4S-Classifier: Empowering Conservation through Semi-Supervised Learning for Rare and Endangered Species

Hongyang He[1], Hongyang Xie[1], Guodong Shen[1], Boyang Fu[1], Haochen You[2], Victor Sanchez[1]

[1]Signal and Information Processing (SIP) Lab
Department of Computer Science, University of Warwick
Coventry, CV4 7AL, UK
[2]Graduate School of Arts and Sciences, Columbia University
120 West 105th Street, New York, NY 10025, USA
{hongyang.he, hongyang.xie, guodong.shen, boyang.fu, v.f.sanchez-silva}@warwick.ac.uk, hy2854@columbia.edu

Abstract

The survival of numerous endangered wildlife species is increasingly jeopardized by drastic climate changes, ecological disturbances, and human activities, leading to rapid population declines. Identifying endangered and rare species is essential for biodiversity conservation. However, many rare species are recognizable only by specialized biologists, making the labeling process both challenging and costly. Moreover, even with labeled data, training models on these limited categories often results in inherent learning biases. To address high labeling costs and challenges associated with sparse label learning, we propose a novel semi-supervised learning framework, 4S-Classifier, which comprises two key modules: a Rare Species Bank (RSBank) and an Attention-based Rare Species Embedding (RSEmbed). The RSBank module stores embeddings of rare species across multiple training epochs, using clustering, kernel density estimation, and confidence scores to enhance the learning of underrepresented categories. For the sparse samples within the RSBank, the RSEmbed module acts as a fusion-based augmentation approach utilizing embeddings to improve model performance on sparse and rare species data. It is applied to improve training effectiveness. By integrating both modules, our framework achieves a classification accuracy of 88.37% on an endangered species dataset (iNaturalist) and 91.56% on a wildlife dataset (Wildlife Insights) with only 25% labeled data, demonstrating its outstanding performance.

Code — https://github.com/SIPLab24/4S-Classifier

Introduction

In the field of species conservation, especially for rare species detection based on computer vision, the scarcity and imbalance of labeled data have long been key factors limiting model performance. Figure 1 shows the imbalance in labeled data of the Wildlife Insights dataset. To overcome these challenges, Semi-Supervised Learning (SSL) has emerged as an effective solution, leveraging a large amount of unlabeled data and a small amount of labeled data to enhance the model's generalization capabilities. However, traditional SSL methods face several issues, particularly when handling classes with limited examples, e.g., rare species, where the imbalance of labeled data and the bias of pseudo-labels often lead to limited model performance.

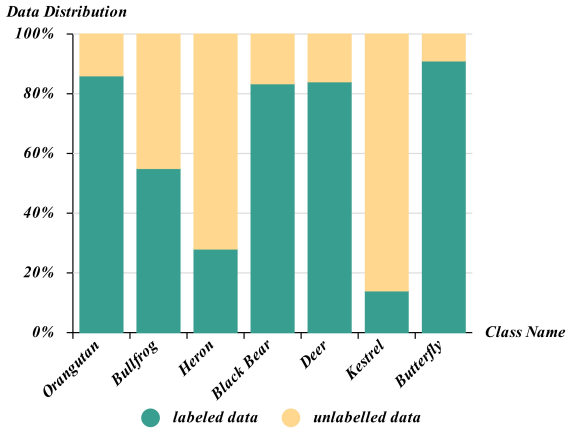

Figure 1: The distribution of labeled and unlabeled data across different species categories. This figure illustrates the imbalanced representation of species labels in the Wildlife Insights dataset (Ahumada et al. 2020), highlighting the challenges faced in training models effectively due to the scarcity of labeled samples for certain endangered species.

To address the issues of sparse samples and high annotation costs for rare species, we propose a new semi-supervised learning framework that combines the RSBank and RSEmbedd modules. The RSBank (Rare Species Bank) module is designed to reveal the distribution of rare species. It effectively integrates the embedding vectors of these species with other data by constructing distributional features for rare species. On the other hand, the RSEmbedd (Rare Species Embedding) module uses the Wide ResNet network to generate embedding vectors for rare species. By combining labeled data with pseudo-labeled unlabeled data, it enhances the expression ability of rare species. In this way, the framework can effectively learn the features of rare species with limited labeled data while mitigating the negative impacts of pseudo-label bias and class distribution imbalance.

The innovation of this new framework lies in its capa-

bility to maximize the use of distributional information and embedding vectors of rare species through the combined use of the RSBank and RSEmbedd modules. This significantly improves the accuracy and robustness for rare species detection. Experimental results show that our framework has significant advantages in rare species detection tasks, as it handles labeled data imbalance and pseudo-label challenges more effectively compared to traditional methods, providing an efficient solution for species conservation and ecological research. Our contributions are summarized as follows:

- We propose 4S-Classifier, a semi-supervised learning framework specifically designed for the classification of images depicting rare and endangered species.
- We introduce an Attention-based Rare Species Embedding module to specifically focus on and capture the distinctive features of rare and endangered wildlife images during training.
- We introduce a Rare Species Bank module to track and store labels of sparse species images, mitigating the severe class imbalance in the data.
- In the field of vision-based wildlife classification, our work offers a solution to address the challenges of sample scarcity and high labeling costs for rare and endangered species images, with validated efficiency on multiple datasets.

Interestingly, the name ***4S-Classifier*** carries a dual significance: not only does it imply **For Species**, but it also stands for a " **S**emi-**S**upervised **S**ystem for **S**pecies " coincidentally encapsulating four "**S**"s. This naming is both intentional and meaningful, reflecting the core focus and methodology of our approach.

