse information for representing each class. The generated descriptions are then processed by WildIng using the text encoder. To obtain the final embedding, the model computes the centroid of the resulting embeddings. Specifically, let $\mathbf{P}^{(c)} \in \mathbb{R}^{M_c \times F}$ be the set of M_c embeddings obtained from the LLM-generated descriptions for class c. The centroid for each class c is computed as:

$$\mathbf{t}^{(c)} = \frac{1}{M_c} \sum_{i=1}^{M_c} \mathbf{P}_i^{(c)}, \tag{1}$$

where $\mathbf{P}_i^{(c)}$ represents the i-th row of $\mathbf{P}^{(c)}$, corresponding to an individual textual description embedding. Finally, the output of the text embedding component in WildIng is a matrix:

$$\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_{|\mathcal{C}^D|}]^\top \in \mathbb{R}^{|\mathcal{C}^D| \times F}, \tag{2}$$

which contains the final embeddings for all classes in $\mathcal{C}^D$.

3.4 Image Encoder

3.4.1 Pre-processing

We employ an object detection model to process our camera trap datasets, aiming to extract image crops that contain relevant information for analysis.

3.4.2 Image Embeddings

WildIng employs an image encoder to extract feature embeddings from cropped images. The images are processed in mini-batches of size B, where each image is transformed into an embedding of dimension F using the image encoder. The output of this stage is a matrix:

$$\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_B]^\top \in \mathbb{R}^{B \times F}, \tag{3}$$

where $\mathbf{v}_i$ represents the visual embedding of the i-th image in the mini-batch. This matrix is used as input for the subsequent stages of our framework, where text and image embeddings are aligned and contrasted.

3.5 Image-text Encoder

In the image-text branch of WildIng, we use the mini-batch of cropped images as input. To generate textual descriptions of the animals in these images, we utilize an image-text encoder, as illustrated in Figure 3. This encoder consists of three main components: a VLM, a text encoder, and an MLP. First, the VLM generates textual descriptions based on the input images, using a prompt similar to the one described in (Fabian et al., 2023) and provided in the Appendix Appendix A. Therefore, these textual descriptions are processed using the text encoder. Finally, WildIng applies an MLP to refine the extracted embeddings by introducing trainable parameters. As demonstrated in Section 4, incorporating trainable parameters improves the model's performance. However, effective alignment between the image embeddings and the projected representations requires a dedicated similarity mechanism and a contrastive loss function, as detailed in Section 3.6 and Section 3.7.

The output of the image-text encoder of WildIng is a matrix:

$$\mathbf{L} = [\mathbf{l}_1, \mathbf{l}_2, \ldots, \mathbf{l}_B]^\top \in \mathbb{R}^{B \times F}, \tag{4}$$

where $\mathbf{l}_i$ represents the transformed embedding of the i-th image description in the mini-batch.

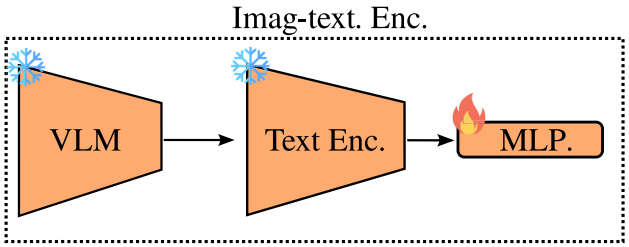

Fig. 3 Detailed illustration of the image-text module, which consists of a VLM, a text encoder, and an MLP. This module processes input images and converts them into image-text embeddings

3.6 Similarity Mechanism

The embeddings from the three types of modalities, text (**T**), image (**V**), and image-text (**L**), are the inputs for the similarity method. The process incorporates two stages: i) similarity computation and ii) weighted integration. In the first stage, WildIng computes the cosine similarities between text and image embeddings, as well as text and image-text embeddings. Specifically, let $\mathbf{W} \in \mathbb{R}^{B \times |\mathcal{C}^D|}$ be the matrix of cosine similarities between the text and image embeddings, computed as follows:

$$\mathbf{W}_{ij} = \frac{\langle \mathbf{v}_i, \mathbf{t}_j \rangle}{\|\mathbf{v}_i\| \|\mathbf{t}_j\|} \; \forall \, 1 \leq i \leq B, 1 \leq j \leq |\mathcal{C}^D|, \tag{5}$$

where $\mathbf{W}_{ij}$ represents the (i, j) item of the matrix, $\langle \cdot, \cdot \rangle$ denotes inner product, and $\| \cdot \|$ is the ℓ_2 norm of a vector. Along the same process, WildIng calculates the cosine similarities between the text and image-text embeddings as follows:

$$\mathbf{Q}_{ij} = \frac{\langle \mathbf{l}_i, \mathbf{t}_j \rangle}{\|\mathbf{l}_i\| \|\mathbf{t}_j\|} \; \forall \, 1 \leq i \leq B, 1 \leq j \leq |\mathcal{C}^D|, \tag{6}$$

where $\mathbf{Q} \in \mathbb{R}^{B \times |\mathcal{C}^D|}$ is the matrix of cosine similarities between the text and image-text embeddings.

Both cosine similarities are combined using a weighted average between the matrices $\mathbf{W}$ and $\mathbf{Q}$, where the weights are controlled by the hyperparameter $\alpha \in [0, 1]$. Specifically, the output of the weighted integration is a matrix $\mathbf{S} \in \mathbb{R}^{B \times |\mathcal{C}^D|}$, defined as follows:

$$\mathbf{S} = \alpha \mathbf{W} + (1 - \alpha)\mathbf{Q}. \tag{7}$$

Since $\alpha \in [0, 1]$, the resulting matrix $\mathbf{S}$ is a convex combination of $\mathbf{W}$ and $\mathbf{Q}$. As a result, each element $\mathbf{S}_{ij}$ in the matrix is also between 0 and 1.

3.7 Contrastive Loss

We train our model using a contrastive loss function, $\mathcal{L}$, which takes the matrix $\mathbf{S}$ as input. The loss function is calculated for each mini-batch as follows:

$$\mathcal{L}(\mathbf{S}) = \frac{1}{B}\sum_{i=1}^{B} -\log \frac{\exp(\mathbf{S}_{ik}/\tau)}{\sum_{j=1}^{|\mathcal{C}^D|}\exp(\mathbf{S}_{ij}/\tau)}, \tag{8}$$

where τ is a temperature hyperparameter and k is the index of the class in $\mathcal{C}^D$ of the ith image in the mini-batch. This loss function aims to ensure that the embeddings corresponding to the same species category are brought closer together in the feature space.

4 Experiments and Results

In this section, we describe the datasets used in this work, the evaluation protocol, implementation details, results, and a discussion of WildIng. We compare our proposal with CLIP (Radford et al., 2021), CLIP-Adapter (Gao et al., 2024), Long-CLIP (Zhang et al., 2024a), BioCLIP (Stevens et al., 2024), WildCLIP (Gabeff et al., 2024), and some adaptations of Long-CLIP and BioCLIP. Additionally, we conduct ablation studies to analyze the contribution of each component of WildIng, such as the image encoder, the image-text encoder, and LLM. We explore the effect of incorporating a template set to introduce task-specific information and evaluate different LLMs to assess their impact on performance. Finally, we investigate the sensitivity of WildIng to the hyperparameter α in the similarity mechanism, and to the number of LLM-prompted sentences in the text encoder. All evaluations are reported using accuracy, and macro F1-score.

