IET Computer Vision

WILEY

The Institution of Engineering and Technology

ORIGINAL RESEARCH OPEN ACCESS

Performance of Computer Vision Algorithms for Fine-Grained Classification Using Crowdsourced Insect Images

Rita Pucci[1] | Vincent J. Kalkman[1] | Dan Stowell[1,2]

[1]Naturalis Biodiversity Center, Leiden, Netherlands | [2]Department of Cognitive Science and AI, Tilburg University(NL), Tilburg, Netherlands

Correspondence: Rita Pucci (rita.pucci@naturalis.nl)

Received: 29 February 2024 | **Revised:** 14 January 2025 | **Accepted:** 5 February 2025

Handling Editor: Tilo Burghardt

Funding: EU Horizon Europe projects MAMBO programme under grant agreement No.101060639, TETTRIs Grant Agreement 101081903, and Observation.org.

Keywords: computer vision | image classification | natural scenes

ABSTRACT

With fine-grained classification, we identify unique characteristics to distinguish among classes of the same super-class. We are focusing on species recognition in Insecta as they are critical for biodiversity monitoring and at the base of many ecosystems. With citizen science campaigns, billions of images are collected in the wild. Once these are labelled, experts can use them to create distribution maps. However, the labelling process is time consuming, which is where computer vision comes in. The field of computer vision offers a wide range of algorithms, each with its strengths and weaknesses; how do we identify the algorithm that is in line with our application? To answer this question, we provide a full and detailed evaluation of nine algorithms among deep convolutional networks (CNN), vision transformers (ViT) and locality-based vision transformers (LBVT) on 4 different aspects: classification performance, embedding quality, computational cost and gradient activity. We offer insights that we have not yet had in this domain proving to which extent these algorithms solve the fine-grained tasks in Insecta. We found that ViT performs the best on inference speed and computational cost, whereas LBVT outperforms the others on performance and embedding quality; the CNN provide a trade-off among the metrics.

1 | Introduction

Fine-grained classification task is aimed at distinguishing between different classes that belong to the same super-class. In the field of biodiversity monitoring, species are considered as classes, and the super-class is the order [1]. Taxonomists, who are domain experts, define the species based on their morphology or molecular data. Because the species within an order are closely related and share similar characteristics, such as colours and traits, fine-grained classification tasks in this field are particularly challenging. In this paper, we will consider the taxonomic class of Insecta and in particular the orders of Coleoptera and Odonata focusing on the European species. Coleoptera and Odonata are two of the oldest insect orders and play important roles in our environment. However, they are still not fully understood because of the complexity involved in identifying the different species. Coleoptera can be further divided into four suborders: Archostemata, Myxophaga, Adephaga and Polyphaga, with over 130,000 species present in Europe alone [2]. The order Odonata can be divided into two suborders: Epiprocta and Zygoptera, with over 200 species present in Europe alone [3].

Many species within both orders have similar physical characteristics making it difficult to differentiate between them. Figure 1 shows sample images for the Observation.org dataset

FIGURE 1 | Images in each group belong to the same order, images in each row belong to the same genus and each image represents a unique species from the Coleoptera and Odonata.

[4] from the two classes which exhibit low inter-species variations when the species are from the same genus and large intra-species variations otherwise. Additionally, in Coleoptera, many species are small, and their distinguishing features are often hard to discern. Other factors that make identification challenging include within-species variability arising from differences in life stages, sexes and regional or seasonal variations.

The ability to identify the insects that inhabit ecosystems is one of the main steps to understanding them. Despite its significance, the fine-grained task in biodiversity has posed two key challenges: (1) inter-class variances are often extremely subtle, thus requiring highly discriminative representation for effective classification; (2) as the rarity of a species increases, there are fewer training samples per category, impeding the performance of large-data favoured methods. The conventional identification technique is to cross-validate the image with the regional field guides, online sources and field experts. The majority of the images of these insects are collected by citizen scientists. The use of tools such as Obsidentify [5] helps them in collecting images which are going to be part of the datasets studied by expert taxonomists and available on data platforms such as GBIF [6]. This ongoing work is crucial to improve the knowledge of the current state of biodiversity. Now, billions of images are available for insects and conventional identification techniques cannot stand alone since they are highly time-consuming and unaffordable for the common person. There is increasing interest in the investigation of new deep learning fine-grained methods for biodiversity monitoring. Early and fast identification techniques are crucial and the fast developing of deep learning technologies in computer vision have shown impressive solutions to many real-world problems such as animal identification [7]. At the state of the art, the convolutional neural network (CNN) for computer vision is an algorithm based on an inductive bias of locality and shift-invariance. These two main features make CNN a highly effective deep learning algorithm in image classification. Many variations of the CNN algorithm are available addressing different limitations and proposing new and advanced structures. We have seen an increased interest in the application of transformers for the same tasks to which CNN was historically devoted. Vision transformer (ViT) [8] enables multi-head self-attention to capture long-range dependencies within an image and thus can extract diverse feature patterns for discriminative classification. Unfortunately, ViT is data hungry and the lack of training data may impede its application in fine-grained tasks. As for CNN also with ViT, at the state of the art, we can find different approaches to overcome the limits such as knowledge distillation [9]. Recently, a merged group of algorithms is taking space in this challenge: the locality-based vision transformers (LBVT). As the name suggests, these algorithms are based on vision transformers and then improved with modules of a structure composed of convolutional layers [10]. The obtained algorithms benefit from the inductive bias of locality and shift invariance from the convolutional layers while being able to capture long-range dependencies with self-attention modules. On the other hand, the structure and order of such layers require a deeper understanding.

With their pros and cons, these algorithms are all good candidates for fine-grained tasks for insect images but to what extent? Is the classification performance a sufficient metric to evaluate these algorithms?

In this paper,

- We take into consideration these three groups of deep learning models and we delve into their behaviour when applied to fine-grained tasks in the biodiversity monitoring domain.
- We evaluate each model on four aspects each of which will put the algorithms on different prospects giving us an intuition on what to expect from the model. The aspects are as follows:
  - the classification performance.
  - the embedding quality.
  - the computational cost.
  - the gradient activity.
- We present an overall and a per-species classification performance analysis to observe their behaviour with the long tail distribution of species and in particular with rare species.
- Based on the compressed input representation, each of these models creates an embedding space. We evaluate the quality of the embedding space based on its ability to capture and represent the underlying structure and relationships in the data.
- We consider one critical key point which influences the use of an algorithm which is the computational costs as the resources required, the computational time and the computational complexity.
- To answer the question of which part of the image affects the prediction of the model, we will analyse the gradient activation with GradCam.

For this study, we select algorithms at the state of the art which obtain the best performance or are commonly used in image classification: Inception_v3 [11](`Incpt`), EfficientNet_v2 [12] (`EffNet`), ResNet 50 [13](`ResNet`) for the convolutional

neural network (CNN); T2TViT_14 [14](T2TViT) and ViT trained in knowledge distillation for vision transformers (ViT); ConViT [15](ConViT) and ViTAEv2 [16](ViTAE) for the locality-based vision transformers (LBVT).

For training and validation, we consider datasets collected by citizen science and stored in Observation.org [4]. For the classification performance, we evaluate the models on Artportalen [17] limited to Odonata and Coleoptera from Europe which are collected from different communities of citizen science than training. All four analyses present the results to address the fine-grained task at the species level.

Fine-grained accuracy for biodiversity monitoring is a difficult task which is why our comprehensive evaluation of 9 different computer vision models based on 4 distinct aspects provides a unique contribution to the field. Our evaluation offers a detailed assessment of the competing paradigms for neural network architectures, which is something that has been missing until now. We think that by leveraging our results, we can advance the development of more effective and efficient classification techniques.

2 | Related Work

Insecta are the most diverse taxonomic class of animals on earth but their small size and high diversity have always made them challenging to study. Extensive work has been done to monitor different species in different orders for example Lepidoptera, Coleoptera, Odonata, Orthoptera and Hymenoptera. Monitoring activities determine which insects are at risk, how insect populations fluctuate in natural areas and which management actions are most beneficial to the ecosystems [1, 18–24]. An important resource is the data collected by citizen sciences around the earth but that implies an enormous amount of data to classify. In this context, the application of deep learning algorithms, such as convolutional neural networks (CNN), has seen increased popularity for the ability of automated feature extraction and a high accuracy rate in fine-grained classification. In this paper, we will focus on the vision models since the datasets considered are only visual but promising results are obtained also with the vision–language models [25]. CNN is now popularly used for insect identification and presents a wide range of models applied to classify species in different case studies. Works propose species classification in Lepidoptera order [1, 20, 23] which reach high performance in accuracy. Customised models are proposed for generic species from different orders in the class Insecta [21, 24], also specifically to classify bees in real time [19, 26], Orthoptera for mobile application [27] and Odonata [22]. Even if we have a prolific application of dedicated CNN in fine-grained tasks for insects, we demonstrate in the paper that these models have limitations on the identification of rare species, extraction of compressed input representation and memory efficiency resulting in enormous limitations in practical applications. It is still an open challenge that requires investigation. Moreover, we do not observe equal interest in the application of transformer-based models in this task. An interesting comparison between very simple CNN and transformer-based algorithms for fine-grained tasks among species of different kingdoms identifies the ViT model as outperforming the CNN-based models [28]. A customised transformer model is proposed for insect pest recognition highlighting the need to integrate some of the CNN features into the transformer structure making the model focus more on global coarse-grained information rather than local fine-grained information [29].

Though a vast amount of work has been done in the domain of insect identification, we have not found any published research on a comparative evaluation of algorithms from three of the main groups of deep neural networks for fine-grained identification in biodiversity monitoring. Furthermore, there is no experimentation on the most modern models from computer vision for this task.

