Journal of the Royal Society of New Zealand

ISSN: 0303-6758 (Print) 1175-8899 (Online) Journal homepage: www.tandfonline.com/journals/tnzr20

Automatic species identification from images for Aotearoa

Hongyu Wang, Paul Schlumbom, Eibe Frank, Varvara Vetrova, Geoffrey Holmes, Bernhard Pfahringer, Nick Lim & Albert Bifet

To cite this article: Hongyu Wang, Paul Schlumbom, Eibe Frank, Varvara Vetrova, Geoffrey Holmes, Bernhard Pfahringer, Nick Lim & Albert Bifet (16 Jul 2025): Automatic species identification from images for Aotearoa, Journal of the Royal Society of New Zealand, DOI: 10.1080/03036758.2025.2525161

To link to this article: https://doi.org/10.1080/03036758.2025.2525161

Published online: 16 Jul 2025.

JOURNAL OF THE ROYAL SOCIETY OF NEW ZEALAND
https://doi.org/10.1080/03036758.2025.2525161

Taylor & Francis
Taylor & Francis Group

REVIEW ARTICLE

OPEN ACCESS Check for updates

Automatic species identification from images for Aotearoa

Hongyu Wang[a], Paul Schlumbom[b], Eibe Frank [a], Varvara Vetrova[c], Geoffrey Holmes[a], Bernhard Pfahringer [a], Nick Lim [b] and Albert Bifet[b]

[a]School of Computing and Mathematical Sciences, University of Waikato, Hamilton, New Zealand; [b]Artificial Intelligence Institute, University of Waikato, Hamilton, New Zealand; [c]School of Mathematics and Statistics, University of Canterbury, Christchurch, New Zealand

ABSTRACT

Image classification for species identification has applications in areas such as conservation and education. Given New Zealand's geographic isolation and the relatively small number of species present on its islands, there is an opportunity to apply machine learning to enable accurate automatic species identification for Aotearoa, even on mobile devices without Internet access. We present neural network-based image classification models trained to classify organisms present in New Zealand. The data for model development and evaluation, obtained from the crowd-sourcing website iNaturalist, comprises 14,991 species, including 6,216 Animalia, 6,173 Plantae, and 2,407 Fungi species, alongside a small set of observations of Bacteria, Chromista, Protozoa, and Viruses. It contains organisms observed in the natural environment as well as captive and cultivated organisms. The trained models achieve over 76% classification accuracy across all species and produce class probability estimates, calibrated using temperature scaling, that can be used to gauge confidence in their classifications. Input attribution methods can be used to interpret a model's inferences by highlighting its areas of focus on images. The models are available to the public as downloadable model files and as part of both web and mobile applications for species identification that are distributed as open-source software.

ARTICLE HISTORY
Received 18 October 2024
Accepted 18 June 2025

KEYWORDS
Image classification; species identification; convolutional neural networks; computer vision; transfer learning; finetuning

Introduction

New Zealand has a diverse ecological system. Many indigenous species of flora, fauna, and fungi are unique to its islands, but there are also many exotic species: some co-exist harmoniously, while others harm the country's biodiversity and economy. There is strong academic, commercial, and public interest in studying and conserving the ecological system of New Zealand (Dymond 2013; Reid and Rout 2020).

Automatic classification of local species, both indigenous and exotic, has applications ranging from environmental conservation, e.g. invasive species identification, to

CONTACT Eibe Frank eibe@waikato.ac.nz

education, e.g. biodiversity awareness. Species identification software can help local conservation efforts and facilitate public engagement. It also contributes to the global research ecosystem for environmental conservation and biodiversity. In this paper, we build on work considering the use of machine learning to classify images of New Zealand organisms, such as Mo et al. (2017) and Vetrova et al. (2018).

Machine learning requires training data. We use crowd-sourced images and labels from the iNaturalist (iNaturalistorg 2021) platform, which enables naturalists, citizen scientists, and biologists to share observations of organisms and discuss their findings. Thousands of photographs taken in New Zealand have been uploaded and classified by the iNaturalist community, which includes Mātaki Taiao – iNaturalist NZ operated by the New Zealand Bio-Recording Network Trust.

Correct labels in the training and validation data used for machine learning are important to ensure high-quality models that produce accurate classifications. The Global Biodiversity Information Facility (GBIF) (GBIForg 2021) has compiled a dataset from submissions to the iNaturalist website that have been assigned to the research-grade category by iNaturalist and are available under an appropriate Creative Commons license. By querying the database with location information, a dataset of observations made in New Zealand can be obtained.

A drawback of the research-grade dataset is that it contains only observations of organisms in the natural environment, which means it excludes domesticated fauna and cultivated flora. Observations of captured and cultivated organisms can be obtained from iNaturalist directly using its export tool (iNaturalistorg 2021). We merge the captive/cultivated observations into the research-grade dataset for our experiments, as we consider classification of captive/cultivated organisms to be of interest to users.

The data available for model development and evaluation comprises 14,991 species, including 6,216 Animalia, 6,173 Plantae, and 2,407 Fungi species, alongside a small set of available observations of Bacteria, Chromista, Protozoa, and Viruses. The dataset is highly imbalanced, with a small number of species containing thousands of images each and thousands of species containing fewer than ten images each, rendering this a challenging problem for machine learning.