4.1 Datasets

We evaluate WildIng using two publicly available camera trap datasets from different geographical regions: Snapshot Serengeti (Swanson et al., 2015), collected in savanna environments in Africa using Scoutguard cameras, and Terra Incognita (Beery et al., 2018), collected in the southwest of the United States, where the predominant environment is semi-arid desert and pinyon–juniper woodland (Archer & Predick, 2008). Information about the specific camera trap models used in the Terra Incognita dataset is not specified. This dataset presents several visual challenges, such as poor illumination (especially at night), motion blur due to low shutter speed, occlusions from vegetation or frame edges, and forced perspective when animals appear very close to the camera. Examples of cropped images from these datasets are shown in Figure 4 and their class distributions are shown in Figure 5 and Figure 6.

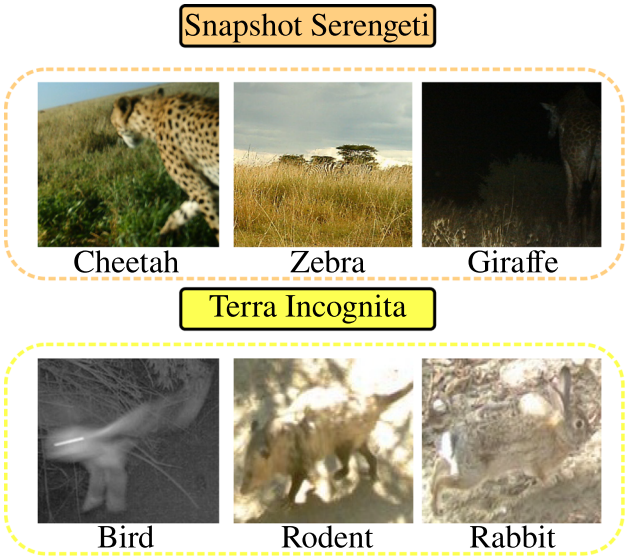

Fig. 4 Cropped images from the Snapshot Serengeti and Terra Incognita datasets where we observe the geographical domain shift and the difference in classes (different taxonomic groups)

- *Snapshot Serengeti* (Swanson et al., 2015). We use the version of the Snapshot Serengeti dataset adopted in WildCLIP (Gabeff et al., 2024), which consists of 46 classes. This dataset version contains 380 × 380 pixel image crops, generated by the MegaDetector model from the Snapshot Serengeti project, using a confidence threshold above 0.7. Only images containing single animals were selected. The dataset includes a total of 340, 972 images, with 230, 971 for training, 24, 059 for validation, and 85, 942 for testing.
- *Terra Incognita* (Beery et al., 2018). This dataset consists of 16 classes and introduces two testing groups: Cis-locations and Trans-locations. Cis-locations contain images similar to the training data, while Trans-locations feature images from different environments. These partitions were originally designed to assess the robustness of computer vision models in an in-domain evaluation setting. We filter the images in this dataset using the MegaDetector model from the PyTorch-Wildlife library (Hernandez et al., 2024). The dataset contains a total of 45, 912 images, distributed as follows: 12, 313 for training, 1, 932 for Cis-validation, 1, 501 for Trans-validation, 13, 052 for Cis-test, and 17, 114 for Trans-test.

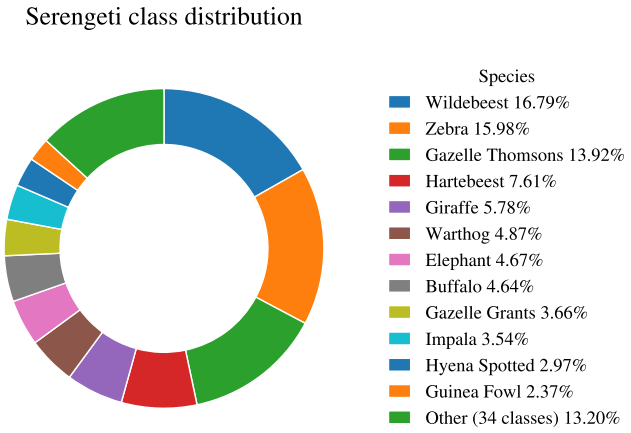

Fig. 5 Class distribution of the Serengeti dataset

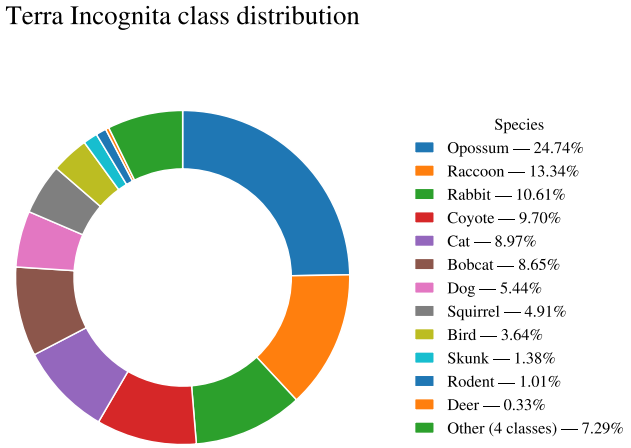

Fig. 6 Class distribution of the Terra Incognita dataset

4.2 Evaluation Protocol

We conduct two experiments to assess the performance of our model in comparison to state-of-the-art (SOTA) methods. In the first experiment, we use the Snapshot Serengeti dataset for training and validation and the Terra Incognita dataset for testing. This cross-dataset setup allows us to evaluate the model's performance under geographical domain shift, i.e., geographical out-of-domain evaluation. Formally, we define $\mathcal{D}$ as the Snapshot Serengeti dataset and $\mathcal{S}$ as the Terra Incognita dataset. Snapshot Serengeti was collected in various protected areas in Africa, whereas Terra Incognita originates from the southwest of the United States. This experimental setup introduces two key challenges: i) a distribution shift between the datasets $\mathcal{D}$ and $\mathcal{S}$ (geographical domain shift), and ii) a discrepancy in the set of classes, where $\mathcal{C}^D \neq \mathcal{C}^S$. These challenges are illustrated in Figure 4. Due to the difference in class sets, closed-set SOTA methods cannot be used for comparison. We report accuracy results on the Cis-Test and Trans-Test subsets of Terra Incognita.

In the second experiment, we modify our problem definition from Section 3 and use the same dataset for training, validation, and testing to evaluate the model without the challenges introduced by the geographical domain shift and novel classes, i.e., under in-domain evaluation. Specifically, we train and evaluate the model on either the Snapshot Serengeti or the Terra Incognita datasets. This approach allows us to assess the model's accuracy and robustness within a consistent domain.

4.2.1 Implementation Details

For pre-processing, we employ the MegaDetector model (Beery et al., 2019). WildIng is implemented using the 3.5 version of ChatGPT for the LLM, the LongCLIP-B version of Long-CLIP for the text and image encoder, and the 1.5-7B version of LLaVA for the VLM. For training our model in both experiments, we set $\tau = 0.1$. The MLP architecture for the first experiment consists of a single hidden layer with a dimension of 793 and employs the Rectified Linear Unit (ReLU) as the activation function. Additionally, a skip connection is implemented between the input and output layers. Additionally, we train the model for 30 epochs and set $\alpha = 0.5$. Optimization is performed using the Stochastic Gradient Descent (SGD) algorithm with a learning rate of 0.09, a momentum of 0.80, and a batch size of 128. We perform our experiments on GPUs Tesla P100-PCIE-16GB.

