A Step Towards Worldwide Biodiversity Assessment: The BIOSCAN-1M Insect Dataset Supplementary Materials

Zahra Gharaee[3*], **ZeMing Gong**[4*], **Nicholas Pellegrino**[3*], **Iuliia Zarubiieva**[2,5], **Joakim Bruslund Haurum**[7], **Scott C. Lowe**[5,8], **Jaclyn T.A. McKeown**[1,2], **Chris C.Y. Ho**[1,2], **Joschka McLeod**[1,2], **Yi-Yun C Wei**[1,2], **Jireh Agda**[1,2], **Sujeevan Ratnasingham**[1,2], **Dirk Steinke**[2†], **Angel X. Chang**[4,6†], **Graham W. Taylor**[2,5†], **Paul Fieguth**[3†]
[1]Centre for Biodiversity Genomics, [2]University of Guelph, [3]University of Waterloo, [4]Simon Fraser University, [5]Vector Institute for AI, [6]Alberta Machine Intelligence Institute (Amii), [7]Aalborg University and Pioneer Centre for AI, [8]Dalhousie University
https://biodiversitygenomics.net/1M_insects/

S1 Data collection and organization

The BIOSCAN-1M Insect dataset consists of insect RGB images and a metadata file containing taxonomic annotation, DNA barcode sequences, and an assigned Barcode Index Number (BIN). In the following sections, we describe the resources available within the dataset.

S1.1 RGB images

To facilitate different levels of visual processing we created 6 packages of color images of varying sizes. These packages are as follows:

Original full-size RGB images. The original images are converted to JPEG image format. These images each have a resolution of 2880×2160, and they are typically around 5 MB in size, however, some images are smaller at 600–800 kB. The package is structured as 113 zip files, each of which contains 10,000 images except the last (zip file 113 contains 8,131 original full-size images). The total size of this package is 2.5 TB. All 113 zip files are stored within the BIOSCAN project space in GoogleDrive as described in Section S2 inside a folder named **BIOSCAN_original_images** and the zip files named as **bioscan_images_original_full_part<n>** where **n** is the partition ID and is in the range of 1 to 113.

Cropped RGB images. The images in this package are cropped by our cropping tool as described in the main body of the paper and available in the accompanying BIOSCAN-1M code repository. The package is structured into six zip files where each file contains 20 partitions (20×10,000 files), except the last zip file which contains 13 partitions. The total size of this package is 151 GB. All six zip files are stored within the BIOSCAN project space in GoogleDrive as described in Section S2 inside a folder named **BIOSCAN_cropped_images** and the zip files named as **bioscan_images_cropped_part<m-n>** where **m-n** indicate the start and end partition ID, in the range of 1–113.

Resized original RGB images. This package is available in two archive formats (zip and HDF5). The package contains downscaled versions of the original images, requiring reduced storage space. The resizing was done such as to reduce the smaller dimension of the image to 256 pixels (and the longer side scaled to preserve the aspect ratio of the original image) and then saved in JPEG format. The total size of these packages are approximately 27 GB, and they are named as **original_256.zip** and **original_256.hdf5**.

*Joint first author.
†Joint senior/last author.

37th Conference on Neural Information Processing Systems (NeurIPS 2023) Track on Datasets and Benchmarks.

Resized cropped RGB images. This package is also available in two archive formats (zip and HDF5). The package contains resized versions of the cropped images. The resizing was done such as to reduce the smaller dimension of the image to 256 pixels (and the longer side scaled to preserve the aspect ratio of the cropped image) and then saved in JPEG format. The total size of these packages are approximately 7 GB, and they are named as **cropped_256.zip** and **cropped_256.hdf5**.

S1.2 Metadata file

To enhance the metadata of our published dataset, we incorporated structured metadata following Web standards. The metadata file for our dataset is named **BIOSCAN_Insect_Dataset_metadata**. We created two versions of this file: one data frame in TSV format (**.tsv**) and the other in JSON-LD format (**.jsonld**). The JSON-LD file was validated using the *Google Inspection Tool*.

The metadata file is a table with 22 columns, which contain content as described below. Note that if a sample was not labelled by a taxonomist, for each taxonomy ranking group (columns 4–13) the corresponding annotation is listed as **not_classified** instead. Similarly, if a sample has no association with an experiment shown by columns 16–21, then the sample's role is shown as **no_split**.

1. **sampleid**: An identifier given by the collector.
2. **processid**: A unique number assigned by BOLD to each record.
3. **uri**: Barcode Index Number (BIN).
4. **name**: Taxonomy ranking classification label.
5. **phylum**: Taxonomy ranking classification label.
6. **class**: Taxonomy ranking classification label.
7. **order**: Taxonomy ranking classification label.
8. **family**: Taxonomy ranking classification label.
9. **subfamily**: Taxonomy ranking classification label.
10. **tribe**: Taxonomy ranking classification label.
11. **genus**: Taxonomy ranking classification label.
12. **species**: Taxonomy ranking classification label.
13. **subspecies**: Taxonomy ranking classification label.
14. **nucraw**: Nucleotide barcode sequence.
15. **image_file**: Image file name stored in structured packages.
16. **large_diptera_family**: Image association with the training, validation, and test split of experiment-1.
17. **medium_diptera_family**: Image association with the training, validation, and test split of experiment-2.
18. **small_diptera_family**: Image association with the training, validation, and test split of experiment-3.
19. **large_insect_order**: Image association with the training, validation, and test split of experiment-4.
20. **medium_insect_order**: Image association with the training, validation, and test split of experiment-5.
21. **small_insect_order**: Image association with the training, validation, and test split of experiment-6.
22. **chunk_number**: A unique ID to locate the corresponding images within the dataset packages.

S2 Informational content

The link to access the dataset and its metadata is `https://biodiversitygenomics.net/1M_insects/`.

S3 Ethics and responsible use

The BIOSCAN project started by the International Barcode of Life (iBOL) Consortium, has collected a large dataset of hand-labelled images of insects. Each record is taxonomically classified by human experts, and accompanied by genetic information.

The publication of the BIOSCAN-1M Insect dataset is a common effort made by researchers from the University of Waterloo, Simon Fraser University, Aalborg University, Dalhousie University and the University of Guelph with support from the Vector Institute for Artificial Intelligence, Alberta Machine Intelligence Institute, Pioneer Centre for AI, and the Centre for Biodiversity Genomics.

The availability of the BIOSCAN-1M Insect dataset presents an immense opportunity for scientific advancement and understanding of insect biodiversity. However, it is important to emphasize the ethical and responsible use of this data.