Related Work

Advances in Wildlife Detection Technology

Wildlife detection, which utilizes artificial intelligence technologies to monitor animals, has become an increasingly prominent research area for ecological conservation. Recent works in this area can be generally divided into two categories: supervised learning-based methods (Bakana, Zhang, and Twala 2024; Liu et al. 2024; Kellenberger, Marcos, and Tuia 2018; Peng et al. 2020; Yang et al. 2023; Zhang and Cai 2023; Lu and Lu 2023) and unsupervised learning-based methods (Chabot, Stapleton, and Francis 2022; Zheng et al. 2022b; Manohar, Kumar, and Kumar 2016; Seraj, Ayele, and Meratnia 2019). Object detection methods generally adopt either one-stage or two-stage detection frameworks. One-stage frameworks, such as YOLO (Redmon 2016) and SSD (Liu et al. 2016), combine object classification and localization into a single step, delivering high speed but often sacrificing accuracy. Two-stage frameworks, such as Faster R-CNN (Girshick et al. 2014) and Feature Pyramid Networks (FPN) (Lin et al. 2017), separate region proposal and classification steps, allowing for more precise detection at the expense of higher computational costs. For instance, (Chabot, Stapleton, and Francis 2022) introduced a multimodal approach that combines the semantics of animal images and sounds, achieving impressive performance on the Baffin Bay Wildlife dataset. (Zheng et al. 2022b) introduced an attention mechanism-based approach that can effectively distinguish real objects from dynamic backgrounds, thus enhancing detection accuracy even in complex scenarios with high false positive rates.

Unsupervised learning-based detection methods rely primarily on unlabeled data for training by leveraging template matching techniques. Existing unsupervised frameworks typically adhere to a common routine. They initially segment images through the analysis of feature distributions, utilizing segmentation techniques like thresholding, edge detection, histogram analysis, and wavelet transforms. Following segmentation, they proceed to feature extraction and dimensionality reduction, employing techniques such as non-negative matrix factorization (NMF), principal component analysis (PCA), and t-SNE. With the extracted compact features, they classify the segmented images using clustering algorithms such as K-means and K-nearest neighbors (KNN). For instance, (Christiansen et al. 2014) used a KNN classifier to identify animal regions in airborne infrared images. Similarly, (Guerrero et al. 2023) used thermal imaging to detect seals by marking points above a temperature threshold and distinguishing between adult and juvenile groups based on size and average cluster temperature. Although unsupervised methods bypass the need for labeled data during training, they often fall short in robustness and accuracy compared to supervised methods, limiting their broader applicability.

Semi-supervised Learning with Sparse Samples

Semi-supervised learning aims to effectively leverage unlabeled data to overcome the challenge of sparse samples. Consequently, several frameworks have been proposed to efficiently process and learn from large quantities of unlabeled data. For example, (Chen et al. 2023b) proposed Expectation Maximization Learning (EML) and Adaptive Negative Learning (ANL) to predict the distribution of unlabeled data and adaptively filter noisy data. This approach allows for the expansion of unlabeled data while reducing noise and enhancing data reliability. (Fini et al. 2023) introduced a self-supervised clustering approach that groups unlabeled data into reliable clusters. This approach refines pseudo-label quality, minimizes noisy samples, and provides more consistent data for semi-supervised learning. (Chen et al. 2023a) developed the SoftMatch mechanism, which incorporates soft labels to avoid overfitting on uncertain data. This approach strikes a balance between managing noisy data and maximizing data utilization, leading to superior performance under sparse sample conditions. (Wang et al. 2022b) proposed a self-adaptive thresholding technique that allows the model to dynamically adjust its confidence levels for unlabeled data. This technique makes the training process more robust and adaptive to varying noise levels in sparse data. (Wang et al. 2022a) tackled the inherent data imbalance by employing pseudo-labels. Specifically, they proposed a debiasing mechanism to adjust class distributions in pseudo-labeled data. According to their findings, this mechanism can dynamically expand or shrink the selection of soft labels, enabling more fine-grained learning under sparse

Algorithm 1: Distribution Detection and Rare Species Embedding Selection in the RSBank module

Input: $\{E_j\}_{j=1}^{n}$
Parameter: KDE bandwidth $h(\mathbf{e}_i)$, GMM, density threshold τ_f
Output: $\{E_{\text{rare}}\}$ in RSBank

```
Clustering: GMM on {E_j}_{j=1}^n to form clusters
Identify low-density clusters for rare species
for each E_i in low-density clusters do
    Calculate h(E_i), apply KDE to estimate f(E_i)
end for
Generate DPM with density estimates f(E_i)
Set threshold τ_f based on DPM percentiles
for each E_i in DPM do
    if f(E_i) < τ_f then
        Add E_i to {E_rare}
    end if
end for
Store {E_rare} in RSBank
return {E_rare}
```

sample conditions.

Proposed Framework

Figure 2 illustrates the overall pipeline of our 4S-Classifier framework. It mainly consists of the RSEmbed module and the RSBank module. The RSBank module is built based on a distribution detection method for sparse class labels. It leverages both clustering in the embedding space and probability density estimation to uncover the label distribution of sparse species. The RSEmbed module is designed to extract and integrate rich representations and key features from images and pseudo-label embeddings, guided by a pre-trained attention map. The following sections delve into the details of these two modules, the complete 4S-Classifier architecture, and the loss function used for training.

Rare Species Bank (RSBank)

The RSBank module is embedded within the unlabeled data training branch of the 4S-Classifier. It acts as a "Memory Bank" for rare and endangered species; in other words, it clusters the image embeddings of rare species to build a Kernel Density Estimation (KDE) model (Kim and Scott 2012). This model identifies rare species based on probability maps of their distribution, generated after multiple epochs of training. Note that the RSBank module operates under an assumption that is clearly observed in the wild: the data distribution for rare and endangered species heavily deviates from that of common species.

For each species i, we compute an embedding vector $\mathbf{e}_i$ in a high-dimensional space from the final hidden layer of Wide ResNet28-8 (Wu, Shen, and Van Den Hengel 2019). Subsequently, the embeddings are clustered based on their similarities in the embedding space. Due to the severe data imbalance between rare and common species, the embedding vectors are unevenly distributed in the embedding space. Thus, we use Gaussian Mixture Models (GMM)(Reynolds et al. 2009) for clustering:

$$GMM(\mathbf{e}_i) = c_k, \tag{1}$$

where c_k denotes the k^{th} cluster. For each cluster, we calculate the density as the number of embeddings per unit volume around the cluster centroid, denoted by V_k. For example, the density d_k for cluster c_k with centroid μ_k is estimated as:

$$d_k = \frac{|c_k|}{V_k}. \tag{2}$$

Identifying low-density (sparse) clusters is crucial, as these are most likely to represent rare species. Let C_R be the set of rare classes, which can be used to identify categories with a significantly high proportion of rare species:

$$C_R = \left\{ c_k \mid \frac{\text{Number of rare species in } c_k}{\text{Total instances in } d_k} > \text{threshold} \right\}, \tag{3}$$

where d_k represents the sample distribution of class c_k, and the threshold determines the significance of rare species in each class. Eq. 3 effectively filters out classes where rare species are highly concentrated.