In the second experiment, we fine-tune WildIng by unfreezing the image encoder and adjusting key hyperparameters, including the $\alpha = 0.6$. Specifically, we use a batch size of 256 and train the model for 57 epochs using SGD with a momentum of 0.82 and a learning rate of $1e{-}3$. For the Snapshot Serengeti dataset, we apply an MLP with a single hidden layer with 256 dimensions. For the Terra Incognita dataset, we set an MLP with a single hidden layer of 733 dimensions. Early stopping is applied with a patience of 5 epochs.

To optimize the hyperparameters, we use a random search. Furthermore, we evaluate the best hyperparameter combination using 100 different random seeds for the first experiment and 3 random seeds for the second experiment. For details on the search space and additional information, please refer to Appendix Appendix B.

4.3 Quantitative Results

4.3.1 Comparison with the SOTA in Out-of-domain Evaluation

Table 1 presents the zero-shot performance evaluation of various models trained on datasets such as ShareGPT4V and Snapshot Serengeti and evaluated on the Cis-Test and Trans-Test sets of the Terra Incognita dataset. Models such

Table 1 Zero-shot performance results of WildIng and other foundation models on the Terra Incognita dataset (out-of-domain evaluation). All methods are trained on data different from the test dataset. The best method is highlighted in **bold**, and the second-best is underlined. Results are reported in accuracy (%) and macro F1 score (F1-M)

Model	Training	Cis-Test	F1-M	Trans-Test	F1-M
CLIP	OpenAI data	39.14	0.39	34.67	0.32
Long-CLIP	ShareGPT4V	42.41	0.41	37.55	0.34
BioCLIP	TREEOFLIFE-10M	21.12	0.20	14.53	0.15
CLIP Adapter	Snapshot Serengeti	$27.45_{\pm 2.84}$	$0.25_{\pm 0.03}$	$16.17_{\pm 3.37}$	$0.18_{\pm 0.03}$
Long-CLIP Adapter	Snapshot Serengeti	$30.40_{\pm 1.58}$	$0.28_{\pm 0.02}$	$18.73_{\pm 1.54}$	$0.19_{\pm 0.02}$
BioCLIP Adapter	Snapshot Serengeti	$14.21_{\pm 3.30}$	$0.12_{\pm 0.03}$	$8.59_{\pm 3.02}$	$0.08_{\pm 0.01}$
WildCLIP	Snapshot Serengeti	$41.62_{\pm 0.40}$	$0.39_{\pm 0.01}$	$37.52_{\pm 0.42}$	$0.31_{\pm 0.01}$
WildCLIP-LwF	Snapshot Serengeti	$\underline{43.67}_{\pm 0.12}$	$\underline{0.40}_{\pm 0.01}$	$\mathbf{40.17}_{\pm 0.05}$	$\underline{0.34}_{\pm 0.01}$
WildIng (ours)	Snapshot Serengeti	$\mathbf{50,06}_{\pm 2.39}$	$\mathbf{0.43}_{\pm 0.01}$	$\underline{39.96}_{\pm 1.89}$	$\mathbf{0.36}_{\pm 0.01}$

Table 2 Performance comparison in Snapshot Serengeti (in-domain evaluation). All models use the ViT-B/16 backbone. The best method is highlighted in **bold**, and the second-best is underlined. Results are reported in accuracy (%)

Model	Loss Function	Test
Linear Probe CLIP	Cross-entropy	$\underline{84.84}_{\pm 0.42}$
CLIP Adapter	Contrastive	$84.77_{\pm 0.26}$
Long-CLIP Adapter	Contrastive	$83.97_{\pm 1.03}$
BioCLIP Adapter	Contrastive	$80.47_{\pm 0.89}$
WildCLIP	Contrastive	$68.73_{\pm 0.28}$
WildCLIP-LwF	Contrastive	$69.53_{\pm 0.02}$
WildIng (ours)	Contrastive	$\mathbf{90.74}_{\pm 0,05}$

as CLIP, BioCLIP, and Long-CLIP (first three rows) are reported without their standard deviation, as we used the pre-trained models provided by the authors of each model.

For the remaining cases, we report the results obtained by training the model with 100 different random seeds. The results indicate that CLIP, trained on OpenAI data, achieves a Cis-Test accuracy of 39.14% and a Trans-Test accuracy of 34.67%, while Long-CLIP, trained on ShareGPT4V, improves these metrics to 42.41% and 37.55%, respectively. BioCLIP, despite being trained on TREEOFLIFE-10M, struggles to generalize to the Terra Incognita dataset, achieving only 21.12% on Cis-Test and 14.53% on Trans-Test. Based on CLIP Adapter (Gao et al., 2024), we evaluate this strategy and extend it to Long-CLIP and BioCLIP. The results for these adapter-based models show lower performance compared to models without adapters. Specifically, the original CLIP Adapter achieves an accuracy of 27.45% in Cis-Test. The Long-CLIP Adapter and BioCLIP Adapter variants obtain 30.4% and 14.21% accuracy in Cis-Test, while their Trans-Test accuracies are 16.17%, 18.73%, and 8.59%, respectively. Overall, these results suggest that adapter-based models are highly domain-specific and further widen the gap between domains compared to the original architectures.

WildCLIP and its Learning without Forgetting (LwF) variant, both trained on Snapshot Serengeti, show better performance, achieving 41.62% and 43.67% on Cis-Test, and 37.52% and 40.17% on Trans-Test, respectively. Our proposed method, WildIng, outperforms all previously evaluated models on Cis-Test, achieving 50.06% accuracy, and obtains the second-best accuracy on Trans-Test with 39.96%, demonstrating its effectiveness in geographical out-of-domain evaluation. Furthermore, the macro F1 scores highlight that WildIng is less biased toward majority classes (see Fig. 6 for the class distribution of Terra Incognita), this behavior is observed in both test sets (Cis-Test and Trans-Test). WildIng is the only model that improves the macro F1 score relative to its base model (LongCLIP in our case), increasing it from 0.41 to 0.45 on Cis-Test and from 0.34 to 0.37 on Trans-Test. These results highlight that WildIng not only generalizes well across domains, but also improves per-class performance despite being trained on a highly imbalanced dataset such as Snapshot Serengeti (see Fig. 5).

In addition, the standard deviation of our proposal is comparable to that of SOTA models such as CLIP Adapter, Long-CLIP Adapter, and BioCLIP Adapter. These results highlight the advancements of WildIng in learning geographical domain-invariant representations and improving open-set recognition. By leveraging additional semantic information to represent the input, WildIng surpasses previous SOTA models in handling geographical domain shifts and recognizing unseen classes.

4.3.2 Trainable Parameters and Computational Cost

In our comparison of trainable parameters, we exclude the zero-shot models CLIP, Long-CLIP, and BioCLIP, since these models are not trained as part of this work. WildIng has a total of 813,339 trainable parameters. This is significantly lower than the WildCLIP and WildCLIP-LwF models, which have approximately 86 million trainable parameters. This difference is because those models unfreeze the CLIP image encoder during training. Nevertheless, WildIng achieves best

performance in out-of-domain evaluation on the Cis-Test set, with an accuracy of 50.06% and ranks second on the Trans-Test set with an accuracy of 39.96% (see Table 1). On the other hand, the adapter-based models (CLIP Adapter, Long-CLIP Adapter, and BioCLIP Adapter) have the smallest number of trainable parameters, with 262,914. However, these adapter-based models achieved the worst performance, as shown in Table 1. Finally, it is clear that WildIng offers a good trade-off between model complexity and performance.