First and foremost, researchers and institutions must prioritize the protection of individuals' privacy and adhere to data protection regulations and guidelines. To our knowledge, there is no personal or identifiable information in the dataset. However, any such information associated with the dataset should be treated with utmost care and reported to the authors.

Furthermore, the researchers and organizations who utilize the BIOSCAN-1M Insect dataset should ensure transparency in their methodologies and practices. This includes clearly stating the purpose of their research, obtaining informed consent when applicable, and maintaining integrity in the interpretation and reporting of the results.

The responsible use of the BIOSCAN-1M Insect dataset entails promoting open collaboration and sharing of knowledge within the scientific community. Researchers should foster an environment that encourages an exchange of ideas, methodologies, and findings while giving credit to the original dataset creators. It is essential to acknowledge and respect the contributions of the human experts who hand-labelled the images by taxonomically classifying specimens. Proper attribution and recognition should be given to these individuals, as their expertise and efforts are instrumental in the creation and accuracy of the dataset.

S4 Dataset availability and maintenance

The BIOSCAN-1M Insect dataset and all its content described in the previous sections are available on a GoogleDrive folder named **1M_Image_project**. To access the BIOSCAN-1M Insect dataset, please visit `https://biodiversitygenomics.net/1M_insects/`. We've published a code repository for dataset manipulation, including tasks like downloading dataset packages, image and metadata reading, image cropping, dataset subsampling, partitioning into train, validation, and test sets, and running the classification experiments presented in the BIOSCAN-1M Insect paper. To access the BIOSCAN-1M code repository, please visit `https://github.com/zahrag/BIOSCAN-1M`.

S5 Licensing

Table S1 shows the copyright associations related to the BIOSCAN-1M Insect dataset with the corresponding names and contact information.

Table S1: Copyright associations related to the BIOSCAN-1M Insect dataset

Copyright Associations	**Name & Contact**
Image Photographer	CBG Robotic Imager
Copyright Holder	CBG Photography Group
Copyright Institution	Centre for Biodiversity Genomics (email:CBGImaging@gmail.com)
Copyright License	Creative Commons-Attribution Non-Commercial Share-Alike (CC BY-NC-SA 4.0)
Copyright Contact	collectionsBIO@gmail.com
Copyright Year	2021

S6 Experiment details and results

S6.1 Backbone models

We utilized two distinct pretrained backbone models for our experiments with the BIOSCAN-1M-Insect dataset. A comprehensive comparison between these models is presented in this section and in Table S2.

ResNet-50 [3] is a deep convolutional neural network, which includes residual blocks that allow for the training of very deep networks without falling into the vanishing gradient problem. ViT-B/16 [8, 2] signifies that the ViT model is designed to process images with a resolution of 224x224 pixels. Each image is divided into smaller patches of size 16x16 pixels, which are then fed into the transformer layers. Each transformer layer includes multi-head self-attention mechanisms and feed-forward neural networks.

Table S2: A comparison between the two pretrained backbone models used in our experiments: ResNet-50 and the ViT-Base-Path16-224. CNN and FC denote Convolutional Neural Network, and Fully Connected layers, respectively.

Features	**ResNet-50**	**ViT-B/16**
Layers	50	12
Based Networks	CNN, Pooling and FC	Transformer
Number of parameters	25.6 M	86 M

Overall ResNet-50 has a deeper architecture with more layers than ViT-B/16. This depth can enable it to learn hierarchical features in the data, while ViT's strength lies in capturing relationships between patches by applying self-attention mechanisms, which enables it to capture long-range dependencies in images thus making it suitable for both local and global context understanding. Moreover, due to its transformer architecture, ViT can parallelize training more effectively, which can result in faster convergence times despite its higher number of parameters.

S6.2 Validation results

Table S3 shows the performance of all 24 experiments conducted with 3 different seeds using the validation set. According to the validation results, ViT-B/16 with the Cross-Entropy loss function consistently outperforms other models.

Comparing Focal loss to Cross-Entropy, we found that Cross-Entropy produced slightly better results. This could be due to insufficient fine-tuning of Focal loss hyperparameters (alpha and gamma). Furthermore, addressing class imbalance could involve selectively oversampling the less frequent classes during training. This strategy boosts their representation in the training process. For Focal loss, limited exposure to rarer classes might hinder the effectiveness of the re-weighting mechanism.

The presented results of table S3 depict the mean accuracy across various seeds, accompanied by the standard deviation from the average values of each model. The outcomes reveal a notable consistency in the performance of almost all models.

The six models highlighted in bold in Table S3 are used for inference in the test experiments and to report the final results. Pretrained classification checkpoints of these six models, which achieved the best validation accuracy, are available in the GoogleDrive project folder under the directory named **BIOSCAN_1M_Insect_checkpoints**.

S6.3 Confusion Matrix

For an in-depth analysis of the performance of models trained under various configurations, we provide detailed Confusion Matrices for the classification experiments conducted at the order and family levels. These experiments were carried out using the model employing ViT-B/16 and the Cross-Entropy loss function. The evaluation was performed on the test set of the Large dataset. You can refer to Figures S1 and S2 for a visual representation of the respective Confusion Matrices.

Table S3: The table displays Micro-Average-Top-1 and Macro-Average-Top-1 validation accuracy across 24 experiments conducted using 3 distinct seeds. These experiments encompass varying data sizes (Small, Medium, Large), loss functions (Cross-Entropy, Focal), and pretrained backbone models (ResNet-50, ViT-B/16). The experiments utilize consistent hyperparameters and extend to both Insect-Order and Diptera-Family classification levels.

			Micro-Top-1		**Macro-Top-1**	
Dataset	**Backbone**	**Loss Fn**	**Insect-Order**	**Diptera-Family**	**Insect-Order**	**Diptera-Family**
Large	ResNet-50	CE	**99.65**±0.10	97.30±0.02	86.26±0.30	89.98±0.27
		Focal	99.62±0.06	97.15±0.00	84.66±0.21	89.42±0.58
	ViT-B/16	CE	99.58±0.21	**97.67**±0.01	**87.36**±1.20	91.47±0.31
		Focal	99.52±0.27	97.58±0.02	85.80±1.75	**91.54**±0.21
Medium	ResNet-50	CE	98.98±0.04	96.24±0.05	87.30±1.29	91.24±0.33
		Focal	98.85±0.04	95.92±0.04	86.61±0.51	90.64±0.22
	ViT-B/16	CE	**99.14**±0.04	**96.74**±0.06	**88.40**±1.17	**92.83**±0.16
		Focal	99.11±0.04	96.55±0.02	86.75±1.46	92.23±0.35
Small	ResNet-50	CE	97.79±0.08	93.23±0.24	87.37±0.56	91.43±0.36
		Focal	97.62±0.09	92.57±0.07	86.55±0.60	90.68±0.20
	ViT-B/16	CE	**98.34**±0.10	**94.46**±0.15	**88.74**±1.16	**92.93**±0.33
		Focal	98.26±0.03	94.42±0.04	88.61±0.09	92.92±0.16

Table S4: The table presents the Micro-F1-Score and Macro-F1-Score of our trained models, evaluated on the validation set, and then averaged across different seeds.