Image classification using machine learning is commonly performed using artificial neural networks (Krizhevsky et al. 2012). Deep convolutional neural networks (CNNs) have been shown to be highly effective in image recognition tasks. EfficientNetV2 (Tan and Le 2021) is a carefully adjusted CNN architecture based on the first version of EfficientNet (Tan and Le 2019), achieving state-of-the-art performance at the time of its publication, while being more efficient than a large number of contemporary computer vision models. We use fine-tuned EfficientNetV2 models, named *S* (small), *M* (medium), and *L* (large) after the relative model sizes, performing fine-tuning and validation on mutually exclusive splits of the iNaturalist dataset. In addition to overall accuracy across all classes (i.e. species),[1] we consider per-class (i.e. per-species) accuracy. Furthermore, an abstention mechanism is considered to limit the models' predictions to ones with relatively high certainty when computing accuracy. Different abstention thresholds are evaluated and compared.

Our trained models can be downloaded from https://github.com/Waikato/aotearoa-species-classifier and applied to user-supplied images at https://what-is-this.cms.waikato.ac.nz. The *S* model has also been implemented in a mobile app, enabling on-

device inference without Internet access. This app has been made freely available on both the Apple App Store and the Google Play store under the name 'Aotearoa Species Classifier'.

Related work

We summarise related work on image-based species identification with machine learning performed in the New Zealand context before briefly reviewing the literature pertaining to the EfficientNetV2 model. Beyond the literature we review here, there is a substantial amount of research on image-based detection and/or classification of subsets of species using machine learning, exemplified by work published at the CV4Animals workshop 2025 that formed part of the prestigious computer vision conference CVPR, including work that considers the detection of albatrosses present in New Zealand Rogers et al. (2025).

Automatic species identification from images

Mo et al. (2017) used an InceptionResNetV2 (Szegedy et al. 2017) model pretrained on ImageNet (Russakovsky et al. 2015) and fine-tuned it on the NatureWatch (GBIForg 2021; iNaturalistorg 2021) dataset.[2] The dataset contained 1,214,141 image instances belonging to 19,027 classes after sanitation and was then randomly partitioned into training and validation sets in a 80–20 split. Mo et al. (2017) adopted a multi-view model architecture that makes predictions based on two views of an image: the full image and a central crop of it, as it was reasoned that both the overall shape and the high-frequency details may have a significant impact on the prediction. Auxiliary classifiers were also used to optimise the predictive power of the model's lower layers' feature representations, as was common practice for Inception networks (Szegedy et al. 2017). Hierarchical knowledge transfer was applied during fine-tuning, where the model was first fine-tuned until convergence to predict the kingdom of each instance; subsequently, its targeted taxon level became increasingly fine-grained, e.g. phylum, class, etc., until it was fine-tuned to convergence to predict the species of each instance. The fine-tuned model of Mo et al. (2017) achieved 55.8% species-level accuracy. Automatic specificity adjustments were implemented to predict higher taxon labels if species-level confidence was too low. The model achieved approximately 90% accuracy at an average taxon depth of 5.1, i.e. approximately family level, with automatic specificity control.

Vetrova et al. (2018) performed multi-class and one-class classification of (1) 17 plant species of the cryptic genus Coprosma and (2) 10 species of native and invasive moth species from New Zealand. The images were of laboratory quality, but the species themselves were difficult to distinguish even for expert taxonomists. Additionally, Vetrova et al. (2018) evaluated a simple CNN trained as a Siamese network (Chopra et al. 2005) with triplet margin ranking loss on the moth data.

Contemporary convolutional neural networks

Residual networks (ResNets) are deep neural networks utilising residual connections to facilitate gradient-based optimisation of network parameters (He et al. 2016). Although

problems with vanishing or exploding gradients (Glorot and Bengio 2010) can be addressed by normalised parameter initialisation (He et al. 2015) and intermediate batch normalisation layers (Ioffe and Szegedy 2015), it was observed that neural networks were prone to degrade in performance beyond a certain depth. He et al. (2016) proposed attaching residual connections to a deep network to combat this, enabling optimisation to adopt propagation paths similar to those of shallower networks. It was also shown that bottleneck convolutional blocks with residual links can be used to construct deep CNNs that are computationally efficient.

Tan and Le (2019) proposed EfficientNet, a deep CNN architecture with residual connections that was scaled up in depth, width, and resolution from a baseline network using a simple and effective compound coefficient. The baseline network was obtained using neural architecture search (Zoph and Le 2017; Tan et al. 2019) by optimising both accuracy on validation data and the number of processing operations needed. Tan and Le (2019) derived eight different-sized EfficientNet model from the baseline network, termed 'EfficientNet-Bn' ($n \in [0, 7]$). The EfficientNet models achieved better accuracy-FLOPS trade-off than standard ResNets (He et al. 2016).

Tan and Le (2021) proposed EfficientNetV2 with various improvements. Training bottlenecks were analysed, and Tan and Le (2021) identified and addressed three outstanding issues: (1) exceedingly large image sizes led to slow training, which was addressed by progressively increasing training image size during training and adjusting regularisation in tandem; (2) depthwise convolutions were slow in early layers but effective in later stages, which was addressed by using neural architecture search to determine the best combination of MBConv (Sandler et al. 2018) and Fused-MBConv (Xiong et al. 2021) building blocks; and (3) a non-uniform scaling strategy was applied to gradually scale up later stages more in terms of layers. EfficientNetV2 models achieved state-of-the-art performance on the ImageNet benchmark data while being smaller and faster to train than most of their contemporary competitors.

