

# TaxaNet: Harnessing a Hierarchical Loss Function for Insect Classification Using Deep Learning\*

Chirag Padubidri<sup>1,3,\*</sup>, Heleen Visserman<sup>4</sup>, Andreas Lanitis<sup>1,3</sup> and Andreas Kamilaris<sup>1,2</sup>

<sup>1</sup>CYENS Center of Excellence, Nicosia, Cyprus

<sup>2</sup>University of Twente, Enschede, The Netherlands

<sup>3</sup>Cyprus University of Technology, Limassol, Cyprus

<sup>4</sup>Radboud University Nijmegen, Netherlands

## Abstract

Insects have the largest percentage of all living organisms worldwide, playing a pivotal role in maintaining essential ecosystem services such as pollination, pest control, nutrient cycling, and food provisioning. However, recent studies have reported alarming declines in insect populations globally, highlighting an urgent need for automated methods to estimate and quantify these populations, to better understand the reasons of their decline and to take proper measures. The ability to automatically estimate insect populations is crucial for shaping appropriate environmental policies. Advances in AI and computer vision techniques are revolutionizing the study of insects through non-invasive camera traps. However, the diversity of insects, close resemblances of many species, and multi-level taxa classification remain significant challenges in image-based insect monitoring.

In this work, we propose TaxaNet a deep learning model for multi-level insect taxa classification, utilizing a pretrained EfficientNet as a feature extractor, followed by six classification blocks. Each block predicts one of the six taxonomic levels: Kingdom, Class, Order, Family, Genus, and Species. This hierarchical design and the loss function improves lower-level taxa predictions by leveraging the higher-level features available. A class-weighted hierarchical loss function, alongside the standard class-wise loss, allows the model to understand the relationships between taxonomic levels while maintaining classification accuracy. Trained on the Diopsis insect camera trap dataset containing 31,000 training images and 7,900 test images, the model achieved an average precision of 0.85 and a recall of 0.86 across five taxonomic levels. These results demonstrate the effectiveness of our approach in harnessing multi-level insect taxonomy to achieve multi-level insect classification.

## Keywords

insect classification, hierarchical loss, computer-vision,

## 1. Introduction

Insects, constitute the largest group of organisms on Earth and they play critical roles in sustaining essential ecosystem services such as pollination, pest control, nutrient cycling, and food provisioning [1, 2]. Despite their ecological importance, recent studies have highlighted significant declines in insect populations at a global scale, raising concerns about the biodiversity loss and potential disruption of these vital ecosystem functions [3, 4]. This alarming trend underscores the need for accurate and efficient methods to monitor insect populations, which is crucial for more informed environmental policies, understanding the drivers of decline and ensuring biodiversity conservation [5].

Traditional methods for insect population monitoring include manual counting and identification, but they are labor-intensive, time-consuming, and prone to human error [6]. The advent of AI and computer vision technologies, such as deep learning, offer promising solutions by enabling automated, non-invasive monitoring through camera traps and vision-based analytics [7, 8]. Hence, modern methods for insect population consider highly automated and semi- or fully autonomous camera traps, which allow visual monitoring of insects and AI techniques for detecting/classifying different insect individuals and species. In spite of the significant advancements in the field, several challenges remain, particularly due to the vast diversity of insect species, the high morphological similarities among many species, and

---

4th International Workshop on Camera Traps, AI, and Ecology, September 5 - 6, 2024, Hagenberg, Austria

\*Corresponding author.

✉ [c.padubidri@cyens.org.cy](mailto:c.padubidri@cyens.org.cy) (C. Padubidri)

🌐 <https://cpadubidri.github.io/> (C. Padubidri)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

the need for multi-level taxonomic classification that reflects the hierarchical structure of biological taxonomy [9]. Existing approaches for insect classification often rely on "single-level" computer vision classification models that do not account for the hierarchical relationships between different taxonomic levels [10]. While effective to some extent, these models fail to fully leverage the taxonomic hierarchy, limiting their overall accuracy and generalization capabilities. Hierarchical classification methods which consider the evolutionary relationships between species, have been explored in various domains and show promise in addressing the limitations of single-level classification [11].