We apply KDE to estimate the probability density $p(\mathbf{e})$, quantifying the likelihood of sampling an embedding $\mathbf{e}$ from the rare clusters C_R as follows:

$$p(\mathbf{e}) = \frac{1}{|C_R|h} \sum_{e_i \in C_R} K\left(\frac{\mathbf{e} - \mathbf{e}_i}{h}\right). \tag{4}$$

Note that KDE can model the distribution of rare species in the embedding space. Specifically, Eq. 4 estimates the density $p(\mathbf{e})$ at any given point $\mathbf{e}$, representing the likelihood of observing a rare species at that location in the embedding space. The Gaussian kernel function $K(\cdot)$ ensures a smooth and continuous density estimate, while the bandwidth h controls the level of smoothness, balancing fine-grained details and generalization. By incorporating the embeddings of rare species, this method effectively highlights regions in the embedding space where rare species are more likely to exist, aiding in the accurate detection and classification of these species.

To capture the sparsity more accurately, we use an adaptive bandwidth h_i in Eq. 4 for each rare species, where h_i is smaller for more isolated species (increasing density precision) and larger for those close to common clusters:

$$h_i = \frac{1}{d(\mathbf{e}_i, \mathbf{e}_c)}, \tag{5}$$

where $\mathbf{e}_c$ refers to the nearest embedding of a common species, representing the proximity of rare species $\mathbf{e}_i$ to the most similar common cluster. This simplifies the formula while retaining clarity, with $\mathbf{e}_c$ specifically denoting the nearest common reference point in the embedding space. Eq. 4 generates a density probability map (DPM) that represents the likelihood of encountering rare species, revealing the spatial distribution of sparse classes. Algorithm 1 shows the implementation of the RSBank module. In addition, Fig.3 shows the density probability map of embeddings for the rare species class and common species class in a 3D space for the Wildlife Insights dataset.

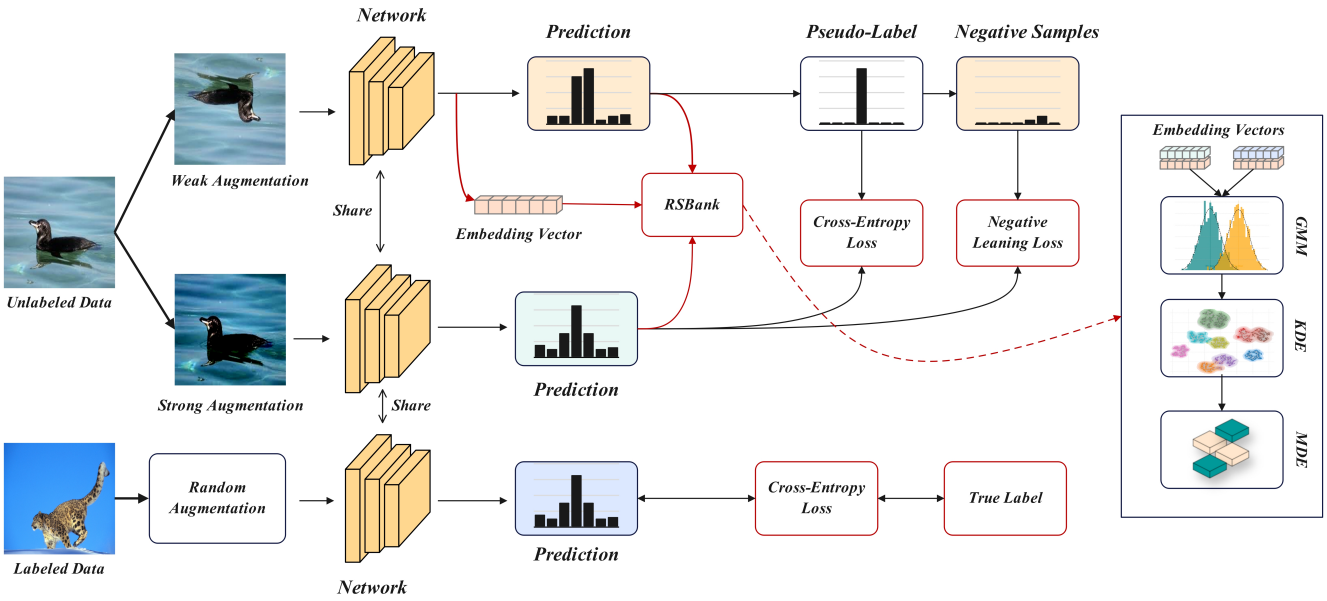

Figure 2: Overview of proposed 4S-Classifier, which builds upon and improves FullMatch (Chen et al. 2023b) by retaining only the ANL (Adaptive Negative Learning) and incorporating two modules: RSEmbed and RSBank. The figure shows the RSBank module.

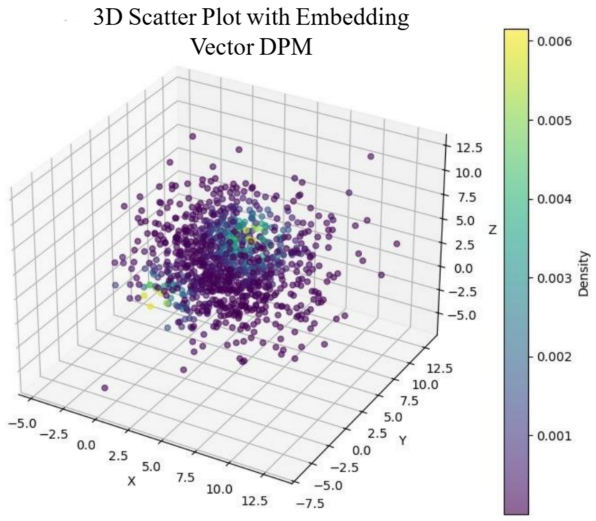

Figure 3: 3D DPM of embedding vectors. This plot shows the distribution of embedding vectors, likely associated with species or other classification entities. It provides insights into the separability and distributional layout of embeddings generated by the model, which confirms the effectiveness of the RSEmbed module.