Training EfficientNetV2 on NZ species data

We first describe how our experimental data was acquired and sanitised before explaining how the pre-trained EfficientNetV2 models were fine-tuned on the data.

Data selection and sanitation

GBIF (GBIForg 2021) hosts a database of research-grade iNaturalist (iNaturalistorg 2021) observations collected globally. This dataset is under the 'CC-BY-NC' license. An observation requires an identification agreed on by the community to qualify as 'research-grade'. By querying the database by country, a subset can be obtained of all observations made in New Zealand. It is worth noting that one observation may contain multiple individual instances, for example, given a bird observed in the wild, multiple photographs may have been taken in quick succession, and the bird's calls may have been recorded as a sound file–all these files are considered to belong to the same observation.

The research-grade data contains only observations made in the natural environment, excluding observations of organisms kept and maintained by humans, such as pets, livestock, and cultivated plants. These observations can be acquired using the iNaturalist

export tool (iNaturalistorg 2021) under the captive/cultivated category. The export tool allows querying by country, and our experiments use only observations made in New Zealand. Observations available via the export tool are under various different licenses, and we only used observations under licenses no stricter than the 'CC-BY-NC' license of the research-grade data. The captive/cultivated observations were merged with the research-grade data, leading to the raw dataset for sanitation.

The raw dataset contains categories at different taxonomic ranks, ranging from class to subspecies, which is not ideal for training a species-level classifier. In order to sanitise the classes, all observations labelled at high-than-species ranks, i.e. class, order, family, and genus, were removed, and all observations labelled at lower-than-species ranks i.e. subspecies, form, variant, etc., were relabelled using their species. Corrections were also applied to outdated or ambiguous species information in the data records.

The sanitised dataset was split into a training set and a validation set. To prevent leakage of correlated instances between the two sets, the instances were grouped as observations. Each observation, with all its image instances, was only in either the training *or* the validation set but never in both. The split into training set and validation set was stratified by species class: a set percentage of observations belonging to each class was split off into the training set, and the rest form the validation set.

EfficientNetV2

We used EfficientNetV2 models of three different sizes, referred to as models *S*, *M*, and *L* in Tan and Le (2021). The PyTorch Image Model library (Wightman 2019) provides these models, with parameters pretrained on the ImageNet benchmark dataset. The model architectures are shown in Table 1.

The pretrained models were used as feature extractors; before we performed training, a randomly initialised logistic regression classifier was appended to each model to convert feature vectors into logits. This classifier was first fine-tuned for a small number of iterations with the feature extractor's weights frozen, before all weights in the feature extractor and classifier were fine-tuned together. This was to preserve knowledge gained during pretraining until the classifier was well-tuned, by preventing change of pretrained weights when the classifier's performance was still poor.

After a fine-tuned model was validated, the validation data was used to calibrate the model's confidence before deployment, using temperature scaling (Guo et al. 2017). Temperature scaling does not change the class rankings of predictions but adjusts the probability of the top prediction, i.e. the model's confidence, to be more consistent with the model's estimated accuracy in that confidence range.

Experimental specifications

Our experimental specifications are divided into three subsections that provide details on the data, training, and validation, respectively.

Data

The version of research-grade New Zealand species data (GBIForg 2023) used in our experiment was downloaded from GBIF on 30/08/2023. It contains 15,579 classes,

Table 1. The three EfficientNetV2 architectures.

Stage	Operator	Stride	Channels	Layers
EfficientNetV2-S architecture				
0	Conv3x3	2	24	1
1	Fused-MBConv1, k3x3	1	24	2
2	Fused-MBConv4, k3x3	2	48	4
3	Fused-MBConv4, k3x3	2	64	4
4	MBConv4, k3x3, SE0.25	2	128	6
5	MBConv6, k3x3, SE0.25	1	160	9
6	MBConv6, k3x3, SE0.25	2	256	15
7	Conv1x1 & Pooling & FC	–	1280	1
EfficientNetV2-M architecture				
0	Conv3x3	2	24	1
1	Fused-MBConv1, k3x3	1	24	3
2	Fused-MBConv4, k3x3	2	48	5
3	Fused-MBConv4, k3x3	2	80	5
4	MBConv4, k3x3, SE0.25	2	160	7
5	MBConv6, k3x3, SE0.25	1	176	14
6	MBConv6, k3x3, SE0.25	2	304	18
7	MBConv6, k3x3, SE0.25	1	512	5
8	Conv1x1 & Pooling & FC	–	1280	1
EfficientNetV2-L architecture				
0	Conv3x3	2	32	1
1	Fused-MBConv1, k3x3	1	32	4
2	Fused-MBConv4, k3x3	2	64	7
3	Fused-MBConv4, k3x3	2	96	7
4	MBConv4, k3x3, SE0.25	2	192	10
5	MBConv6, k3x3, SE0.25	1	224	19
6	MBConv6, k3x3, SE0.25	2	384	25
7	MBConv6, k3x3, SE0.25	1	640	7
8	Conv1x1 & Pooling & FC	–	1280	1