2.1 | Deep Learning Models

Among all the groups of models available in computer vision, we select three groups of deep neural network models that are widely used at the state of the art for classification in computer vision for ecology. For each of these, we select the latest algorithms with the best performance on ImageNet1K [30] and some of the most used algorithms for image classification. For a fair comparison, none of these models are specialised for fine-grained tasks.

2.1.1 | Convolutional Neural Networks

In the group of CNN, we consider models that are mainly based on the convolutional layers and fully connected layers [31]. We are interested in models that are competitive with ViT in inference speed, and model size. We choose Inception_v3 [32], EfficientNet_v2_medium [12] and ResNet50 [33]. In the paper, we refer to these models respectively as: Incpt, EffNet and ResNet. Incpt is the result of dealing with the trade-off between performance in classification tasks and computational cost. The structure is thought to scale up in a way that aims to utilise factorised convolutions to reduce the computational bottleneck because of fast and extreme compression of feature maps (convolution with kernels bigger than 3x3). This idea makes the network wider instead of deeper in favour of an efficient computation [32]. EffNet has the structure and connections optimised for speed, based on floating point operations per second (FLOPS), and for parameter efficiency. In particular, EffNet consists of convolutional-based layers [34, 35] designed to better utilise mobile or server accelerators. Both these models represent good competitors to the transformer-based models. ResNet is not a model of the same complexity as the previous two but it is one of the most used models as the backbone or the main model in ecology for classification [13, 36–38]. The CNN models are naturally equipped with intrinsic inductive bias, shift-invariance and hierarchical structure to extract multi-scale features and locality. These are proper advantages in extracting representative features which are used to identify the species in the images. Even if CNN models are commonly used as the backbone of many image classification models at the state of the art, they are not well suited to model long-range dependency

because of their structure being focused on extracting local features from low level to high level progressively. This can affect the performance in the fine-grained tasks: these models are less inclined to identify relations among details of the subject. The details and their relationship are typically the characteristics used by taxonomists to distinguish species.

2.1.2 | Vision Transformers

Models based only on attention [39] are here referred to as fully transformer models. The vision transformer (ViT) [8] is the first fully transformer model applied for image classification demonstrating that transformers are promising for vision tasks. ViT is based on the self-attention mechanism which allows the model to capture global contextual information, enabling it to learn long-range dependencies and relationships between image tokens (patches). The self-attention mechanism weighs the importance of different tokens in the sequence when processing the input data. In this paper, we consider for comparison the recent evolution of ViT, the Token-2-Token ViT 14 [14] (`T2TViT`) which uses a progressive tokenisation module to aggregate neighbouring tokens into one token. In the first layer, a token is a patch of the image, whereas in the intermediate layers, a token is a patch of the feature maps. The model can extract local information reducing the length of the token iteratively. This architecture reduces the data hunger and boosts the performance relative to the vanilla ViT. In addition to `T2TViT`, we also consider the vanilla ViT_small_patch16 [40] model trained via knowledge distillation [9, 41]. *Knowledge distillation* is a model compression method in which a small model (the student) is trained to mimic a pre-trained larger model (the teacher). In this paper, we follow the cross-architecture knowledge distillation [10], and we explore the use of homologous (both CNNs or transformers) or not homologous architectures. We propose this technique in two flavours which consist of distillation from a CNN model, in particular `EffNet`, ResNet like-model and distillation from an LBVT model, `ViTAE`. In both, the ViT model is used as the student model. In the paper, we refer to these models respectively as: `ViTdEfN`, `ViTdVAE`. For these models, the teachers are trained on the training split of the case studies considered in the paper. In particular, the teachers are the `EffNet` and `ViTAE` trained for this paper. Finally, we consider the data-efficient image transformers [40] `DeiT`. In this case, the teacher is a CNN model pre-trained on ImageNet1K.

2.1.3 | Locality-Based Vision Transformer

Finally we consider locality-based vision transformer models (LBVT), which create a collaboration between the convolutional and the transformer layers. With this intent, the CNN structures are included in the vision transformers since the convolution kernels help the model capture the local information. As such, adding locality from CNN improves the data efficiency of vision transformers, resulting in a better performance on a small dataset [42]. In this paper, we consider the convolution-like ViT basic [15], and ViT advanced by exploring inductive bias version2 basic [16]. We refer to these models as `ConViT` and `ViTAE`. The `ConViT` model includes gated positional self-attention (GPSA) that can be initialised as a convolutional layer [43] for capturing the local information at the beginning of the training stage. As such, `ConViT` can utilise the advantages of the soft inductive bias without being limited to CNN. GPSA allows ViT to be the same as CNN to improve the data efficiency on small datasets. `ViTAE` implements inductive bias and the scale-invariance properties into a transformer architecture. To obtain such a result, the algorithm exploits multiple parallel convolutional layers to create the scale invariance and inductive bias, and the transformer layers to create long-range dependencies among the extracted features.

3 | Methods

3.1 | Datasets

The Coleoptera and Odonata datasets used in this paper are available on Observation.org [4], the largest nature platform in the Netherlands for nature observation. Each of these datasets is split into train and test datasets, by using only the test set we wish to evaluate fine-grained classification in highly unbalanced natural datasets, with as few as 2 images per species in many cases. The models are trained only on the data available on the training set. The level of fine-grained tasks considered in this paper is the species level. In the quantitative analysis Section 3.3, we evaluate the models on two datasets: the test split of the Observation.org datasets, and the Odonata and Coleoptera from Artportalen.se [17]—the Swedish nature conservation portal. With the test on the Artportalen.se datasets, we evaluate the generalisation ability of the trained models on a test set taken from an entirely different data source. In both orders, we consider only European species. The images are collected with mobile phones by citizen scientists. It is worth noting that many species of Coleoptera and all species of Odonata present sexual dimorphism and metamorphism because of different stages in life. Figure 2 shows the challenges introduced with the dimorphism. For the same species, the animal can appear in completely different forms. With species, we refer to the taxonomical full name of the species which consists of the order (e.g. Odonata), the infraorder (e.g. Anisoptera), the family (e.g.

FIGURE 2 | Metamorphism of Coleoptera (top row) and Odonata (bottom row) at different life stages. Each image, in the first row refers to the same species (*Harmonia axyridis f. conspicua*), and the species is represented at different stages of the life cycle (from left to right): adult, pupa, larva, egg. The second row presents similar information for a species in Odonata order (*Chalcolestes viridis*).

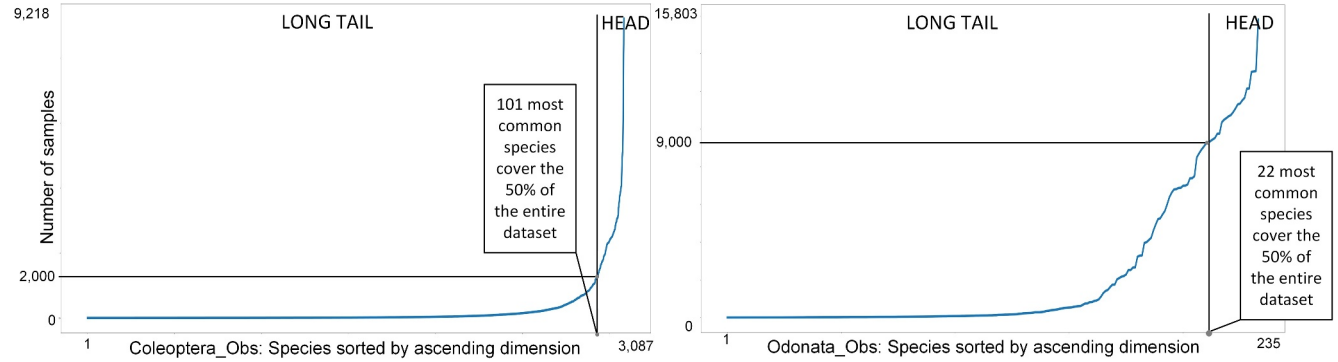


FIGURE 3 | Long tail and head split of the Coleoptera_Obs and Odonata_Obs distributions. The split identifies the most common species which cover 50% of the entire dimension of the datasets.

Aeshnidae), the genus (e.g. *Aeshna*) and the species (e.g. *affinis*) (Figure 3).

3.1.1 | Coleoptera_Obs

The dataset, from Observation.org, consists of 849,296 images over 3087 species. We split the dataset in train and test with the ratio of 80:20 samples per species (674,441:174,855 samples). The dataset is unbalanced with a minimum of 2 samples and a maximum of 9218 samples per species. The dataset consists of species from 122 families, and we have samples from the Polyphaga and Adephaga suborders and 1344 genera with a total of 3087 species. In the dataset, there are samples of 13 morphs: imago, imago brachypterous, imago macropterous, imago micropterous, unknown, gall, exuviae, deviant, larva/nymph, mine, egg, pupa, queen; and three sexes: male, female and unknown.

3.1.2 | Odonata_Obs

The Odonata dataset from Observation.org contains 628,189 images from 235 wild Odonata species. The ratio of train data and test data is roughly 80:20 per species (502,467:125,722 samples). The dataset is unbalanced with a minimum of 2 samples and a maximum of 15,803 samples per species. The dataset consists of species from both Epiprocta, in particular from Anisoptera and Zygoptera infraorders. For the Anisoptera infraorder, we have samples from six families for a total of 153 species, and for the Zygoptera infraorder, we have five families for a total of 82 species, we name this dataset with Odonata_Obs. For this dataset, information is available about morph and sex for further observation in the results. The dataset consists of samples of eight morphs: imago, unknown, fresh imago, exuviae, deviant, larva/nymph, prolarva and egg; and three sexes: male, female and unknown.

3.1.3 | Coleoptera_Art

The Artportalen [44] consists of 3426 species of Coleoptera. Among them, 1574 are used to validate the models and are available in the train split of Coleoptera_Obs. The dataset is unbalanced with a total of 118,464 samples. There are more than 400 species with less than 10 samples each and less than 30 species with more than 500 samples each.