For computational cost, training the full model of WildIng on the Snapshot Serengeti dataset takes around 30 minutes using a single Tesla P100-PCIE-16GB GPU. For comparison, training the standard version of WildCLIP (Gabeff et al., 2024) in the same dataset on A100-PCIE-40GB takes 13.4 hours. For that reason, it is clear that our proposal offers a lightweight alternative for addressing geographical domain shift, instead of training a large model such as WildCLIP.

4.3.3 In-domain Performance Comparison in the Snapshot Serengeti Dataset

Table 2 shows a comparison of multiple models trained in the Snapshot Serengeti dataset. The Linear Probe CLIP model, trained with a cross-entropy loss, achieves a test accuracy of 84.84%. Despite this performance, it lacks open vocabulary capabilities due to its supervised training approach. Among the adapter-based models, the CLIP Adapter, Long-CLIP Adapter, and BioCLIP Adapter achieve test accuracies of 84.77%, 83.97%, and 80.47%, respectively. These results suggest that adapter-based methods are a good option for in-domain evaluation. WildCLIP achieves a test accuracy of 68.73%, demonstrating moderate performance but performing worse than both adapter-based models and our method. The WildCLIP-LwF variant improves this performance, reaching 69.53%. The LwF strategy contributes positively to retaining learned information, but its improvement remains limited compared to other models like CLIP Adapter and WildIng. The proposed method, WildIng, achieves the highest test accuracy of 90.74%, outperforming all previously evaluated models while maintaining a low standard deviation.

4.3.4 In-domain Performance Comparison in the Terra Incognita Dataset

Similar to Section 4.3.3, Table 3 presents a comparison of different models in the geographical in-domain evaluation. All models in this comparison are trained and evaluated on the Terra Incognita dataset.

The Linear Probe CLIP model achieves a Cis-Test accuracy of 78.09% and a Trans-Test accuracy of 67.23%. Among the adapter-based methods, the CLIP Adapter, Long-CLIP Adapter, and BioCLIP Adapter achieve Cis-Test accuracies of 79.45%, 79.11%, and 77.09%, respectively, while their Trans-Test accuracies are 69.10%, 69.15%, and 63.45%. These results indicate that adapter-based models provide competitive performance in in-domain evaluation.

Table 3 Performance comparison in Terra Incognita (in-domain evaluation). All models use the ViT-B/16 backbone. The best method is highlighted in **bold**, and the second-best is underlined. Results are reported in accuracy (%).

Model	Cis-Test	Trans-Test
Linear Probe CLIP	$78.09_{\pm 0.34}$	$67.23_{\pm 2.31}$
CLIP Adapter	$79.45_{\pm 1.21}$	$69.10_{\pm 2.70}$
Long-CLIP Adapter	$79.11_{\pm 0.47}$	$69.15_{\pm 1.33}$
BioCLIP Adapter	$77.09_{\pm 1.19}$	$63.45_{\pm 0.85}$
WildCLIP	$\mathbf{91.70}_{\pm 0.21}$	$\mathbf{84.47}_{\pm 0.24}$
WildCLIP-LwF	$\underline{88.93}_{\pm 0.02}$	$\underline{82.91}_{\pm 0.01}$
WildIng (ours)	$84.10_{\pm 3.03}$	$75.80_{\pm 5.38}$

WildCLIP achieves a Cis-Test accuracy of 91.70% and a Trans-Test accuracy of 84.47%, demonstrating the highest performance among all evaluated models. The LwF variant of WildCLIP slightly underperforms its base version, reaching a Cis-Test accuracy of 88.93% and a Trans-Test accuracy of 82.91%.

The proposed method achieves a Cis-Test accuracy of 84.10%, outperforming the adapter-based models and the Linear Probe CLIP. In the Trans-Test scenario, WildIng achieves 75.80%, showing competitive performance but lower than WildCLIP and WildCLIP-LwF. However, we observe a significant increase in the standard deviation due to the smaller size of the Terra Incognita dataset, which limits the robustness when evaluated with only three random seeds. This effect is further amplified by unfreezing the image encoder and fine-tuning it in this experiment. It is well known in the domain generalization literature that there is an inherent trade-off between improving accuracy on a specific domain and achieving effective domain alignment for generalization across domains (Nguyen et al., 2022; Yu et al., 2024). To address this limitation, few-shot adaptation and training-free methods could be explored to reduce the dependence on large datasets (Zhang et al., 2022; Zanella & Ben Ayed, 2024; Bendou et al., 2025). In future work, we will explore these approaches, as their investigation falls beyond the scope of the current study.

4.4 Ablation Studies

We conduct several ablation studies on (i) the impact of incorporating a template set to introduce task-specific information; (ii) the impact of the image encoder, the image-text encoder, and the LLM; and (iii) the influence of different LLMs on per-

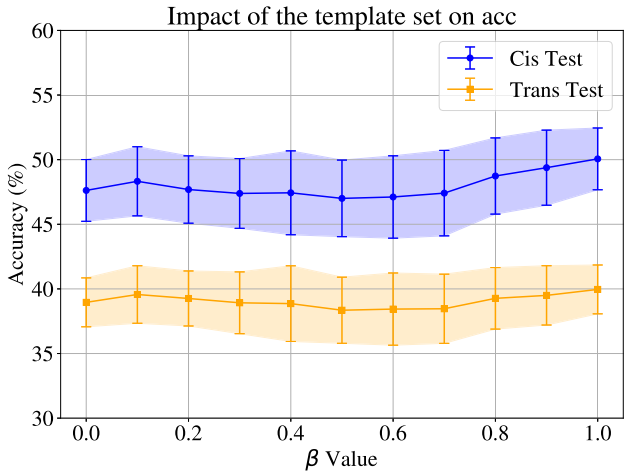

Fig. 7 Evaluation of the template set contribution to the WildIng in out-of-domain performance

formance. This analysis is conducted for the out-of-domain evaluation.

4.4.1 Evaluating the Incorporation of a Template Set

We evaluate the impact of using a predefined template set to describe the dataset categories, following the original configuration used in CLIP (Radford et al., 2021). This approach aims to incorporate task-specific information captured by the template set and serves as a potential alternative to LLM-generated descriptions in scenarios where their use is impractical or computationally expensive. The template set adds context to the descriptions by specifying that the images were captured by camera traps. For example, a template might describe a category as: "A photo captured by a camera trap of a { }". For a detailed list of the template set, refer to Appendix Appendix C.

To integrate template-based descriptions with LLM descriptions, we introduce a hyperparameter β, which controls the contribution of each source of knowledge to the final text embedding for each class. For both the template set and LLM-generated descriptions, we compute the centroid as defined in equation (1). The final text embedding is obtained as a weighted combination of these two centroids $\mathbf{t}_c = (1-\beta)\mathbf{m}_1^{(c)} + \beta\mathbf{m}_2^{(c)}$, where $\mathbf{m}_1^{(c)}$ represents the centroid of the template-based descriptions, and $\mathbf{m}_2^{(c)}$ represents the centroid of the LLM-based descriptions. The parameter $\beta \in [0, 1]$ determines the relative influence of each source.

Figure 7 presents the impact of the template set on model performance by varying the hyperparameter β. Figure 7 shows that the best results are achieved when $\beta = 1$, which corresponds to excluding the template set from the text embedding calculation.