1,094,237 observations, and 2,113,946 individual files. We obtained additional captive/cultivated instances using the export tool of iNaturalist. All instances downloaded are published under one of 'CC-BY', 'CC-BY-NC', and 'CC0' licenses. This yielded a further 39,740 instances belonging to 3,768 classes. Of these classes, 2,486 are present in the research-grade dataset, and the remaining 1,282 are only present in the captive/cultivated dataset. Species such as Canis familiaris (dogs) are only present in the captive/cultivated dataset and not in the research-grade dataset. Although species like Felis catus (cats) are present in both datasets, the research-grade data contains snapshots of feral cats as well as photos of their tracks and droppings in natural environments; the commonly expected photos of domesticated cats in household environments are contained in the captive/cultivated data.

The research-grade and captive/cultivated datasets were merged. The majority of the instances are image files such as JPEG and PNG files, but the dataset also contains some sound and video files. For the purpose of training an image classifier, only image files whose formats can be recognised by the Torchvision library (Paszke et al. 2019) were kept, and the rest of the instances were removed from the dataset.

There are 2,141,684 instances belonging to 1,128,849 observations in 14,991 species-level classes in the dataset after sanitation. All images were resized so that each image's short side was 512 pixels long while the long side was adjusted to maintain the original aspect ratio. All resized images were saved losslessly as PNG files.

The sanitised data was split into two partitions: one for training, and the other for validation. As indicated earlier, the split was stratified by class and performed at observation

level, which means that all instances belonging to the same observation were allocated to the same split to prevent correlated instances from leaking between the training and validation splits. If a class contained only one observation, it was allocated to the training split to avoid empty training classes; conversely, if a class contained at least two observations, most of the observations were allocated to the training split, and (the floor of) 10% of the observations were allocated to the validation split, while guaranteeing that the validation split had at least one observation. Given a class with N observations, the number of validation observations was computed as

$$V = \max\left(\lfloor 0.1 \times N \rfloor, 1\right).$$

Correspondingly, the number of training observations is given by

$$T = N - V.$$

The training split of the sanitised dataset contained 1,915,282 instances belonging to 1,010,473 observations, and the validation split contained 226,402 instances belonging to 118,376 observations.

Training

We conducted training using PyTorch (Paszke et al. 2019) using the EfficientNetV2 (Tan and Le 2021) models implemented in the PyTorch Image Models library (Wightman 2019). The models are available in three different sizes: small, medium, and large, and each has been pretrained on either ImageNet's (Russakovsky et al. 2015) ILSVRC2012 (1,000 classes) or 21k (21,841 classes) versions. Through experimentation, it was determined that the ImageNet21k-pretrained models exhibited better performance on the New Zealand species data. Therefore, we used the three different-sized models, termed *S*, *M*, and *L*, pretrained on ImageNet21k as starting points for fine-tuning on the species data.

S was fine-tuned with a batch size of 1024 instances, *M* was fine-tuned with a batch size of 384 instances, and *L* was fine-tuned with a batch size of 192 instances. *S* received input of size 300x300 pixels, while *M* and *L* received input of size 384x384 pixels. The input sizes are consistent with the specifications of Tan and Le (2021), which we deem suitable for our application. Training instances were preprocessed with AutoAugment (Cubuk et al. 2019), a data augmentation technique, using its ImageNet configuration. All models were fine-tuned with the RMSProp optimiser with decay 0.9, momentum 0.9, weight decay $1e-5$, and initial learning rate $1e-6$ per 16 instances in a mini-batch, i.e. $6.4e-5$ for *S*, $2.4e-5$ for *M*, and $1.2e-5$ for *L*. The learning rate was decayed exponentially by 1% per epoch.

In order to fit the initial models pretrained on ImageNet21k to the species data, a randomly initialised 14,991-class logistic regression classifier was attached to the pretrained feature extractor. The classifier was first trained using the training split for five epochs with the feature extractor frozen. The feature extractor was unfrozen afterwards, and the entire model was fine-tuned on the training split for 495 epochs.

Validation

During fine-tuning, a checkpoint was saved per five epochs. Checkpoints were evaluated using the validation split of the species data. Validation results were used to determine hyperparameter configurations–learning rate, batch size, early stopping point, etc. The New Zealand species dataset's attributes and the models' use cases required that the final configuration be selected not simply based on accuracy alone. Therefore, we assessed the following criteria:

(1) Top-5 accuracy. Differences between species in the same genus or family may be subtle, and it was therefore considered acceptable for the correct label to fall within a model's top-5 predictions.
(2) Accuracy of different class groups, where classes are grouped into 'bins' by their respective numbers of observations. The species data is highly unbalanced, with its biggest classes containing thousands of observations and its smallest classes containing fewer than five observations. Generally, better accuracy can be expected for bins containing populous classes. However, fine-tuned models that show clear signs of overfitting should be avoided, e.g. when the model's accuracy for the smaller classes is worse than a random guess.
(3) Abstention performance. A model should be able to abstain from making a prediction when it is sufficiently uncertain about the instance to be classified. Given an image, the neural network provides a probability estimate for each species. For the purposes of abstention, we used the highest probability assigned to any of the species, i.e. top-1 probability. Given an abstention threshold, an abstention rate can be calculated, as well as accuracy on instances classified by the model (i.e. instances where the highest probability is above the threshold).
(4) Performance at the Kingdom level, to investigate how accuracy of the model depends on the Kingdom that a species belongs to (Animalia, Plantae, etc.).