In this work, we propose TaxaNet, which is a computer vision model based on deep learning for multi-level insect classification that utilizes hierarchical classification properties of insects. Our TaxaNet model is built upon a pretrained EfficientNet architecture, which serves as a feature extractor, followed by a series of classification blocks designed to predict taxonomic levels such as Kingdom, Class, Order, Family, Genus, and Species. Our main contribution is the introduction of a hierarchical loss function that complements the standard class-wise loss which is mainly used in literature, enhancing a model's ability to capture the hierarchical relationships between levels, thereby guiding the model to make more accurate and consistent predictions across the taxonomic hierarchy [12]. The proposed model was evaluated using the ARISE Diopsis insect camera trap dataset [13]. The challenge focuses on developing algorithms for classifying insect species within a public dataset of 39,445 insect images labeled with a multi-level taxonomy across five levels. The labels are incomplete, meaning not all images have all five taxonomic levels labeled. The challenge requires participants to submit their results to a portal where the algorithms are tested on a private test dataset. This task is made even more difficult by the highly imbalanced and fine-grained dataset, where image quality is often poor, and the data is organized hierarchically by taxonomic levels. This paper aims to demonstrate the effectiveness of the hierarchical classification approach, achieving high precision and recall across multiple taxonomic levels using ARISE Diopsis challenge.

## 2. Literature Review

Deep learning (DL) has become an essential technique for classifying insects and other biological organisms, particularly with the rise of convolutional neural networks (CNNs). Traditional methods often employ "single-level" classification models that do not consider the inherent hierarchical relationships between taxa, as seen in studies such as [9, 14, 15, 16]. The limitation of this single-level classification approach is that it fails to leverage the additional information available when the full taxonomic labels of the organisms under study are known.

In the context of animal monitoring, taxonomic hierarchical classification offers a more nuanced approach by considering the evolutionary relationships between species, which are typically structured in a tree-like hierarchy. In the context of the classification of insects and other arthropods, hierarchical classification has been explored using Deep Learning (DL) models like YOLOv3 [17], R-CNN [18], Faster R-CNN [19], SSD [20], and EfficientDet [21]. The work in [22] highlights the effectiveness of using hierarchical classification in the context of automated image analysis and deep learning, particularly for challenging biological datasets that often feature class imbalance, high similarity between species, and limited training examples. The study employed a two-step process where YOLOv3 was first used to detect and classify arthropods into broad taxonomic categories, followed by a separate model for species-level classification. This approach, tested on a custom dataset with 22 classes of arthropods, demonstrated significant improvements over a global detector trained without hierarchical classification, achieving a mAP of 75% compared to 48%. Despite its success, the method leverages hierarchical information in a segmented manner, with separate models for each taxonomic rank, rather than fully integrating hierarchical relationships within a single model.

The concept of integrating hierarchical classification within a single model has gained attention due to its potential to enhance the predictability performance by harnessing multiple taxonomic levels at the same time. This is achieved by incorporating class labels at each level of the hierarchy within a unified model. A crucial component in this approach is the design of a loss function that accounts for

the hierarchical dependencies, penalizing incorrect predictions more severely when they occur higher in the hierarchy. In the realm of hierarchical classification, various loss functions have been designed to leverage the structured information inherent in taxonomies. For instance, triplet loss has been explored to capture hierarchical relationships by ensuring that examples closer in the hierarchy are embedded more closely together. However, it faces challenges such as requiring careful construction of triplets and being limited in the extent of hierarchical information utilized at each iteration. To address these limitations, rank-based loss functions have been introduced, which generalize well across hierarchies with any number of levels and can handle incomplete hierarchical labels effectively [23].

For instance, [11] introduced a hierarchical cross-entropy loss function that organizes the output classes into a meaningful hierarchy. However, the approach was primarily evaluated on text classification tasks, where performance varied significantly across datasets, and it did not fully exploit the hierarchical structure within the CNN architecture. Further advancements in hierarchical classification involve embedding hierarchy directly into CNN architectures. Gao et al. [24] proposed a Deep Hierarchical Classification (DHC) framework, which was tested on both text and image datasets, such as Query-Category, Title-Category, and CIFAR-100. DHC demonstrated superior performance compared to models like SVM, FastText, TextCNN, and KerasCNN across different hierarchical levels. The study highlighted that DHC’s hierarchical loss function, combined with layer representation sharing, significantly enhanced the model’s ability to accurately classify across multiple levels of a hierarchy.

LaGrassa et al. [25] introduced a hierarchical classification architecture using deep linear layers with cross-entropy and center loss, optimizing local and global hierarchical relationships while penalizing hierarchy violations. Their model was tested on two datasets: VQA-Med 2019 and the Animals Taxonomy8 dataset, which features a three-level taxonomic structure. The results highlighted the effectiveness of their hierarchical loss combined with cross-entropy and center loss in improving multi-level classification tasks. In another approach, Bjerger et al. [12] applied hierarchical classification to the task of insect classification by incorporating multitask learning and anomaly detection into a single model architecture. Their work is particularly notable for integrating a hierarchical loss function that effectively models the taxonomic relationships between different insect species. This method leverages both the hierarchical structure and multitask learning, resulting in improved classification accuracy across multiple taxonomic levels. The inclusion of anomaly detection further enhances the model’s ability to handle previously unseen species, making it a robust approach for real-world insect classification tasks. The inclusion of anomaly detection further enhances the model’s ability to handle previously unseen species, contributing to an overall improvement in performance.

