CATALOG: A Camera Trap Language-guided Contrastive Learning Model

Julian Santamaria, Claudia Isaza, Jhony Giraldo

CATALOG: A Camera Trap Language-guided Contrastive Learning Model

Julian D. Santamaria[1], Claudia Isaza[1], Jhony H. Giraldo[2]
[1] SISTEMIC, Faculty of Engineering, Universidad de Antioquia-UdeA, Medellín, Colombia.
[2] LTCI, Télécom Paris, Institut Polytechnique de Paris, Palaiseau, France.
{julian.santamaria, victoria.isaza}@udea.edu.co, jhony.giraldo@telecom-paris.fr

Abstract

*Foundation Models (FMs) have been successful in various computer vision tasks like image classification, object detection and image segmentation. However, these tasks remain challenging when these models are tested on datasets with different distributions from the training dataset, a problem known as domain shift. This is especially problematic for recognizing animal species in camera-trap images where we have variability in factors like lighting, camouflage and occlusions. In this paper, we propose the **Ca**mera **Tra**p **L**anguage-guided **Co**ntrastive Learnin**g** (CATALOG) model to address these issues. Our approach combines multiple FMs to extract visual and textual features from camera-trap data and uses a contrastive loss function to train the model. We evaluate CATALOG on two benchmark datasets and show that it outperforms previous state-of-the-art methods in camera-trap image recognition, especially when the training and testing data have different animal species or come from different geographical areas. Our approach demonstrates the potential of using FMs in combination with multi-modal fusion and contrastive learning for addressing domain shifts in camera-trap image recognition. The code of CATALOG is publicly available at https://github.com/Julian075/CATALOG.*

1. Introduction

In recent years, the field of deep learning has seen remarkable progress, driven in part by the emergence of a new class of models known as Foundation Models (FMs) [5, 6, 24, 25, 34]. These models are characterized by their large size, depth, and the vast amounts of data on which they have been trained, sometimes in the order of billions of data samples. In computer vision, FMs have demonstrated exceptional performance in a wide range of tasks, including zero-shot image classification, object detection, and image segmentation [10, 24, 33, 35]. By leveraging the knowledge acquired during pre-training, FMs have enabled significant advances in the state-of-the-art of the classical computer vision tasks and opened up new possibilities for real-world applications. One key advantage of some FMs, particularly those that incorporate text as an input modality, is their ability to handle open vocabulary tasks [8, 32]. These models are not restricted to a fixed set of categories but can recognize a wide range of objects based on descriptive text inputs [19, 23, 38]. This capability is particularly important in diverse and dynamic environments, such as wildlife monitoring, where the range of observable species can be extensive and unpredictable.

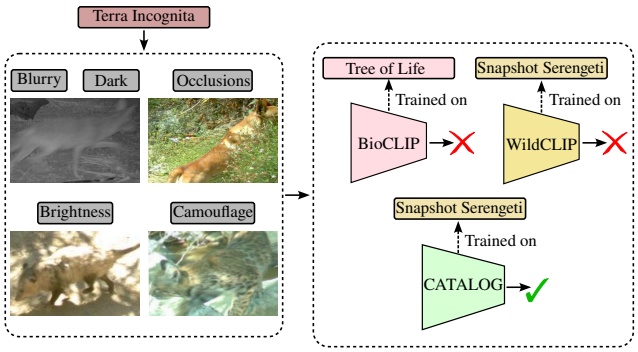


Figure 1. Comparison of CATALOG, BioCLIP [28], and WildCLIP [11] under challenging camera-trap conditions. CATALOG demonstrates superior performance.

Despite the success of FMs, adapting them to specific domains that differ significantly from the original training data remains a challenging task [4, 22, 37]. This difficulty is exacerbated when there is limited data, as it is often the case in specialized applications such as camera-trap image classification [3, 9, 13, 26]. Camera traps are remote devices that are triggered by motion or heat to capture images or videos of wildlife in their natural habitat [12]. While the use of camera traps has become increasingly popular in wildlife research and conservation efforts, collecting very large-scale datasets for this domain is still a significant challenge [3, 29, 31].

Due to these challenges, previous studies often adapt existing pre-trained FMs to the specific datasets instead of cre-

ating new models from scratch [9, 11, 28]. For example, WildCLIP is an adaptation of the Contrastive Language-Image Pre-Training (CLIP) model [24], which uses Vision Language Models (VLMs) tailored to camera-trap data [11]. Similarly, BioCLIP adapts CLIP for the tree-of-life dataset to improve biological image analysis [28]. However, in the case of camera-trap images, we observe that: i) FMs often perform poorly on images that differ significantly from the training dataset, and ii) FMs do not generalize well in images that exhibit substantial variability in factors such as lighting, camouflage, and occlusions [21, 26, 31]. These observations are exemplified in Fig. 1.

In this paper, we introduce the **Ca**mera **Tra**p **L**anguage-guided **Co**ntrastive Learnin**g** (CATALOG) model to recognize animal species in camera-trap images. Our approach combines multiple FMs, including a Large Language Model (LLM) [5], CLIP [24], LLaVA (Large Language-and-Vision Assistant) [18], and BERT (Bidirectional Encoder Representations from Transformers) [6], to learn domain-invariant features from text and image modalities. We introduce three key technical novelties: first, we combine text information from various sources using the centroid in the embedding space; second, we align the multi-modal features using a convex combination of the different sources; and third, we train our model using a contrastive loss to facilitate the learning of domain-invariant features for camera-trap images. We train our model on the Snapshot Serengeti dataset [29] and evaluate it on the Terra Incognita dataset [3]. The results demonstrate that CATALOG outperforms previous general-purpose and domain-specific FMs for camera-trap image recognition, especially when the domain of the training set differs from that of the testing set. Our main contributions can be summarized as follows:

- We introduce a novel CATALOG model that integrates several FMs for camera-trap image recognition.
- When tested on datasets that differ from its training data, CATALOG outperforms previous FMs in recognizing animal species from camera-trap images.
- We conduct a series of ablation studies to confirm the effectiveness of each component in our model.