Results

We first present the models' top-k accuracy. We then show their accuracy in five bins of classes based on the number of observations per class. Lastly, we discuss the models' abstention performance in terms of the trade-off between acceptance rate and classification accuracy.

Summary results

Top-k accuracy curves of models S, M, and L, along with loss on the training and validation data, are shown in Figures 1, 2, and 3 respectively. S achieves best top-1 accuracy 76.86% at epoch 470, and its top-1 accuracy at the end of training is 76.62%. M achieves best top-1 accuracy 78.91% at epoch 395, and its top-1 accuracy at the end of training is 78.59%. L achieves best top-1 accuracy 74.95% at epoch 50, and its top-1 accuracy at the end of training is 69.19%. S and M converge in accuracy as training progresses, while the accuracy of L reaches a peak and then declines. This difference in behaviour is also observed in the loss on the validation data. This is likely due to overfitting, as the high capacity of model L may require more training data to fit adequately.

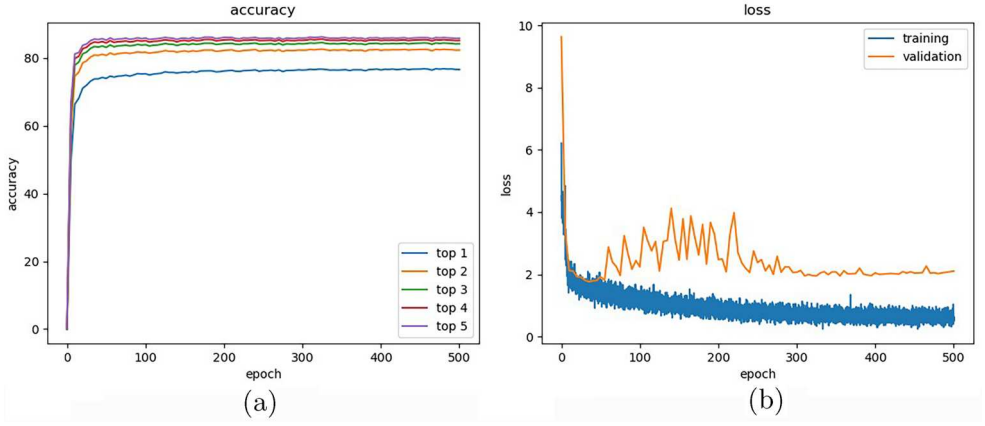

Figure 1. Accuracy and loss of model *S* over 500 training epochs. **A,** Top-*k* validation accuracy. **B,** Training and validation loss.

Binned results

The classes are partitioned into five bins depending on the number of training observations in each class: 1 to 4, 5 to 9, 10 to 19, 20 to 49, and 50 or more. The models' accuracy in each bin is shown in Figure 4, and statistics on the bins are presented in Table 2.

Intuitively, higher accuracy is achieved in bins with more observations per class. Even for classes with only one to four observations, the mean accuracy of 2% or 3% is still substantially better than random guess in the total 14,991 classes.

Kingdom results

Figure 5 shows the validation results of model *S* by the kingdom of the species, in particular: (a) accuracy over the instances whose true label belongs to each kingdom, (b) instance-level confusion matrix by kingdom, and (c&d) class-average precision and recall by kingdom.

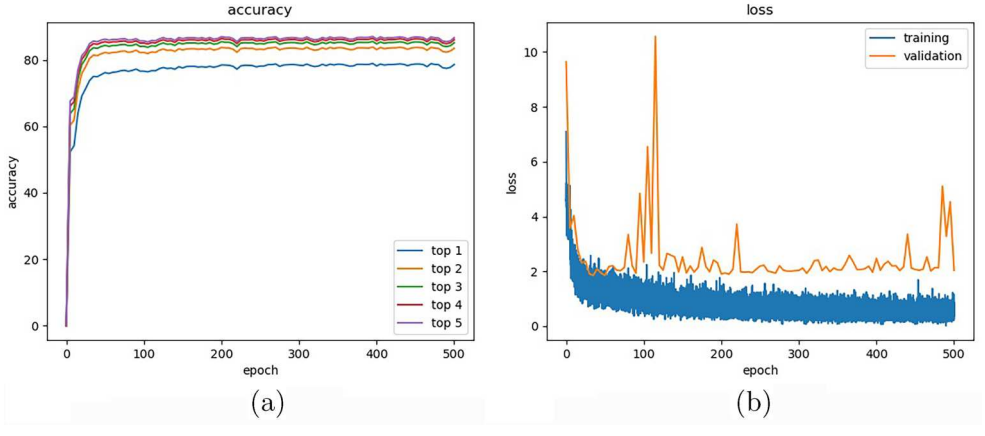

Figure 2. Accuracy and loss of model *M* over 500 training epochs. **A,** Top-*k* validation accuracy. **B,** Training and validation loss.