Our approach differs from [12] by tailoring a hierarchical loss function to our dataset with incomplete labels at deeper taxonomic levels, using a weighted sum of layer-wise entropy and dependency losses. This strategy emphasizes consistent higher taxonomic levels while still considering lower levels, making it particularly useful for incomplete datasets.

## 2.1. Our Contribution

Based on the literature review, our contributions can be summarized as follows:

1. We propose a pre-processing method to address datasets with high imbalance among insect classes by employing a combination of under-sampling and over-sampling techniques. This approach effectively balances the dataset, ensuring that it captures the diversity across all classes while ensuring adequate representation for minority classes.
2. Inspired by the work in [12], we propose a hierarchical loss function tailored to our dataset, which has incomplete labels at deeper taxonomic levels. Our proposal combines a weighted sum of layer-wise entropy and dependency losses, allowing the model to focus on specific levels of the hierarchy. This approach is particularly useful for incomplete datasets containing missing labels at different levels, as it places more importance on higher taxonomic levels, where labels are more consistent, while still considering the lower levels as well.
3. We propose TaxaNet, a model that utilizes EfficientNet as the backbone and is specifically adapted to handle multi-level classification. This architecture allows us to effectively implement the

hierarchical loss function, enabling the model to output predictions across multiple taxonomic levels, addressing the weaknesses of incomplete datasets.

Our approach has been effective, as shown by our position as the leader in the classification phase of the ARISE Diopsis Challenge. This result shows that our methods and model can handle complex hierarchical classification tasks well.

### 3. Methodology

#### 3.1. Dataset

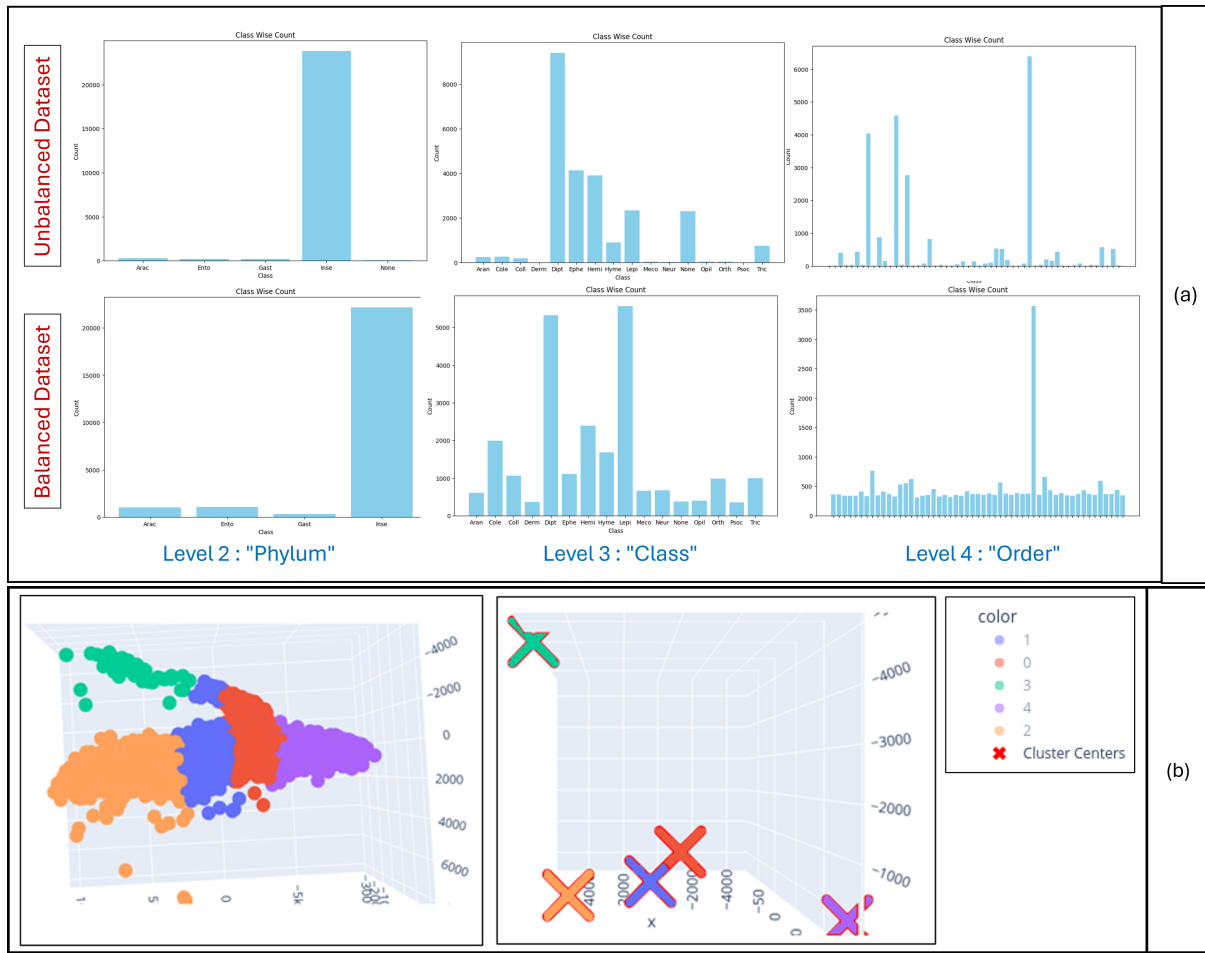
The Insect classification data set from Arise Biodiversity - Diopsis challenge dataset [13] is a comprehensive collection of images for the purpose of insect detection and multi-level classification. The Diopsis dataset includes 3,965 high-resolution images (3280 x 2464 pixels) containing a total of 48,216 labeled insects. Each insect is annotated with a bounding box and a taxonomic label. For the purpose of this work we employ only the classification dataset which contains 39,445 cropped images of individual insects labeled with taxonomic level. These images are labeled at the most specific taxonomic level possible (Fig: 1), identifying each insect to the most precise classification category allowed by the data, resulting in 84 distinct "leaf" classes. A "leaf" class represents the most detailed level of classification, where the taxonomic hierarchy concludes. If an insect can be identified at the species level, that species is used as the label; otherwise, the label may be at a higher level, such as genus or family. Additionally, a CSV file is provided to map each taxon's name to its ancestors, aiding in hierarchical classification.