2. Related Work

Foundation models. In recent years, models trained with vast amounts of data, capable of learning high-level representations and performing complex tasks, have significantly advanced the fields of machine learning and artificial intelligence [15, 16, 30]. These models, powered by huge datasets, have achieved outstanding performance across various domains [7]. Among these models, LLMs and VLMs are particularly remarkable for their exceptional ability to process images and text, while also generating coherent and relevant information [40]. Notable examples of these models include GPT-3 [5] and GPT-4 [1]. Due to their extensive pre-training, these LLMs have demonstrated the ability to generalize even in contexts for which they were not specifically trained [9, 20]. Similarly, CLIP [24] has shown impressive results in image classification by learning joint representations of images and text. Another notable example of FM is the LLaVA model [18], which integrates vision and language modalities to achieve state-of-the-art results in multi-modal tasks.

Foundation models for biology. In biology, FMs have been adapted to address domain-specific challenges. For example, BioCLIP [28] extends the principles of CLIP [24] to biological data, covering diverse categories such as plants, animals, and fungi. BioCLIP also integrates rich structured biological knowledge. This model leverages the TREEOFLIFE-10M dataset [28] and taxonomic names to achieve significant performance improvements in fine-grained classification tasks. This enables the classification and analysis of complex biological images and text data.

Foundation models for camera traps. FMs are increasingly being applied to camera-trap data for wildlife monitoring and conservation. One such model is WildCLIP [11], which uses the strengths of CLIP to accurately identify and classify different animal species in camera-trap images. Another approach is WildMatch [9], which introduces a zero-shot species classification framework. WildMatch adapts vision-language models to generate detailed visual descriptions of camera-trap images using expert terminology, which are then matched against an external knowledge base to identify species. These advances demonstrate the significant potential of FMs in improving wildlife monitoring and conservation efforts.

Previous methods for camera-trap image recognition have made progress in specific domains, but they often struggle when tested in diverse environmental contexts [27] and under challenging conditions, as illustrated in Fig. 1. Our proposed model, CATALOG, addresses this limitation by leveraging feature representation from FMs that are robust against domain shifts. This integration improves species recognition and contextual understanding, making the model less sensitive to variations in environmental conditions and new classes.

3. CATALOG

Problem definition. In this paper, we assume access to an annotated training dataset, 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 $\mathcal{C}^D$. For testing purposes, we have another dataset, $\mathcal{S}$, 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 sets of classes in both datasets

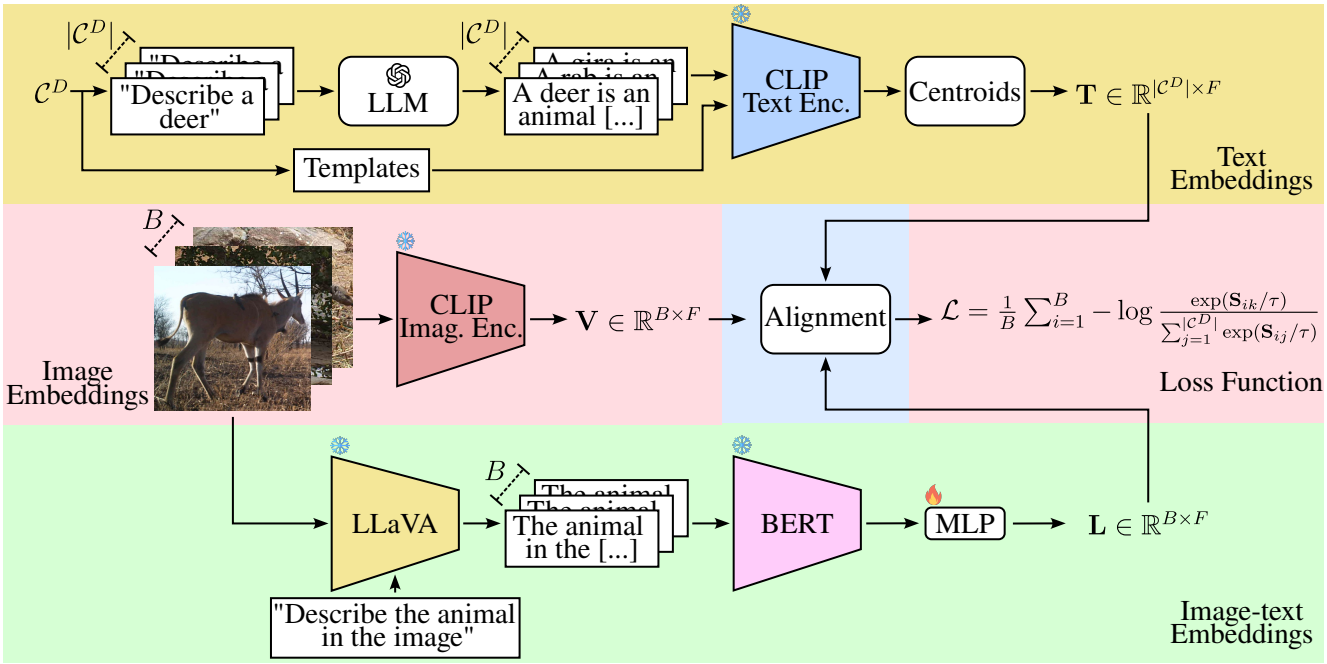


Figure 2. The pipeline of CATALOG. Our model is divided into five parts: i) text embeddings, ii) image embeddings, iii) image-text embeddings, iv) feature alignment, and v) loss function. The textual embeddings are computed using a set of pre-defined templates and the LLM descriptions. The image embeddings are computed using CLIP. The image-text embeddings are calculated using LLaVA and BERT. Finally, we align the multi-modal features using an alignment mechanism and train the model with a contrastive loss function.

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 images may not necessarily come from the same distribution, resulting in a domain shift. Our goal is to train a deep learning model using only the training dataset $\mathcal{D}$ and deploy it on the testing dataset $\mathcal{S}$.

