Multibranch Co-training to Mine Venomous Feature Representation: A Solution to SnakeCLEF2024

Peng Wang[1,†], Yangyang Li[1,†], Bao-Feng Tan[1,†], Yi-Chao Zhou[1], Yong Li[1] and Xiu-Shen Wei[2,*]

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China
[2]School of Computer Science and Engineering, and Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications, Southeast University, Nanjing, China

Abstract

The SnakeCLEF2024 competition aims to develop an advanced algorithm capable of automatically identifying snake species from images. Accurate identification of snake species in snakebite cases can assist doctors in administering targeted antivenom, which is crucial for effective treatment. In this paper, we propose a multibranch co-training strategy based on Convolutional Neural Networks (CNNs) as the solution. During the training phase, our method consists of three branches which can be trained end-to-end. The first branch is used for the classification of all species and generates a gating coefficient. The second branch specifically focuses on venomous snakes, while the third branch concentrates on harmless species. The gating coefficient determines which of these branches will be utilized. During the inference phase, we only retain the first branch. Our solution significantly enhances the model's ability to distinguish between venomous and harmless snake species and achieve an accuracy of 69.83% and scored 83.57% on the track1 on the private leaderboard, which is the 1st place among all participants. The code is available at https://huggingface.co/pengdadaaa/SnakeCLEF2024.

Keywords

Snake Species Identification, Fine-grained image recognition, Long-tailed, SnakeCLEF

1. Introduction

The SnakeCLEF2024 [1] competition, co-hosted as part of the LifeCLEF2024 [2] within the CLEF2024 conference and the FGVC11 workshop in conjunction with the CVPR2024 conference, aims to advance the development of robust algorithms for snake species identification from images. Each year, snakebites result in an annual mortality of between 81,000 and 138,000 people, and an additional 400,000 victims suffer from incurable physical and psychological disabilities [3, 4]. Accurate identification of snake species is crucial for administering the correct antivenom, which can significantly reduce the number of fatalities and disabilities caused by snakebites. Furthermore, snake species identification can improve the protection of harmful snakes, reducing the number of snakes killed out of fear. This objective is profoundly significant for biodiversity conservation and is a crucial aspect of human health preservation.

Compared to SnakeCLEF2023 [5], the test data of SnakeCLEF2024 contains only image information without metadata, making it more practical but also more challenging for accurate recognition. Unlike [6], we focus on enhancing the model's capacity to mine distinguishable features for recognizing venomous and harmless species and provide an efficient solution. Specifically, we use the first three stages of the CNN as the basic feature extractor. The fourth stage and a fully connected layer are considered as experts responsible for making predictions, constructing a model similar to a mixture of experts. Experimental results show that through end-to-end co-training, our method effectively improves model performance and achieves significant improvements in multiple metrics.

This paper follows the structure as outlined: We first describe the related work in this field in Section 2. Then, in Section 3, we analyze the competition data and challenges in detail. In Section 4, we describe

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France
*Corresponding author.
†Under the supervision of Xiu-Shen Wei.
wangpeng@njust.edu.cn (P. Wang); lyylyyi599@njust.edu.cn (Y. Li); tanbf@njust.edu.cn (B. Tan); wingegg_313@126.com (Y. Zhou); yong.li@njust.edu.cn (Y. Li); weixs.gm@gmail.com (X. Wei)

our method. Section 5 provides detailed experimental details and results. Finally, we summarize this work briefly.

2. Related Work

The problem of automatic snake recognition has been studied for a long time. Early research was based on manually designed rules to propose features beneficial for snake classification, intended for use by computer scientists and herpetologists [7]. A. Amir et al. [8] was the first to use texture-based features along with various machine learning algorithms for automatic snake recognition. With the development of deep learning, CNN networks have made tremendous progress in image classification tasks [9, 10, 11]. I. S. Abdurraza et al. [12] successfully developed a CNN-based automatic snake classification algorithm, achieving high accuracy. During the same period, many other snake recognition algorithms based on deep learning were also proposed.