3.1.4 | Odonata_Art

The Artportalen [45] has 73 species from both Anisoptera and Zygoptera infraorders of which 69 species are available in the train split of Observation.org. The dataset consists of 55,680 samples and it is unbalanced with 12 species with less than 100 samples and 20 species with more than 1000 samples.

3.2 | Experimental Configuration

3.2.1 | Preparation of the Models

All the models considered in this paper are pre-trained on ImageNet1k with exceptions for the models which serve as teachers in `ViTdEfN` and `ViTdVAE`. We used the checkpoints available online for the `ViTAE` [16] and `T2TViT` [14]. For the pre-trained initialisation of `EffNet`, `Incpt`, `ResNet`, `ConViT`, `DeiT` and the ViT model in `ViTdEfN` and `ViTdVAE`, we used the checkpoints available on PyTorch image models (timm) [46]. All the models are modified in the head layer which is replaced with a linear layer for the number of classes to be identified. The models are fine-tuned on the dataset described in Sec.3.1. We perform a complete fine-tuning which means all the parameters of the architectures are updated during the training phase. All the experiments are executed on a NVIDIA A40 GPU. We trained the model for a maximum of 310 epochs. We apply early stopping regularisation based on training loss to avoid overfitting. We used a batch size of 32 samples, 5×10^{-4} as the learning rate, 0.065 weight decay, with AdamW [47] as the optimiser with cosine learning rate decay [48] and 10 warm-up epochs. Because of limited resources we analysed only one run.

3.2.2 | Augmentation and Data Preparation

For a fair comparison, we implement the same training scheme for all three models. We set the image size as 224×224 and apply augmentation methods, such as mixup [49] and cutmix [50], for all the models. We do not apply any balancing process

and we evaluate the models on the test set available for the datasets.

3.3 | Classification Metrics

The performance in the classification of the models is evaluated with the entire test set of the dataset and with each species separately. The evaluation of the models on the dataset is presented for the test set of Observation.org and Artportalen datasets. We take into consideration two metrics to evaluate the models: average batch accuracy (avgACC), and F1score. We consider avgACC top-1, hereafter named top-1, the model prediction with the highest probability must be exactly the expected answer; and avgACC top-5, hereafter named top-5, which considers any of our models' top-5 highest probability answers to match with the expected answer. The F1score evaluates the weighted average mean of precision and recall. To evaluate the performance of our models, we analyse their accuracy when applied to each species individually. We calculate the average accuracy achieved on all samples of a given species. Our analysis includes results for both rare and common species and also identifies the number of species that were not recognised by the models. Additionally, we examine the confusion matrices of each model to understand the nature of any misclassifications.

3.4 | Embedding Metrics

Clustering refers to the process of grouping the data in a dataset into different groups, called clusters. The data points in each cluster share similar characteristics or properties. Why is it important in this context? We consider the clusters in the embedding space provided by each model. The embedding space is a relatively low-dimensional space into which it is translated an high-dimensional vectors. For each input, the model compresses the information in an embedding which populates the embedding space. We evaluate if the embedding of each of the samples in the test set is organised in distinctive clusters. As described in Section 3.1, all our data are annotated, and the annotation is made with taxonomic names where both the genus and species are available. We consider the correspondence between clusters obtained with genus and species labels to evaluate the models. We present, in 4.2, a quantitative evaluation of the clusters formed in the embedding space. With this intent, we use the silhouette score which aids in the assessment of clustering performance. For each data point, it considers the average distance of the point to all other data points in the same cluster (intra-cluster distance), and the average distance of the point to all data points in the nearest cluster (inter-cluster distance). The values presented in the section are the overall silhouette score computed as the average value among all the points. We then present a visual representation of the embedding distribution for each model obtained with uniform manifold approximation and projection (UMAP) [51]. UMAP is a dimension-reduction technique that can be used for visualisation. UMAP reduces the data after learning the manifold. It is based on parameters such as *n_neighbours*—the size of the local neighbourhood used in learning the manifold, *min_dist*—how tight the points are packed together and *n_components*—dimensionality of the reduced dimension space (2D/3D). We set these parameters based on our empirical analysis with values: $n\_neighbours = 50$, $min\_dist = 0.5$ and $n\_components = 2$.

3.5 | Computational Cost Metrics

The computational cost is the measure of the amount of resources a neural network uses in training or inference. This analysis is important to evaluate the feasibility of the models. We will analyse metrics which refer to the performance of the models in terms of the time demand, the computing power and the memory space needed by each model. The models are here evaluated based on the structural information which is invariant with the case studies considered, and the time and memory demand which is closely related to that. In this paper, we evaluate the number of layers, FLOPS, inference and training times and the number of parameters. The number of layers, in convolutional-based models, expresses the capacity of the model to compress the features while extracting them. The floating point operations per second (FLOPS) is a measure of the computational complexity of deep learning models. It describes how many operations are required to run a single instance of a given model. We report the FLOPS results presented in the papers at the state-of-the-art (venue). The inference time is here computed as the time needed for a model to provide a prediction of a batch of one image. We compute the inference time as the mean out of three predictions. We load the fine-tuned model, we load an image and we apply the transformations required (resize and cast in a tensor) and we execute the model in evaluation modality, by using the utility of PyTorch. The inference time is related only to the computation of the prediction. We consider the number of parameters that are trained in the structure, these parameters define the memory required by the model to be loaded. The number of trainable parameters affects the training time which is here considered as a metric to evaluate the computational cost of the models. The training time is computed as the time needed to compute the prediction, the loss, make the backward step to accumulate the gradient for each parameter and the optimisation step to update all the parameters based on the current gradients. The inference and the training time are computed as the mean time required on three executions on our hardware.

3.6 | Gradient Activity Metrics

To explain how the model behaves with the input provided, we use the gradient-weighted class activation mapping (Grad-CAM) [52] technique. Grad-CAM utilises the gradients of the classification score concerning the final layers of each model, to identify the parts of an input image that most impact the classification score. The places where the gradient is large are where the final score depends most strongly on the data. With models in the CNN group, the Grad-CAM uses the feature maps produced by the batch normalisation layer in the last convolutional layer of a CNN. With models in the ViT and LBVT groups, we use the output of the normalisation layer before the attention block. This output is a tensor $<B, W, H>$, where B is the batch dimension, W consist of the class token, and the patches that

make up the image and finally H are the channels. To reshape the tensor to the 2D spatial images, we use a reshape_transform function as defined in reference [52].

4 | Results

4.1 | Classification Performance

We first consider the average per-species of the metrics, then the distribution of the performance versus the amount of data available per-species and we discuss the different behaviour of the models relative to the long tail distribution of the species. In the latter case, we discuss how the rarity of the species (the lack of samples) affects the performance of each model. Finally, we test the robustness of the models on the Artportalen.se dataset to observe the robustness of the models in generalising the new conditions.

4.1.1 | Results Per-Dataset

Table 1 shows the results obtained in the evaluation with the test datasets of Observation.org. The Coleoptera_Obs dataset is heavily unbalanced with a high number of species and a low number of samples. There is a minimal difference in performance on top-1, top-5 and F1score among `ViTAE`, `EffNet` and `T2TViT`. It is interesting to observe that the `T2TViT` model reaches a similar performance of `ViTAE` and `EffNet` while having a less complex structure. All the other models show an accuracy lower by 2%. In particular, the distilled models `ViTdVAE` and `ViTdEfN` reach a plateau in the training phase after 30–40 epochs and they are not able to improve which is visible in the results shown in Table 1. On the contrary, `DeiT` reaches high accuracy. The Odonata_Obs dataset is also unbalanced but they consist of a high number of samples and a low number of species. We observe a similar behaviour with this dataset, where `ViTAE`, `EffNet` and `T2TViT` obtained similar performances outperforming the other models. In particular, the `ViTAE` and the `EffNet` outperform the others in all metrics for both datasets. We can conclude that these models can be better candidates compared to the other models on the average species accuracy if the focus is the accuracy performance.

4.1.1.1 | Ablation Study on Knowledge Distillation.

We wished to understand the contribution of knowledge distillation to the model and so compared these against the same Vit trained in knowledge distillation. Table 2 shows the results of vanilla ViT compared to the other models obtained by knowledge distillation. We observe that the models trained in knowledge distillation outperform the vanilla ViT. That can be explained by the fact that the datasets are small and cannot alone satisfy the needs of ViT but together with the distillation of knowledge from models with high performance we can obtain better results. For this reason, in this paper, we consider the `ViTdEfN` and `ViTdVAE` rather than the vanilla ViT.

4.1.1.2 | Results on Artportalen Dataset.

In Section 4.1.1, we evaluated the models on the test set of the Observation.org dataset. The test set belongs to the same distribution as the train split even if the images are completely unknown to the models. To evaluate their robustness, we evaluate the models computing the avgACC top-1 and top-5 and F1score with Coleoptera_Art and Odonata_Art (Table 3). With Coleoptera_Art, the overall performance drops by 20% compared to the one observed for Coleoptera_Obs. This behaviour underlines a lack of robustness of all these models to different distributions. Among the models, we observe that

TABLE 2 | AvgACC top-1/top-5 and F1 score on Coleoptera_Obs and Odonata_Obs for the ablation study on ViT models, in bold the best result of the column.

Model	Top 1(%)	Top 5(%)	F1(%)
Coleoptera_Obs			
ViT	81.70	93.70	81.42
ViTdVAE	86.60	96.10	86.11
ViTdEfN	**86.70**	**96.20**	**86.40**
DeiT	84.10	95.10	83.90
Odonata_Obs			
ViT	88.80	97.5	88.40
ViTdVAE	90.90	98.30	90.33
ViTdEfN	**91.10**	**98.30**	**90.56**
DeiT	89.70	97.70	89.32

TABLE 1 | AvgACC top1/top5 and F1score on the validation-split of Coleoptera_Obs and Odonata_Obs, in bold the best results of the column.