Attention-based Rare Species Embedding (RSEmbed)

The embedding vectors of selected rare species as identified by the RSBank module are fused with the embedding vectors of labeled and unlabeled data, a process executed by the RSEmbed module. It is important to note that the embedding vectors of labeled data are associated with their ground truth labels, while the embedding vectors of unlabeled data are linked to the pseudo-labels predicted by Wide ResNet28-8 (Wu, Shen, and Van Den Hengel 2019). After these embeddings are appropriately aligned and fused using the RSEmbed module, they are used for subsequent training. In other words, the RSEmbed modules is an augmentation mechanism that enhances the 4S-Classifier's generalization capability by aligning the representations of rare species across different data types (Gao et al. 2020). An overview of the fusion mechanism of the RSEmbed module is shown in Fig. 4.

Instead of calculating attention weights based solely on similarity scores, we use a pre-trained SENet (Hu, Shen, and Sun 2018) to dynamically learn these weights. SENet generates attention scores by first applying global average pooling to "squeeze" spatial information into a channel-wise descriptor. To compute the attention score for each embedding E_j, we create a joint representation by concatenating the rare species embedding E_{rare} with E_j, It is important to note that $E_{\text{rare}}$and E_j here are the embeddings processed through RSBank, which are different from $\mathbf{e}_c$ and $\mathbf{e}_i$ mentioned earlier in RSBank method. This joint representation is processed by the pretrained SENet to obtain a scalar attention score that reflects the importance of E_j with respect to E_{rare}, which is formulated as follows:

$$s_j = \text{SENet}_{\text{pre}}(\text{concat}(E_{\text{rare}}, E_j)). \tag{6}$$

SENet is trained end-to-end as an integral part of the fusion process, dynamically adjusting attention scores based on the input data during training. The obtained attention scores are then normalized using the softmax function:

$$\alpha_{\text{rare}} = \frac{\exp(s_{\text{rare}})}{\exp(s_{\text{rare}}) + \sum_{k=1}^{n} \exp(s_k)}, \tag{7}$$

$$\alpha_j = \frac{\exp(s_j)}{\exp(s_{\text{rare}}) + \sum_{k=1}^{n} \exp(s_k)}, \tag{8}$$

where α_{rare} and α_j are attention weights computed by the softmax function. They determine the overall importance of the rare species embedding E_{rare} and the other embeddings E_j in the final fusion. s_{rare} and s_j are attention scores, which are intermediate values that reflect the raw importance of the embeddings before the softmax normalization. They are computed dynamically by passing the embeddings through an attention network.

Instead of using a simple weighted sum, we apply gates to control each embedding's contribution to the final fused embedding. These gates are learnable parameters that decide, at the feature level, how much information from each embedding should be retained in the fusion. This allows the model to selectively emphasize important features within each embedding while suppressing less relevant ones. This process can be illustrated as:

$$g_{\text{rare}} = \sigma(W_{\text{gate}} E_{\text{rare}}), \tag{9}$$

$$g_j = \sigma(W_{\text{gate}} E_j), \tag{10}$$

where W_{gate} is a learned weight matrix, σ is the sigmoid activation function, and g_{rare} and g_j are gating vectors computed by the sigmoid function. Each gating vector adjusts the contribution of features within an embedding at a granular feature-wise level.

For each embedding, either E_{rare} or E_j, the gating vector g produces values between 0 and 1 for each feature, allowing the model to retain, suppress, or amplify specific feature dimensions in the embedding. The gated embeddings are computed as:

$$\tilde{E}_{\text{rare}} = g_{\text{rare}} \odot E_{\text{rare}}, \tag{11}$$

$$\tilde{E}_j = g_j \odot E_j, \tag{12}$$

where $\odot$ represents element-wise multiplication.

Next, we combine the gated embeddings using the attention weights α_{rare} and α_j computed from Eq. 7 and Eq. 8:

$$E_{\text{fused}} = \alpha_{\text{rare}} \cdot \tilde{E}_{\text{rare}} + \sum_{j=1}^{n} \alpha_j \cdot \tilde{E}_j. \tag{13}$$

By allowing for a selective focus on the most relevant features within each embedding, the gates improve the quality of the fused embedding by emphasizing or diminishing certain feature dimensions. Algorithm 2 details the implementation of the RSBank module.

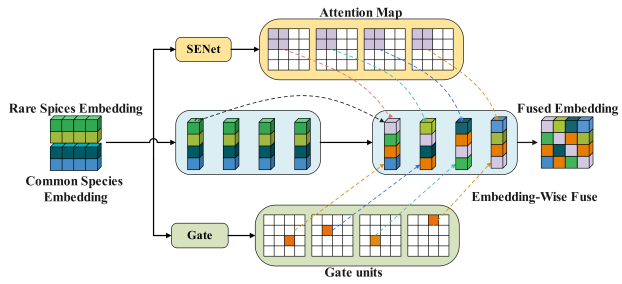

Figure 4: The RSEmbed module. The attention map from the pre-trained SENet and the gates focus on specific dimensions, and their weights are used to fuse the embedding vectors of rare species samples with those of other samples. The upper path shows the generation of the feature map, while the lower path shows the generation of the gate weight matrix.

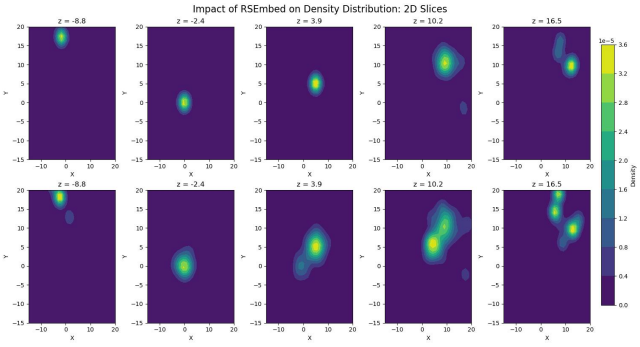

Figure 5: Impact of the RSEmbed module on density distribution of Rare Species Categories. The figure shows 2D density slices of rare species embeddings before (top row) and after (bottom row) using the RSEmbed module. Each subplot represents a density slice at a specific z-coordinate (e.g., z=-8.8, -2.4, 3.9, 10.2, 16.5). Post-RSEmbed slices (bottom row) display enhanced clustering and separability, highlighting RSEmbed's impact on *organizing* the embedding space for rare species.