The winning method of SnakeCLEF 2021 [13, 14] combined object detection with an EfficientDet-D1 [15] model, and an EfficientNet-B0 classifier as well as likelihood weighting to fuse image and location information. The best model reached a macro-averaging F1 score of 90.30%. In SnakeCLEF 2022 [16], one team [17] used YOLOv5 [18] to first detect the specific location of the snake in the image, and then used a CNN network for classification, while also utilizing metadata to statistically determine the regional distribution of snake species. They also employed various strategies such as test-time augmentation and model ensembling. In SnakeCLEF 2023 [19], the winning team [6] used CLIP [20] to process metadata and leveraged intermediate layer features from CNNs to aid in the final classification decision. Additionally, they designed a post-processing strategy to determine whether the snake was venomous. In previous competitions, some teams also used attention-based models such as MetaFormer [21], ViT [22], and VOLO [23].

3. Competition Description

Understanding datasets and metrics is essential for participating in this competition. Within this section, we aim to introduce our comprehension of the datasets and provide an overview of the evaluation metrics employed by the competition organizers.

3.1. Dataset

The organizers provide a dataset, consisting of 103,404 recorded snake observations, supplemented by 182,261 high-resolution images. These observations encompass a diverse range of 1,784 distinct snake species.

Fine-gained Image This dataset presents a challenging fine-grained image classification task, as illustrated in Figure 1. Our objective is to accurately identify different species. While these species share many visual similarities, they exhibit only subtle differences in fine-grained features. Accurately distinguishing these species demands models capable of identifying subtle yet significant differences.

Long-tailed Distribution It is worth noting that the provided training dataset is in a heavily long-tailed distribution, as shown in Figure 2. In this distribution, the most frequently encountered species are represented by 1,891 images. However, the least frequently encountered species is captured by a mere 3 images, highlighting its exceptional rarity within the dataset. The number of images for venomous snakes is 32,379, while the number of images for harmless snakes is 135,348, which also represents an imbalanced distribution.

(a) Ahaetulla_malabarica (b) Ahaetulla_nasuta (c) Ahaetulla_oxyrhynca (d) Ahaetulla_prasina

Figure 1: Examples of images belonging to different species. Different species of snakes have very similar appearances, making the classification task more challenging.

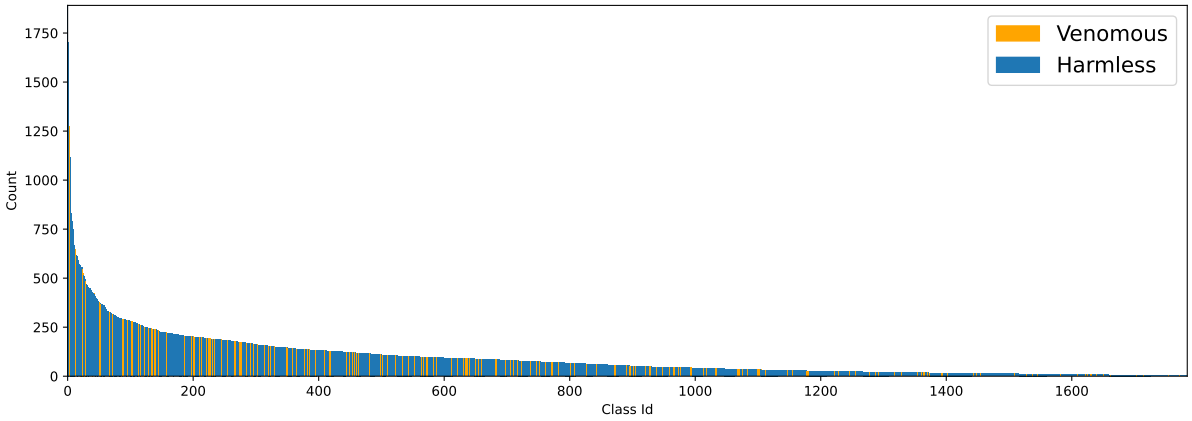

Figure 2: Long-tailed distribution of the SnakeCLEF2024 training dataset. The blue color represents venomous species. The orange color represents harmless species.