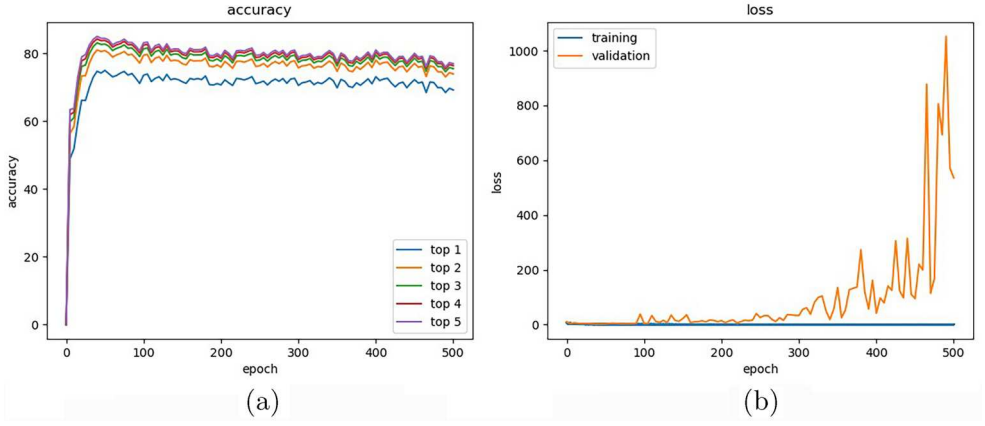

Figure 3. Accuracy and loss of model *L* over 500 training epochs. **A,** Top-*k* validation accuracy. **B,** Training and validation loss.

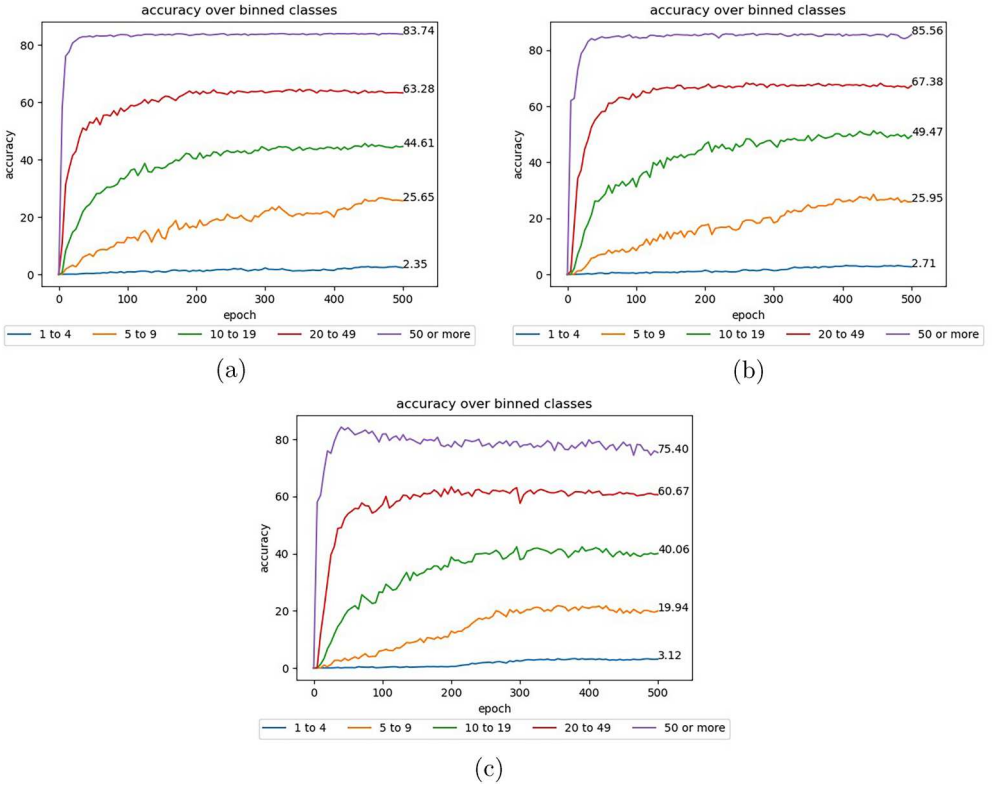

Figure 4. Binned validation accuracy over 500 training epochs. **A,** Model *S*. **B,** Model *M*. **C,** Model *L*.

The majority of training and validation instances belong to the Animalia, Plantae, and Fungi kingdoms, and the model performs better in these three kingdoms than the other four kingdoms. Only the Virus kingdom has too few instances for the model to identify them, and even the other kingdoms with relatively few instances elicit positive results

Table 2. The number of species, the mean number of observations per species, and the mean number of instances per species in each bin.

bins	1 to 4	5 to 9	10 to 19	20 to 49	over 50
number of species	7224	1806	1399	1563	2999
mean number of observations	1.53	6.98	14.04	31.80	305.93
mean number of instances	3.84	17.28	34.38	71.56	565.65