4S-Classifier Architecture

During training, the 4S-Classifier framework utilizes advanced Adaptive Negative Learning (ANL) (Chen et al. 2023b), a technique originally proposed in the Fullmatch work for negative sample learning. ANL assigns negative pseudo-labels as additional targets for loss computation (Chen et al. 2023b). As shown in Fig. 2, labeled data, after undergoing random augmentation, is trained in a supervised manner using Wide ResNet28-8 (Wu, Shen, and Van Den Hengel 2019; Chen et al. 2023b). The weights learned from the supervised Wide ResNet28-8 are shared with its semi-supervised counterpart through Exponential Moving Average (EMA) (Cai et al. 2021).

Unlabeled data undergoes weak and strong augmentations separately before being fed into the Wide ResNet28-8 with shared weights for classification. Weakly-augmented samples are pseudo-labeled using Wide ResNet28-8 (Wu, Shen,

Algorithm 2: Advanced Embedding Fusion with Attention and Gating Mechanisms

Input: $E_{\text{rare}}, \{E_j\}_{j=1}^{n}$
Parameter: $\texttt{SENet}, W_{\text{gate}}$
Output: E_{fused}

$s_{\text{rare}}, s_j \leftarrow \texttt{SENet}(E_{\text{rare}}), \texttt{SENet}(E_j), \forall E_j$
Normalize $s_{\text{rare}}, s_j \rightarrow \alpha_{\text{rare}}, \alpha_j$
$g_{\text{rare}}, g_j \leftarrow \sigma(W_{\text{gate}} E_{\text{rare}}), \sigma(W_{\text{gate}} E_j), \forall E_j$
$\tilde{E}_{\text{rare}}, \tilde{E}_j \leftarrow g_{\text{rare}} \odot E_{\text{rare}}, g_j \odot E_j$
$E_{\text{fused}} \leftarrow \alpha_{\text{rare}} \cdot \tilde{E}_{\text{rare}} + \sum_{j=1}^{n} \alpha_j \cdot \tilde{E}_j$
return E_{fused}

and Van Den Hengel 2019; Chen et al. 2023b). Notably, for pseudo-labeled samples with high confidence, the embedding vectors from the final layer of Wide ResNet28-8 serve as input to the RSBank module. This module then identifies and filters out sparsely distributed classes with limited labeled data. These filtered classes are subsequently passed to the RSEmbed module, along with labeled data, unlabeled data, and high-confidence pseudo-labeled data. The augmented samples after the RSEmbed fusion are used for further semi-supervised training. Figure 5 illustrates the changes in the feature embedding vectors of rare species classes after the RSEmbed module.

Since the fused embedding vectors also undergo weight allocation, the significance of ANL becomes evident here. Its composite dual loss of positive and negative predictions enables the model to better infer classification probabilities for rare classes. Through the learning and fusion of embedding vectors, issues of data imbalance and labeling in sparse samples are alleviated, thereby reducing the model's learning bias.

Loss Function in 4S-Classifier

The 4S-Classifier is trained using a composite loss function, formulated as:

$$\mathcal{L}_{\text{sum}} = \mathcal{L}_s + \mathcal{L}_{us} + \alpha \cdot \mathcal{L}_{anl} + \beta \cdot \mathcal{L}^{rs}_{\text{cons.}} + (1-\beta) \cdot \mathcal{L}^{rs}_{\text{ce}}, \quad (14)$$

where $\mathcal{L}_s$ is the fully supervised learning loss for labeled data (Chen et al. 2023a), and $\mathcal{L}_{us}$ is the semi-supervised learning loss for pseudo-labeled data (Chen et al. 2023a), both derived from traditional semi-supervised learning frameworks (Zhang et al. 2021; Berthelot et al. 2021). $\mathcal{L}_{anl}$ refers to the ANL loss described earlier.

In addition, we introduce a consistency loss $\mathcal{L}^{rs}_{\text{cons.}}$ and a cross-entropy loss $\mathcal{L}^{rs}_{\text{ce}}$ to enhance the model's focus on rare species. These loss terms are defined as follows:

$$\mathcal{L}^{rs}_{\text{cons.}} = \left\| f\left(E^{\text{aug1}}_{\text{fused}}\right) - f\left(E^{\text{aug2}}_{\text{fused}}\right) \right\|^2, \quad (15)$$

$$\mathcal{L}^{rs}_{\text{ce}} = -\sum_{RS} \hat{y}^{rs}_{\text{pseudo}} \log(\hat{y}^{(rs)}), \quad (16)$$

where $f(\cdot)$ represents the Wide ResNet28-8 model, $E^{\text{aug1}}_{\text{fused}}$ and $E^{\text{aug2}}_{\text{fused}}$ are two augmented versions of the fused embedding, $\hat{y}^{(rs)}_{\text{pseudo}}$ is the pseudo-label for the rare species class rs, and $\hat{y}^{(rs)}$ is the Wide ResNet's prediction for the rare species class rs based on the fused embedding.

Since the consistency loss $\mathcal{L}^{rs}_{\text{cons.}}$ is highly influenced by the quality of pseudo-labels (Zhang and Sabuncu 2018), we use parameters β to adjust the weight of each loss. When the predicted pseudo-label probabilities fall below a threshold (Chen et al. 2023b), β is increased during backpropagation, emphasizing the consistency loss for rare species. Conversely, when pseudo-label confidence is high, the model prioritizes the Cross-Entropy Loss.

Experiments

Experimental Setup

Datasets. We evaluate our 4S-Classifier on two wildlife datasets: iNaturalist 2017/18 (Zhang et al. 2023) and Wildlife Insights (Van Horn et al. 2018). The iNaturalist 2017/18 dataset, with over 675,000 images across 5,089 species, is widely used for wildlife detection and is notable for capturing the significant class imbalance observed in the wild. The Wildlife Insights dataset focuses on endangered species with a high richness of rare species but includes only partially labeled data. In our experiments, we use 20% labeled data and 80% unlabeled data from iNaturalist, as well as 10% labeled data and 90% unlabeled data from Wildlife Insights, making it ideal for testing semi-supervised frameworks.