3.2. Evaluation Metric

To motivate research in recognition scenarios with uneven costs for different errors, such as mistaking a venomous snake for a harmless one, this competition will again go beyond the 0-1 loss common in classification. This year's competition incorporates a evaluation metric, denoted as "track1" on the leaderboard. This metric combines the F1-Score with an assessment of the confusion errors related to venomous species. It is calculated as a weighted average, incorporating both the macro F1-score and the weighted accuracy of various types of confusions:

$$M = \frac{w_1 F_1 + w_2 \left(100 - P_1\right) + w_3 \left(100 - P_2\right) + w_4 \left(100 - P_3\right) + w_5 \left(100 - P_4\right)}{\sum_i^5 w_i}, \tag{1}$$

where $w_1 = 1.0, w_2 = 1.0, w_3 = 2.0, w_4 = 5.0, w_5 = 2.0$ are the weights of individual terms. The metric incorporates several percentages, namely F_1 representing the macro F1-score, P_1 denoting the percentage of harmless species misclassified as another harmless species, P_2 indicating the percentage of harmless species misclassified as a venomous species, P_3 reflecting the percentage of venomous species misclassified as another harmless species, and P_4 representing the percentage of venomous species misclassified as another venomous species.

3.3. Challenges of the Competition

Past iterations of this competition have witnessed remarkable accomplishments by deep learning models [13, 14, 24, 16, 25, 26, 27]. To achieve a better solution, we summarize the competition challenges this year based on the above analysis:

- Fine-grained image recognition: The field of fine-grained image analysis [28, 29, 30] has long posed a challenging problem within the FGVC workshop, meriting further investigation and study. This year's competition lacks available metadata for the test images, increasing the requirements for understanding subtle image features and making the task more challenging.
- Long-tailed distribution: This dataset has a heavily long-tail distribution. The imbalance of data in the tail class leads to insufficient generalization ability of models in these categories, making it difficult for models to effectively learn and recognize tail class instances.
- Identification of venomous and harmless species: The distinction between venomous and harmless snake species is meaningful, as venomous snake bites lead to a large number of deaths each year.
- Limited computational resources: We need to process approximately 10,000 images within one hour on a server with an Nvidia T4, small 4vCPU, 15GB RAM, and 16GB VRAM.

4. Method

In this section, we provide a detailed description of our method.

4.1. Data Preprocessing

Data preprocessing plays a crucial role in machine learning, as it not only influences the final performance but also affects the feasibility of problem resolution. Upon obtaining the dataset provided by the competition organizers, we encountered several issues. For instance, certain images listed in the metadata CSV file were nonexistent in the corresponding image folders. To address this, we generated a new CSV file by eliminating the affected rows from the original file.

Data augmentation plays a vital role in image classification tasks by expanding the scale and diversity of training data through a series of algorithms and techniques, effectively addressing the issue of overfitting. By applying a variety of image transformation operations, data augmentation significantly enhances the diversity of datasets, enabling models to learn more robust and comprehensive feature representations. In our method, we leverage fundamental image augmentation methods from Albumentations [31], including RandomResizedCrop, Transpose, HorizontalFlip, VerticalFlip, ShiftScaleRotate, RandomBrightnessContrast, PiecewiseAffine, HueSaturationValue, OpticalDistortion, ElasticTransform, Cutout, and GridDistortion. Furthermore, we incorporate data mixing augmentation techniques such as CutMix [32] and TokenMix [33] during the competition. These methods provide strong regularization to models by softening both images and labels, thus preventing model overfitting on the training dataset. During the inference stage, we also employ Test-Time Augmentation (TTA) by applying various augmentation methods to each input image, generating multiple augmented versions. These augmented images are then individually processed by the model to obtain multiple sets of predictions. Finally, these predictions are averaged to produce the final prediction.

4.2. Model

Throughout the competition, we explored various models, incorporating both classical and state-of-the-art architectures such as Convolutional Neural Networks and Vision Transformers. The models employed during the competition included ConvNeXt [34], ConvNeXt-v2 [35], and EVA-02 [36]. The implementation of these models was facilitated by the use of the timm library [37]. Considering the limitations on model parameters and the need for robust model representation capabilities, we selected ConvNeXt [34] or ConvNeXt-v2 [35] as the backbone architectures for our final method.