			Micro-F1-Score		**Macro-F1-Score**	
Dataset	**Backbone**	**Loss Fn**	**Insect-Order**	**Diptera-Family**	**Insect-Order**	**Diptera-Family**
Large	ResNet-50	CE	99.67±0.07	97.44±0.03	87.50±0.04	90.73±0.23
		Focal	99.63±0.06	97.28±0.01	84.66±0.87	90.22±0.26
	ViT-B/16	CE	**99.68**±0.06	**97.68**±0.01	**87.94**±1.59	**92.01**±0.12
		Focal	99.62±0.14	97.58±0.01	86.98±2.06	91.91±0.22
Medium	ResNet-50	CE	99.00±0.04	96.26±0.06	87.43±1.02	92.61±0.03
		Focal	98.88±0.05	95.98±0.05	86.77±1.21	91.77±0.5
	ViT-B/16	CE	**99.14**±0.04	**96.75**±0.04	**89.33**±1.21	**93.58**±0.08
		Focal	99.12±0.03	96.56±0.01	87.52±1.01	93.20±0.21
Small	ResNet-50	CE	97.84±0.11	93.27±0.35	87.89±0.77	92.10±0.51
		Focal	97.63±0.04	92.78±0.06	87.52±0.62	91.37±0.12
	ViT-B/16	CE	**98.31**±0.10	**94.54**±0.16	**88.92**±0.74	**93.56**±0.23
		Focal	98.28±0.03	94.42±0.06	88.36±0.28	93.48±0.05

S6.4 Qualitative analysis

In this section, we provide a qualitative analysis of the performance results from the order classification experiment on the Small dataset. We aim to shed light on the misclassifications made by our model by visually examining some of the misclassified images.

Surprisingly, roughly 57% of the misclassifications in order-level classification experiments on the Small dataset, using 10,000 test samples, can be traced back to low-quality insect images. This is evident when examining the examples shown in Figure S3, where image quality hampers accurate classification. A similar analysis revealed that approximately 45% of the misclassifications in order-level experiments with the Large dataset, using 225,660 test samples, were also attributed to low-quality insect images.

Another observation shows that a large proportion of misclassifications are the insects belonging to different orders that are all incorrectly classified as one of the dominant classes of our Small dataset. As an example, there are 16.2% of misclassifications in order-level classification experiments on the Small dataset where insects belonging to different orders are all incorrectly classified as Diptera (flies or mosquitoes), which is the dominant class. This observation, illustrated in Figure S4, highlights specific instances where the model struggles to differentiate between various orders and tends to favour Diptera as the predicted classification.

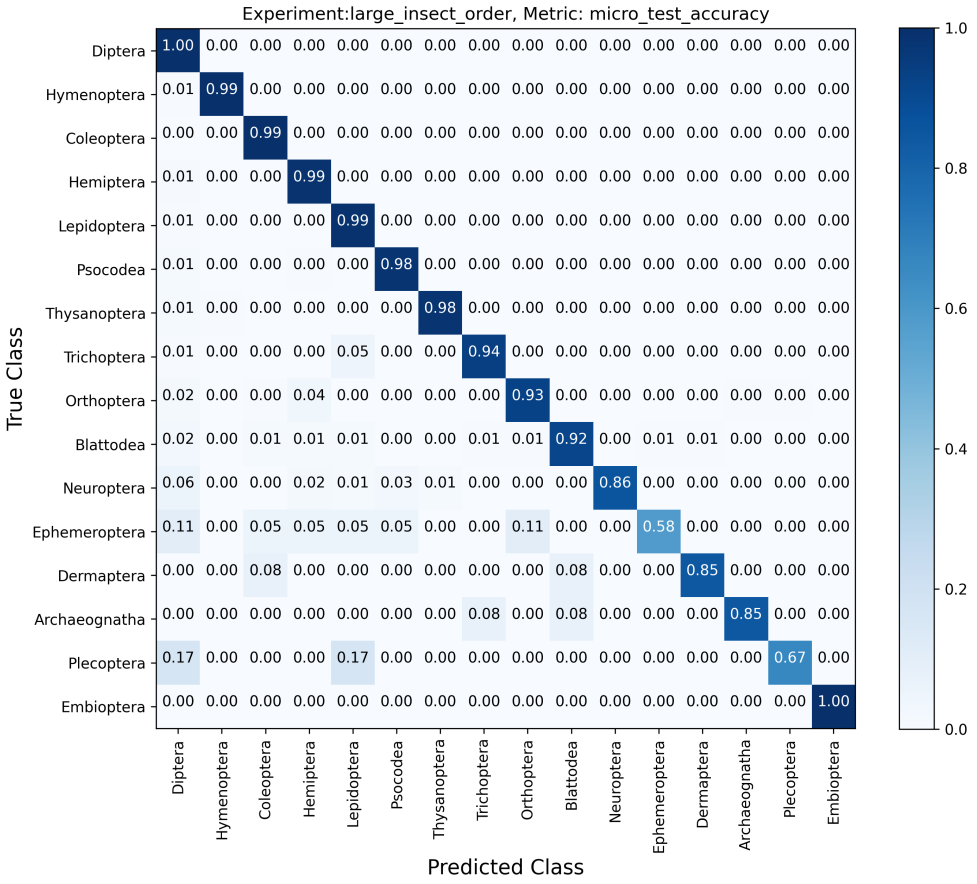

Figure S1: The Confusion Matrix displays the per-class predictions of the order level classification using the Large dataset of the BIOSCAN-1M Insect dataset. The test evaluation is performed on the best model achieved from validation performance results presented in Table S3.

By examining these qualitative analyses, we gain insights into the challenges faced by our model in correctly classifying insect orders, especially when dealing with low-quality images and distinguishing between similar orders when these orders have a low number of training samples.

Our classification experiments have an important application in data cleaning. By identifying low-quality images that have been misclassified, we can effectively detect and remove them from the dataset. This process plays a crucial role in enhancing the overall quality and reliability of the data, as it ensures that only high-quality images of insects are retained.