3.1. Overview of the Approach

Fig. 2 presents the pipeline of our proposed framework, CATALOG. Our approach consists of three main components: i) text, ii) image, and iii) image-text embeddings. For the text component, we input our dictionary of classes, $\mathcal{C}^D$, and use an LLM with predefined templates to generate descriptions for each category. Then, we utilize CLIP's text encoder to obtain embeddings for each textual description. To ensure a unique embedding for each class in $\mathcal{C}^D$, we apply a technique to combine the embeddings (Sec. 3.2), resulting in a single embedding of dimension F. For the image component, we use CLIP's image encoder to extract embeddings from a mini-batch of B images (Sec. 3.3). Meanwhile, for the image-text component, we employ the VLM LLaVA coupled with BERT and a Multi-Layer Perceptron (MLP) to compute image-text embeddings from the mini-batch of images (Sec. 3.4). To ensure the embeddings from all three components are aligned, we use an alignment mechanism (Sec. 3.5). Finally, we utilize the output of the alignment mechanism to compute a contrastive loss, which is used to train our model (Sec. 3.6). We let all FMs frozen (❄) and only train (🔥) the MLP.

3.2. Text Embeddings

To generate textual descriptions for each category in our dataset $\mathcal{C}^D$, we utilize an LLM that can provide detailed information about the animals without requiring expert inputs. We also create multiple descriptions using predefined templates tailored to our specific task of camera-trap image recognition such as "a photo captured by a camera trap of a { }". These templates add context to the descriptions by specifying that the images were captured by camera traps. Examples of the precise prompts and templates used can be found in the supplementary material. We process two types of textual descriptions using CLIP's text encoder: one generated by the LLM and $M-1$ manually crafted templates. As a result, for each class in $\mathcal{C}^D$, we obtain M embeddings, each with a dimension of F. This allows us to represent each class using a set of textual embeddings that capture the semantic meaning of the category.

To obtain the final embedding for each class $c \in \mathcal{C}^D$, we compute the centroid of the M embeddings generated for that class. Specifically, let $\mathbf{P}^{(c)} \in \mathbb{R}^{M \times F}$ be the set of M embedding for the class $c \in \mathcal{C}^D$. The final embeddings $\mathbf{t}_c \in \mathbb{R}^F$ of c is calculated as the average of these M embeddings,

as follows:

$$\mathbf{t}_c = \frac{1}{M} \sum_{i=1}^{M} \mathbf{P}_{i:}^{(c)}, \tag{1}$$

where $\mathbf{P}_{i:}^{(c)}$ represents the ith row of $\mathbf{P}^{(c)}$. The output of the text embedding part of CATALOG is a matrix $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_{|\mathcal{C}^D|}]^\top \in \mathbb{R}^{|\mathcal{C}^D| \times F}$, which contains the final embeddings for all classes in $\mathcal{C}^D$.

3.3. Image Embedding

Pre-processing. We utilize the MegaDetector model [2] to process our camera trap datasets. The purpose of using this model is to extract crops from the camera-trap images that contain relevant information.

CLIP embeddings. We use CLIP's image encoder [24] to extract embeddings from the cropped images. We process the images in mini-batches of size B. For each image, we extract an embedding of dimension F using the CLIP encoder. The output of this stage is a matrix $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_B]^\top \in \mathbb{R}^{B \times F}$, where $\mathbf{v}_i$ corresponds to the visual embedding of the ith image in the mini-batch. This matrix is then used in the subsequent stages of our framework to align and contrast the text and image embeddings.

3.4. Image-text Embeddings

In the image-text branch of CATALOG, we use the mini-batch of cropped images as input. We employ the LLaVA model [18] to generate textual descriptions of the animals present in the cropped images, using a prompt similar to the one described in [9] (provided in the supplementary material). These textual descriptions are processed using the BERT model, which we selected because it provides 512 possible tokens [36]. We also utilize Long CLIP [39], but the results are not satisfactory (see Sec. 4.2). BERT generates text embeddings of dimension F'. Since F' is not equal to the dimension F of the CLIP embeddings, we feed these BERT embeddings into an MLP to match the dimensions.

The MLP serves to project each BERT embedding into a F-dimensional space to match the CLIP embedding dimension. However, it does not perform the alignment between the two different embeddings. The actual alignment between BERT and CLIP embeddings is achieved through the alignment mechanism and loss function, which we describe in detail in Sec. 3.5 and Sec. 3.6.

The output of the image-language branch of CATALOG is a matrix $\mathbf{L} = [\mathbf{l}_1, \mathbf{l}_2, \ldots, \mathbf{l}_B]^\top \in \mathbb{R}^{B \times F}$, where $\mathbf{l}_i$ is the transformed BERT embedding of the ith image in the mini-batch.

3.5. Alignment Mechanism

The alignment method takes the text ($\mathbf{T}$), image ($\mathbf{V}$), and image-text ($\mathbf{L}$) embeddings from each branch as input. The feature alignment process consists of two parts: i) similarity computation and ii) fusion mechanism. For the similarity computation, we calculate 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:

$$\mathbf{W}_{ij} = \frac{\langle \mathbf{v}_i, \mathbf{t}_j \rangle}{\|\mathbf{v}_i\| \|\mathbf{t}_j\|} \; \forall \, 1 \leq i \leq B, 1 \leq j \leq |\mathcal{C}^D|, \tag{2}$$

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. Similarly, we compute the cosine similarities between the text and image-text embeddings as follows:

$$\mathbf{Q}_{ij} = \frac{\langle \mathbf{l}_i, \mathbf{t}_j \rangle}{\|\mathbf{l}_i\| \|\mathbf{t}_j\|} \; \forall \, 1 \leq i \leq B, 1 \leq j \leq |\mathcal{C}^D|, \tag{3}$$

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

The fusion mechanism is implemented as a weighted average between the matrices $\mathbf{W}$ and $\mathbf{Q}$, where the weights are determined by the hyperparameter $\alpha \in [0, 1]$. Specifically, the output of the fusion method is a matrix $\mathbf{S} \in \mathbb{R}^{B \times |\mathcal{C}^D|}$, defined as follows:

$$\mathbf{S} = \alpha \mathbf{W} + (1 - \alpha) \mathbf{Q}. \tag{4}$$

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

3.6. 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:

$$\mathcal{L}(\mathbf{S}) = \frac{1}{B} \sum_{i=1}^{B} -\log \frac{\exp(\mathbf{S}_{ik}/\tau)}{\sum_{j=1}^{|\mathcal{C}^D|} \exp(\mathbf{S}_{ij}/\tau)}, \tag{5}$$

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. The intuition behind this loss function is to bring the image-text embeddings close to the text and image embeddings in the feature space when they correspond to the same species category in the dataset. Conversely, we aim to push apart the multi-modal embeddings from other species in the mini-batch. This encourages the model to learn a shared embedding space where the embeddings from different modalities are similar for the same species.