However, relying solely on the visual backbone and training it with the classical classification strategy is insufficient for effectively addressing the task at hand. To make the model focus on distinguishable features that can differentiate between venomous and harmless species, we propose a multibranch co-training method as our final submission. The model architecture is illustrated in Figure 3. Inspired by [38, 39, 40], our method primarily involves three branches, which are processed sequentially from

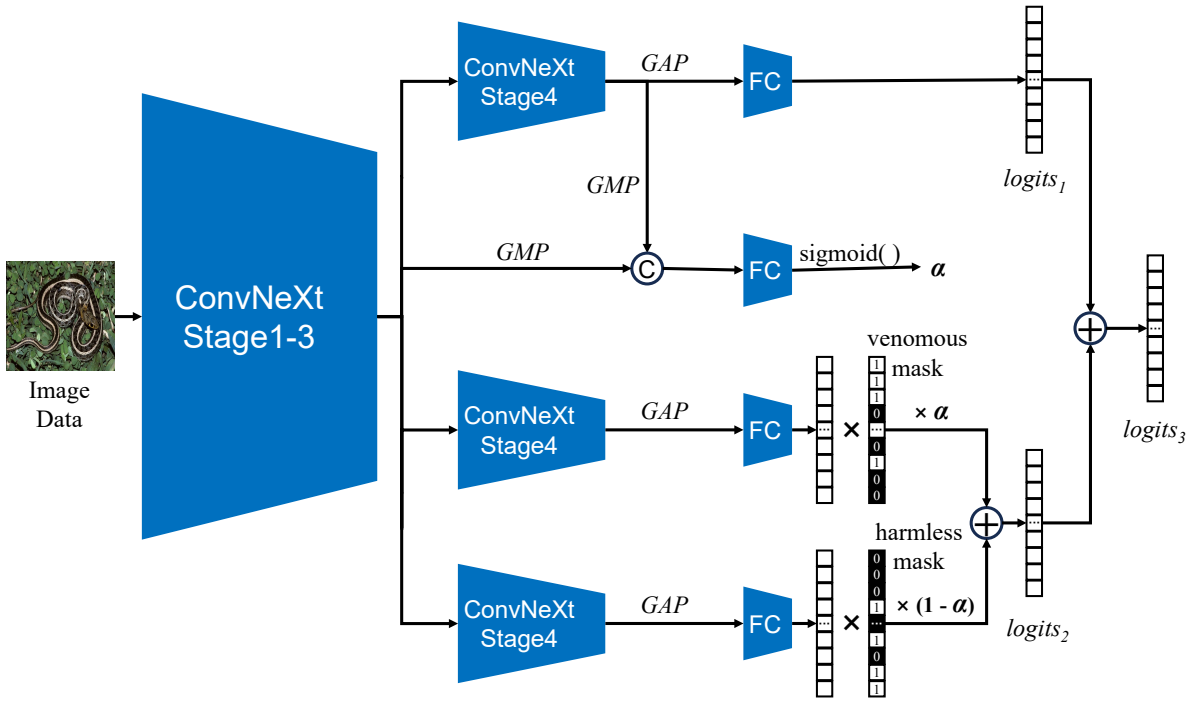

Figure 3: The architecture of our model take ConvNeXt [34] as the backbone, and consist of three branch which share weights with the first three stages. "GAP" is short for global average pooling and "GMP" is short for global max pooling.

top to bottom as shown in Figure 3. Each branch uses the same residual network structure and shares weights for the first three stages.

Given an image I, we obtain feature maps from the first three stages and from the fourth stage, denoted as X_3 and X_4 respectively, after processing it through the first branch. The feature map from the fourth stage(X_4) undergoes global average pooling and is passed through a classification head to obtain $logits_1$. Additionally, we concatenate the features obtained from global max pooling of X_3 and X_4, and pass them through a fully connected layer and sigmoid function to obtain α, which acts as a gating coefficient to select between the 2nd and 3rd branches.In our method, the 1st branch, serving as the primary branch, can identify all snake species and generate the gating coefficient α.