Furthermore, our classification experiments also enable us to validate the taxonomic classifications performed by human experts. By examining instances of false predictions, we can investigate whether a sample has been incorrectly annotated, providing valuable insights into the accuracy of the taxonomic classification process.

S6.5 Discussion

S6.5.1 Dataset: Generation, Curation and Growth

The BIOSCAN project is currently in its initial stages, with its primary goal being the facilitation of a global biodiversity assessment. In this section, we clarify certain aspects and procedures accomplished in the generation of the BIOSCAN-1M Insect dataset.

All samples of the BIOSCAN-1M Insect dataset were processed at one facility using the same workflows and imaging equipment. This should exclude all potential biases with respect to data collection.

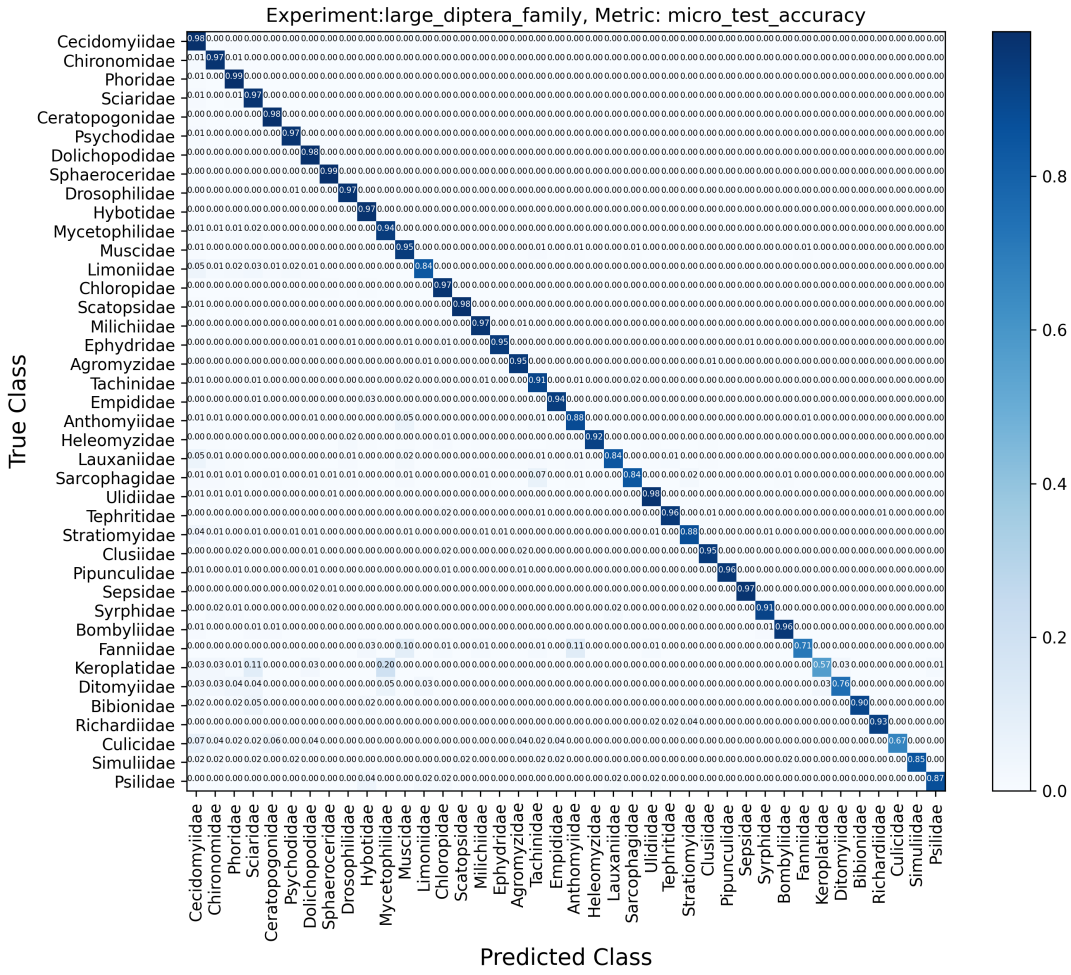

Figure S2: The Confusion Matrix displays the per-class predictions of the family level classification using the Large dataset of the BIOSCAN-1M Insect dataset. The test evaluation is performed on the best model achieved from validation performance results presented in Table S3.

Regarding the dataset labelling procedure, the order-level classification is conducted by taxonomists and entomologists primarily relying on morphology rather than barcode matching. For family and finer-grained classifications, a combination of approaches may be employed, often supported by BIN assignment.

The annotators responsible for labelling the data were personnel affiliated with the Centre for Biodiversity Genomics, including technicians and research scientists, all engaged in full-time roles at the institute and receiving equitable compensation. A diverse team of approximately 15 to 20 individuals participated in providing labels for the dataset. These labels were derived from the images, forming the foundation for establishing higher-level taxonomies such as order and family. Additionally, the annotators conducted visual examinations of specimens for finer-grained taxonomic ranks, utilizing primary literature as a reference where feasible. Taxonomic experts (human professionals) engaged in the barcoding process have varying levels of involvement, which tends to increase when the placement of a specimen is more contentious. Additionally, these experts are responsible for describing species, a task not handled by the machine. They also supply the reference ID that enables us to establish matches.

Several factors contribute to why most samples of the dataset are classified only to the family and finer-grained classes are not provided, with the primary one being the time constraints associated with completing the assignment. The complexity of the task arises from the fact that relying solely