**Figure 1:** Visualization of insect samples with multiple taxonomic labels. The table at the bottom of each image displays the taxonomic classification with the following levels: Kingdom, Phylum, Class, Order, Family, and Genus.

##### 3.1.1. Data Preprocessing:

The dataset presents several challenges, including the following: a) a significant imbalance in the number of training examples per species, b) a relatively poor image quality due to trade-offs between camera cost and resolution. To address these challenges, we implemented a combination of data under-sampling and over-sampling techniques, explained below.



**Figure 2:** (a) Visualization of class distribution before and after applying the oversampling technique. The graph illustrates how the oversampling method increased the representation of underrepresented classes to achieve a more balanced dataset. (b) Visualization of the undersampling process showing the formation of 5 clusters after applying PCA to reduce the data to four principal components, which explain 81% of the variance. The graph plots the first three principal components for visualization, highlighting the cluster centers and the distribution of samples within each cluster.

1. **Undersampling:** To address the issue of data imbalance, we employed a data under-sampling technique to reduce redundancy while maintaining the diversity of the dataset. The RGB images were subjected to Principal Component Analysis (PCA) to reduce dimensionality, retaining the first four principal components, covering a variance of 0.8193. These four components were then used to create clusters using K-Means clustering, with the number of clusters experimentally determined to be 5. Within each cluster, the distance of samples to their respective cluster centers were calculated as in Fig:2. To maximize diversity and eliminate redundancy, samples that were close to each other within the same cluster were removed. This undersampling technique reduced the dataset size from 39,446 to 27,273, ensuring that we retained the most diverse and representative samples.
2. **Oversampling:** To further address class imbalance, we implemented additionally an over-sampling technique based on weighted random sampling. This approach was applied at the level 4 of the taxonomic hierarchy (Order), where the dataset contains the largest number of classes. The core idea behind this method was to assign sampling probabilities inversely proportional to the class frequencies. Specifically, class weights were calculated as the inverse of the total number of items per class, ensuring that under-represented classes were given higher sampling probabilities. This technique effectively balanced the dataset by increasing the representation of minority classes without altering the original data distribution. By generating a sample set with



replacement, we maintained the total number of samples while ensuring a more uniform class distribution across the dataset.

These preprocessing steps helped to address data imbalance and was crucial in enhancing the model’s ability to learn effectively from all classes, particularly those with fewer examples, thereby improving the overall classification performance.

### 3.2. Model

We named our classification model as TaxaNet. It is based on the EfficientNet-B0 architecture [26], a convolutional neural network (CNN) known for its efficiency in both computational resources and accuracy. EfficientNet-B0 was pretrained on the ImageNet dataset, to exploit transfer learning in our specific insect classification task. TaxaNet leverages the EfficientNet-B0 backbone, which outputs a 1280-dimensional feature vector after passing through the series of its convolutional layers. To adapt this backbone for our multi-level classification task, we modified the classifier head of the network. The original fully connected layer of EfficientNet-B0, which outputs a 1000-dimensional vector for ImageNet classification, was replaced by a custom classifier. This custom classifier is a sequential block that first reduces the feature vector to 512 dimensions using a linear layer, followed by a ReLU activation and a dropout layer with a dropout rate of 0.5, to prevent overfitting. Finally, another linear layer maps the 512-dimensional vector to the desired number of classes.