NN		Coleoptera_Obs			Odonata_Obs		
Model	Version	Top 1(%)	Top 5(%)	F1(%)	Top 1(%)	Top 5(%)	F1(%)
Incpt	base_v3	86.10	95.60	85.75	89.90	97.40	88.54
EffNet	m_v2	88.00	96.70	87.78	92.60	98.50	**94.30**
ResNet	50	86.70	95.90	86.40	90.80	97.80	90.41
T2TViT	14	88.10	96.70	87.97	91.50	98.40	93.65
DeiT	B_p_16	84.10	95.10	83.90	89.70	97.70	89.32
ViTdVAE	S_p_16	86.60	96.10	86.11	90.90	98.30	90.33
ViTdEfN	S_p_16	86.70	96.20	86.40	91.10	98.30	90.56
ConViT	B	85.20	95.50	84.87	89.90	97.90	89.47
ViTAE	B_v2	**89.80**	**97.50**	**89.53**	**93.60**	**98.80**	93.29

all the models reach 64% on average for top-1 and 80% for top-5, whereas F1score is on average lower than 65%. The performance follows the pattern observed with Coleoptera_Obs in terms of the single performance of each model. Among the CNN group, the `EffNet` reaches the better performance on all the metrics, in the ViT group the `T2TViT` shows higher results and in the LBVT group, the `ViTAE` outperforms the others. There is a drop in performance also with Odonata_Art, even if it is limited to 10% and only with `EffNet`. All the other models show robust behaviour with the distribution of Artportalen, with performance that is in line with the one obtained with Odonata_Obs. Among the CNN group, the `ResNet` shows high performance, in the ViT group, the `T2TViT` reaches the better one, and in the LBVT the `ViTAE` is the model with the highest performance. The `ViTAE` is the model which outperforms all the other models in both datasets and is a good candidate if we consider classification performance.

4.1.2 | Results Per-Species

The results presented in Section 4.1.1 demonstrate the significant advantage of introducing transformers which decisively outperform CNN models in terms of accuracy. The fine-grained classification tasks focus on the details at the class level. Our analysis focuses on fine-grained classification tasks, examining accuracy distributions among species, particularly for rare species. Figure 3 shows the performance of the models, distinguishing between the 'long tail' and 'head'. The "head" includes the most common classes, accounting for 50% of the dataset, whereas the 'long tail' contains the rest. For the Coleoptera_Obs dataset, the head consists of the 101 most common species, with the smallest class having 2000 samples, whereas the head of the Odonata_Obs dataset features the 22 most common species, each with at least 9000 samples. We focus on the cumulative average accuracy (c-avgACC) for top-1 results from both the long tail and head as shown in Figure 4.

TABLE 3 | AvgACC top1/top5 and F1score on datasets from Coleoptera_Art and Odonata_Art, in bold the best result of the column.

	Coleoptera_Art			Odonata_Art		
Model	**Top 1(%)**	**Top 5(%)**	**F1(%)**	**Top 1(%)**	**Top 5(%)**	**F1(%)**
Incpt	65.20	81.90	65.56	86.30	95.10	86.11
EffNet	67.30	82.90	67.68	69.20	75.20	68.51
ResNet	65.30	82.30	65.81	87.50	95.40	87.50
T2TViT	66.80	83.40	67.16	87.40	96.30	87.49
DeiT	62.70	79.60	62.80	85.90	95.00	86.06
ViTdVAE	64.20	79.60	62.80	86.90	96.30	86.67
ViTdEfN	66.20	82.50	66.66	86.70	96.20	86.33
ConViT	63.30	80.30	63.63	85.40	95.50	85.22
ViTAE	**69.30**	**84.80**	**69.73**	**90.60**	**96.80**	**90.55**

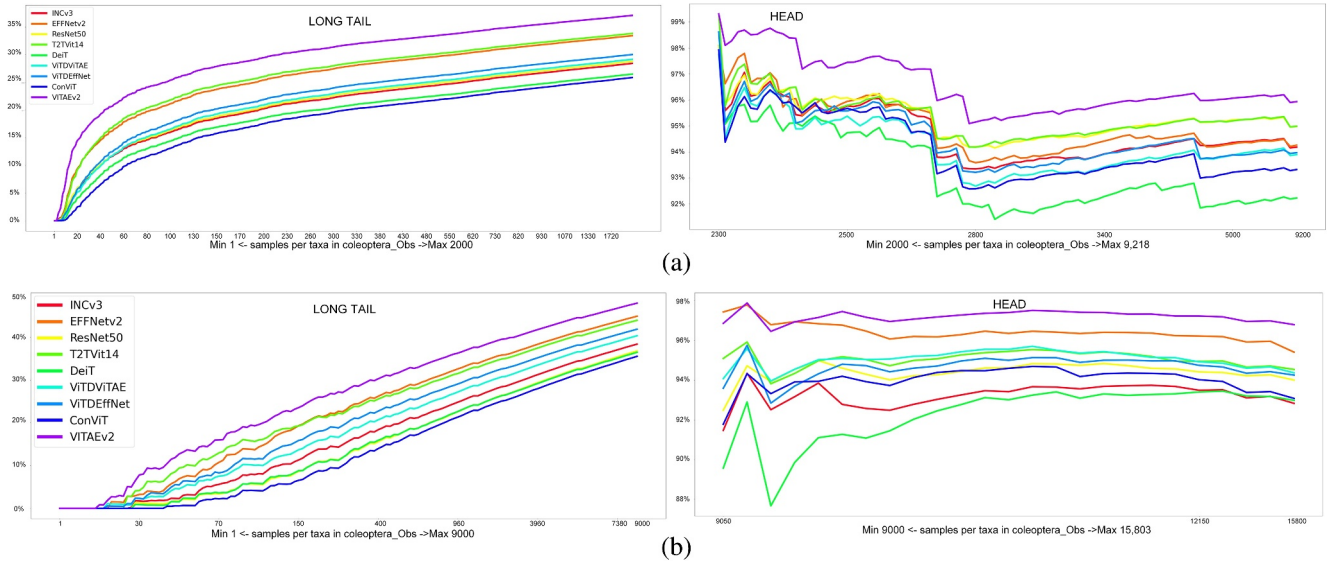
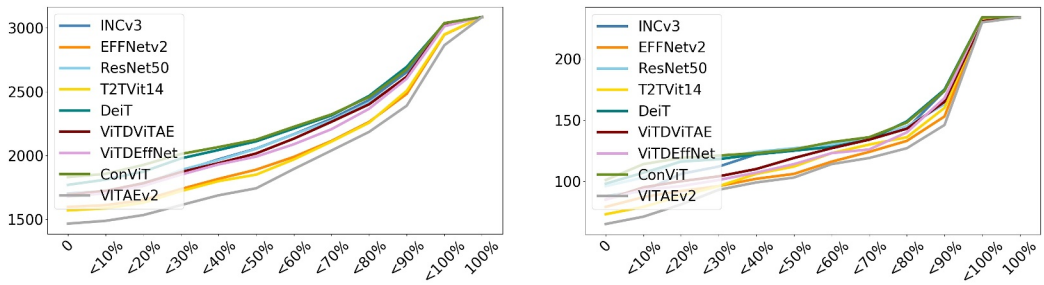


FIGURE 4 | Cumulative AvgACC top-1 (*y*-axis) results obtained per species in ascending order with each model for the species-wise (*x*-axis) based on the number of samples available in the training split. The results are presented for the long tail and head split identified in Figure 3. In (a and b) we present results obtained with Coleoptera_Obs and Odonata_Obs respectively.

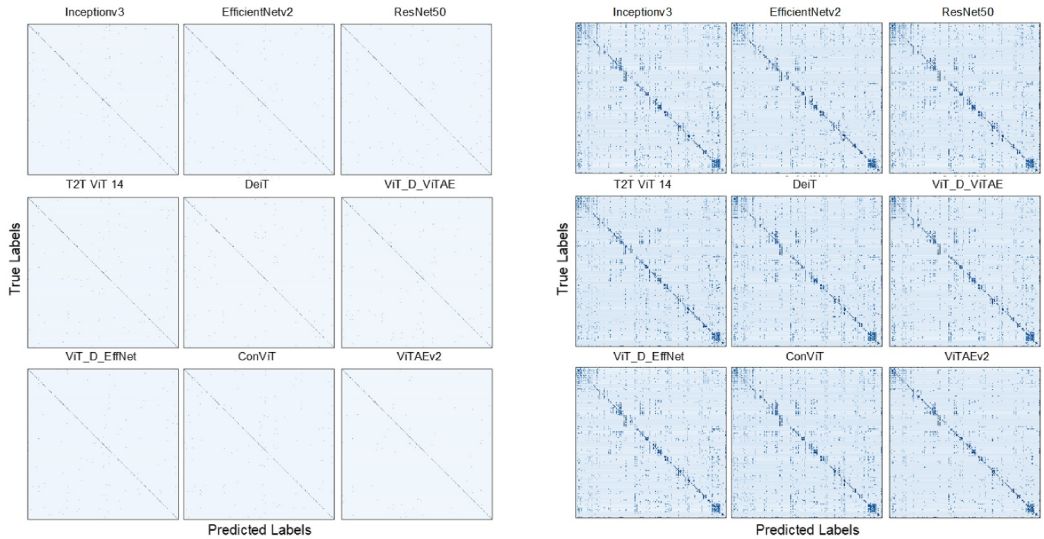
The models were trained on the original distribution without balancing among species, enabling us to evaluate their effectiveness in recognising rare species. Each point on the y-axis in the graphs represents the c-avgACC top-1 for models trained on species with up to x samples in the training set. In the long tail graphs in Figure 4a, the model ViTAE achieves an accuracy of 15%–20%, while other models score below 15% with species that have fewer than 20 samples. As the number of training samples increases, all models improve, with EffNet, T2TViT and ViTAE outperforming others by over 5%. EffNet performs competitively but falls slightly behind transformer-based models. In Figure 4b, which focuses on rare species with fewer than 30 samples, ViTAE shows a superior ability to learn class characteristics. Performance improves across all models with more training samples, with ViTAE consistently leading, followed by T2TViT and EffNet.