The 2nd branch focuses on venomous species (venomous branch), while the 3rd branch focuses on harmless species (harmless branch). To make these branches concentrate on their respective tasks, we use binary masks generated from the coarse labels (venomous or harmless) to stop the gradient. We combine the outputs of the 2nd and 4rd branches according to the gating coefficient α to obtain $logits_2$, and by summing $logits_1$ and $logits_2$, we can directly derive $logits_3$. All three obtained logits are utilized during the training phase. However, in the inference phase, we select only one of them, which is then passed through the $\mathrm{softmax}(\cdot)$ function to produce the predicted probability for an input image (refer to the Table 3 for specific selection).

4.3. Optimization Procedure

For the classification, the most widely adopted Cross-Entropy(CE) Loss can be written as:

$$L_{ce}(\mathbf{z}) = -\sum_{i=1}^{C} y_i \log(\sigma_i), \quad \text{with } \sigma_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}, \tag{2}$$

where $\mathbf{z} = [z_1, z_2, \ldots, z_C]$ and $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_C]$ are the predicted logits and probabilities of the classifier, respectively. And $y_i \in \{0, 1\}, 1 \leq i \leq C$ is the one-hot ground truth label. However, the classifier trained by the widely applied CE Loss is highly biased on long-tailed datasets, resulting in

much lower accuracy of tail classes than head classes. To tackle this challenge, we extensively explored various techniques implemented in [41, 42, 43]. In our final submission, we incorporated the seesaw loss [44] as a key component. The seesaw loss formulation can be expressed as follows:

$$L_{\text{seesaw}}\left(\mathbf{z}\right) = -\sum_{i=1}^{C} y_i \log\left(\widehat{\sigma}_i\right) \ , \ \text{with } \widehat{\sigma}_i = \frac{e^{z_i}}{\sum_{j\neq i}^{C} \mathcal{S}_{ij} e^{z_j} + e^{z_i}} \, . \tag{3}$$

The hyper-parameters $\mathcal{S}_{ij}$ are carefully set based on the distribution characteristics inherent in the dataset. As shown in Figure 3, for a input image processed by the model, we obtain 3 predicted logits. For each predicted logits, we calculate the loss using either CE loss or Seesaw loss (refer to the Table 3 for specific configurations). The final loss is:

$$L = \frac{L_1(logits_1) + L_2(logits_2) + L_3(logits_3)}{3} \, . \tag{4}$$

In addition to the choice of loss functions, the selection of an optimizer and an appropriate learning rate decay strategy are important in the training of our models. For optimization, we adopt the AdamW optimizer [45]. To enhance convergence speed and overall performance, we implement cosine learning rate decay [46] coupled with warmup techniques during the training process. These strategies collectively facilitate more effective and efficient model convergence.

5. Experiments

In this section, we will introduce our implementation details and main results.

5.1. Experiment Settings

The proposed method was developed using the PyTorch framework [47]. All the pretrained weights used in our experiments come from the timm library [37]. Fine-tuning of these models was conducted across four Nvidia RTX 3090 GPUs. The total number of training epochs was set to 15, with the first epoch dedicated to warm-up. To optimize the model parameters, we utilized the AdamW optimizer [45] in conjunction with a cosine learning rate scheduler [46]. During inference on the test dataset, considering that an observation may consist of multiple images, we average the predicted probabilities from different images of the same ID to obtain the final prediction for each observation.

5.2. Main Results

In this section, we present our primary experiment results. Unless otherwise specified, the model is trained by using Seesaw loss and performs inference in float32. First, we present some basic experimental results. Table 1 and Table 2 respectively show the results of different backbones on the validation set or the public leaderboard. Based on our experimental results and the experiences of past winners, we chose the ConvNeXt series models [34, 35] as our backbone and used a resolution of 512×512.