Table 4 Ablation studies for performance variations for different design choices of WildIng. All models are trained on Snapshot Serengeti and evaluated on Terra Incognita (out-of-domain evaluation). The best combination is highlighted in **bold**, and the second-best is underlined. Results are reported in accuracy (%)

Img Enc	Img-txt Enc	LLM	Cis-Test	Trans-Test
✓	✗	✗	46.67	37.18
✓	✗	✓	40.32	39.11
✗	✓	✗	$28.14_{\pm 2.53}$	$21.72_{\pm 1.79}$
✗	✓	✓	$33.57_{\pm 2.81}$	$26.66_{\pm 2.96}$
✓	✓	✗	$47.62_{\pm 2.39}$	$38.96_{\pm 1.89}$
✓	✓	✓	$\mathbf{50.06}_{\pm 2.39}$	$\mathbf{39.96}_{\pm 1.89}$

Table 5 Performance comparison with different LLMs. The best combination is highlighted in **bold**, and the second-best is underlined. Results are reported in accuracy (%)

LLM	Cis-Test Acc(%)	Trans-Test
LLAMA	$28.82_{\pm 0.84}$	$21.31_{\pm 0.60}$
Qwen	$29.70_{\pm 0.44}$	$22.88_{\pm 0.55}$
Phi	$29.47_{\pm 0.62}$	$19.96_{\pm 0.66}$
ChatGPT	$\mathbf{50.06}_{\pm 2.39}$	$\mathbf{39.96}_{\pm 1.89}$

4.4.2 Image Encoder, Image-text Encoder, and LLM

We evaluate the performance of WildIng by removing the image encoder, the image-text encoder, and the textual descriptions generated by the LLM. Table 4 presents the geographical out-of-domain evaluation results for Cis-Test and Trans-Test under different design choices in the model. Our findings show that the lowest performance occurs when both the image encoder and LLM-generated descriptions are removed (third row in Table 4). This setting is equivalent to setting $\alpha = 0$ in (7) and training the model using the camera trap template set introduced in Section 4.4.1. Under this configuration, the model achieves accuracy scores of 28.14% in the Cis-Test and 21.72% in the Trans-Test. Similarly, removing only the image encoder also results in poor performance (fourth row in Table 4). These results highlight the crucial role of the image encoder in WildIng, indicating that the image-text encoder alone cannot fully replace it.

When the image encoder is included, the model shows a significant improvement, achieving accuracies of 46.67% in the Cis-Test and 37.18% in the Trans-Test (first row in Table 4). Similar to the configuration in the first row of Table 4, adding LLM-generated descriptions improves performance in the Trans-Test, increasing accuracy to 39.11%. However, in the Cis-Test, the accuracy decreases to 40.32% (second row in Table 4). These results suggest that while textual descriptions can be beneficial, their effectiveness depends on the test dataset and the type of text information used.

This suggests that using only the image encoder is insufficient to capture geographically invariant features. Standard deviations are not reported in the first two rows in Table 4, as we exclusively used the pre-trained model provided in Long-CLIP (Zhang et al., 2024a).

When both image and image-text encoders are included, the model provides more robust results across both test sets (fifth and sixth row in Table 4). In particular, incorporating the image encoder, image-text encoder, and LLM-generated descriptions leads to the highest accuracy, with 50.06% in the Cis-Test and 39.96% in the Trans-Test. This highlights the importance of integrating image and image-text embeddings more effectively to capture relationships between images and their categorical descriptions, helping to construct invariant representations against geographical domain shifts.

4.4.3 Evaluating different LLMs

Table 5 shows a comparison of WildIng when trained with descriptions generated by different LLMs, including LLAMA (Touvron et al., 2023), Qwen (Yang et al., 2024), Phi (Abdin et al., 2024), and ChatGPT. The key difference among these models is the quality of the generated descriptions for each category.

We observe that WildIng achieves an accuracy of 28.82% in the Cis-Test and 21.31% in the Trans-Test when trained with descriptions from LLAMA. Similarly, poor results are obtained with Qwen and Phi. In contrast, when using ChatGPT-generated descriptions, the model reaches 50.06% in the Cis-Test and 39.96% in the Trans-Test. This suggests that the descriptions generated by ChatGPT are significantly better than those produced by the other models in this specific task. These results highlight the importance of generating high-quality descriptions that provide the model with relevant information about each category. More informative descriptions enhance the model's ability to distinguish between different categories, which in turn results in better performance across both test scenarios.

4.5 Sensitivity to the Hyperparameter α

Figure 8 illustrates how variations in the parameter α between 0 and 1 in (7) affect the performance of WildIng for Cis-Test and Trans-Test in Terra Incognita for out-of-domain evaluation. We observe that the information from both matrices, $\mathbf{Q}$ and $\mathbf{W}$, is complementary. Figure 8 shows that the optimal value of α is 0.5, indicating that giving nearly equal importance to both matrices results in the highest accuracy. When α deviates from this optimal value, the accuracy declines, suggesting that overemphasizing either matrix leads to a loss of useful information for classification. This trend is consistent across both evaluation sets, demonstrating the robustness of the model when incorporating information from both matrices.

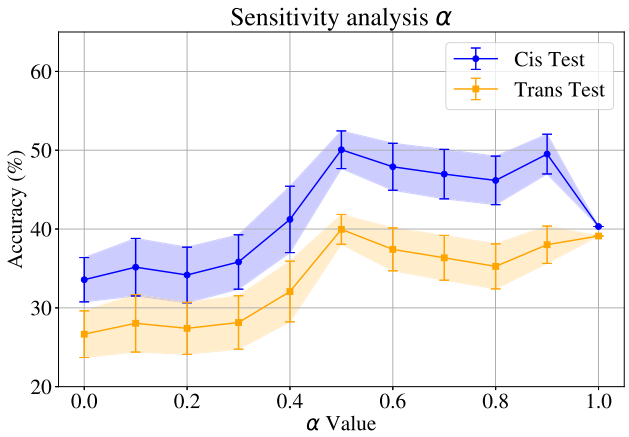

Fig. 8 Sensibility analysis of the hyperparameter α of WildIng in the Terra Incognita dataset for out-of-domain evaluation

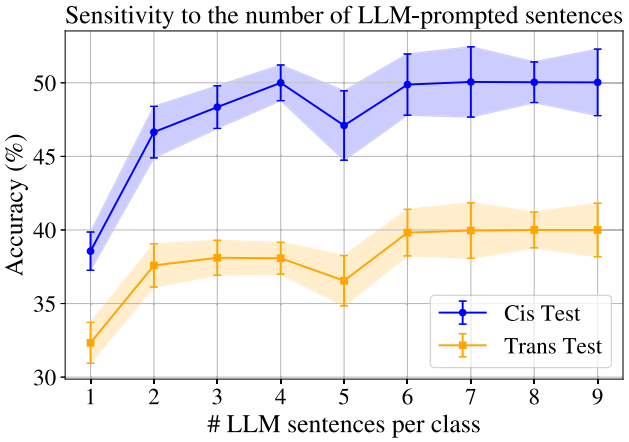

Fig. 9 Evaluation of the sensitivity of WildIng to the number of LLM-prompted sentences per class

4.6 Sensitivity to the number of LLM-prompted sentences

The Figure 9 present the performance of WildIng when we vary M_c, the number of short description per class c in $\mathcal{C}^D$. We observe a consistent improvement in accuracy with more descriptions, especially on the Cis-Test split. These results suggest that WildIng benefits from a larger number of short descriptions that are diverse and clearly describe each class.

4.7 Limitations

Although FMs have shown promising results in recognizing animal species in camera trap images, there is still a difference between their performance on out-of-domain (Table 1) and in-domain (Table 3) geographical evaluation.