Given the hierarchical nature of the dataset, where class labels are organized across multiple taxonomic levels, the TaxaNet model is designed to output predictions corresponding to these levels, such as species, genus, and family. The output vector is split into segments corresponding to these different taxonomic levels by connecting the feature vector to separate fully connected layers, each representing a different taxonomic level. The size of each segment, and the number of neurons in each fully connected layer, is determined by the number of classes at that particular taxonomic level, with the size directly corresponding to the one-hot encoded labels. This one-hot encoding of class labels ensures that each segment accurately represents the classes at its respective level, facilitating precise class predictions. This design not only ensures efficient computation but also effectively captures the hierarchical relationships between different classes, enhancing the model’s performance across various taxonomic levels. Additionally, the architecture is specifically tailored to handle a hierarchical loss function, which will be discussed in Sec: 3.3, ensuring that the training process is aligned with the hierarchical structure of the dataset.

### 3.3. Loss function

The loss function used in TaxaNet is designed to handle the hierarchical nature of the classification task, ensuring that the predictions are accurate at each taxonomic level and consistent across levels. We implemented a combination of Cross Entropy Loss and a Hierarchical Dependency Loss, inspired by the approach proposed in [12]. A summary of these components and their integration is described below.

#### 3.3.1. Cross Entropy Loss

As a baseline, we initially trained our model using only Cross Entropy Loss, a standard choice for classification tasks. For each taxonomic level  $l$ , the Cross Entropy Loss  $lloss_l$  is defined as:

$$lloss_l = - \sum_{j=0}^N y_{lj} \log(\hat{y}_{lj})$$

where  $y_{lj}$  is the true label and  $\hat{y}_{lj}$  is the predicted probability for the  $j$ th sample. The predicted probabilities  $\hat{y}_{lj}$  are derived from a softmax function applied to the model’s output at level  $l$ .

**Table 1**

Evaluation results comparing three models trained on different loss functions: M-L1 (cross-entropy), M-L2 (hierarchical), and M-L3 (layer-wise weighted hierarchical).

Model	Data	Loss	Precision				Recall			
			Level2	Level3	Level4	Level5	Level2	Level3	Level4	Level5
M-L1	Train	0.025	1.00	0.99	0.99	0.97	1.00	0.99	0.97	0.97
	Valid	0.536	0.81	0.71	0.66	0.75	0.84	0.83	0.71	0.82
M-L2 [12]	Train	0.287	0.99	0.99	0.97	0.98	1.00	0.98	0.97	0.98
	Valid	0.735	0.91	0.85	0.88	0.80	0.90	0.82	0.76	0.82
M-L3	Train	0.327	1.00	1.00	0.99	0.97	0.99	0.99	0.99	0.97
	Valid	0.814	0.91	0.86	0.87	0.79	0.92	0.84	0.85	0.82

Note: Metrics begin at Level 2 since Level 1 contains only one class and is not predicted by the model

### 3.3.2. Hierarchical Dependency Loss

In addition to Cross Entropy Loss, we incorporated a Hierarchical Dependency Loss [12] to penalize inconsistent predictions across taxonomic levels. This loss ensures that predictions at lower levels (e.g., species) align with those at higher levels (e.g., genus, family). The dependency loss  $dloss_l$  for each level  $l$  is defined as:

$$dloss_l = \exp(D_l) - 1, \quad D_l = \{1 \text{ if } \hat{y}_l \not\rightarrow \hat{y}_{l-1}, 0 \text{ otherwise}\}$$

Here,  $D_l$  checks if the predicted label at level  $l$  is a child of the predicted label at level  $l-1$ . A violation of this relationship results in a penalty, guiding the model to respect and consider the hierarchical consistency.

### 3.3.3. Weighted Sum with Decaying Weights

The method proposed in [12] assumes a well-labeled dataset with complete labels at all taxonomic levels. However, our dataset contains missing labels at different levels, making it necessary to adapt our approach accordingly. Thus, we used a Weighted Sum with Decaying Weights approach. This method assigns decreasing weights  $w_i$  to each taxonomic level, with higher levels (closer to the root) receiving more emphasis. The total loss is calculated as:

$$\text{Total Loss} = \sum_{i=1}^5 w_i \cdot (\text{Entropy Loss}_i + \text{Dependency Loss}_i)$$

Where:  $w_1 > w_2 > \dots > w_5$ , ensuring higher levels receive more emphasis. After some trial-and-error experiments, we chose  $w = [0.25, 0.25, 0.25, 0.15, 0.1]$  to prioritize higher levels while still considering the lower levels. Entropy and dependency losses at each level are scaled according to these weights.