After selecting the basic backbone, we conducted experiments using the multibranch co-training strategy proposed in the Section 4. We used $logits_1$ and $logits_3$ with $\mathrm{softmax}(\cdot)$ to obtain final prediction results respectively. The experimental results are shown in Table 3. Based on the experimental results, we use $logits_1$ for the final prediction and directly drop the last three branches during the inference stage to reduce computational overhead. Multiple evaluation metrics indicate that our solution can effectively improve model performance.

We ensemble the two best-performing models from Table 3, using the average of the output probabilities from the two models as the final submission result. Considering the limited computational resources, we use half-precision (float16) during inference. The ensembled result achieved first place on both the public leaderboard and the private leaderboard.

Table 1
Results on the validation set without multibranch co-training.

Backbone	Resoulution	Validation Metrics			Comments
		track1	acc	F1	
ConvNeXt-L [34]	512×512	88.53	76.65	62.72	cutmix+tta2
EVA-02-L [36]	336×336	84.62	68.47	54.68	cutmix+tta2
EVA-02-L [36]	336×336	86.15	72.44	59.26	tokenmix+tta2

Table 2
Results on public leaderboard without multibranch co-training.

Backbone	Resoulution	Public Test Metrics			Comments
		track1	acc	F1	
ConvNeXt-v2-B [35]	512×512	79.42	64.31	29.65	cutmix+tta3
ConvNeXt-v2-B [35]	512×512	80.98	66.00	29.45	cutmix+tta3
EfficientNet-B5 [48]	512×512	79.70	65.05	31.84	cutmix+tta3
ConvNeXt-L [34]	512×512	81.06	67.11	34.11	cutmix+tta3
EVA-02-L [36]	336×336	79.65	64.83	32.40	tokenmix+tta3

Table 3
Results on public leaderboard with multibranch co-training.

Backbone	Resoulution	loss			Public Test Metrics						Comments
					$logits_1$			$logits_3$			
		L_1	L_2	L_3	track1	acc	F1	track1	acc	F1	
ConvNeXt-L [34]	512×512	seesaw	CE	CE	84.45	70.64	39.19	80.04	68.06	36.79	cutmix+tta2
ConvNeXt-v2-B [35]	512×512	seesaw	CE	CE	83.17	68.58	37.62	-	-	-	cutmix+tta2
ConvNeXt-v2-B [35]	512×512	seesaw	seesaw	seesaw	84.07	69.98	39.66	80.03	67.99	36.54	cutmix+tta2
ensemble	512×512	-	-	-	85.63	43.66	72.04	-	-	-	float16+tta2

6. Further Discussion

Here, we briefly discuss our method. Our primary motivation is to enhance the model's ability to distinguish between venomous and harmless snake species. Building on this motivation, in addition to classifying all snake species (a total of 1784), our method also indirectly addresses a binary classification problem. The mask in our method serves as the supervisory information for this binary classification task. Specifically, when a venomous image is input, optimizing $logits_2$ using CE loss or Seesaw loss will increase α (for a harmless image, it increases $1-\alpha$). By generating α with GMP and introducing the binary classification supervisory information, we actually apply a constraint to the parameters of the first branch, ensuring that the maximum activation value is directly associated with being venomous or harmless. We guess that this constraint enables the network to effectively mine the feature representation indicative of venomousness, leading to the improvement of performance. We did not explore the method in greater depth to demonstrate its interpretability. However, we believe that further exploration into fully utilizing the binary classification supervisory information is worthwhile.

7. Conclusion

This paper focused on addressing the snake classification problem. In our solution, we used the GMP operation and a fully connected layer to generate the gating coefficient α, which determines the maximum activation value of the feature map associated with whether a snake is venomous and trained three branches end-to-end. Our multibranch co-training strategy has demonstrated significant effectiveness in this competition, achieving a track1 score of 83.57% on the private leaderboard.

References

[1] L. Picek, M. Hruz, A. M. Durso, Overview of SnakeCLEF 2024: Revisiting snake species identification in medically important scenarios, in: Working Notes of CLEF 2024 - Conference and Labs of the Evaluation Forum, 2024.

[2] A. Joly, L. Picek, S. Kahl, H. Goëau, V. Espitalier, C. Botella, B. Deneu, D. Marcos, J. Estopinan, C. Leblanc, T. Larcher, M. Šulc, M. Hrúz, M. Servajean, et al., Overview of lifeclef 2024: Challenges on species distribution prediction and identification, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2024.