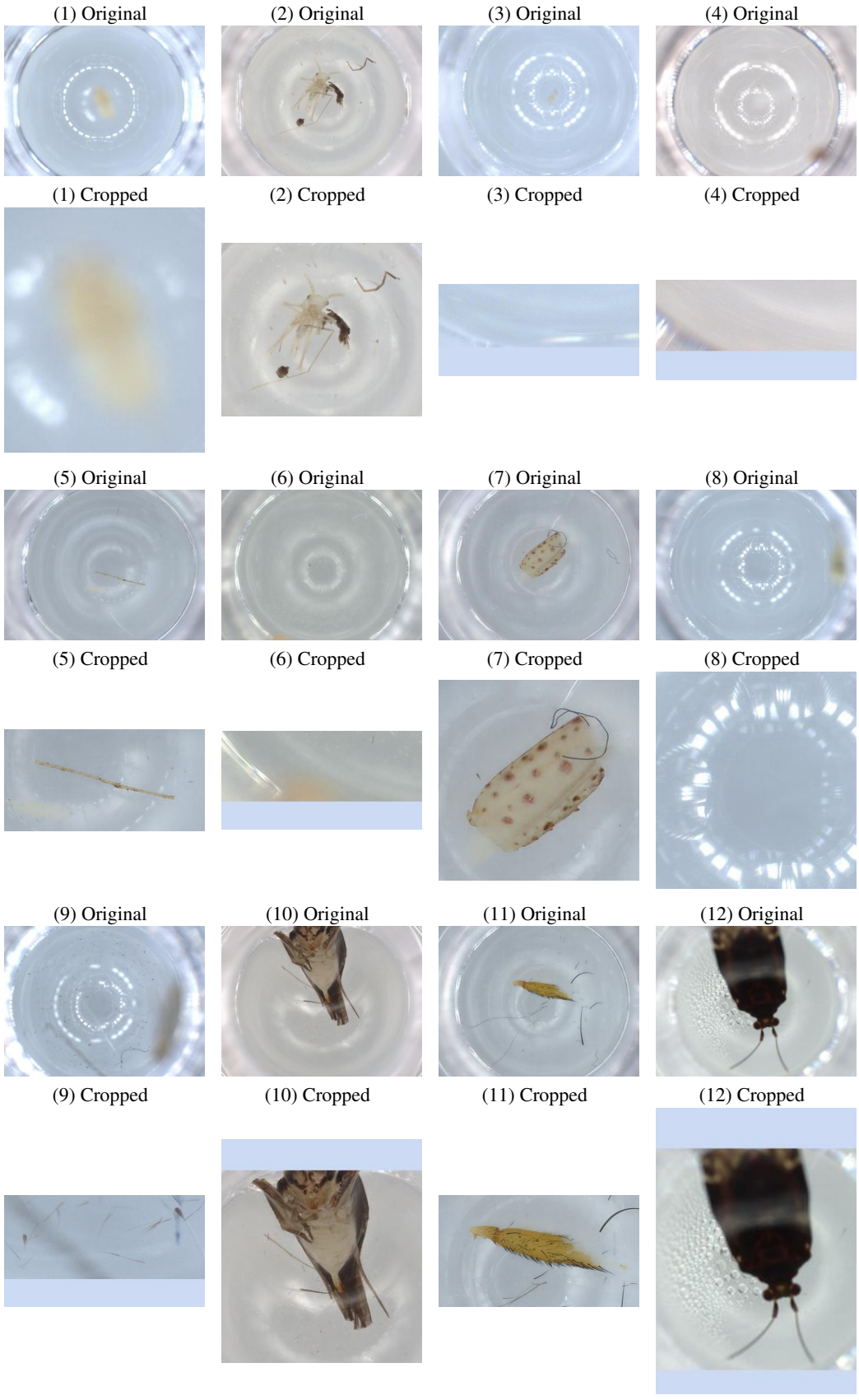

Figure S3: Examples of misclassifications caused by low quality images photographed from insects.

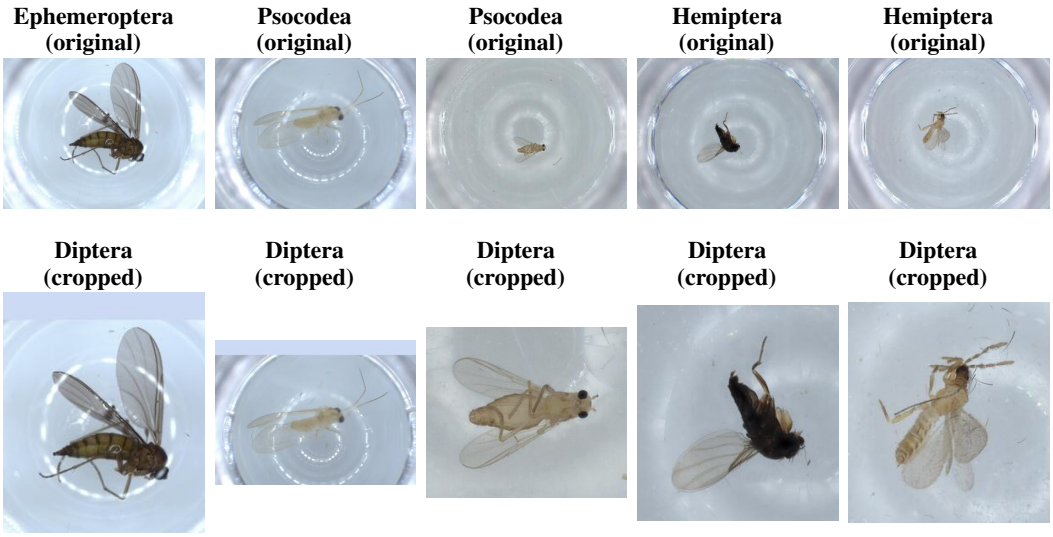

Figure S4: Examples misclassified as the dominant class Diptera (flies).

on a barcode is often insufficient due to potential ambiguities, as discussed earlier. Each sample's labeling requires verification through visual inspection (images), or in some cases, examination of the original specimen, before proceeding with further classification. This process is not easily scalable, prompting the adoption of BINs as a species proxy.

However, the process utilized to expand the dataset remains consistent with the methodology employed for the BIOSCAN-1M Insect dataset. The Dataset will be retrained at regular intervals and older versions archived and stored on Zenodo and date stamped. Similarly, we will use GitHub's release mechanism to version the accompanying code.

S6.5.2 Application: Model and Tasks

In this article, the baseline problems and methods we explored were chosen to be simple and accessible and as a result, limited. They are not the focus of the paper as our primary focus is to release the dataset and showcase its inherent potential. We expect future works will use the dataset for interesting problems such as hierarchical classification, zero-shot classification, set-valued classification and methods that improve performance in the fine-grained and long-tailed label regime.

We believe that the most promising methods will be hierarchical classifiers that yield uncertainty estimates over multiple taxonomic levels. Improving performance on minority classes and reliably delineating novel operational taxonomic units is also important. To get there, we believe the most promising areas of investigation from the ML side will be semi-parametric methods that use reference libraries at test time, set-valued classification as a natural means of expressing uncertainty, and zero-shot classification.

The utilization of the BIOSCAN-1M Insect dataset in conjunction with other large biological datasets from various domains becomes feasible by harnessing the preprocessing module proposed in this paper. By employing tools like our cropping tool and applying machine learning techniques for domain adaptation/generalization, one can capitalize on the capabilities of a pre-trained model on BIOSCAN-1M Insect images to effectively tackle classification challenges in out-of-distribution scenarios.