In Figure 5, we display the number of species each model classifies within a given range of avgACC top-1. For the Coleoptera_Obs, ViTAE has the lowest number of species with 0% accuracy, whereas ConViT has the highest number of species with 0% accuracy. With ViTAE, EffNet and T2TViT almost 2000 species have an accuracy lower than 70%, although, with all the other models, there are around 2500 species. ViTAE has the highest number of species with 100% accuracy, around 200 species while all the other models have less than 100 species with the same accuracy. With Odonata_Obs, we observe a similar trend in the performances of the models. ViTAE, EffNet and T2TViT have a low amount of species with 0% accuracy, in particular for ViTAE with have less than 20 species. For all the models, around 160 species have an accuracy lower than 90%. In Odonata_Obs and Coleoptera_Obs, we observed a different behaviour based on the number of data in training but a similar behaviour on the number of the species recognised. ViTAE proves to be less sensitive to the low amount of data available ending with fewer species with 0% top-1. Even if the other models are more sensitive to the data available in the training dataset, they reach good performance. Finally, we observe the confusion matrices, to evaluate the performance of the models based on the number of true positives, true negatives, false positives and false negatives. Figure 6 shows the confusion matrices with the test datasets of Coleoptera_Obs and Odonata_Obs. In the confusion matrix, the rows/columns close to each other are species which share the genus. For Coleoptera_Obs, we do not observe visible clusters. The predictions provided by the models highlight the diagonal of the matrices,



(a) Coleoptera_Obs (b) Odonata_Obs

FIGURE 5 | The lines show the number of species (x-axis) by the accuracy top-1 (y-axis) obtained by each model. For example: in Figure 5a the EffNet classifies almost 2100 species with avgACC top-1 between 60% and 70%.



(a) Coleoptera_Obs (b) Odonata_Obs

FIGURE 6 | Confusion matrices per order and models. We do not report the names of species in figures to maintain the matrices' readability. The classes are ordered by taxonomic name, which, in this paper, is a pair <genus_species>.

which is supported by the high overall accuracy that all the models obtained. Taking a closer look, it is visible that DeiT, ResNet, Incpt and ConViT show many predictions not lying on the diagonal. This is supported by the F1score metrics shown in Table 1, where these three models present lower results. With Odonata_Obs, there are some visible clusters shared among models which denote confusion among species from the same genus that is the difficulty of the fine-grained task. This underlines how all these models behave similarly for some classes. Also in this case study, ConViT, DeiT, ResNet and Incpt show more prediction far from the diagonal.

4.2 | Embedding Performance

We provide a quantitative and visual analysis of the embedding spaces learnt by the trained models. Figure 7 shows the visualisation of the embedding space of each model for the Observation.org datasets, obtained with UMAP. In Figure 7a, the embedding space aims to distinguish among 3087 classes while in Figure 7b, the classes are 235. We guide the evaluation of the visual representation taking into consideration the quantitative evaluation of the embedding space with the silhouette score. In Table 4, we separately report the mean silhouette score considering the genus and species levels. For Coloeptera_Obs, the ResNet obtains the lowest (worst) results both on genus and species and, in 7a, the embedding space of ResNet is collapsed in a central focal point. DeiT, ConViT and Incpt obtain a higher score, and also in these cases the embedding spaces appear collapsed on lateral focal points. The EffNet, T2TViT, ViTdEfN and ViTdVAE obtained similar results, whereas ViTAE outperforms all the other models at genus and species levels. These results are confirmed by the visual analysis where the embedding spaces appear distributed in the space. With Odonata_Obs in Figure 7b, Incpt obtained the lowest results at both levels and the embedding space shows multiple overlaps among species and does not present a structure. ConViT, EffNet and T2TViT obtained good results at the species level, the overlapping is less present and the points look spread in the space. Finally, ViTAE outperforms all the models at the species level and obtained good results together with ViTdEfN at the genus level where there is a minimum overlap but the overall results in a good distancing of the species. Combined with the visual analysis, the quantitative analysis suggests that the T2TViT and ViTAE generate embedding spaces with a better distribution of embedding in the space.

TABLE 4 | Embedding dimension for each model and average silhouette score obtained on the Odonata_Obs and Coleoptera_Obs embedding space, in bold the best result of the column.

		Silhouette score			
		Coleoptera_ Obs		Odonata_ Obs	
Model	Embedding Dim	Genus	Species	Genus	Species
Incpt	[1, 1024]	0.1293	0.1239	0.0407	0.0130
EffNet	[1, 1024]	0.1333	0.1577	0.1669	0.2165
ResNet	[1, 2048]	0.0439	0.0491	0.1453	0.1448
T2TViT	[1, 384]	0.1502	0.1739	0.1716	0.2345
DeiT	[1768]	0.1217	0.1039	0.1039	0.1704
ViTdVAE	[1, 768]	0.1403	0.1469	0.1844	0.2258
ViTdEfN	[1, 768]	0.1518	0.1689	**0.2063**	0.2482
ConViT	[1, 768]	0.1337	0.1147	0.1855	0.2157
ViTAE	[1, 1024]	**0.2189**	**0.2517**	0.1925	**0.3217**

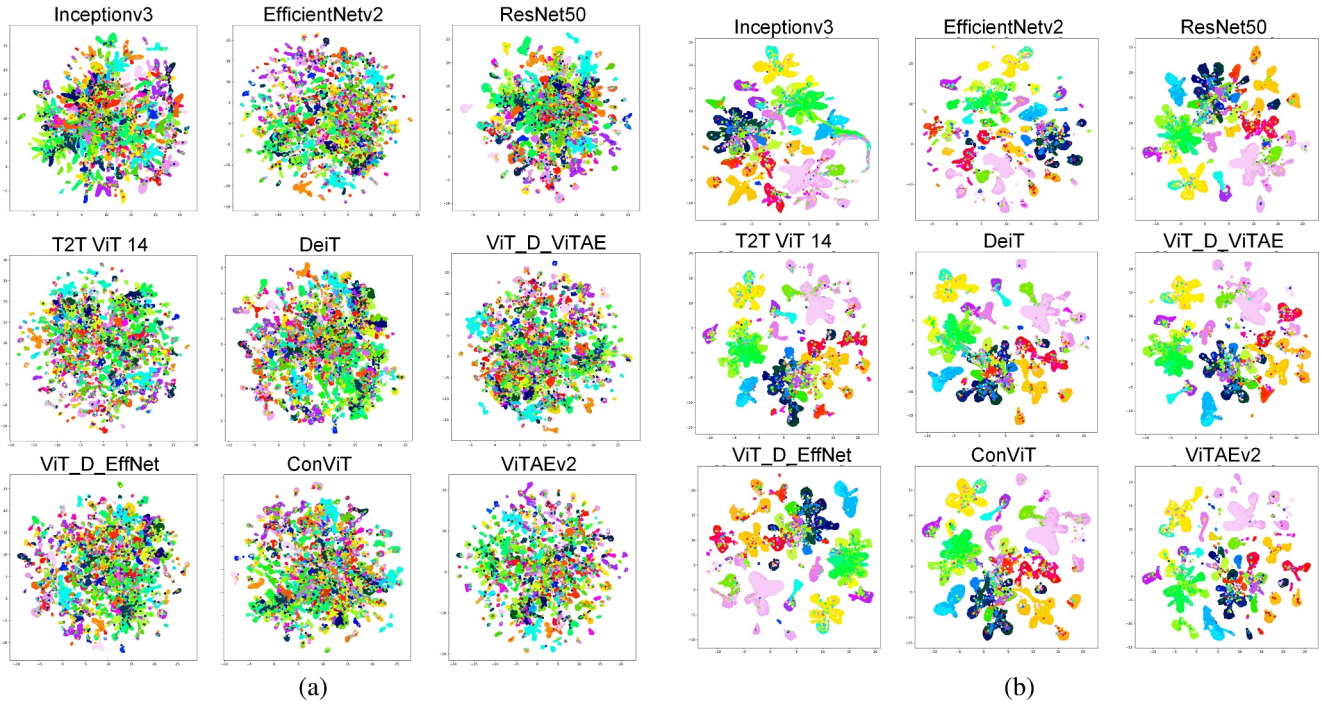


FIGURE 7 | UMAP visualisation of the embedding space generated at the species level by each of the trained models. In (a and b) we present results obtained with Coleoptera_Obs and Odonata_Obs respectively.

TABLE 5 | Structural information, inference/training time and the number of parameters of each model for Odonata_Obs and Coleoptera_Obs case studies, in bold the best result of the column.