Implementation Details. We conduct experiments to optimize our framework, focusing particularly on the proposed RSBank and RSEmbed modules, in a semi-supervised manner similar to FullMatch. All input images are resized to 224x224 pixels during training, with a batch size of 64. Both modules use 512-dimensional embeddings. Specifically, the RSBank module has an embedding update rate of 0.5. The RSEmbed modules uses a learning rate of 0.01, a pseudo-label update rate of 0.3, and a dropout rate of 0.5. The consistency loss weight β is dynamically adjusted based on pseudo-label confidence, with a threshold of 0.75 for low-confidence pseudo-labels. In addition, L2 regularization is applied to the embeddings with a strength of 0.0001. Across all experiments, our framework utilizes a cosine learning rate decay schedule and an SGD optimizer with a momentum of 0.9 and an initial learning rate of 0.03 (Chen et al. 2023b). The whole implementation is built on TorchSSL (Zhang et al. 2021).

Results and Analysis

Table 1 compares the Top-1 accuracy of our framework against other state-of-the-art works on the iNaturalist and Wildlife Insights datasets, including one fully-supervised framework (Bakana, Zhang, and Twala 2024) and multiple semi-supervised frameworks (Duan et al. 2023; Girshick et al. 2014; Zhao et al. 2022; Fan et al. 2022; Zheng et al. 2022a; Li, Xiong, and Hoi 2021; Wang et al. 2022b). On the iNaturalist dataset, we experiment with labeled data proportions of 5%, 10%, and 25%. The 4S-Classifier achieves the highest classification accuracy under all three proportions. Compared to the baseline FullMatch, the 4S-Classifier delivers accuracy improvements of 1.53% on the 5% labeled

	iNaturalist			Wildlife Insights		
Label Amount	5%	10%	25%	5%	15%	25%
FlexMatch	59.76±1.35	62.07±1.20	83.94±1.65	53.61±0.21	67.50±0.56	83.87±1.25
AdaMatch	55.61±0.05	58.93±1.24	84.43±0.19	57.25±1.33	67.20±0.35	82.57±0.46
DC-SSL	56.35±0.22	62.11±0.27	79.61 ±0.08	55.89±1.18	68.07±0.01	75.55±0.12
CoSSL	-	-	84.15±1.50	55.24±0.59	63.30±0.30	79.67±0.25
Debiased Learning	59.12±1.32	61.98±1.15	82.75±1.20	57.93±0.53	68.05±1.35	83.42±1.46
SimMatch	50.97±2.25	62.26±1.40	81.24±0.62	50.17±2.33	68.21±1.74	83.46±1.38
FreeMatch	57.58±1.24	62.33±1.75	83.62±1.18	59.98±1.11	68.09±1.23	85.58±2.56
SoftMatch	59.35±2.17	62.39±1.98	84.07±1.21	57.45±3.86	68.22±0.32	84.59±2.37
Fixmatch+PGR	60.63±1.15	62.19±1.45	84.12±2.34	60.77±1.54	70.65±1.76	86.53±2.38
DST (FixMatch)	61.32±1.25	63.27±3.27	84.02±1.52	**62.19**±1.86	70.20±2.34	86.23±1.25
FixMatch	60.54±2.28	62.67±1.81	83.79±1.32	61.08±1.83	69.20±2.45	85.57±1.21
WildARe-YOLO†	-	91.15±1.50	-	-	89.61±0.55	-
FullMatch	62.03±1.55	63.22±0.82	85.18±0.01	60.86±0.62	72.39±1.23	86.17±1.35
4S-Classifier (ours)	**63.56**±1.23	**65.16**±0.05	**88.37**±0.23	61.23±1.42	**73.82**±1.38	**91.56**±2.05

Table 1: The Top-1 accuracy (%) of the 4S-Classifier and other models on the iNaturalist and Wildlife Insights datasets with labeled data amounts of 5%, 10%, and 25%. † Fully supervised learning model (Bakana, Zhang, and Twala 2024).

dataset, 1.94% on the 10% labeled dataset, and 3.19% on the 25% labeled dataset. These results demonstrate that our proposed framework can significantly outperform FullMatch. Moreover, Fig. 6 illustrates the changes in accuracy over 400 training epochs for both the 4S-Classifier and FullMatch, exhibiting the faster and more stable convergence of the 4S-Classifier compared to FullMatch.

On the Wildlife Insights dataset, where the labeled data and classes are equally sparse, and the distribution of endangered species is completely random, our 4S-Classifier still achieves important improvements. Specifically, the 4S-Classifiersets a new state-of-the-art accuracy of 91.56% with only 25% labeled data, firmly outperforming FullMatch by 5.39%. Note that since the Wildlife Insights dataset contains unlabeled data, we trained the fully supervised learning models only using the labeled portion of this dataset.

Table 2 presents the cross-dataset performance of FullMatch, WildARe-YOLO, and 4S-Classifier, trained with 25% labeled data on the iNaturalist dataset and tested on the Wildlife Insights dataset. Average Precision (AP) and Area Under the Curve (AUC), two commonly used metrics for species recognition, are reported for nine selected endangered species. The experimental results in Table 2 highlight the 4S-Classifier's high detection accuracy across all nine species, including the relatively rare Javan Rhino. Notably, the AP and AUC for rare classes such as Snow Leopard, Mountain Gorilla, Red Panda, CN Alligator, and California Condor are higher than those achieved by WildARe-YOLO (Bakana, Zhang, and Twala 2024). Additionally, the 4S-Classifier's classification performance for other classes also outperforms FullMatch.

Ablation Study

We conducted two sets of ablation experiments to validate the effectiveness of the proposed RSEmbed and RSBank modules. As shown in Table 3, the 4S-Classifier is clearly

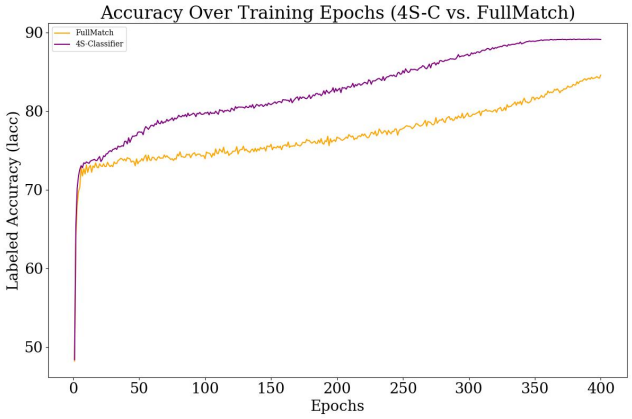

Figure 6: Accuracy over training epochs of the 4S-Classifier and FullMatch on iNaturalist dataset. The convergence of the 4S-Classifier is faster and smoother compared to FullMatch.

advantageous in rare species detection through the integration of these two modules.