Overall, we believe that the unique annotation and metadata including the DNA barcodes will prompt interesting multimodal strategies. We intend to enhance our approach by incorporating DNA barcode sequences and utilizing Barcode Index Numbers (BINs). This strategic direction aims to effectively tackle the limitations associated with the current taxonomic labels of the images. Notably, the utilization of BINs holds promise as each image is inherently associated with a distinct and unique BIN.

Figure S5: Examples of images used to adapt our cropping tool. We include variations of insects' size, color, position and shape.

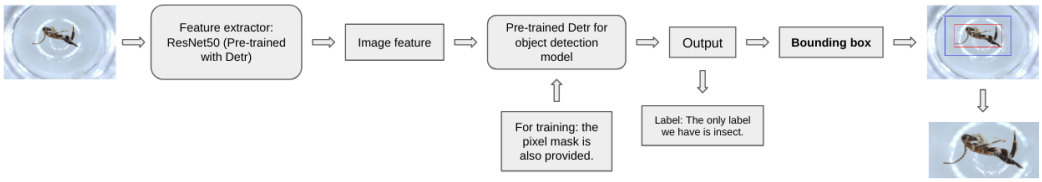

Figure S6: Our DETR [1] based cropping tool takes an input image, extracts features using a ResNet50 [3] backbone, and extracts a tight fitting bounding box for the insect (see red box). We then extend the bounding box (see blue box) to obtain the final cropped image. We use a DETR model pretrained on MSCOCO [5]. To fine-tune the DETR model, we annotate a small set of insect images with their segmentation mask.

S7 Preprocessing: Cropping tool

Our observations showed significant improvement in processing time when we used cropped images rather than original ones. However, cropping is a challenging problem since insect images have varying shapes, sizes, colors which is also shown in Figure S5. The illumination, background color and surface are not the same across the original images.

Furthermore, there are cases in the original images that the insect is photographed in pieces and in such cases the cropping is quite challenging especially when the insect is small, and its less discriminative body parts like legs are distant from the main body so these pieces could be cropped instead.

To address these issues more effectively, we have developed a tool based on the DETR model for automatic identification and cropping of the main insects in images. The primary objective of this tool is to facilitate data storage and subsequent research, such as neural network training. The tool uses the DETR model to accurately locate the main insects in images and crop accordingly. By removing irrelevant background information, the tool optimizes storage space and reduces the time spent on data management. Additionally, the cropped images can be effectively used for tasks such as image classification through neural network training, leading to improved performance in the following image classification task. Our crop tool checkpoint is available in the GoogleDrive project folder under the directory named **BIOSCAN_1M_Insect_checkpoints**.

S7.1 Approach

The cropping tool consists of first detecting a tight bounding box for the insect in the image using an object detector and then cropping the image by extending the bounding box. We show an overview of the cropping tool in Figure S6. To accurately locate the insect in the image, we chose the DETR [1] model which has excellent performance in the task of object detection and the corresponding pre-trained ResNet-50 [3] as the feature extractor. At the beginning, the CNN-based feature extractor extracts a set of image features that are fed into a transformer-based encoder-detector. The detector takes a set of learned positional embeddings as object queries and uses them to attend to the encoder outputs. Each of the output decoder embeddings is then passed to a shared FFN which will predict whether there is "no object" or a detected object with its class and bounding box. Each bounding box is parameterized as (cx, cy, w, h) where (cx, cy) is the center of the bounding box, and (w, h) is the width and height of the box, all normalized to 1.

The DETR network is trained by optimizing a bipartite set loss that matches detected boxes with the ground-truth boxes using the Hungarian algorithm to minimize the overall matching loss between the matched pairs. The pairwise matching loss is a combination of the classification loss and a box regression loss (the bounding box loss is included only when the detected box matches a ground truth box that corresponds to an object, and is a weighted combination of GIOU [7] and L1 loss between the bounding box parameters). In our case, we have only one object class ("insect") so the classification reduces to a binary classification between "insect" and "no object".

Note that other than the ground-truth bounding box, for training the DETR model of the cropping tool, the pixel mask of the insect in the image is also required for the training. This pixel mask is not needed during the inference phase.

Training details. We start with a DETR model pretrained on MSCOCO [5] and fine-tune it on our dataset. We use the AdamW [6] optimizer with a learning rate of 0.0001, weight decay of 0.0001 and a batch size of 8. We train for 10 epochs. On an RTX 2080 Ti with 4 workers, for 1,000 images, training takes 1.5 minutes per epoch and a total of 15 minutes for 10 epochs.

The original DETR is trained with images resized to fit within an 800 ×1,333 tensor. We match that and resize our image (preserving the aspect ratio) so that the shortest side is less than 800 and the longest side is less than 1333. No data argumentation is applied during training.

Cropping. In the cropping phase, With the predicted bounding box (the red bounding box in Figure S6), we can choose to enlarge it using a certain method to include more details or meet specific image aspect ratio requirements. By default, we will choose 0.4 times the longest edge as the target and extend this size in both height and width to produce the final cropping bounding box (the blue bounding box in Figure S6).

To crop the image, we run our fine-tuned DETR model on the input image to identify the tight bounding box around the insect. We assume that each image contains one insect of interest, and during cropping, we take the predicted bounding box with the highest probability that is higher than 0.5. Before cropping, we extend the predicted bounding box by a fixed ratio $R = 1.4$ of the size of the tight bounding box. We extend the height and width by the same number of pixels by computing the extended size as: $\text{ExtendSize} = (R - 1) \times \max(\text{width}, \text{height})$.

If the bounding box is at the edge of the original image, we pad the image by adding pixels of maximum intensity to match the white background. In this way, even if the predicted bounding box does not encompass all the details of the insect, we can still include the entire insect in the cropped image. Furthermore, this maintains a more square aspect ratio, which facilitates downstream tasks such as image classification.

Runtime. The cropping tool can be run in CPU or GPU mode. On a Linux machine with 16 cores and running 4 workers, using CPU only, 10 k images can be cropped in 2 hours and 40 minutes (images loaded and written to local SSD). Using an RTX 2080 Ti GPU, 10K images can be cropped in 30 minutes on the same machine.

S7.2 Data

We developed our tool on two sets of images of insects that are pinned (INSECTS-PINNED) and insects in wells (INSECTS-WELL). Using the Toronto Annotation Suite (TORAS) [4], we annotate each with their segmentation mask. For each set, we annotated a large (1,000 images) and a small

Figure S7: Typical instances of annotated IW (left two columns) and IP (right two columns) images. To obtain an accurate bounding box in reasonable annotation time, we focused on drawing the external outline of the main insect only excluding the small spaces between its legs. Small parts of the insect that are far away from the main body (e.g. the small leg in the first image) are also not included.