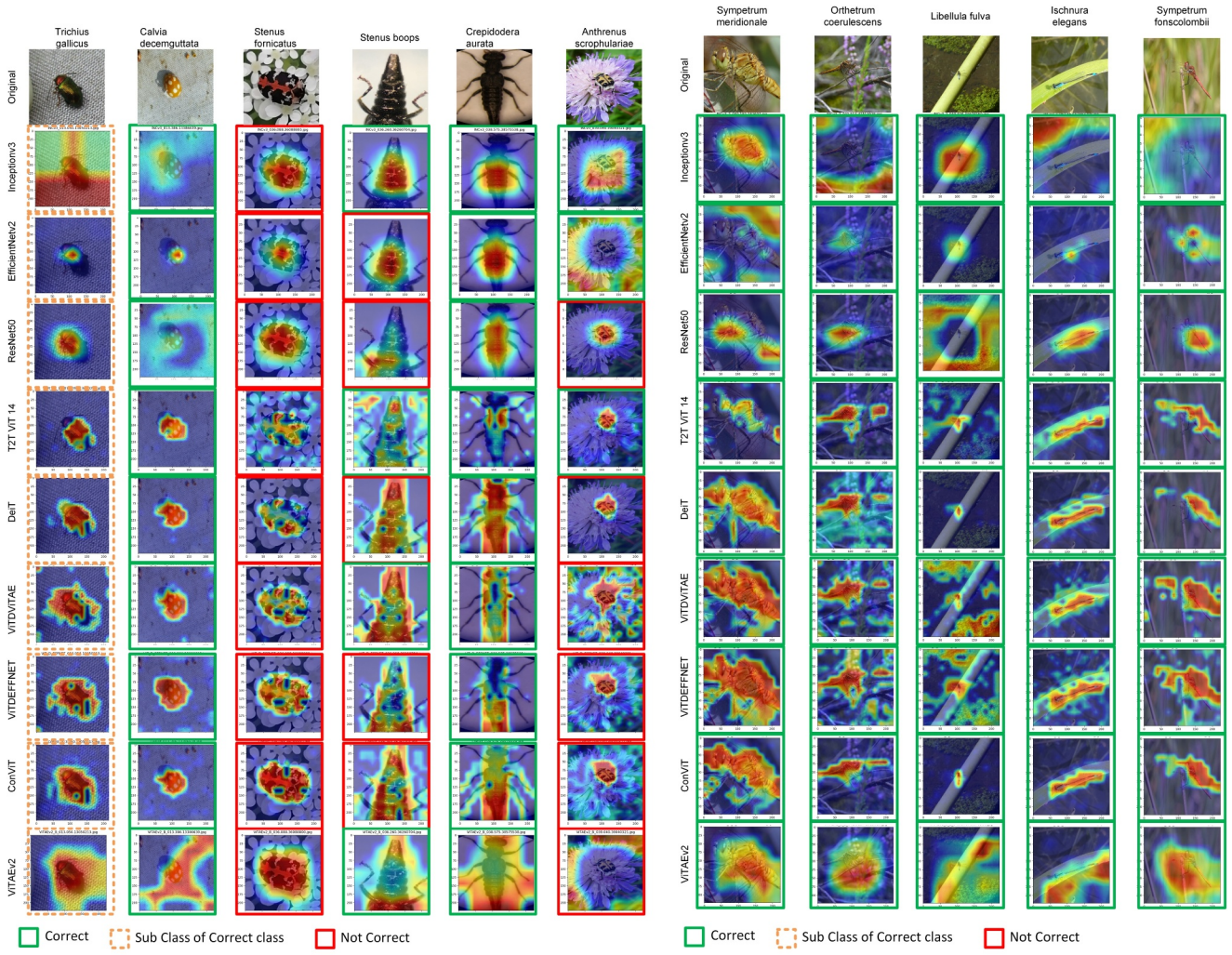
NN structural information				Coleoptera			Odonata		
Model	#Layers	FLOPS (G)	Venue	Inf-time (ms)	Param (M)	Train-time (ms)	Inf-time (ms)	Param (M)	Train-time (ms)
Incpt	48	5.71	CVPR'16	14.6	28.1	97.2	18.0	22.3	87.1
EffNet	154	24.0	ICML'21	28.2	57.7	181.1	27.9	51.6	168.2
ResNet	50	**3.80**	CVPR'16	10.2	29.8	52.5	**8.3**	24.0	44.8
T2TViT	16	5.20	ICCV'21	10.5	**22.3**	52.6	8.4	**21.2**	45.7
DeiT	**12**	17	ICML'21	**6.0**	90.5	**31.3**	34.9	86.2	**35.3**
ViTdVAE	**12**	4.60	ICML'21	25.7	22.9	53.0	25.5	21.8	82.1
ViTdEfN	**12**	4.60	ICML'21	28.9	22.9	60.0	24.2	21.8	87.8
ConViT	32	17.0	ICML'21	17.2	88.1	67.6	11.2	86.0	55.8
ViTAE	41	24.3	IJCV'22	24.3	91.8	165.6	24.4	88.9	151.5
Input–[1,3,224,224]									

4.3 | Computational Cost Performance

We compute the metrics for each model without any routine such as data augmentation or optimisation strategy. In Table 5, we observe that the number of layers strongly correlates with the model's group. The CNN have a deeper structure compared to equally complex fully/partially transformer models. This is not surprizing considering the characteristics of convolutional models versus transformers. Convolutional models require more layers to extract global features, whereas transformers can extract global features with just a few layers using patches. `ResNet` outperforms all the others in terms of FLOPS, followed by `ViTdEfN` and `ViTdVAE` which refer to the ViT model used as a student and `T2TViT`. These four models are the smaller and more portable models analysed in this paper. The two case studies are presented separately. Each has a different head dimension, corresponding to a different number of parameters. `ViTAE` requires almost 2 × the number of trainable parameters for `EffNet`, while if we consider the training time, `EffNet` takes slightly more time than `ViTAE`. `T2TViT` and `ResNet` are the fastest models among the ones considered. Finally, the inference time manifests the same behaviour as described for the training time.

4.4 | Gradients Activity Performance

The visualisation of the gradient activity computed with Grad-Cam is presented in Figure 8 with samples from Coleoptera_Obs and Odonata_Obs. We select inputs from different species with different resolutions to observe how the models react to these variations. In each row, we present a heatmap of the gradient activity at the final layers of each model, as described in Section 3.6, and we map the heatmap on the input images to identify the areas of interest for the model. The heatmaps show the regions with the higher and lower influence on the decision made by the model, respectively, in red and blue. In Figure 8a, the samples are of six distinctive species of Coleoptera, the third and the sixth present a botanical background, whereas all the others present a concrete background. Overall, the models have different gradient focus, and we can observe similarities among models from the same group. Models in the CNN group show areas of interest which are relative to the feature maps which are highly responsive during the classification. Models in the ViT group are more focused on the detailed silhouette of the subject. The two models considered for LBVT have different behaviour, whereas `ConViT` has a focus similar to models from the ViT group and `ViTAE` focus is not limited to the subject but also part of the background is taken into consideration. From the coloured and dashed squares, we observe that some of the samples are misclassified by all the models, in particular, the first and third images. All the models identify the first image with the class *Trichius gallicus*, whereas the class labelled in the data is *T. gallicus*. This behaviour can be due to two main factors: 1—the two species are closely related and present a similar morphology and 2—in the training dataset the *T. gallicus* consists of 427 samples, whereas the *T. gallicus* has 3823 samples. Hence, the models tend to choose the more common species. The third image is misclassified by all the models, we do not observe a pattern shared among all of them. `EffNet`, `Incpt`, `T2TViT` and `ViTdVAE` misclassified the image for a species from the same genus *Stenus ater*. All the other models misclassify the image with classes unrelated to genus or subspecies level. Finally the last image with species *Anthrenus scrophulariae* appears to be confusing for the majority of the models which focus on the flower instead of the insect. Figure 8b shows five samples of Odonata_Obs, the images show different species and the animals are at different distances from the camera. All the samples presented are well classified by all the models considered. The figure shows how the models focus differently while predicting the correct class. We observe that the overall behaviour of the models is similar to that observed with Coleoptera. With models in CNN, the focus areas are not always on the subject but rather on the background. The models in the ViT group and the `ConViT` model show a well-defined focus on the subject of the images. Finally, `ViTAE` focuses on the subject and background to compute the classification.



(a) Coleoptera_Obs

(b) Odonata_Obs

FIGURE 8 | Visualisation of gradient activation with GradCam. In the first row, there are the original inputs presented to the models; in each row, the heatmaps of the gradient activity show the regions with the higher (red) and lower (blue) influence on the decision made by the model.

5 | Discussion

In this section, we consider all the analyses presented in the paper and we present the overall view of the performance obtained by the models in the presented case study of fine-grained classification of images of insects. Table 6 summarises the various metrics discussed in the paper. We use a dictionary of symbols to indicate the best and the worst models in each category. We distinguish between focus on areas and details, not as positive or negative aspects but more as the ability of the models. The type of focus is an aspect to evaluate in relation to the dataset of interest to answer the question: are the details that can distinguish the species or is it more the overall image that can help in the classification? From the Table 6 we observe several characteristics of the models:

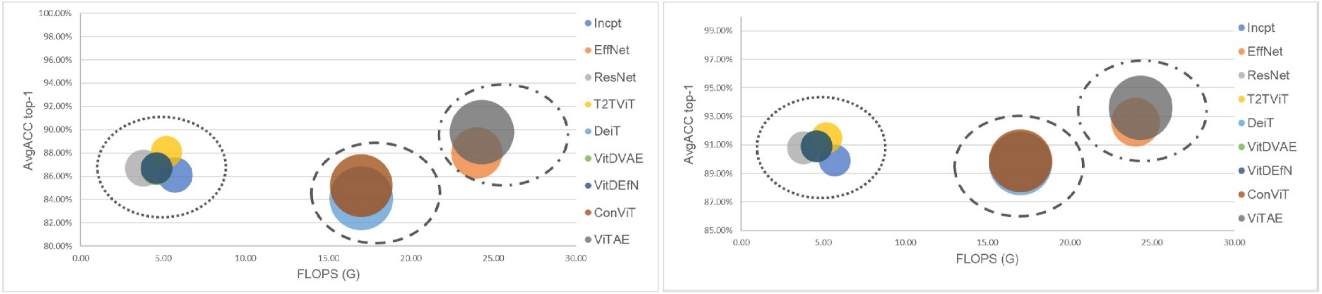
- `Incpt` does not show any particularly positive remarks in any of the two datasets. This model focuses on areas of the image.
- `EffNet` is the third best model for avgACC top-1 and its gradient activation is on areas rather than details with both datasets.
- `ResNet` shows to be fast in inference and train time but with low results in embedding space generation. It has not a particular focus.
- `T2TViT` has positive remarks on classification performance, embedding space generation and inference and train time while using the lowest amount of memory.
- `DeiT` is the fastest among the models selected but with low performance in classification and embedding.
- `ViTdVAE` has negative points on inference time and focus on areas of the pictures but it focuses on details and requires a low amount of memory space. This model has an evident focus on details while a low focus on wide areas.
- `ViTdEfN` is one of the best models on the embedding space generation and the memory demand.
- `ConViT` shows a high focus on details of the subject in the focus. On the Odonata_Obs dataset, it has good performance on classification and inference time.
- `ViTAE` has the best performance in classification and embedding space with a wide focus on the subject and

background. While its performance on computational cost is the lowest.