4. Experiments and Results

This section presents the datasets used in the current work, the evaluation protocol, the implementation details,

the results, and the discussion of CATALOG. We compare our algorithm against CLIP [24], BioCLIP [28], and WildCLIP [11]. We also perform a set of ablation studies to analyze each component of CATALOG, including the set of prompts and textual information we use to describe the different categories, different architectural choices like the VLM, CLIP's image encoder, and the loss function. Finally, we study the sensibility of CATALOG regarding the hyperparameter α in the alignment mechanism.

Datasets. We evaluate CATALOG using two public datasets in camera traps: Snapshot Serengeti [29] and Terra Incognita [3]. Some cropped images from these datasets are shown in Fig. 3.

- *Snapshot Serengeti* [29]. We use the version of the Serengeti dataset used in WildCLIP [11], which comprises 46 classes. This dataset version includes 380×380 pixel image crops generated by the MegaDetector model from the Snapshot Serengeti project, applying a confidence threshold above 0.7. Only camera trap images containing single animals were selected for this version. The dataset includes $340,972$ images divided into $230,971$ for training, $24,059$ for validation, and $85,942$ for testing.
- *Terra Incognita* [3]. This dataset comprises 16 classes and introduces two testing groups called Cis-locations and Trans-locations. This means that the images taken from these locations are similar (Cis-locations) or different (Trans-locations) to the training data. These partitions were originally provided to test the robustness of computer vision models trained and evaluated in the same Terra Incognita dataset (in-domain evaluation). We filter the images in the dataset with the MegaDetector model of the library PyTorch-Wildlife [14]. The dataset contains $45,912$ images divided into $12,313$ for training, $1,932$ for Cis-Validation, $1,501$ for Trans-Validation, $13,052$ for Cis-Test, and $17,114$ for Trans-Test.

Evaluation protocol. We conduct two experiments to evaluate the performance of our model compared to the current state-of-the-art models. In the first experiment, we use the Snapshot Serengeti dataset for training and validation, and the Terra Incognita dataset for testing (out-of-domain evaluation). In other words, $\mathcal{D}$ and $\mathcal{S}$ are the Snapshot Serengeti and Terra Incognita datasets, respectively. The Snapshot Serengeti dataset was collected in various protected areas in Africa, while Terra Incognita was collected in the American Southwest. Therefore, we have two main problems in this experimental setup: i) the difference in data distribution between the two datasets $\mathcal{D}$ and $\mathcal{S}$ (domain shift), and ii) the difference in the sets of classes ($\mathcal{C}^D \neq \mathcal{C}^S$). These two challenges are illustrated in Fig. 3. The difference in the sets of classes rules out any closed-set state-of-the-art method for comparison. We report the accuracy results in the Cis-Test and Trans-Test sets of Terra Incognita.

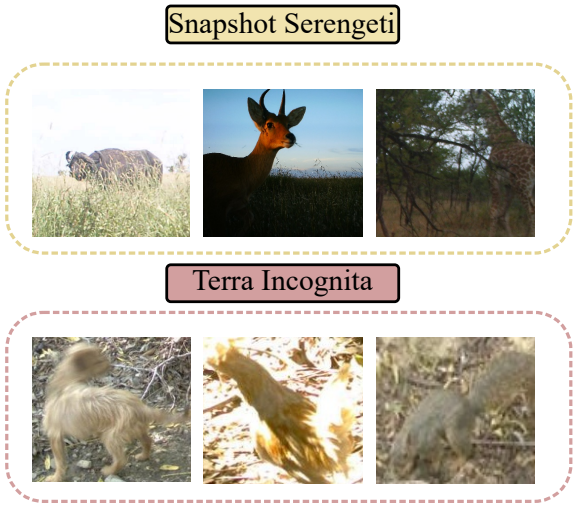


Figure 3. Cropped images from the Snapshot Serengeti and Terra Incognita datasets where we observe the domain shift and the difference in classes (different animal species).

For the second experiment, we modify our problem definition in Sec. 3 and use the same dataset to assess the model's performance without the complications introduced by the domain shift and new classes (in-domain evaluation). More precisely, we use the Snapshot Serengeti or Terra Incognita datasets for training, validation, and testing. This approach allows us to evaluate the model's accuracy and robustness within a consistent domain.

Implementation details. We use the 3.5 version of ChatGPT for the LLM, the ViT-B/16 version of CLIP, the 1.5-7B version of LLaVA, and the BERT-base-uncased version of BERT in our implementation of CATALOG. For training our model in the first experiment, we set $\alpha = 0.6$ and $\tau = 0.1$. The MLP architecture consists of a single hidden layer with dimension $1,045$, and the Gaussian Error Linear Unit (GELU) as the activation function. We train our model for 8 epochs with a dropout rate of 0.27. We use the Stochastic Gradient Descent (SGD) algorithm to train CATALOG with a learning rate of 0.08, momentum of 0.8, and batch size of 48. For the second experiment, we fine-tuned CATALOG by unfreezing the CLIP image encoder and adjusting key hyperparameters: batch size of 100, and training for 86 epochs using SGD with a momentum of 0.8. For the Snapshot Serengeti dataset, we use a 0.4 dropout rate, $1e{-3}$ learning rate, and an MLP with four hidden layers of $1,743$ dimensions. For the Terra Incognita dataset, we use a 0.5 dropout rate, $1e{-4}$ learning rate, and an MLP with a single hidden layer of $1,045$ dimensions. Early stopping was applied with a patience of 20 epochs. We optimize the hyperparameters of our model using random search.