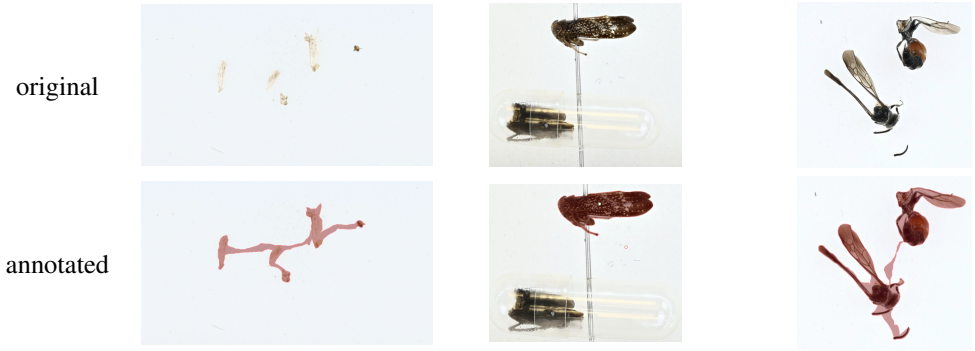

Figure S8: Examples of special annotation cases. Left: for an insect that is broken into multiple parts with even size, we create a mask that covers all of the parts of the insect. The ideal mask should contain minimal background, and keep the edge of the mask as close to the insect's edge as possible (left, right). Middle: for two insects where one is in the container and the other is not, we annotate the insect that is not in the container. Right: for a split insect we annotate all parts.

(100-150 images) training set and another small set for evaluation. The annotation was done by three volunteers and took a total of 4 hours for 1,000 images. The two sets of images are described below (see Figures S7 and S8 for example images and annotated masks):

INSECTS-PINNED (IP). The insect is pinned in these images (or has a pin near it) with a fairly clean white background. The images are taken by a Digital SLR camera (Canon) mounted on a motor-drive positioning system (OpenBuilds ACRO) equipped with stepper motors and a motion control system. Pinned specimens are arrayed in sets of 96 (8 ×12 array) in a large enough distance between them to avoid including parts of neighbouring specimens in the image frame. For this set, we collected 1,000 images to form the large training set (IP-1000-train), 100 images for the small training set (IP-100-train), and another 100 images for the validation set (IP-100-val).

INSECTS-WELL (IW). In these images, the insects are placed in a well. Here the images tend to have a less clean background due to the glass and uneven reflected light. The images are taken using a Keyence VHX-7000 Digital Microscope system with a fully integrated head and automatic stage that permits high-resolution (4 k) microphotography of individual specimens. Because its scanning stage can hold a 96-well plate, the system automatically acquires a high-resolution image of each specimen by controlling movements in the X-Y plane. Also, its capability to control the z-axis position of the stage with a precision of 0.1 m allows it to photograph each specimen at multiple heights before rapidly compiling these images into an in-focus image (depth stacking). For this set, we collected 1,000 images to form the large training set (IW-1000-train), 150 images for the small training set (IW-150-train), and another 150 images for the validation set (IW-150-val).

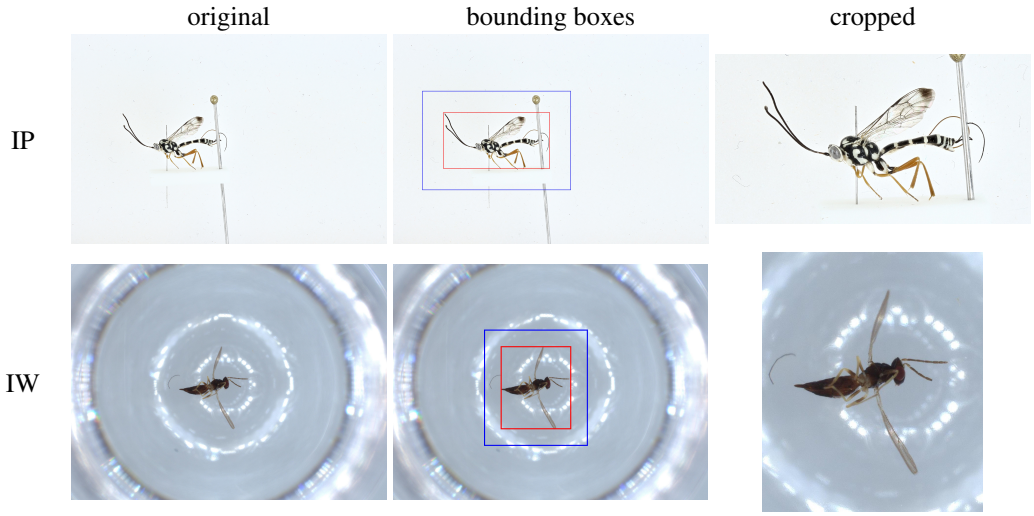

Figure S9: Cropping examples of images from INSECTS-PINNED (IP) and INSECTS-WELL (IW) with the original image, image with detected bounding boxes in red, extended bounding boxes in blue, and final cropped image.

Table S5: The Average Precision (AP) and Average Recall (AR) were computed on the IW-150 val and IP-100-val datasets using the DETR model, which was pre-trained with different training splits.

	INSECTS-PINNED-100-Val		INSECTS-WELL-150-Val	
Training data	AP[0.75]	AR[0.50:0.95]	AP[0.75]	AR[0.50:0.95]
IP-100	0.910	0.893	0.543	0.729
IP-1000	0.949	**0.918**	0.415	0.587
IW-150	0.415	0.587	0.801	0.802
IW-1000	0.665	0.695	0.872	0.835
IP-1000 + IW-1000	**0.964**	0.907	**0.901**	**0.885**

Note that the BIOSCAN-1M Insect Dataset consists only of insects in wells. We include the insects with pins to extend the usefulness of the cropping tool for a broader spectrum of backgrounds that may appear in the process that specimens are acquired in the larger BIOSCAN project.

During annotation, we focus on masking the main insect and we exclude small broken pieces of the insect that are far from its body (see Figure S7). There are also challenging cases where the insect may be broken into pieces or there are multiple insects (see Figure S8). For insects that are broken into multiple pieces of similar size, we create a mask that covers all the pieces. When there are multiple insects, we mask only the central insect.