[3] J. M. Gutiérrez, J. J. Calvete, A. G. Habib, R. A. Harrison, D. J. Williams, D. A. Warrell, Snakebite envenoming, Nature reviews Disease primers 3 (2017) 1–21.

[4] B. Bracke, M. Bagherifar, L. Bloch, C. M. Friedrich, Joint feature learning of image data with embedded metadata to leverage snake species classification (2023).

[5] A. Joly, C. Botella, L. Picek, S. Kahl, H. Goëau, B. Deneu, D. Marcos, J. Estopinan, C. Leblanc, T. Larcher, et al., Overview of lifeclef 2023: evaluation of ai models for the identification and prediction of birds, plants, snakes and fungi, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2023, pp. 416–439.

[6] F. Hu, P. Wang, Y. Li, C. Duan, Z. Zhu, F. Wang, F. Zhang, Y. Li, X.-S. Wei, Watch out venomous snake species: A solution to snakeclef2023, arXiv preprint arXiv:2307.09748 (2023).

[7] A. P. James, B. Mathews, S. Sugathan, D. K. Raveendran, Discriminative histogram taxonomy features for snake species identification, Human-Centric Computing and Information Sciences 4 (2014) 1–11.

[8] A. Amir, N. A. H. Zahri, N. Yaakob, R. B. Ahmad, Image classification for snake species using machine learning techniques, in: Computational Intelligence in Information Systems: Proceedings of the Computational Intelligence in Information Systems Conference (CIIS 2016), Springer, 2017, pp. 52–59.

[9] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Communications of the ACM 60 (2017) 84–90.

[10] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014).

[11] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[12] I. S. Abdurrazaq, S. Suyanto, D. Q. Utama, Image-based classification of snake species using convolutional neural network, in: 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), IEEE, 2019, pp. 97–102.

[13] R. Chamidullin, M. Šulc, J. Matas, L. Picek, A deep learning method for visual recognition of snake species, Working Notes of CLEF (2021).

[14] L. Picek, A. M. Durso, I. Bolon, R. R. de Castañeda, Overview of snakeclef 2021: Automatic snake species identification with country-level focus, Working Notes of CLEF (2021).

[15] M. Tan, R. Pang, Q. V. Le, Efficientdet: Scalable and efficient object detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 10781–10790.

[16] L. Picek, M. Hrúz, A. M. Durso, I. Bolon, Overview of snakeclef 2022: Automated snake species identification on a global scale, Working Notes of CLEF (2022).

[17] L. Bloch, J.-F. Böckmann, B. Bracke, C. M. Friedrich, Combination of object detection, geospatial data, and feature concatenation for snake species identification., in: CLEF (Working Notes), 2022, pp. 1982–2013.

[18] G. Jocher, Yolov5 by ultralytics, 2020. URL: https://github.com/ultralytics/yolov5. doi:10.5281/zenodo.3908559.

[19] L. Picek, M. Šulc, R. Chamidullin, A. Durso, Overview of snakeclef 2023: snake identification in medically important scenarios, CLEF, 2023.

[20] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in:

International conference on machine learning, PMLR, 2021, pp. 8748–8763.
[21] Q. Diao, Y. Jiang, B. Wen, J. Sun, Z. Yuan, Metaformer: A unified meta framework for fine-grained recognition, arXiv preprint arXiv:2203.02751 (2022).
[22] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, Proceedings of the International Conference on Learning Representations (2021).
[23] L. Yuan, Q. Hou, Z. Jiang, J. Feng, S. Yan, Volo: Vision outlooker for visual recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence (2022).
[24] L. Picek, I. Bolon, A. M. Durso, R. R. de Castañeda, Overview of the snakeclef 2020: Automatic snake species identification challenge, Working Notes of CLEF (2020).
[25] L. Bloch, A. Boketta, C. Keibel, E. Mense, A. Michailutschenko, O. Pelka, J. Rückert, L. Willemeit, C. M. Friedrich, Combination of image and location information for snake species identification using object detection and efficientnets, Working Notes of CLEF (2020).
[26] C. Zou, F. Xu, M. Wang, W. Li, Y. Cheng, Solutions for fine-grained and long-tailed snake species recognition in snakeclef 2022, arXiv preprint arXiv:2207.01216 (2022).
[27] F. Hu, P. Wang, Y. Li, C. Duan, Z. Zhu, Y. Li, X.-S. Wei, A deep learning based solution to fungiclef2023, Aliannejadi et al.[1] (2023) 2051–2059.
[28] X.-S. Wei, Y.-Z. Song, O. Mac Aodha, J. Wu, Y. Peng, J. Tang, J. Yang, S. Belongie, Fine-grained image analysis with deep learning: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (2021) 8927–8948.
[29] X.-S. Wei, Y. Shen, X. Sun, P. Wang, Y. Peng, Attribute-aware deep hashing with self-consistency for large-scale fine-grained image retrieval, IEEE Transactions on Pattern Analysis and Machine Intelligence (2023).
[30] X.-S. Wei, J.-H. Luo, J. Wu, Z.-H. Zhou, Selective convolutional descriptor aggregation for fine-grained image retrieval, IEEE transactions on image processing 26 (2017) 2868–2881.
[31] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, A. A. Kalinin, Albumentations: Fast and flexible image augmentations, Information 11 (2020) 125.
[32] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, Y. Yoo, Cutmix: Regularization strategy to train strong classifiers with localizable features, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6023–6032.
[33] J. Liu, B. Liu, H. Zhou, H. Li, Y. Liu, Tokenmix: Rethinking image mixing for data augmentation in vision transformers, in: European Conference on Computer Vision, Springer, 2022, pp. 455–471.
[34] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, S. Xie, A convnet for the 2020s, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2022, pp. 11976–11986.
[35] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon, S. Xie, ConvNeXt V2: Co-designing and scaling convnets with masked autoencoders, arXiv preprint arXiv:2301.00808 (2023).
[36] Y. Fang, Q. Sun, X. Wang, T. Huang, X. Wang, Y. Cao, Eva-02: A visual representation for neon genesis, arXiv preprint arXiv:2303.11331 (2023).
[37] R. Wightman, Pytorch image models, https://github.com/rwightman/pytorch-image-models, 2019.
[38] B. Zhou, Q. Cui, X.-S. Wei, Z.-M. Chen, Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 9716–9725. doi:10.1109/CVPR42600.2020.00974.
[39] X. Wang, L. Lian, Z. Miao, Z. Liu, S. X. Yu, Long-tailed recognition by routing diverse distribution-aware experts, arXiv preprint arXiv:2010.01809 (2020).
[40] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, G. E. Hinton, Adaptive mixtures of local experts, Neural computation 3 (1991) 79–87.
[41] Y. Zhang, X. Wei, B. Zhou, J. Wu, Bag of tricks for long-tailed visual recognition with deep convolutional neural networks, in: Proceedings of AAAI Conference on Artificial Intelligence, 2021, pp. 3447–3455.
[42] Y.-Y. He, J. Wu, X.-S. Wei, Distilling virtual examples for long-tailed recognition, in: Proceedings

of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 235–244.
[43] X.-S. Wei, S.-L. Xu, H. Chen, L. Xiao, Y. Peng, Prototype-based classifier learning for long-tailed visual recognition, Science China Information Sciences 65 (2022) 160105.
[44] J. Wang, W. Zhang, Y. Zang, Y. Cao, J. Pang, T. Gong, K. Chen, Z. Liu, C. C. Loy, D. Lin, Seesaw loss for long-tailed instance segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2021, pp. 9695–9704.
[45] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, arXiv preprint arXiv:1711.05101 (2017).
[46] I. Loshchilov, F. Hutter, Sgdr: Stochastic gradient descent with warm restarts, arXiv preprint arXiv:1608.03983 (2016).
[47] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems, 2019, pp. 8024–8035.
[48] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: International conference on machine learning, PMLR, 2019, pp. 6105–6114.