Model	Backbone	Training	Test	Cis-Test Acc (%)	Trans-Test Acc (%)
CLIP [24]	ViT-B/32	OpenAI data	Terra Incognita	32.18	26.62
CLIP [24]	ViT-B/16	OpenAI data	Terra Incognita	39.14	34.67
BioCLIP [28]	ViT-B/16	TREEOFLIFE-10M	Terra Incognita	21.12	14.53
WildCLIP [11]	ViT-B/16	Snapshot Serengeti	Terra Incognita	40.38	38.90
WildCLIP-LwF [11]	ViT-B/16	Snapshot Serengeti	Terra Incognita	41.60	36.20
CATALOG (ours)	ViT-B/16	Snapshot Serengeti	Terra Incognita	**48.59**	**41.92**

Table 1. Zero-shot performance results of CATALOG and other foundation models in the Terra Incognita dataset (out-of-domain evaluation). All methods are trained in data that differ from the test dataset. The best method is highlighted in **bold**.

4.1. Quantitative Results

Comparison with the state-of-the-art in out-of-domain evaluation. Tab. 1 reports the performance metrics of various models trained in datasets like TREEOFLIFE-10M and Snapshot Serengeti, and tested on the Cis-Test and Trans-Test sets of the Terra Incognita dataset, reflecting the models' accuracy (%) in zero-shot learning scenarios. We observe that CLIP ViT-B/32 achieves an accuracy of 32.18% in Cis-Test and a Trans-Test accuracy of 26.62%, while CLIP ViT-B/16 improves these metrics to 39.14% and 34.67%, respectively. The WildCLIP model improves upon CLIP (ViT-B/16) with 40.38% on Cis-Test and 38.90% on Trans-Test due to its refinement in the Snapshot Serengeti dataset. We also test the version Learning without Forgetting (LwF) of WildCLIP, which slightly alters performance with 41.60% and 36.20% on Cis-Test and Trans-Test, respectively. In contrast, BioCLIP shows lower accuracy levels across both datasets, achieving 21.12% on Cis-Test and 14.53% on Trans-Test. CATALOG outperforms all previous FMs for camera-trap images, achieving 48.59% of accuracy in Cis-Test and 41.92% in Trans-Test. These results highlight the advancements in zero-shot, domain-invariant, and open-set capabilities achieved by CATALOG, surpassing previous state-of-the-art models.

In-domain performance comparison in the Snapshot Serengeti dataset. Tab. 2 shows a comparison of multiple models trained in the Snapshot Serengeti dataset. CLIP-MLP is a modified version of CLIP's image encoder where we add an MLP and we train it with the regular cross-entropy loss. This model achieves a test accuracy of 84.92%. However, it is worth noting that unlike WildCLIP and CATALOG, the CLIP-MLP model lacks open vocabulary capabilities due to the cross-entropy training. Therefore, CLIP-MLP is a strong baseline when no open-set capabilities are required. WildCLIP performs moderately well with a test accuracy of 61.78%, but it lags behind CATALOG. The variant of WildCLIP, WildCLIP-LwF, shows a slight improvement, achieving a test accuracy of 64.39%. The Learning without Forgetting (LwF) approach incorporated in this model appears to contribute positively, although the gain is not enough when compared to CLIP-MLP and CATALOG. CATALOG outperforms both CLIP-MLP and WildCLIP with 90.63%, showing its effectiveness for the case of in-domain evaluation.

Model	Loss Function	Test Acc (%)
CLIP-MLP [24]	Cross-entropy	84.92
WildCLIP [11]	Contrastive	61.78
WildCLIP-LwF [11]	Contrastive	64.39
CATALOG (ours)	Contrastive	**90.63**

Table 2. Performance comparison in Snapshot Serengeti (in-domain evaluation). All models use the ViT-B/16 backbone.

Model	Cis-Test Acc(%)	Trans-Test Acc(%)
CLIP-MLP [24]	77.62	71.88
WildCLIP [11]	**91.72**	**84.52**
WildCLIP-LwF [11]	88.96	82.86
CATALOG (ours)	89.64	84.32

Table 3. Performance comparison in the Terra Incognita dataset (in-domain evaluation). All models have the ViT-B/16 backbone.

In-domain performance comparison in the Terra Incognita dataset. Tab. 3 shows the performance of CLIP-MLP, WildCLIP, WildCLIP-LwF, and CATALOG when trained and evaluated on the Terra Incognita dataset. The CLIP-MLP model performs well with a Cis-Test accuracy of 77.62% and a Trans-Test accuracy of 71.88%. Although this model performs well, its lack of open vocabulary capabilities remains a limitation. WildCLIP achieves the highest Cis-Test accuracy of 91.72% and the best Trans-Test accuracy of 84.52%. WildCLIP-LwF slightly underperforms WildCLIP, with a Cis-Test accuracy of 88.96% and a Trans-Test accuracy of 82.86%. CATALOG achieves a Cis-Test accuracy of 89.64%, outperforming CLIP-MLP and WildCLIP-LwF, also performs competitively in the Trans-Test scenario with an accuracy of 84.32%, slightly lower than WildCLIP. Nevertheless, when we analyze the performance gap between the Cis-Test and Trans-Test in Terra Incognita, it is smaller in CATALOG compared to other methods. This demonstrates that CATALOG is effective at

CLIP	VLM	LLM	Templates	Cis-Test Acc (%)	Trans-Test Acc (%)
✗	✓	✗	✗	2.54	3.00
✗	✓	✗	✓	8, 37	12, 36
✗	✓	✓	✗	11, 20	10, 80
✗	✓	✓	✓	14.51	15.83
✓	✗	✗	✗	39.14	34.67
✓	✗	✓	✗	37.92	36.32
✓	✗	✗	✓	44.26	33.74
✓	✗	✓	✓	44.39	34.73
✓	✓	✓	✓	**48.59**	**41.92**

Table 4. Ablation studies for performance variations for different design choices of CATALOG. CLIP refers to the usage of the image encoder of CLIP. VLM refers to the usage of the Image-text Embedding module. LLM refers to the description generated by ChatGPT for each animal species. Finally, Templates refer to a set of predefined templates customized for our specific task in camera-trap image recognition. All models are trained on Snapshot Serengeti and evaluated on Terra Incognita (out-of-domain evaluation).

handling domain shifts, even within the same dataset.

4.2. Ablation Studies

We conduct several ablation studies about i) the impact of the CLIP's image encoder, ii) the use of the VLM and the different textual descriptions of the categories, iii) the loss function used to train CATALOG, and iv) the impact of using a text encoder capable of processing prompts longer than 77 tokens instead of BERT.