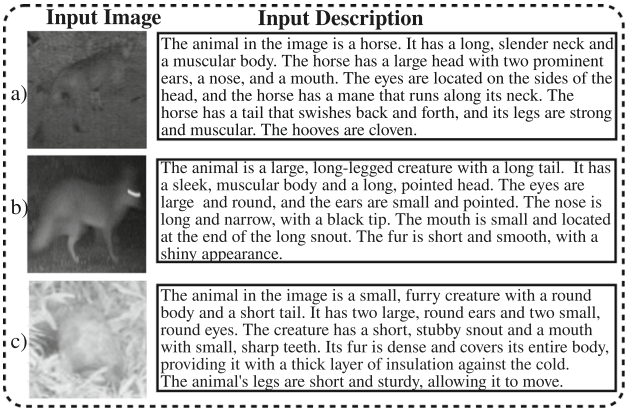

Fig. 10 Failure cases of the WildIng model in camera trap image classification. **Case a:** The VLM generates an incorrect description containing details that do not match the input image. **Case b:** A blurry image leads to a vague and uninformative description. **Case c:** When the input image is highly unclear, the VLM produces a random and unrelated description

Figure 10 illustrates the typical failure cases of WildIng. These misclassifications arise mostly when descriptions are inaccurate or ambiguous. To illustrate the model's sensitivity to input descriptions, Figure 10 presents three examples:

- In **case a**, the VLM generates hallucinated details that are not present in the image. For example, it identifies the animal as a horse, even though the input image does not match this description. This results in a completely incorrect prediction.
- In **case b**, the input image is blurry and difficult to interpret. Consequently, the VLM produces a vague description with minimal useful information. Due to the lack of context, the model fails to make an accurate prediction.
- In **case c**, the image is so unclear that the VLM generates a description unrelated to the input. This random description further misleads the model, leading to an incorrect prediction.

There are diverse Uncertainty Estimation (UE) strategies for handling hallucinated or low-quality captions (Li et al., 2025). Neighborhood consistency can be used to identify likely unreliable model responses (Khan & Fu, 2024). The method proposed in (Khan & Fu, 2024) introduces n variations of an input prompt to the model (e.g., a VLM) and expects the evaluated model to produce consistent outputs for all n cases, highlighting whether the model's deterministic embeddings can effectively handle semantic variability. Another solution, more on the hidden embeddings of the model, is proposed in (Mushtaq et al., 2025), which combines hidden state representations from the model with token-level uncertainty to obtain a more comprehensive measure of reliability. In the future, these approaches could be further explored, as their inclusion may help mitigate issues related to inaccurate or ambiguous descriptions.

5 Conclusions

In this research, we introduce WildIng, a new model that presents a geographical invariant representation to mitigate performance loss caused by geographical domain shifts in camera trap image recognition. WildIng addresses geographical domain shifts by leveraging robust features extracted from the image and image-text encoder while also incorporating enhanced category descriptions generated by an LLM. Our extensive experiments demonstrate that WildIng outperforms state-of-the-art models in camera trap image classification, particularly when there are geographical domain shifts between the training and testing datasets, all while preserving its open-vocabulary capabilities.

Appendix A Prompts

In this section, we provide the prompts for the LLM and LLaVA used in WildIng.

Appendix A.1 Prompt LLM

The prompt used to generate the LLM description of the animal species follows a structured format based on the methodology described in (Pratt et al., 2023) and is presented below.

You are an AI assistant specialized in biology and providing accurate and detailed descriptions of animal species. We are creating detailed and specific prompts to describe various species. The goal is to generate multiple sentences that capture different aspects of each species' appearance and behavior. Please follow the structure and style shown in the examples below. Each species should have a set of descriptions that highlight key characteristics.

Example Structure:

Badger:

- a badger is a mammal with a stout body and short sturdy legs.
- a badger's fur is coarse and typically grayish-black.
- badgers often feature a white stripe running from the nose to the back of the head dividing into two stripes along the sides of the body to the base of the tail.
- badgers have broad flat heads with small eyes and ears.
- badger noses are elongated and tapered ending in a black muzzle.
- badgers possess strong well-developed claws adapted for digging burrows.
- overall badgers have a rugged and muscular appearance suited for their burrowing lifestyle.

Appendix A.2 Prompt LLaVA

The prompt used in LLaVA follows the approach described in (Fabian et al., 2023) and is structured as follows:

SYSTEM: You are an AI assistant specialized in biology and providing accurate and detailed descriptions of animal species.\n ≪ image ≫ \n
USER: You are given the description of an animal species. Provide a very detailed description of the appearance of the species and describe each body part of the animal in detail. Only include details that can be directly visible in a photograph of the animal. Only include information related to the appearance of the animal and nothing else. Make sure to only include information that is present in the species description and is certainly true for the given species. Do not include any information related to the sound or smell of the animal. Do not include any numerical information related to measurements in the text in units: m cm in inches ft feet km/h kg lb lbs. Remove any special characters such as unicode tags from the text. Return the answer as a single paragraph.

Appendix B Hyperparameter Search Space

To select the hyperparameters, we performed a Monte Carlo partitioning of the dataset. We generated three different partitions, each created using a different random seed. For each partition, a subset of the development set (training + validation data) was randomly assigned to the training and validation sets.

We then conducted a random search over a predefined hyperparameter space, testing different hyperparameter combinations. For each combination, we trained the model on all three partitions separately and computed the accuracy for each setting. To determine the final performance of a configuration, we calculated the mean accuracy and standard deviation across the three partitions. This process was repeated 30 times, and the best hyperparameter combination was selected based on the highest mean accuracy. The search space included the following hyperparameters:

- **Batch size**: $b \in \{128, 256\}$
- **Hidden dimension**: $h \in \{253 + 60k \mid k \in \mathbb{Z}, 0 \leq k \leq 11\}$
- **Learning rate**: $\eta \in \{0.01, 0.02, \ldots, 0.09\}$
- **Momentum**: $m \in \{0.80, 0.82, \ldots, 0.98\}$
- **Number of epochs**: $e \sim \mathcal{U}(25, 100)$ (randomly sampled between 25 and 100)
- **Temperature** (τ): $\tau \in \{0.1, 0.01, 0.001\}$
- α: $\alpha \in \{0.4, 0.5, 0.6\}$

To evaluate the robustness of the selected hyperparameter combination, we further examined its consistency by computing the standard deviation in test accuracy across 100 different random seeds for the out-of-domain experiments and 3 different random seeds for in-domain experiments, due to the fact that in this experiment, we unfreeze the image encoder and fine-tune it.

Appendix C Templates

In this section, we provide examples of templates specifically designed for the camera trap image recognition task. These templates are adapted from the ImageNet templates used in CLIP (Radford et al., 2021) and are presented below:

- a photo captured by a camera trap of a {}.
- a camera trap photo of the {} captured in poor conditions.
- a cropped camera trap image of the {}.
- a camera trap image featuring a bright view of the {}.
- a camera trap image of the {} captured in clean conditions.
- a camera trap image of the {} captured in dirty conditions.
- a camera trap image with low light conditions featuring the {}.
- a black and white camera trap image of the {}.
- a cropped camera trap image of a {}.
- a blurry camera trap image of the {}.

- a camera trap image of the {}.
- a camera trap image of a single {}.
- a camera trap image of a {}.
- a camera trap image of a large {}.
- a blurry camera trap image of a {}.
- a pixelated camera trap image of a {}.
- a camera trap image of the weird {}.
- a camera trap image of the large {}.
- a dark camera trap image of a {}.
- a camera trap image of a small {}.