This approach allows explicit control over the importance of each taxonomic level, which is particularly advantageous given the incomplete labeling in our dataset. The flexibility of this method helps the model learn effectively from the available data, improving overall classification performance.

## 4. Results and Discussion

We trained the TaxaNet model using the Adam optimizer with a learning rate of 0.001. The training involved three separate models, each using a different loss function as described in the previous section: the standard Cross Entropy Loss (M-L1), the hierarchical loss proposed by [12](M-L2), and our custom hierarchical loss with decaying weights(M-L3). The model was trained on a dataset comprising 27,273 images, which were split into a 90:10 ratio for training(24,545) and validation (2,727) images. The results from this training process are summarized in Table 1.

**Table 2**

Comparison of model performance on the ARISE Diopsis challenge private test set, showing accuracy and taxon distance metrics [13].

Model	Accuracy	Taxon Distance
M-L1	0.50	4.40
M-L2	0.59	2.34
M-L3	0.62	2.25

The model trained with Cross Entropy Loss (M-L1) exhibits the lowest performance, in terms of precision and recall at deeper taxonomic levels. This outcome is well expected, as Cross Entropy Loss treats each class independently, without considering the hierarchical relationships between classes. As a result, the M-L1 may accurately predict lower-level classes but fail to maintain consistency across levels, leading to reduced overall performance. In contrast, the models trained with hierarchical loss functions (M-L2 and 3) demonstrated significantly higher performance. The hierarchical loss proposed by [12] (M-L2) performed better than the Cross Entropy Loss by incorporating penalties for inconsistent predictions across taxonomic levels. This approach helps the model to learn not only the correct labels but also the relationships between different levels, leading to more coherent predictions throughout the hierarchy.

The model trained with our custom hierarchical loss function with decaying weights (TaxaNet, M-L3) slightly outperformed the hierarchical loss from [12]. This slight improvement can be attributed to the weight applied at different levels of the hierarchy. By assigning higher weights to predictions at higher taxonomic levels (closer to the root of the hierarchical structure of the insects' biological taxonomy), our loss function ensures that TaxaNet places greater emphasis on learning these broader classifications. This approach is particularly beneficial in scenarios where higher-level classes are more critical for classification purposes, or in cases where data labeling is not complete, and the finer-grained classes at lower levels need to be considered for better classification. Moreover, the decaying weights allow for a balanced contribution from each taxonomic level, preventing the model from overfitting to any single level of the hierarchy.

The results in Table 1 are based on a dataset that was split into a 90:10 ratio for training and validation, respectively. While the validation set is slightly smaller than usual, this was a deliberate choice due to the limited data available for training. The aim was to encourage the model to see as many samples as possible during training to improve its accuracy. Despite the small validation set, we performed a more robust evaluation by testing the three models on a private test dataset provided by the ARISE Diopsis Challenge portal. This final evaluation on the private dataset is important for assessing the models' generalization in real-world challenges.

The results, summarized in Table 2, show the accuracy of our models on the ARISE private test set. Table 2 includes two key metrics: overall accuracy and taxon distance. Accuracy refers to the percentage of insects correctly classified, providing an overall measure of model performance, though it tends to be influenced by the majority classes. Taxon distance, on the other hand, is a custom metric specific to hierarchical labels. It quantifies the number of edges that need to be traversed within the taxonomic tree to move from the true taxon to the predicted taxon. This metric accounts for the nature of hierarchical misclassifications. Lower taxon distances indicate predictions that are closer to the correct classification, even if not entirely accurate, thus providing a more nuanced view of model performance. As shown in the table, TaxaNet (Model 3), which utilizes our custom hierarchical loss function with decaying weights, achieved the highest accuracy and the lowest taxon distance. We currently lead the competition at the ARISE Diopsis Challenge, outperforming the other participants in both accuracy and taxonomy distance [13]. We strongly believe that the key elements which allow us to lead the competition are the ones described in this paper.



## 4.1. Future Work

As future work, we intend to further experiment with the loss function by incorporating taxon distance together with the existing entropy and dependency loss, either as a differentiable term or as a regularization term. The taxon distance, used in the Diopsis challenge, seems to be an interesting metric we intend to investigate integrating into our loss function, as it could potentially improve accuracy. By explicitly penalizing predictions that deviate significantly from the correct classification within the taxonomic hierarchy, we aim to enforce greater hierarchical consistency.