Impact of the RSEmbed module: The use of this module alone reaches an AP of 88.09%, highlighting its effectiveness in refining embeddings and thus improving rare species detection. This finding specifically confirms RSEmbed's ability to capture detailed representations of rare species, particularly for rare classes in unbalanced data.

Complementary effect of the RSBank and RSEmbed modules: The combination of these two modules yields the highest AP of 91.56%, suggesting a strong synergy between them. While the RSEmbed module refines embeddings, the RSBank module ensures their appropriate distribution, thereby optimizing the framework's ability to accu-

Model	Class	Objects	AP	AUC
FullMatch	Zebra	1,902	78.12	76.9
	Javan Rhino	507	82.34	77.46
	Snow Leopard	1,285	80.45	80.52
	Mount. Gorilla	1,043	77.89	75.23
	Red Panda	621	79.56	77.64
	Sumatran Tiger	1,819	84.23	79.9
	CN.Alligator	819	81.67	82.34
	Black Rhino	833	75.34	78.25
	California Condor	1,799	83.72	84.31
WildARe-YOLO	Zebra	2358	86.1	89.07
	Javan Rhino	963	90.22	93.6
	Snow Leopard	1,741	85.97	94.56
	Mount. Gorilla	1,499	86.0	89.78
	Red Panda	1,077	94.01	90.76
	Sumatran Tiger	2,275	88.21	86.89
	CN.Alligator	1,275	87.54	94.2
	Black Rhino	1,289	88.92	89.63
	California Condor	1967	88.34	94.86
4S-Classifier	Zebra	2292	83.95	83.84
	Javan Rhino	897	89.59	88.58
	Snow Leopard	1675	**86.22**	89.12
	Mount. Gorilla	1433	**87.93**	89.47
	Red Panda	1019	**88.46**	84.12
	Sumatran Tiger	2209	82.93	82.1
	CN.Alligator	1187	**89.13**	89.79
	Black Rhino	1001	84.81	87.27
	California Condor	1901	**88.62**	87.11

Table 2: Cross-Dataset performance comparison: FullMatch vs. 4S-Classifier vs. WildARe-YOLO. *Objects* refers to the number of samples of the rare species detected by the model.

Model	RSBank	RSEmbed	AP
4S-Classifier	✗	✗	82.67
	✓	-	84.31
	-	✓	88.09
	✓	✓	**91.56**
FullMatch	✗	✗	86.17
	✓	-	86.21
	-	✓	87.74
	✓	✓	**89.16**

Table 3: Ablation experiments to evaluate the individual contributions of the RSBank and RSEmbed modules to model performance on Wildlife Insights dataset.

rately differentiate species.

FullMatch vs. 4S-Classifier backbones: FullMatch, when enhanced with the RSBank and RSEmbed modules, reaches an AP of 89.16%. However, the 4S-Classifier, integrating both modules, attains a higher AP of 91.56%. This indicates that the 4S-Classifier's design is better suited for semi-supervised learning scenarios with unbalanced labels, and is more effective in harnessing the RSBank and RSEmbed modules.

Attention and gate mechanisms: Table 4 summarizes the ablation results for the attention and gate mechanisms in the RSEmbed module on Wildlife Insights dataset. For attention, focusing on rare species improves accuracy, but the *Double Attention* configuration, which balances the focus across rare and common species, achieves the best results. Similarly, for gate mechanisms, focusing on rare species enhances performance, while the *Double Gates* configuration, which incorporates signals from both rare and common species, performs best.

Removing both mechanisms significantly reduces accuracy, while using either alone provides moderate improvement. Combining both mechanisms delivers the highest accuracy, demonstrating the effectiveness of the RSEmbed module in addressing species imbalance and pseudo-label bias.

Model	AP	AUC
RSE Attention (E_{rare})	86.32	84.75
RSE Attention (E_{j})	85.14	83.65
Double Attention	91.56	89.42
w/o Attention	82.71	80.22
Gate (E_{rare})	87.63	86.11
Gate (E_{j})	88.27	87.33
Double Gates	91.56	89.42
w/o Gates	84.5	82.39
w/o Attention + w/ Gates	83.26	83.59
w/o Gates + w/ Attention	85.43	84.11

Table 4: Ablation experiments for the attention mechanism and gate mechanism in the RSEmbed module. *Double Attention* refers to using both attention mechanisms simultaneously, while *Double Gates* refers to using both gate mechanisms simultaneously.

Conclusion

This work presented a new semi-supervised learning framework, 4S-Classifier, to address major challenges in the task of wildlife species visual classification. These challenges include sparse image data for endangered or rare species, incomplete labeling, and the high costs of annotation in real-world scenarios. To mitigate these challenges, the 4S-Classifier introduces two innovative modules, RSEmbed and RSBank. The RSEmbed module is an augmentation method that fuses attention-enhanced embedding vectors for labeled and unlabeled data, while the RSBank module effectively captures rare species' distribution patterns and stores them to aid in embedding integration and classification. Extensive experiments on the iNaturalist and Wildlife Insights datasets demonstrate the 4S-Classifier's exceptional performance, significantly improving the classification accuracy of rare species. This work is of practical significance for identifying endangered species and supporting their future protection.

References

Ahumada, J. A.; Fegraus, E.; Birch, T.; Flores, N.; Kays, R.; O'Brien, T. G.; Palmer, J.; Schuttler, S.; Zhao, J. Y.; Jetz, W.; et al. 2020. Wildlife insights: A platform to maximize the potential of camera trap and other passive sensor wildlife data for the planet. *Environmental Conservation*, 47(1): 1–6.

Bakana, S. R.; Zhang, Y.; and Twala, B. 2024. WildARe-YOLO: A lightweight and efficient wild animal recognition model. *Ecological Informatics*, 80: 102541.

Berthelot, D.; Roelofs, R.; Sohn, K.; Carlini, N.; and Kurakin, A. 2021. Adamatch: A unified approach to semi-supervised learning and domain adaptation. *arXiv preprint arXiv:2106.04732*.