CLIP image encoder, VLM, LLM, and templates. We evaluate the performance of CATALOG when we remove the VLM, CLIP image encoder, the descriptions generated by the LLM, and the set of predefined templates customized for the specific task in camera-trap image recognition. Tab. 4 shows the results in the Cis-Test and Trans-Test for different design choices in CATALOG. Our findings show that the lowest performance occurs when we remove CLIP, the LLM descriptions, and templates (first row in Tab. 4). This is equivalent to setting $\alpha = 0$ in (4) and using for text descriptions the base prompt "A photo of a { }". This results in scores of 2.54% and 3.00% in Cis-Test and Trans-Test, respectively. Similarly, removing CLIP and the templates or CLIP and the LLM descriptions also leads to very poor performance (second and third rows in Tab. 4). When we remove CLIP and include VLM, the LLM descriptions, and the templates, we observe a little increase in performance, obtaining accuracies of 14.51% in the Cis-Test and 15.83% in the Trans-Test sets (fourth row in Tab. 4). These poor results highlight the important role of CLIP's image encoder in CATALOG, meaning that our VLM alone cannot replace CLIP image encoder.

We observe a significant performance increase when we include the CLIP image encoder (fith row in Tab. 4), with scores of 39.14% and 34.67% in Cis-Test and Trans-Test, respectively. This case reduces to the original CLIP model with the base prompt "A photo of a { }". When we incorporate the descriptions generated by the LLM (sixth row in Tab. 4), we obtain an increase in performance for the Trans-Test score (36.32%) but a decrease for the Cis-Test score (37.92%). When we include the templates (seventh row in Tab. 4), we observe an increase in performance in Cis-Test (44.26%) but a slight decrease in Trans-Test (33.74%). Finally, the combination of the LLM and template descriptions (eight row in Tab. 4) offers more robust results across Cis-Test (44.39%) and Trans-Test (34.73%). These results suggest that including textual descriptions can be beneficial, but their effectiveness may vary depending on the test dataset and the choice of text information.

Tab. 4 shows that the best performance is achieved by incorporating the VLM model, CLIP image encoder, the LLM descriptions, and templates (last row in Tab. 4), resulting in scores of 48.59% (Cis-Test) and 41.92% (Trans-Test). This highlights the importance of integrating textual embedding techniques to better capture the relationships between images and their categorical descriptions. The results also demonstrate how integrating FMs enhances the model's generalization performance. Furthermore, incorporating the image-text embeddings with the VLM and CLIP model provides an additional boost in accuracy, underscoring the effectiveness of this module for camera-trap image recognition.

Loss function	Cis-Test Acc(%)	Trans-Test Acc (%)
Sup. contrastive loss	45.64	37.02
Contrastive loss	**48.59**	**41.92**

Table 5. Ablation study on the choice of the loss function to train CATALOG in out-of-domain evaluation.

Evaluating different loss functions. Tab. 5 shows the comparison in performance of CATALOG when trained with the contrastive loss in (5) and the well-known supervised contrastive loss defined in [17]. The main difference between these two approaches is the elements we use as negative pairs. We observe that CATALOG obtains 45.64% in Cis-

Long CLIP	CLIP+BERT	Cis-Test Acc(%)	Trans-Test Acc (%)
✓	✗	28.55	18.05
✗	✓	**48.59**	**41.92**

Table 6. Ablation study on the choice of the text encoder to use in CATALOG for out-of-domain evaluation.

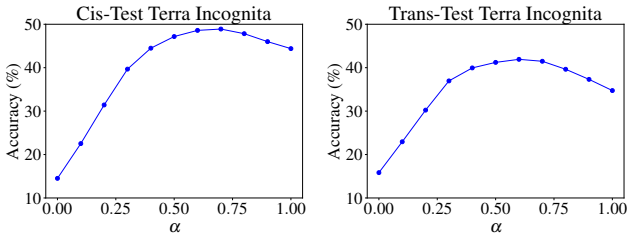
Test and 37.02% in Trans-Test when trained with the supervised contrastive loss. In contrast, our model achieves 48.59% and 41.92% in Cis-Test and Trans-Test, respectively when trained with the contrastive loss. This indicates that the contrastive loss is effective in improving the model's ability to distinguish between different categories, leading to higher performance in both test scenarios. Even though the supervised contrastive loss provides reasonable performance, it does not match the effectiveness of the standard contrastive loss for camera-trap image recognition.

Evaluating text encoder. We evaluate the impact of using different text encoders on the performance of CATALOG in out-of-domain evaluation scenarios. Specifically, we compare the performance of the CLIP [24] and BERT [6] text encoder combination against the Long CLIP text encoder [39], which is capable of processing longer textual prompts (248 tokens). The study also uses the Long CLIP image encoder, ensuring a consistent comparison between the models. All hyperparameters were kept unchanged and optimized based on the values found for the out-of-domain evaluation.

Tab. 6 shows the Cis-Test and Trans-Test accuracy results for the two encoder setups. The CLIP and BERT text encoder combination achieved the highest performance, with Cis-Test and Trans-Test accuracies of 48.59% and 41.92%, respectively. In contrast, using the Long CLIP text encoder significantly reduced performance, achieving only 28.55% accuracy on the Cis-Test and 18.05% on the Trans-Test. This performance drop suggests that simply increasing the text encoder's capacity to process more tokens does not guarantee improved alignment and performance in out-of-domain scenarios.

4.3. Sensitivity to the Hyperparameter α

Fig. 4a and 4b show the change of CATALOG's performance to variations of the parameter α between 0 and 1 in (4) 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 $\mathbf{W}$, are complementary. Both Fig. 4a and 4b show that the optimal value for α is 0.6. This indicates that giving nearly equal importance to both matrices provides the best accuracy. When α deviates from the optimal value, the accuracy decreases, suggesting that overemphasizing either matrix leads to a loss of valuable information for classification. This consistency across both evaluation sets highlights the robustness of the model's performance when information from both matrices is used.



(a) Cis-Test set. (b) Trans-Test set.

Figure 4. Sensibility analysis of the hyperparameter α of CATALOG in the Terra Incognita dataset for out-of-domain evaluation.