Our work has already been compared with the approach by Bjerger et al. [12], as shown in Table 1. Moving forward, we plan to extend our comparison to include the work by LaGrassa et al. [25], who introduced a hierarchical classification architecture using deep linear layers with cross-entropy and center loss. Their focus on optimizing both local and global hierarchical relationships while penalizing hierarchy violations offers another valuable perspective. Comparing our work with theirs will help us further refine our loss function and overall model architecture.

We will also work on preprocessing the data to better handle the varying image resolutions in our dataset. Currently, we resize all images to a uniform size, which may result in the loss of important information. Future work will involve exploring techniques that can preserve more detail and improve model performance.

Additionally, we plan to extend our work on new datasets, which can also be incomplete, and focus on other insect species. This will allow us to assess how well TaxaNet generalizes to other problems, challenges, and datasets. A candidate such dataset is one that we are currently collecting at our field sites. This new dataset will present unique challenges and opportunities, particularly in dealing with diverse environmental conditions, insect species, and variations in image quality.

## 5. Conclusion

In this work, we proposed a method for hierarchical insect classification using the TaxaNet model, built on top of the EfficientNet architecture. Our approach involved data preprocessing based on under- and over-sampling techniques, as well as the proposal of a novel loss function, which is based on a custom hierarchical loss with decaying weights, to improve classification accuracy and maintain consistency across taxonomic levels. The model was trained and evaluated on an incomplete dataset with many challenges, demonstrating superior performance, in comparison to existing approaches.

The experimental results showed that TaxaNet performs particularly well in scenarios where hierarchical relationships between classes are critical and at the same time the datasets have incomplete information in some taxonomical levels. The inclusion of hierarchical dependencies in the loss function proved to be highly effective, with our model achieving higher accuracy and lower taxon distance compared to models trained with standard Cross Entropy Loss. This indicates that TaxaNet not only predicts the correct classes but also respects and harnesses the hierarchical structure, resulting in more meaningful and accurate predictions. As the current leader in the open competition of the ARISE Diopsis Challenge, the proposed method demonstrates significant potential for application in large-scale insect classification and conservation efforts, providing a robust tool for biologists and ecologists to monitor species with greater accuracy and efficiency.

## 6. Funding

This research work has been partially funded from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 739578 and from the Government of the Republic of Cyprus through the Deputy Ministry of Research, Innovation and Digital Policy.

## 7. Acknowledgments

The authors would like to express their gratitude to Savvas Karatsiolis for his invaluable assistance and insights throughout the project.

## 8. Code Availability

The code used in this study is hosted on GitHub at <https://github.com/cpadubidri/insect-classification>. While the repository is currently private to protect the integrity of an ongoing challenge, it will be made publicly accessible by October 2024 to ensure reproducibility.

## References

- [1] J. E. Losey, M. Vaughan, The economic value of ecological services provided by insects, *Bioscience* 56 (2006) 311–323.
- [2] L. H. Yang, C. Gratton, Insects as drivers of ecosystem processes, *Current Opinion in Insect Science* 2 (2014) 26–32.
- [3] F. Sánchez-Bayo, H. A. Wyckhuys, Worldwide decline of the entomofauna: A review of its drivers, *Biological Conservation* 232 (2019) 8–27.
- [4] C. A. Hallmann, M. Sorg, E. Jongejans, H. Siepel, N. Hofland, H. Schwan, W. Stenmans, A. Müller, H. Sumser, T. Hörren, D. Goulson, H. de Kroon, More than 75 percent decline over 27 years in total flying insect biomass in protected areas, *PLOS ONE* 12 (2017) e0185809.
- [5] P. Cardoso, S. R. Leather, Predicting a global insect apocalypse, *Insect Conservation and Diversity* 12 (2019) 263–267.
- [6] Y. Basset, L. Cizek, P. Cuénoud, R. K. Didham, F. Guilhaumon, O. Missa, V. Novotny, F. Ødegaard, T. Roslin, J. Schmidl, et al., Arthropod diversity in a tropical forest, *Science* 338 (2012) 1481–1484.
- [7] N. Larios, H. Deng, A. Zhang, M. Sarpola, J. Yuen, M. Shatnawi, D. Shasha, V. Mazzanti, T. G. Dietterich, et al., Automated insect identification through concatenated histograms of local appearance features: Feature vector generation and region detection for insect images, *Machine Vision and Applications* 21 (2010) 797–810.
- [8] C. Padubidri, A. Kamilaris, A. Charalambous, A. Lanitis, M. Constantinides, The be-hive project—counting bee traffic based on deep learning and pose estimation, in: K. Arai (Ed.), *Intelligent Systems and Applications*, Springer Nature Switzerland, Cham, 2024, pp. 531–545.
- [9] D. Xia, P. Chen, B. Wang, J. Zhang, C. Xie, Insect detection and classification based on an improved convolutional neural network, *Sensors* 18 (2018). URL: <https://www.mdpi.com/1424-8220/18/12/4169>. doi:10.3390/s18124169.
- [10] E. Rodner, A. Freytag, J. Denzler, Fine-grained recognition in the noisy wild: Sensitivity analysis of convolutional neural networks approach, *International Journal of Computer Vision* 122 (2017) 41–52.
- [11] C. Wu, M. Tygert, Y. LeCun, A hierarchical loss and its problems when classifying non-hierarchically, *PLOS ONE* 14 (2019) 1–17. URL: <https://doi.org/10.1371/journal.pone.0226222>. doi:10.1371/journal.pone.0226222.
- [12] K. Bjerge, Q. Geissmann, J. Alison, H. M. Mann, T. T. Høye, M. Dyrmann, H. Karstoft, Hierarchical classification of insects with multitask learning and anomaly detection, *Ecological Informatics* 77 (2023) 102278. URL: <https://www.sciencedirect.com/science/article/pii/S1574954123003072>. doi:<https://doi.org/10.1016/j.ecoinf.2023.102278>.
- [13] A. Biodiversity, Arise diopsis classification dataset, <https://diopsis.challenges.arise-biodiversity.nl/diopsis/>, 2024. Accessed: 2024-08-08.
- [14] O. L. P. Hansen, J.-C. Svenning, K. Olsen, S. Dupont, B. H. Garner, A. Iosifidis, B. W. Price, T. T. Høye, Species-level image classification with convolutional neural network enables insect identification from habitus images, *Ecology and Evolution* 10 (2020) 737–747. URL: <https://doi.org/10.1002/ece3.5888>.