For each template, we replace "{ }" by the specific category in $\mathcal{C}^D$.

Acknowledgements This work was supported by Universidad de Antioquia - CODI (project 2024-73410), by the ANR (French National Research Agency) under the JCJC project DeSNAP (ANR-24-CE23-1895-01), and by the Academic Grant from NVIDIA AI.

Funding Open Access funding provided by Colombia Consortium

Data Availability Snapshot Serengeti data and Terra Incognita are publicly available on LILA BC at [https://lila.science/datasets/snapshot-serengeti] and on Caltech Camera Traps at [https://beerys.github.io/CaltechCameraTraps/]. For reproducibility, we use the preprocessed image data of Snapshot Serengeti as provided by WildCLIP [https://doi.org/10.1007/s11263-024-02026-6].

References

Abdin, M., Aneja, J., Awadalla, H., Awadallah, A., Awan, A.A., Bach, N., Bahree, A., Bakhtiari, A., Bao, J., & Behl, H., et al. (2024). Phi-3 technical report: A highly capable language model locally on your phone. arXiv preprint arXiv:2404.14219.

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). GPT-4 technical report. arXiv preprint arXiv:2303.08774.

Archer, S. R., & Predick, K. I. (2008). Climate change and ecosystems of the southwestern united states. *Rangelands, 30*(3), 23–28.

Awais, M., Naseer, M., Khan, S., Anwer, R. M., Cholakkal, H., Shah, M., Yang, M.-H., & Khan, F. S. (2025). Foundation models defining a new era in vision: a survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 47*(4), 2245–2264.

Beery, S., Morris, D., & Yang, S. (2019). Efficient pipeline for camera trap image review. arXiv preprint arXiv:1907.06772.

Beery, S., Van Horn, G., & Perona, P. (2018). Recognition in Terra Incognita. In IEEE/CVF European Conference on Computer Vision, 456–473.

Bendou, Y., Ouasfi, A., Gripon, V., & Boukhayma, A. (2025). Proker: A kernel perspective on few-shot adaptation of large vision-language models. In Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition Conference, 25092–25102.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Conference of the North American Chapter of the Association for Computational Linguistics, 4171–4186.

Duan, J., Chen, L., Tran, S., Yang, J., Xu, Y., Zeng, B., & Chilimbi, T. (2022). Multi-modal alignment using representation codebook. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 15651–15660.

Fabian, Z., Miao, Z., Li, C., Zhang, Y., Liu, Z., Hernandez, A., Arbelaez, P., Link, A., Montes-Rojas, A., & Escucha, R., et al. (2023). Knowledge augmented instruction tuning for zero-shot animal species recognition. In NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following.

Fang, Z., Lu, J., & Zhang, G. (2025). Out-of-distribution detection with non-semantic exploration. *Information Sciences, 705*, Article 121989.

Gabeff, V., Rußwurm, M., Tuia, D., & Mathis, A. (2024). WildCLIP: Scene and animal attribute retrieval from camera trap data with domain-adapted vision-language models. *International Journal of Computer Vision, 132*(9), 3770–3786.

Gadot, T., Istrate, S., Kim, H., Morris, D., Beery, S., Birch, T., & Ahumada, J. (2024). To crop or not to crop: Comparing whole-image and cropped classification on a large dataset of camera trap images. *IET Computer Vision, 18*(8), 1193–1208.

Gao, P., Geng, S., Zhang, R., Ma, T., Fang, R., Zhang, Y., Li, H., & Qiao, Y. (2024). Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision, 132*, 581–595.

Garau-Luis, J. J., Bordes, P., Gonzalez, L., Roller, M., de Almeida, B., Blum, C., Hexemer, L., Laurent, S., Lang, M., Pierrot, T., et al. (2025). Multi-modal transfer learning between biological foundation models. *In Advances in Neural Information Processing Systems, 37*, 78431–78450.

Giraldo, J. H., Salazar, A., Gomez-Villa, A., & Diaz-Pulido, A. (2019). Camera-trap images segmentation using multi-layer robust principal component analysis. *The Visual Computer, 35*, 335–347.

Gomez-Villa, A., Salazar, A., & Vargas, F. (2017). Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks. *Ecological informatics, 41*, 24–32.

Hernandez, A., Miao, Z., Vargas, L., Dodhia, R., & Lavista, J. (2024). Pytorch-Wildlife: A collaborative deep learning framework for conservation. arXiv preprint arXiv:2405.12930.

Hogeweg, L. E., Gangireddy, R., Brunink, D., Kalkman, V.J., Cornelissen, L., & Kamminga, J.W. (2024). Cood: Combined out-of-distribution detection using multiple measures for anomaly & novel class detection in large-scale hierarchical classification. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3971–3980.

Huang, S., Dong, L., Wang, W., Hao, Y., Singhal, S., Ma, S., Lv, T., Cui, L., Mohammed, O. K., Patra, B., et al. (2024). Language is not all you need: Aligning perception with language models. *In Advances in Neural Information Processing Systems, 36*, 72096–72109.

Jiang, A.Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D.S., Casas, D. d.l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al. (2023). Mistral 7b. arXiv preprint arXiv:2310.06825.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A.,

et al. (2021). Highly accurate protein structure prediction with alphafold. *Nature, 596*, 583–589.

Jung, S.J., Kim, H., & Jang, K.S. (2024). Llm based biological named entity recognition from scientific literature. In IEEE International Conference on Big Data and Smart Computing (BigComp), 433–435.

Kempf, E., Schrodi, S., Argus, M., & Brox, T. (2025). When and how does clip enable domain and compositional generalization? arXiv preprint arXiv:2502.09507.

Khan, Z., & Fu, Y. (2024). Consistency and uncertainty: Identifying unreliable responses from black-box vision-language models for selective visual question answering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10854–10863.

Lam, H. Y. I., Ong, X. E., & Mutwil, M. (2024). Large language models in plant biology. *Trends in Plant Science, 29*, 1145–1155.

Li, J., Li, D., Savarese, S., & Hoi, S. (2023). BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In International Conference on Machine Learning, 19730–19742.

Li, J., Li, Y., Fu, Y., Liu, J., Liu, Y., Yang, M., & King, I. (2025a). Clip-powered domain generalization and domain adaptation: A comprehensive survey. arXiv preprint arXiv:2504.14280.

Li, S., Xu, X., Meng, W., Song, J., Peng, C., & Shen, H. T. (2025). Mitigating hallucinations in large vision-language models via reasoning uncertainty-guided refinement. *IEEE Transactions on Multimedia, 27*, 7380–7391.

Li, X., Tian, H., Piao, Z., Wang, G., Xiao, Z., Sun, Y., Gao, E., & Holyoak, M. (2022). cameratrapr: An r package for estimating animal density using camera trapping data. *Ecological Informatics, 69*, Article 101597.

Liang, W., Mao, Y., Kwon, Y., Yang, X., & Zou, J. (2023). Accuracy on the curve: On the nonlinear correlation of ml performance between data subpopulations. In IEEE/CVF International Conference on Machine Learning, 20706–20724.

Liu, H., Li, C., Wu, Q., & Lee, Y. J. (2024). Visual instruction tuning. *In Advances in Neural Information Processing Systems, 36*, 34892–34916.