Among all the models, we identify `ViTAE`, `EffNet` and `T2TViT` as the models with the best classification performance; `ViTdEfN`, `ViTdEfN` and `T2TViT` with the best embedding space; `ResNet` and `T2TViT` the fast models both at inference and training time; `T2TViT`, `ViTdEfN` and `ViTdVAE` as the smaller models. Figure 9 shows the FLOPS demand with the avgACC top-1 for each of the models considered in the paper. In both Odonata_Obs and Coleoptera_Obs, we identify three sets of models highlighted in the Figure 9a,b with circles. In the circle of dots and dash, there are models with the highest FLOPS demand and highest top-1, these models are the `ViTAE` and `EffNet`. In the circle of the dash, some models do not show high accuracy and not a low FLOPS demand, `DeiT` and `ConViT`. Finally, in the circle of dots, there are all the other models which obtain good top-1, even if they do not outperform the others, and low FLOPS demand, these models are a good trade-off between cost and performance. In this last group, the `T2TViT` has the highest top-1, and it is a good candidate in case we have limits on the computational cost. Figure 10 present the avgACC top-1 and the silhouette score (Sec. 4.2) for each of the models. The behaviour of almost all the models is consistent among the two datasets. We observe that the behaviour of `ResNet` in Figure 10a and `DeiT` in Figure 10b is not consistent,

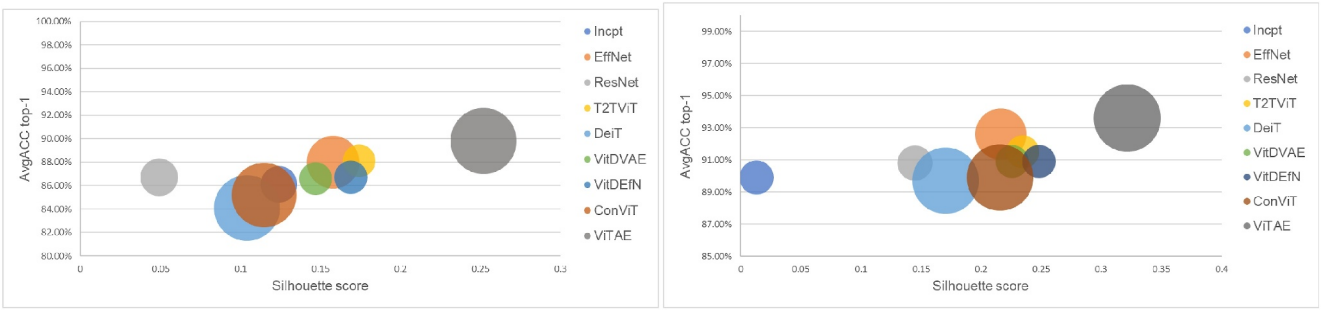
TABLE 6 | Summary of the metrics discussed in the paper for an overall visualisation: avgACC top-1, silhouette score (SC) at genus and species levels, inference and train time (inf/train T), number of parameters (mem space) and gradient activation (GA) on details and areas. Symbols: {+ + +: best; + +: second best; +: third best; =: average; −: third to worst; − −: second to worst; − − −: worst}.

Model	C: Top-1	SC: Genus	SC: Species	Inf T	Train T	Mem space	GA details	GA area
Coleoptera_Obs								
Incpt	=	=	=	=	−	=	−−	+
EffNet	+	=	=	−−	−−−	=	−	++
ResNet	=	−−−	−−−	++	++	=	−−−	=
T2TViT	++	+	++	+	=	+++	=	=
DeiT	−−−	−	−	+++	+++	−	=	=
ViTdVAE	−	=	=	−	+	++	++	−
ViTdEfN	=	++	+	−−−	=	++	+++	−
ConViT	−−	−	−	=	=	−	+	−−−
ViTAE	+++	+++	+++	=	−−	−−−	=	+++
Odonata_Obs								
Incpt	−−	−−−	−−−	=	=	=	−−	+
EffNet	++	=	=	−−−	−−−	=	−	++
ResNet	=	−−	−−	+++	++	=	−−−	=
T2TViT	+	+	+	++	+	+++	=	=
DeiT	−−−	−	−	=	+++	−−	=	=
ViTdVAE	=	=	=	−−		++	++	−−
ViTdEfN	=	+++	++	=	−	++	+++	−
ConViT	−−	++	=	+	=	−	+	−−−
ViTAE	+++	++	+++	−	−−	−−−	=	+++



(a) Coleoptera_Obs (b) Odonata_Obs

FIGURE 9 | Contextualise FLOPS demand with avgACC top-1 of each model. The dimension of the circles expresses the number of parameters (M) required by the models, Table 5. The x-axis is the FLOPS (G) shown in the Table 5 while the y-axis is the avgACC top-1 shown in the Table 1.



(a) Coleoptera_Obs (b) Odonata_Obs

FIGURE 10 | Contextualise embedding space quality through silhouette score results with avgACC top-1 of each model. The dimension of the bubbles is based on the number of parameters (as in Figure 9).

this can be due to the differences between the two datasets (number of species) or the limit of the network. There is no evidence of a connection between the number of parameters and the silhouette score, in fact in both the figures the models do not show any related trend. We do notice a relationship between the avgACC top-1 and the silhouette score: the growths of the avgACC top-1 and the silhouette score are directly proportional. This observation suggests that we need models with higher embedding performance to obtain better results in the fine-grained classification. The silhouette score directly evaluates the ability to form compact clusters in the embedding space. Hence, this gives credit to the Silhouette score as a metric to evaluate models applied for fine-grained classification.

6 | Conclusion

In the paper, we assess the performance of models at the fine-grained classification tasks for three groups of deep learning models: the convolutional neural network, the vision transformer and the locality-based vision transformer. For each of these groups, we consider models which at the state of the art are the newest with high classification performance or the most used ending up with nine models. For this study, we use the datasets of images of insects collected by citizen scientists and available on Observation.org [4]. At the state of the art, models from the convolutional neural network group are often applied without any investigation of models which belong to the other groups. We demonstrate that only consideration of the classification performance is not enough to evaluate a model in this delicate task, we need to consider its inner ability to deal with rare species, and how feasible it is to use that model in terms of computational requirement. Also, models trained on classification tasks are often used as the backbone for more specific tasks and with this perspective, the evaluation of the embedding space is an important aspect to consider. We also show that models based on transformers can satisfy more of these aspects while obtaining higher accuracy and for that, they are good candidates for the fine-grained classification tasks. Finally, we need to consider the end use of these models on datasets of Odonata and Coleoptera and for these case studies we observe that if the performance and the robustness are the features mainly required, the `ViTAE` and `EffNet` models are the most suitable for the fine-grained tasks with a preference for `ViTAE`; if the focal point is for public use so the inference speed is to be considered, `T2TViT` demonstrated to achieve good performance faster than the others and it shows a promising trade-off between performance and costs.

Author Contributions

Rita Pucci: investigation, software, validation, writing – original draft. **Vincent J. Kalkman:** funding acquisition, project administration, supervision, writing – review & editing. **Dan Stowell:** methodology, project administration, supervision, writing – review & editing.

Acknowledgements

This research was supported by the EU Horizon Europe projects MAMBO programme under Grant Agreement No.101060639, TETTRIs Grant Agreement 101081903, and Observation.org.

Conflicts of Interest

Uniud and Unipi.

Data Availability Statement

Data openly available in a public repository that issues datasets with DOIs: Artportalen Coleoptera, DOI:10.15468/DL.Q4MV7V, https://www.gbif.org/occurrence/download/0014398-230530130749713; Artportalen Odonata, DOI: 10.15468/DL.YKU9Z9, https://www.gbif.org/occurrence/download/0014782-230530130749713; Data available on request from the authors: Odonata and Coleoptera datasets from Observation.org are part of an internal database owned by Observation.org.

References

1. Qi Chang, H. Qu, P. Wu, and J. Yi, *Fine-Grained Butterfly and Moth Classification Using Deep Convolutional Neural Networks* (New Brunswick, NJ, USA: Rutgers University, 2017).

2. P. Audisio, M.-A. A. Zarazaga, A. Slipinski, et al., "Fauna Europaea: Coleoptera 2 (excl. Series Elateriformia, Scarabaeiformia, Staphyliniformia and Superfamily Curculionoidea)," *Biodiversity Data Journal* 3 (2015), https://doi.org/10.3897/bdj.3.e4750.

3. F. Suhling, G. Sahlén, S. Gorb, V. J. Kalkman, K. D. B. Dijkstra, and J. van Tol, *Chapter 35 - Order Odonata of Thorp and Covich's Freshwater Invertebrates*. 4th ed. (Boston: Academic Press, 2015), 893–932.