- //onlinelibrary.wiley.com/doi/abs/10.1002/ece3.5921. doi:<https://doi.org/10.1002/ece3.5921>. arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1002/ece3.5921>.
- [15] T. Kasinathan, D. Singaraju, S. R. Uyyala, Insect classification and detection in field crops using modern machine learning techniques, *Information Processing in Agriculture* 8 (2021) 446–457. URL: <https://www.sciencedirect.com/science/article/pii/S2214317320302067>. doi:<https://doi.org/10.1016/j.inpa.2020.09.006>.
- [16] V. Kittichai, T. Pongsakul, K. Chumchuen, Y. Samung, P. Sriwichai, N. Phatthamolrat, T. Tongloy, K. Jaksukam, S. Chuwongin, S. Boonsang, Deep learning approaches for challenging species and gender identification of mosquito vectors, *Scientific Reports* 11 (2021) 4838. URL: <https://doi.org/10.1038/s41598-021-84219-4>. doi:10.1038/s41598-021-84219-4.
- [17] J. Redmon, A. Farhadi, Yolov3: An incremental improvement, in: arXiv preprint arXiv:1804.02767, 2018.
- [18] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 580–587.
- [19] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, in: *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 91–99.
- [20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, Ssd: Single shot multibox detector, in: *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016, pp. 21–37.
- [21] 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 (CVPR)*, 2020, pp. 10781–10790.
- [22] P. Tresson, D. Carval, P. Tixier, W. Puech, Hierarchical Classification of Very Small Objects: Application to the Detection of Arthropod Species, *IEEE Access* 9 (2021) 63925–63932. URL: <https://hal-lirmm.ccsd.cnrs.fr/lirmm-03477540>. doi:10.1109/ACCESS.2021.3075293.
- [23] I. Nolasco, D. Stowell, Rank-based loss for learning hierarchical representations, in: *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2022. URL: <http://dx.doi.org/10.1109/ICASSP43922.2022.9746907>. doi:10.1109/icassp43922.2022.9746907.
- [24] D. Gao, Deep hierarchical classification for category prediction in E-commerce system, in: S. Malmasi, S. Kallumadi, N. Ueffing, O. Rokhlenko, E. Agichtein, I. Guy (Eds.), *Proceedings of the 3rd Workshop on e-Commerce and NLP*, Association for Computational Linguistics, Seattle, WA, USA, 2020, pp. 64–68. URL: <https://aclanthology.org/2020.ecnlp-1.10>. doi:10.18653/v1/2020.ecnlp-1.10.
- [25] R. L. Grassa, I. Gallo, N. Landro, Learn class hierarchy using convolutional neural networks, *Applied Intelligence* 51 (2021) 6622–6632. URL: <https://doi.org/10.1007/s10489-020-02103-6>. doi:10.1007/s10489-020-02103-6.
- [26] M. Tan, Q. V. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, *CoRR* abs/1905.11946 (2019). URL: <http://arxiv.org/abs/1905.11946>. arXiv:1905.11946.