4.4. Limitations

Although FMs have shown promising results in recognizing animal species in camera-trap images, there is still a noticeable difference between their performance on out-of-domain (Tab. 1) and in-domain (Tab. 3) data evaluation in Terra Incognita. This highlights the need to enhance the generalization capabilities of these models for camera-trap image recognition. In the supplementary material, we provide a more detailed analysis of the limitations of the CATALOG model. Two potential directions for addressing this issue include: i) collecting a large-scale dataset of image-text pairs of camera-trap images to train a foundation model like CLIP, and ii) leveraging expert knowledge through textual information about each animal species to fill the domain gap. These ideas warrant further exploration and are left for future work.

5. Conclusions

In this paper, we introduced CATALOG, a new model that integrates multiple FMs to address the performance loss caused by domain changes in camera-trap image recognition. CATALOG tackles domain shifts by using robust features extracted by CLIP, LLaVA, and BERT, combined with stronger category descriptions generated by an LLM and predefined templates specific to the camera-trap context. Our extensive experiments show that CATALOG outperforms state-of-the-art models in camera-trap image classification, especially in the case when there are domain shifts between the training and testing dataset, all while maintaining its open vocabulary capabilities.

Acknowledgments. This work was supported by Universidad de Antioquia - CODI and Alexander von Humboldt Institute for Research on Biological Resources (project 2020-33250), and by the ANR (French National Research Agency) under the JCJC project DeSNAP (ANR-24-CE23-1895-01).

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2

[2] Sara Beery, Dan Morris, and Siyu Yang. Efficient pipeline for camera trap image review. *arXiv preprint arXiv:1907.06772*, 2019. 4

[3] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in Terra Incognita. In *Proceedings of the European Conference on Computer Vision*, 2018. 1, 2, 5

[4] Yasser Benigmim, Subhankar Roy, Slim Essid, Vicky Kalogeiton, and Stéphane Lathuilière. Collaborating foundation models for domain generalized semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 2020. 1, 2

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, 2019. 1, 2, 8

[7] Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. A survey of vision-language pre-trained models. *arXiv preprint arXiv:2202.10936*, 2022. 2

[8] Yu Du, Fangyun Wei, Zihe Zhang, Miaojing Shi, Yue Gao, and Guoqi Li. Learning to prompt for open-vocabulary object detection with vision-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1

[9] Zalan Fabian, Zhongqi Miao, Chunyuan Li, Yuanhan Zhang, Ziwei Liu, Andrés Hernández, Andrés Montes-Rojas, Rafael Escucha, Laura Siabatto, Andrés Link, et al. Multimodal foundation models for zero-shot animal species recognition in camera trap images. *arXiv preprint arXiv:2311.01064*, 2023. 1, 2, 4

[10] Ruohuan Fang, Guansong Pang, and Xiao Bai. Simple image-level classification improves open-vocabulary object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024. 1

[11] Valentin Gabeff, Marc Rußwurm, Devis Tuia, and Alexander Mathis. WildCLIP: Scene and animal attribute retrieval from camera trap data with domain-adapted vision-language models. *International Journal of Computer Vision*, 2024. 1, 2, 5, 6

[12] Jhony H. Giraldo, Augusto Salazar, Alex Gomez-Villa, and Angélica Diaz-Pulido. Camera-trap images segmentation using multi-layer robust principal component analysis. *The Visual Computer*, 2019. 1

[13] Alex Gomez-Villa, Augusto Salazar, and Francisco Vargas. Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks. *Ecological informatics*, 41, 2017. 1

[14] Andres Hernandez, Zhongqi Miao, Luisa Vargas, Rahul Dodhia, and Juan Lavista. Pytorch-Wildlife: A collaborative deep learning framework for conservation, 2024. 5

[15] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Barun Patra, et al. Language is not all you need: Aligning perception with language models. *Advances in Neural Information Processing Systems*, 2024. 2

[16] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. 2

[17] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 2020. 7

[18] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in Neural Information Processing Systems*, 2024. 2, 4

[19] Utkarsh Mall, Bharath Hariharan, and Kavita Bala. Zero-shot learning using multimodal descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1

[20] Zhongqi Miao, Yuanhan Zhang, Zalan Fabian, Andres Hernandez Celis, Sara Beery, Chunyuan Li, Ziwei Liu, Amrita Gupta, Md Nasir, Wanhua Li, et al. New frontiers in ai for biodiversity research and conservation with multimodal language models. 2024. 2

[21] Danielle L Norman, Philipp H Bischoff, Oliver R Wearn, Robert M Ewers, J Marcus Rowcliffe, Benjamin Evans, Sarab Sethi, Philip M Chapman, and Robin Freeman. Can CNN-based species classification generalise across variation in habitat within a camera trap survey? *Methods in Ecology and Evolution*, 2023. 2

[22] Poojan Oza, Vishwanath A Sindagi, Vibashan Vishnukumar Sharmini, and Vishal M Patel. Unsupervised domain adaptation of object detectors: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[23] Sarah Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 1

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. 1, 2, 4, 5, 6, 8

[25] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022. 1

[26] Stefan Schneider, Saul Greenberg, Graham W Taylor, and Stefan C Kremer. Three critical factors affecting automated image species recognition performance for camera traps. *Ecology and Evolution*, 2020. 1, 2

[27] Fanny Simões, Charles Bouveyron, and Frédéric Precioso. DeepWILD: Wildlife identification, localisation and estimation on camera trap videos using deep learning. *Ecological Informatics*, 2023. 2

[28] Samuel Stevens, Jiaman Wu, Matthew J Thompson, Elizabeth G Campolongo, Chan Hee Song, David Edward Carlyn, Li Dong, Wasila M Dahdul, Charles Stewart, Tanya Berger-Wolf, et al. BioCLIP: A vision foundation model for the tree of life. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1, 2, 5, 6

[29] AB Swanson, M Kosmala, CJ Lintott, RJ Simpson, A Smith, and C Packer. Data from: Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna, 2015. 1, 2, 5

[30] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2

[31] Devis Tuia, Benjamin Kellenberger, Sara Beery, Blair R Costelloe, Silvia Zuffi, Benjamin Risse, Alexander Mathis, Mackenzie W Mathis, Frank van Langevelde, Tilo Burghardt, et al. Perspectives in machine learning for wildlife conservation. *Nature Communications*, 2022. 1, 2