Luo, G., Zhou, Y., Sun, X., Wu, Y., Gao, Y., & Ji, R. (2024). Towards language-guided visual recognition via dynamic convolutions. *International Journal of Computer Vision, 132*, 1–19.

Mushtaq, E., Fabian, Z., Bakman, Y.F., Ramakrishna, A., Soltanolkotabi, M., & Avestimehr, S. (2025). Harmony: Hidden activation representations and model output-aware uncertainty estimation for vision-language models. In Proceedings of the Computer Vision and Pattern Recognition Conference, 1663–1668.

Nguyen, T., Lyu, B., Ishwar, P., Scheutz, M., & Aeron, S. (2022). Trade-off between reconstruction loss and feature alignment for domain generalization. In 2022 21st IEEE International Conference on Machine Learning and Applications, 794–801.

Norman, D. L., Bischoff, P. H., Wearn, O. R., Ewers, R. M., Rowcliffe, J. M., Evans, B., Sethi, S., Chapman, P. M., & Freeman, R. (2023). Can CNN-based species classification generalise across variation in habitat within a camera trap survey? *Methods in Ecology and Evolution, 14*(1), 242–251.

Pantazis, O., Brostow, G., Jones, K., & Mac Aodha, O. (2022). SVL-Adapter: Self-Supervised Adapter for Vision-Language Pretrained Models. In British Machine Vision Conference.

Pollock, L. J., Kitzes, J., Beery, S., Gaynor, K. M., Jarzyna, M. A., Mac Aodha, O., Meyer, B., Rolnick, D., Taylor, G. W., Tuia, D., et al. (2025). Harnessing artificial intelligence to fill global shortfalls in biodiversity knowledge. *Nature Reviews Biodiversity, 1*, 166–182.

Pratt, S., Covert, I., Liu, R., & Farhadi, A. (2023). What does a platypus look like? generating customized prompts for zero-shot image classification. In IEEE/CVF International Conference on Computer Vision, 15691–15701.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. *In International Conference on Machine Learning, 139*, 8748–8763.

Reynolds, S. A., Beery, S., Burgess, N., Burgman, M., Butchart, S. H., Cooke, S. J., Coomes, D., Danielsen, F., Di Minin, E., Durán, A. P., et al. (2024). The potential for ai to revolutionize conservation: a horizon scan. *Trends in Ecology & Evolution, 40*, 191–207.

Riz, L., Saltori, C., Wang, Y., Ricci, E., & Poiesi, F. (2024). Novel class discovery meets foundation models for 3d semantic segmentation. *International Journal of Computer Vision, 133*, 527–548.

Santamaria, J.D., Isaza, C., & Giraldo, J.H. (2025). CATALOG: A camera trap language-guided contrastive learning model. In IEEE/CVF Winter Conference on Applications of Computer Vision, 1197–1206,.

Santamaria P, J.D., Giraldo, J.H., Diaz-Pulido, A., & Isaza, C. (2024). Audio vs. visual approach to monitor the critically endangered species atlapetes blancae: Developing deep learning models with limited data. In IARIA Annual Congress on Frontiers in Science, Technology, Services, and Applications, 72–80.

Schneider, S., Greenberg, S., Taylor, G. W., & Kremer, S. C. (2020). Three critical factors affecting automated image species recognition performance for camera traps. *Ecology and Evolution, 10*(7), 3503–3517.

Simões, F., Bouveyron, C., & Precioso, F. (2023). DeepWILD: Wildlife identification, localisation and estimation on camera trap videos using deep learning. *Ecological Informatics, 75*, Article 102095.

Stevens, S., Wu, J., Thompson, M.J., Campolongo, E.G., Song, C.H., Carlyn, D.E., Dong, L., Dahdul, W.M., Stewart, C., Berger-Wolf, T., et al. (2024). BioCLIP: A vision foundation model for the tree of life. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 19412–19424.

Swanson, A., Kosmala, M., Lintott, C., Simpson, R., Smith, A., & Packer, C. (2015). Data from: Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna.

Tan, M. & Le, Q. (2021). Efficientnetv2: Smaller models and faster training. In IEEE/CVF International Conference on Machine Learning, 10096–10106.

Tang, L., Jiang, P.-T., Xiao, H., & Li, B. (2025). Towards training-free open-world segmentation via image prompt foundation models. *International Journal of Computer Vision, 133*, 1–15.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., & Rodriguez, A. (2023). LLaMA: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.

Tuia, D., Kellenberger, B., Beery, S., Costelloe, B. R., Zuffi, S., Risse, B., Mathis, A., Mathis, M. W., van Langevelde, F., Burghardt, T., et al. (2022). Perspectives in machine learning for wildlife conservation. *Nature Communications, 13*(1), 792.

Wald, Y., Feder, A., Greenfeld, D., & Shalit, U. (2021). On calibration and out-of-domain generalization. *In Advances in Neural Information Processing Systems, 34*, 2215–2227.

Wang, Q., Lin, Y., Chen, Y., Schmidt, L., Han, B., & Zhang, T. (2024). A sober look at the robustness of clips to spurious features. *In Advances in Neural Information Processing Systems, 37*, 122484–122523.

Wang, Y., & Kang, G. (2025). Attention head purification: A new perspective to harness clip for domain generalization. *Image and Vision Computing, 157*, Article 105511.

Wu, W., Sun, Z., Song, Y., Wang, J., & Ouyang, W. (2024). Transferring vision-language models for visual recognition: A classifier perspective. *International Journal of Computer Vision, 132*, 392–409.

Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., et al. (2024). Qwen2. 5 technical report. arXiv preprint arXiv:2412.15115.

Yang, L., Zhang, R.-Y., Chen, Q., & Xie, X. (2025). Learning with enriched inductive biases for vision-language models. *International Journal of Computer Vision, 133*, 3746–3761.

Yang, Z., Tian, Y., Wang, L., & Zhang, J. (2025). Enhancing generalization in camera trap image recognition: Fine-tuning visual language models. *Neurocomputing, 634*, Article 129826.

Yu, H., Zhang, X., Xu, R., Liu, J., He, Y., & Cui, P. (2024). Rethinking the evaluation protocol of domain generalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 21897–21908.

Yu, R., Liu, S., Yang, X., & Wang, X. (2023). Distribution shift inversion for out-of-distribution prediction. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, 3592–3602.

Zanella, M., & Ben Ayed, I. (2024). Low-rank few-shot adaptation of vision-language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1593–1603.

Zang, Y., Li, W., Han, J., Zhou, K., & Loy, C. C. (2024). Contextual object detection with multimodal large language models. *International Journal of Computer Vision, 133*, 825–843.

Zhang, B., Zhang, P., Dong, X., Zang, Y., & Wang, J. (2024a). Long-CLIP: Unlocking the long-text capability of CLIP. In IEEE/CVF European Conference on Computer Vision, 310–325.

Zhang, J., Huang, J., Jin, S., & Lu, S. (2024). Vision-Language Models for Vision Tasks: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 46*(8), 5625–5644.

Zhang, R., Zhang, W., Fang, R., Gao, P., Li, K., Dai, J., Qiao, Y., & Li, H. (2022). Tip-adapter: Training-free adaption of clip for few-shot classification. In Proceedings of the IEEE/CVF European conference on computer vision, 493–510.

Zhu, L., Yin, W., Yang, Y., Wu, F., Zeng, Z., Gu, Q., Wang, X., Zhou, C., & Ye, N. (2024). Vision-language alignment learning under affinity and divergence principles for few-shot out-of-distribution generalization. *International Journal of Computer Vision, 132*, 3375–3407.