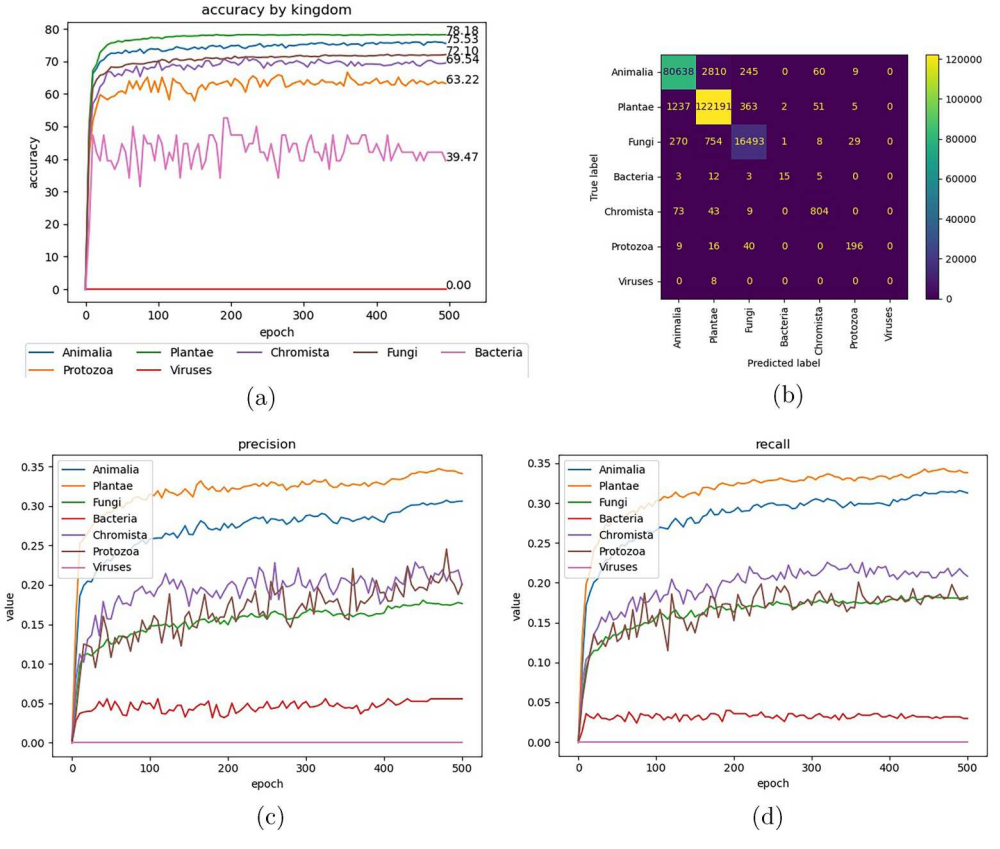

Figure 5. Validation results by kingdom. **A,** Accuracy by kingdom. **B,** Confusion matrix. **C,** Class-average precision by kingdom. **D,** Class-average recall by kingdom.

from the model that are significantly better than random guess, indicating that the model can learn from imbalanced classes.

Abstention results

It is useful to consider the classification accuracy of our models when they are allowed to abstain from making a classification if they are insufficiently confident. Figure 6 shows the trade-off between acceptance rate and classification accuracy on the validation data for the three models when they achieve maximum top-1 accuracy. The rightmost point of each curve reflects accuracy with no abstention, i.e. the top-1 predicted probability threshold is 0, and all validation instances are accepted for

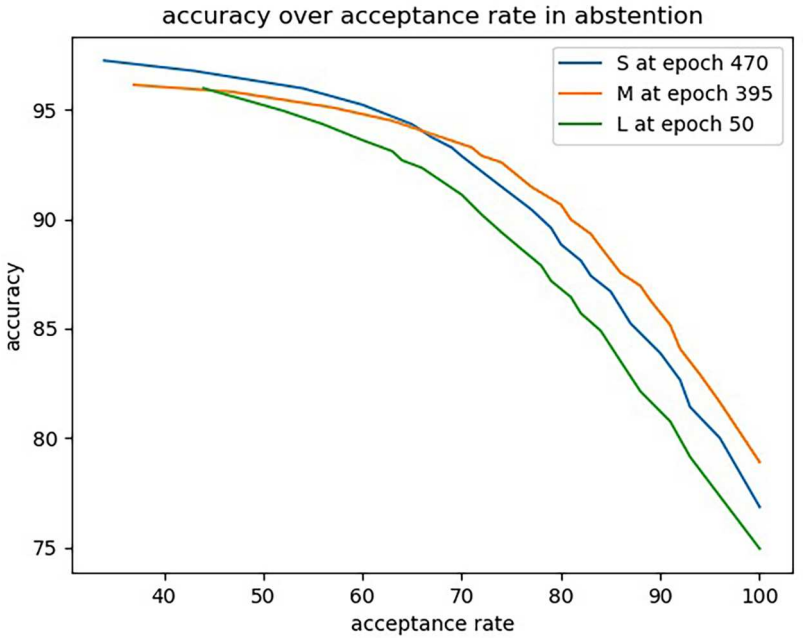

Figure 6. Acceptance rate and accuracy of the models at the best-performing epochs.

classification. The leftmost point of each curve reflects accuracy with an extreme abstention threshold of 99.9%, i.e. only validation instances predicted with a top-1 probability of over 99.9% are accepted, and classification of all other instances is abstained from.

In terms of top-1 accuracy, model *M* is superior with a low abstention threshold that accepts over 70% of validation instances; model *S* becomes superior with a higher abstention threshold that accepts less than 60% of validation instances; model *L* shows signs of overtaking model *M* with very high abstention thresholds. The acceptance rates of the leftmost points of the curves show that model *L* is generally the most confident, followed by model *M* and lastly model *S*.