Cai, Z.; Ravichandran, A.; Maji, S.; Fowlkes, C.; Tu, Z.; and Soatto, S. 2021. Exponential moving average normalization for self-supervised and semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 194–203.

Chabot, D.; Stapleton, S.; and Francis, C. M. 2022. Using Web images to train a deep neural network to detect sparsely distributed wildlife in large volumes of remotely sensed imagery: A case study of polar bears on sea ice. *Ecological Informatics*, 68: 101547.

Chen, H.; Tao, R.; Fan, Y.; Wang, Y.; Wang, J.; Schiele, B.; Xie, X.; Raj, B.; and Savvides, M. 2023a. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. *arXiv preprint arXiv:2301.10921*.

Chen, Y.; Tan, X.; Zhao, B.; Chen, Z.; Song, R.; Liang, J.; and Lu, X. 2023b. Boosting Semi-Supervised Learning by Exploiting All Unlabeled Data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7548–7557.

Christiansen, P.; Steen, K. A.; Jørgensen, R. N.; and Karstoft, H. 2014. Automated detection and recognition of wildlife using thermal cameras. *Sensors*, 14(8): 13778–13793.

Duan, Y.; Zhao, Z.; Qi, L.; Zhou, L.; Wang, L.; and Shi, Y. 2023. Towards semi-supervised learning with non-random missing labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 16121–16131.

Fan, Y.; Dai, D.; Kukleva, A.; and Schiele, B. 2022. Cossl: Co-learning of representation and classifier for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14574–14584.

Fini, E.; Astolfi, P.; Alahari, K.; Alameda-Pineda, X.; Mairal, J.; Nabi, M.; and Ricci, E. 2023. Semi-Supervised Learning Made Simple with Self-Supervised Clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3187–3197.

Gao, J.; Li, P.; Chen, Z.; and Zhang, J. 2020. A survey on deep learning for multimodal data fusion. *Neural Computation*, 32(5): 829–864.

Girshick, R.; Donahue, J.; Darrell, T.; and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587.

Guerrero, M. J.; Bedoya, C. L.; López, J. D.; Daza, J. M.; and Isaza, C. 2023. Acoustic animal identification using unsupervised learning. *Methods in Ecology and Evolution*, 14(6): 1500–1514.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132–7141.

Kellenberger, B.; Marcos, D.; and Tuia, D. 2018. Detecting mammals in UAV images: Best practices to address a substantially imbalanced dataset with deep learning. *Remote sensing of environment*, 216: 139–153.

Kim, J.; and Scott, C. D. 2012. Robust kernel density estimation. *The Journal of Machine Learning Research*, 13(1): 2529–2565.

Li, J.; Xiong, C.; and Hoi, S. C. 2021. Comatch: Semi-supervised learning with contrastive graph regularization. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9475–9484.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; and Berg, A. C. 2016. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, 21–37. Springer.

Liu, Y.; Che, S.; Ai, L.; Song, C.; Zhang, Z.; Zhou, Y.; Yang, X.; and Xian, C. 2024. Camouflage detection: Optimization-based computer vision for Alligator sinensis with low detectability in complex wild environments. *Ecological Informatics*, 83: 102802.

Lu, X.; and Lu, X. 2023. An efficient network for multi-scale and overlapped wildlife detection. *Signal, Image and Video Processing*, 17(2): 343–351.

Manohar, N.; Kumar, Y. S.; and Kumar, G. H. 2016. Supervised and unsupervised learning in animal classification. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 156–161. IEEE.

Peng, J.; Wang, D.; Liao, X.; Shao, Q.; Sun, Z.; Yue, H.; and Ye, H. 2020. Wild animal survey using UAS imagery and deep learning: modified Faster R-CNN for kiang detection in Tibetan Plateau. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169: 364–376.

Redmon, J. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Reynolds, D. A.; et al. 2009. Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663).

Seraj, F.; Ayele, E.; and Meratnia, N. 2019. Unsupervised learning of wildlife behaviour for activity-driven opportunistic beacon networks. In *2019 13th International Conference on Sensing Technology (ICST)*, 1–6. IEEE.

Van Horn, G.; Mac Aodha, O.; Song, Y.; Cui, Y.; Sun, C.; Shepard, A.; Adam, H.; Perona, P.; and Belongie, S. 2018.

The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8769–8778.

Wang, X.; Wu, Z.; Lian, L.; and Yu, S. X. 2022a. Debiased learning from naturally imbalanced pseudo-labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14647–14657.

Wang, Y.; Chen, H.; Heng, Q.; Hou, W.; Fan, Y.; Wu, Z.; Wang, J.; Savvides, M.; Shinozaki, T.; Raj, B.; et al. 2022b. Freematch: Self-adaptive thresholding for semi-supervised learning. *arXiv preprint arXiv:2205.07246*.

Wu, Z.; Shen, C.; and Van Den Hengel, A. 2019. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern recognition*, 90: 119–133.

Yang, W.; Liu, T.; Jiang, P.; Qi, A.; Deng, L.; Liu, Z.; and He, Y. 2023. A forest wildlife detection algorithm based on improved YOLOv5s. *Animals*, 13(19): 3134.

Zhang, B.; Wang, Y.; Hou, W.; Wu, H.; Wang, J.; Okumura, M.; and Shinozaki, T. 2021. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34: 18408–18419.

Zhang, Y.; and Cai, Z. 2023. CE-RetinaNet: A channel enhancement method for infrared wildlife detection in UAV images. *IEEE Transactions on Geoscience and Remote Sensing*.

Zhang, Y.; Kang, B.; Hooi, B.; Yan, S.; and Feng, J. 2023. Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9): 10795–10816.

Zhang, Z.; and Sabuncu, M. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31.

Zhao, Z.; Zhou, L.; Duan, Y.; Wang, L.; Qi, L.; and Shi, Y. 2022. Dc-ssl: Addressing mismatched class distribution in semi-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9757–9765.

Zheng, M.; You, S.; Huang, L.; Wang, F.; Qian, C.; and Xu, C. 2022a. Simmatch: Semi-supervised learning with similarity matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14471–14481.

Zheng, Z.; Zhao, Y.; Li, A.; and Yu, Q. 2022b. Wild terrestrial animal re-identification based on an improved locally aware transformer with a cross-attention mechanism. *Animals*, 12(24): 3503.