[32] Jianzong Wu, Xiangtai Li, Shilin Xu, Haobo Yuan, Henghui Ding, Yibo Yang, Xia Li, Jiangning Zhang, Yunhai Tong, Xudong Jiang, et al. Towards open vocabulary learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 1

[33] Xiaoshi Wu, Feng Zhu, Rui Zhao, and Hongsheng Li. CORA: Adapting CLIP for open-vocabulary detection with region prompting and anchor pre-matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1

[34] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1

[35] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1

[36] Kisu Yang, Dongyub Lee, Taesun Whang, Seolhwa Lee, and Heuiseok Lim. Emotionx-ku: Bert-max based contextual emotion classifier. *arXiv preprint arXiv:1906.11565*, 2019. 4

[37] Xiangyu Yue, Zangwei Zheng, Shanghang Zhang, Yang Gao, Trevor Darrell, Kurt Keutzer, and Alberto Sangiovanni Vincentelli. Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1

[38] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1

[39] Beichen Zhang, Pan Zhang, Xiaoyi Dong, Yuhang Zang, and Jiaqi Wang. Long-CLIP: Unlocking the long-text capability of CLIP. In *Proceedings of the European Conference on Computer Vision*, 2024. 4, 8

[40] Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2

CATALOG: A Camera Trap Language-guided Contrastive Learning Model

Julian D. Santamaria[1], Claudia Isaza[1], Jhony H. Giraldo[2]
[1] SISTEMIC, Faculty of Engineering, Universidad de Antioquia-UdeA, Medellín, Colombia.
[2] LTCI, Télécom Paris, Institut Polytechnique de Paris, Palaiseau, France.
{julian.santamaria, victoria.isaza}@udea.edu.co, jhony.giraldo@telecom-paris.fr

This supplementary material includes some examples of the predefined templates and detailed examples of specific prompts used to generate the category descriptions with the LLM [**?**]. Additionally, it provides the prompt used to create textual descriptions of animals present in cropped images using LLaVA [**?**].

A. Templates

In this section, we present some examples of templates designed for the camera-trap image recognition task. These templates have been adapted from the ImageNet templates used in CLIP [**?**] and are shown 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 $\mathcal{C}^D$.

B. Prompts

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

B.1. Prompt LLM

The prompt utilized to get the LLM description of the animal species follows a structure based on the methodology discussed in [**?**] and is shown 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.

B.2. Prompt LLaVA

The prompt used in LLaVA aligns with the approach employed in [**?**] and is structured as follows:

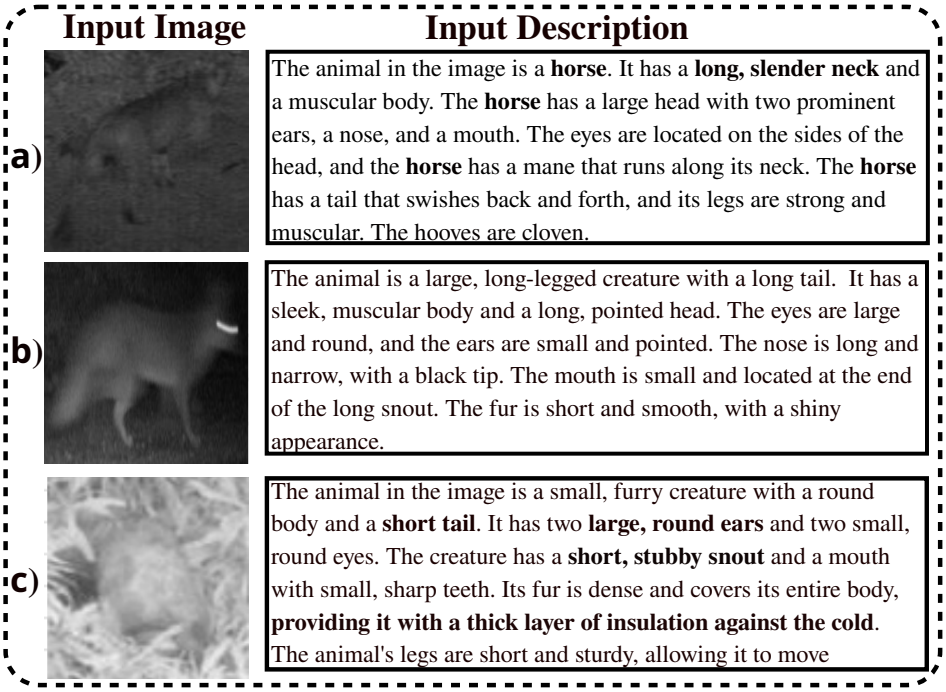


Figure 1. Failure cases of the CATALOG model for camera trap image classification. **case a:** The VLM generates an incorrect description with details that do not match the input image. **case b:** A blurry image results in a vague and unhelpful description. **case c:** When the input image is very unclear, the VLM generates a random and irrelevant description. These examples show that when the descriptions are wrong or not informative, the model makes incorrect predictions.

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.

C. Analysis of CATALOG's Errors

In this section, we provide a more detailed analysis of the limitations of the CATALOG model. Fig. 1 illustrates how sensitive the model is to the input descriptions generated for the VLM. These descriptions provide additional information to help the model make better classifications. However, when the descriptions are wrong or unclear, the model makes incorrect predictions.

To clarify the model's sensitivity to input descriptions, Fig. 1 presents three examples:

- In **case a**, the VLM hallucinates by adding details that are not in the image. For example, it describes the animal as a horse, even though the input image does not match this description. This leads to a completely incorrect prediction.
- In **case b**, the input image is blurry and hard to interpret. The VLM generates a vague description with little useful information. As a result, the model does not get enough context to make the correct prediction.
- In **case c**, the image is so unclear that the VLM creates a random description that has no connection to the input. This random description further confuses the model, leading to a wrong prediction.

These examples show that the CATALOG model depends on the accuracy and quality of the descriptions created by the VLM. When the descriptions are not reliable or informative, the model struggles to classify the input correctly. Improving the robustness of the VLM is crucial for handling noisy or unclear inputs.