S7.3 Experiments

S7.3.1 Metrics

The metrics we used are the Average Precision (AP) and the Average Recall (AR) with the IOU of the bounding box equal to [0.75] and [0.50:0.95], as they measure the precision and recall aspects of detection performance. AP reflects the accuracy of detection by considering the overlap between predicted and ground truth bounding boxes, while AR assesses how well the system captures all the ground truth objects.

S7.3.2 Cropping results

We show examples of cropped images in Figure S9. The images show the accurate identification of the insect subject by the DETR model (red bounding box) and the extended bounding box (blue bounding box) used for cropping. In Figure S10, we show cases where the predicted bounding boxes have an intersection over union (IoU) with the ground truth bounding boxes (green bounding box) less than 0.85. From these examples, we observe that the antennae of certain insects and the presence

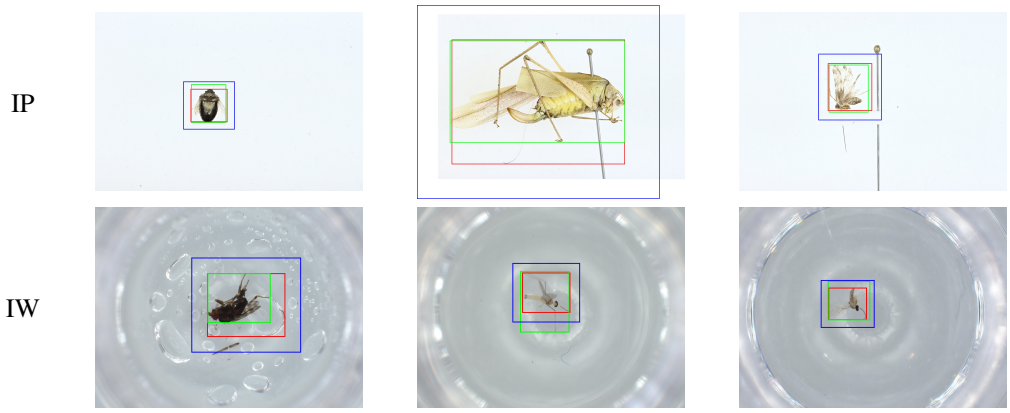

Figure S10: Examples of imperfect insect detection (IOU < 0.85), with ground-truth bounding box in green, detected bounding boxes in red, and extended in blue. In the second image of IP, note that we extend the image with the white background to fit the bounding box that escapes the original image boundaries.

Table S6: Comparison of classification accuracy results on original images vs. cropped images. Both are resized to 256 on the smaller dimension. Overall, we find the cropped images yield slightly higher accuracy.

	Order-level				Family-level			
	Micro-average		Macro-average		Micro-average		Macro-average	
Image type	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
original	0.9626	0.9970	0.8218	0.9964	0.9248	**0.9802**	0.9109	**0.9730**
cropped	**0.9786**	**0.9976**	**0.8757**	**0.9980**	**0.9314**	0.9786	**0.9154**	0.9728

of cluttered backgrounds sometimes can create disturbances to our fine-tuned DETR model. However, by expanding the predicted bounding boxes, we are still able to capture all the desired information within the cropped images.

To evaluate the performance of our cropping tool with different amount and type of data, we trained the DETR model with 5 training splits (IP-100, IP-1000, IW-150, IW-1000 and IP-1000+IW-1000), and evaluated these models on two validation splits(IP-100-val and IW-150-val). Overall, from Table S5, we see that using the mixed training split with 1000 images from IP and 1000 images from IW results in the highest accuracy. This is the model that we use for cropping the images in the BIOSCAN-1M Insect Dataset.

S7.3.3 Insect classification using cropped images

We further evaluate the effectiveness of our auto-cropping tool on a downstream task: insect image classification at the order/family level. In Table S6 we compare the classification performance of the original vs. cropped images on the BIOSCAN small dataset following the training setup we described in the main paper. We use the ResNet-50 backbone with cross-entropy loss and train with the AdamW optimizer with a learning rate of 0.001 and momentum of 0.9 for 100 epochs for order-level classification and 40 epochs for family-level classification. All images are resized such that the shorter side has a size of 256. During training, we apply a random horizontal flip with a probability of 0.5, and random crops of 224×224 are extracted and fed into the backbone to extract image features. During inference, the center 224×224 crop is extracted. We measure the micro and class macro average top-K accuracy at $K = 1$ and $K = 5$.

From Table S6, we see that in most cases, using cropped images to perform training results in higher classification accuracy. In the cases where the original image type outperforms the cropped type, the difference is small.

To further compare the difference between using original images and cropped images for training, we also compare the loss curve during training with original and cropped images. By comparing

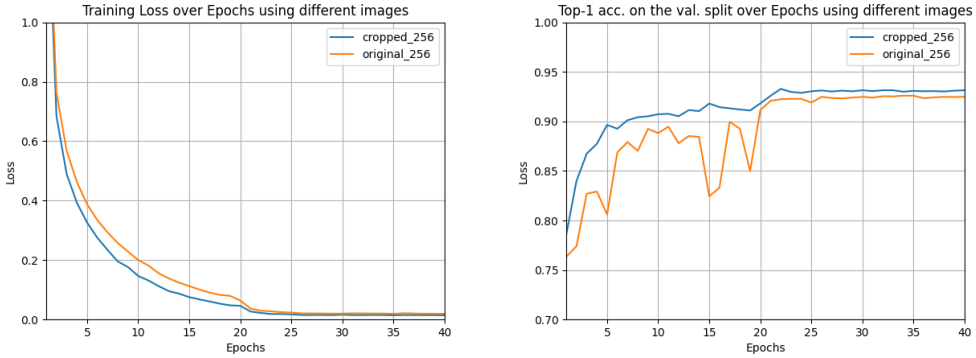

Figure S11: The training loss and Top-1 accuracy on the validation split during the training of family-level classification of images of insects using cropped (blue) and original (orange) images. Both are resized to 256 on the shorter side.

the loss at epochs 10, 15 and 20, we see that using the cropped images can help the model converge faster. Using the cropped images also yields higher top-1 accuracy on the validation split.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proc. of the European Conference on Computer Vision*, pages 213–229. Springer, 2020.

[2] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] Amlan Kar, Seung Wook Kim, Marko Boben, Jun Gao, Tianxing Li, Huan Ling, Zian Wang, and Sanja Fidler. Toronto annotation suite. `https://aidemos.cs.toronto.edu/toras`, 2021.

[5] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proc. of the European Conference on Computer Vision*, pages 740–755. Springer, 2014.

[6] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[7] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019.

[8] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Zhicheng Yan, Masayoshi Tomizuka, Joseph Gonzalez, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*, 2020.