4. Observation.org (2023), https://observation.org/.

5. Obsidentify (2018), https://observation.org/apps/obsidentify/.

6. Global Biodiversity Information Facility (GBIF), https://www.gbif.org/.

7. D. Tuia, B. Kellenberger, S. Beery, et al., "Perspectives in Machine Learning for Wildlife Conservation," *Nature Communications* 13, no. 1 (2022): 792, https://doi.org/10.1038/s41467-022-27980-y.

8. A. Dosovitskiy, L. Beyer, A. Kolesnikov, et al., "An Image Is Worth 16 × 16 Words: Transformers for Image Recognition at Scale," *arXiv preprint arXiv:2010.11929* (2020).

9. G. Hinton, O. Vinyals, and J. Dean, "Distilling the Knowledge in a Neural Network," *arXiv preprint arXiv:1503.02531* (2015).

10. Y. Liu, J. Cao, B. Li, W. Hu, J. Ding, and L. Li, "Cross-Architecture Knowledge Distillation," in *Proceedings of the Asian Conference on Computer Vision* (Macau: Springer, 2022), 3396–3411.

11. C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning."

12. M. Tan and Q. V. Le, "Efficientnetv2: Smaller Models and Faster Training," *arXiv preprint arXiv:2104.00298* (2021).

13. M. Reil, C. Burmester, and M. Kalcher, "Bird Species Classification: ResNet-50 vs ConvNeXt," (2022), https://github.com/avocardio/resnet_vs_convnext.

14. Li Yuan, Y. Chen, T. Wang, et al., "Tokens-to-Token Vit: Training Vision Transformers From Scratch on Imagenet," in *International Conference on Computer Vision (ICCV)*, Virtual (Proceedings of the IEEE/CVF, 2021), 558–567, https://github.com/yitu-opensource/T2T-ViT.

15. S. d'Ascoli, H. Touvron, M. L. Leavitt, A. S. Morcos, G. Biroli, and L. Sagun, "Convit: Improving Vision Transformers With Soft Convolutional Inductive Biases," in *International Conference on Machine Learning (ICML)*, Virtual (Proceedings of the PMLR, 2021), 2286–2296.

16. Q. Zhang, Y. Xu, J. Zhang, and D. Tao, "Vitaev2: Vision Transformer Advanced by Exploring Inductive Bias for Image Recognition and Beyond," *International Journal of Computer Vision* 131, no. 5 (2023): 1141–1162, https://github.com/ViTAE-Transformer/ViTAE-Transformer.

17. artportalen.se (2023), https://www.artdatabanken.se/tjanster-och-miljodata/artportalen/.

18. Z. Miao, K. M. Gaynor, J. Wang, et al., "Insights and Approaches Using Deep Learning to Classify Wildlife," *Scientific Reports* 9, no. 1 (2019): 8137, https://doi.org/10.1038/s41598-019-44565-w.

19. J. Dembski and J. Szymański, "Bees Detection on Images: Study of Different Color Models for Neural Networks," in *Distributed Computing and Internet Technology: 15th International Conference* (India: Springer, 2019), 295–308.

20. W. Ding and G. Taylor, "Automatic Moth Detection From Trap Images for Pest Management," *Computers and Electronics in Agriculture* 123 (2016): 17–28, https://doi.org/10.1016/j.compag.2016.02.003.

21. S. Lim, S. Kim, and D. Kim, "Performance Effect Analysis for Insect Classification Using Convolutional Neural Network," in *International Conference on Control System, Computing and Engineering (ICCSCE)* (Malaysia: IEEE, 2017), 210–215.

22. H. Theivaprakasham, S. Darshana, V. Ravi, V. Sowmya, E. A. Gopalakrishnan, and K. P. Soman, "Odonata Identification Using Customized Convolutional Neural Network," *Expert Systems With Applications* 206 (2022): 117688, https://doi.org/10.1016/j.eswa.2022.117688.

23. H. Theivaprakasham, "Identification of Indian Butterflies Using Deep Convolutional Neural Network," *Journal of Asia-Pacific Entomology* 24, no. 1 (2021): 329–340, https://doi.org/10.1016/j.aspen.2020.11.015.

24. D. Xia, P. Chen, B. Wang, J. Zhang, and C. Xie, "Insect Detection and Classification Based on an Improved Convolutional Neural Network," *Sensors* 18, no. 12 (2018): 4169, https://doi.org/10.3390/s18124169.

25. B. Feuer, A. Joshi, M. Cho, et al., "Zero-Shot Insect Detection via Weak Language Supervision," *Plant Phenome Journal* 7, no. 1 (2024): e20107, https://doi.org/10.1002/ppj2.20107.

26. B. J. Spiesman, C. Gratton, R. G. Hatfield, et al., "Assessing the Potential for Deep Learning and Computer Vision to Identify Bumble Bee Species From Images," *Scientific Reports* 11, no. 1 (2021): 7580, https://doi.org/10.1038/s41598-021-87210-1.

27. P. Chudzik, A. Mitchell, M. Alkaseem, et al., "Mobile Real-Time Grasshopper Detection and Data Aggregation Framework," *Scientific Reports* 10, no. 1 (2020): 1–10, https://doi.org/10.1038/s41598-020-57674-8.

28. Y. Peng and Yi Wang, "CNN and Transformer Framework for Insect Pest Classification," *Ecological Informatics* 72 (2022): 101846, https://doi.org/10.1016/j.ecoinf.2022.101846.

29. Qi Wang, J.J. Wang, H. Deng, X. Wu, Y. Wang, and G. Hao, "AA-Trans: Core Attention Aggregating Transformer With Information Entropy Selector for Fine-Grained Visual Classification," *Pattern Recognition* 140 (2023): 109547, https://doi.org/10.1016/j.patcog.2023.109547.

30. O. Russakovsky, J. Deng, H. Su, et al., "Imagenet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision* 115, no. 3 (2015): 211–252, https://doi.org/10.1007/s11263-015-0816-y.

31. T. Mao, Z. Shi, and D.-X. Zhou, "Theory of Deep Convolutional Neural Networks III: Approximating Radial Functions," *Neural Networks* 144 (2021): 778–790, https://doi.org/10.1016/j.neunet.2021.09.027.

32. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *Computer Vision and Pattern Recognition (CVPR)* (Las Vegas: Proceedings of the IEEE, 2016), 2818–2826.

33. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Computer Vision and Pattern Recognition (CVPR)* (Las Vegas: Proceedings of the IEEE, 2016), 770–778.

34. S. Gupta and M. Tan, "EfficientNet-EdgeTPU: Creating Accelerator-Optimized Neural Networks With AutoML," *Google AI Blog* 2, no. 1 (2019).

35. S. Gupta and B. Akin, "Accelerator-Aware Neural Network Design Using autoML," *arXiv preprint arXiv:2003.02838* (2020).

36. K. Cao and X. Zhang, "An Improved Res-Unet Model for Tree Species Classification Using Airborne High-Resolution Images," *Remote Sensing* 12, no. 7 (2020): 1128, https://doi.org/10.3390/rs12071128.

37. C. Chen, L. Jing, H. Li, and Y. Tang, "A New Individual Tree Species Classification Method Based on the ResU-Net Model," *Forests* 12, no. 9 (2021): 1202, https://doi.org/10.3390/f12091202.

38. C. Zhang, K. Xia, H. Feng, Y. Yang, and X. Du, "Tree Species Classification Using Deep Learning and RGB Optical Images Obtained by an Unmanned Aerial Vehicle," *Journal of Forestry Research* 32, no. 5 (2021): 1879–1888, https://doi.org/10.1007/s11676-020-01245-0.

39. A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention Is All You Need," *Advances in Neural Information Processing Systems* 30 (2017).

40. H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training Data-Efficient Image Transformers & Distillation Through Attention," *International Conference on Machine Learning (ICML)* (2021): 10347–10357.

41. S. Abbasi, M. Hajabdollahi, N. Karimi, and S. Samavi, "Modeling Teacher-Student Techniques in Deep Neural Networks for Knowledge Distillation," in *International Conference on Machine Vision and Image Processing (MVIP)* (Iran: IEEE, 2020), 1–6.

42. Bo-K. Ruan, H.-H. Shuai, and W.-H. Cheng, "Vision Transformers: State of the Art and Research Challenges," *arXiv preprint arXiv:2207.03041* (2022).

43. J.-B. Cordonnier, A. Loukas, and M. Jaggi, "On the Relationship Between Self-Attention and Convolutional Layers," *arXiv preprint arXiv:1911.03584* (2019).

44. GBIF.Org, *GBIF, Biodiversity, Species Occurrences of Artportalen Coleoptera* (The Global Biodiversity Information Facility, 2023), https://doi.org/10.15468/DL.Q4MV7V.

45. GBIF.Org, *GBIF, Biodiversity, Species Occurrences of Artportalen Odonata* (The Global Biodiversity Information Facility, 2023), https://doi.org/10.15468/DL.YKU9Z9.

46. R. Wightman, *PyTorch Image Models* (GitHub repository, GitHub, 2019), https://doi.org/10.5281/zenodo.4414861.

47. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980* (2014).

48. I. Loshchilov and F. Hutter, "Sgdr: Stochastic Gradient Descent With Warm Restarts," *arXiv:1608.03983* (2016).

49. H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond Empirical Risk Minimization," *arXiv preprint arXiv:1710.09412* (2017).

50. S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization Strategy to Train Strong Classifiers With Localizable Features," in *International Conference on Computer Vision (ICCV)* (Korea: Proceedings of the IEEE/CVF, 2019), 6023–6032.

51. L. McInnes, J. Healy, and J. Melville. 2018. "Umap: Uniform Manifold Approximation and Projection for Dimension Reduction," https://umap-learn.readthedocs.io/en/latest/,arXivpreprintarXiv:1802.03426.

52. Jacob Gildenblat and others 2021. "PyTorch Library for CAM Methods," https://github.com/jacobgil/pytorch-grad-cam.

Supporting Information

Additional supporting information can be found online in the Supporting Information section.