Model deployment

The models trained on our data can be used to build species identification tools, e.g. as a starting point for fine-tuning a model for specific biosecurity applications. For generic use, we have made them available through a website and mobile apps.

Website

The small model was deployed to a server-side web service,[3] where inference and feature visualisation methods are implemented.

The web app is built on a Flask framework (Grinberg 2018). Model inference is run on the CPU to minimise operational costs; for the small model this yields an inference time of about 1 s. Inference on the *M* and *L* models takes ten seconds to a few minutes. They

are not included in the current version of the website, as we decided that the minor gains in improvement did not justify the much longer inference times in a practical setting.

Once the predictions are made, the top prediction is presented as the final prediction alongside a confidence score. We have developed a semi-automated approach to creating a database of metadata regarding species; this includes common names for that species in Te Reo Maori and English where they could be found, as well as the introductory Wikipedia paragraphs in either language where available and links to those Wikipedia entries.

We also use pest data from the New Zealand Ministry of Primary Industries (MPI) to indicate if a species is 'unwanted' or 'notifiable'. 'Unwanted' species are those considered harmful to New Zealand's industry and biodiversity, while 'notifiable' species are those that individuals are legally required to report to MPI. Through the inclusion of this data, we have implemented a notification system that alerts the user when such a species of concern is identified.

This demonstrates in principle the possibility of adapting the model for, and integrating it into, targeted solutions for e.g. biosecurity. The model can distinguish, for example, between invasive rainbow/plague skinks (*Lampropholis delicata*) and visually extremely similar endemic species such as the copper skink (*Oligosoma aeneum*), see Figure 7. Given the visual similarity between these example species, we think this is a particularly promising use case as the model could allow even non-experts to identify pest species, and one could envisage for example a crowdsourced biosecurity monitoring system for the country based on this capability. However, such a system could potentially also do considerable damage to endangered species if, for example, the classifier occasionally confused it for a more populous invasive species. In such a case even a small false positive rate might result in a considerable proportion of the vulnerable species being killed. We

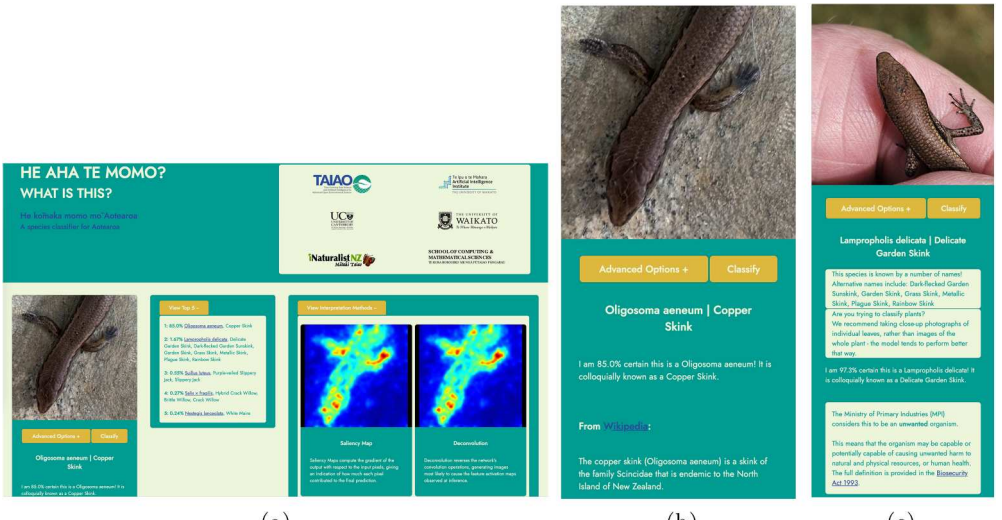

Figure 7. The classifier web interface (left) displays the classification widget, the top 5 predictions widget, and the interpretation methods widget. The classifier is capable of distinguishing endemic species such as the copper skink (center) from visually extremely similar invasive species (right), in this case the rainbow skink, enabling potential applications in biosecurity.

suggest such a system should be extensively tested in the field first to identify common false-positive species, and to take the expected error rate and cost of misclassification into account (e.g. if an endangered species is commonly misclassified as a pest, a higher confidence threshold might be set for that pest).

The metadata also includes the kingdom of each species, which is useful for providing contextual guidance. As much of the iNaturalist data concerning plants depicts close-ups of individual leaves, performance tends to drop when photographing a plant from greater distances. When at least three of the top five predictions are plants, the app provides a notification with tips on photographing plants.

Finally, the app also provides an array of widgets implementing different interpretability methods, to aid in understanding and troubleshooting the predictions. We provide an implementation of a saliency map, as well as five methods from the Captum library (Kokhlikyan et al. 2020): deconvolution, deep lift, guided back propagation, InputXGradient, and GuidedGradCam. Each widget superimposes the resulting heatmaps on a copy of the input image for display.

As deep neural networks have the capacity to overfit their training data, they can focus on incorrect, obscure input elements during prediction, regardless of the prediction's correctness. Therefore, output of these tools should only serve as a guideline to species attribution even when the prediction is correct.

Mobile app

The small model has also been deployed in a mobile app developed with the Flutter framework (Flutter 2024), enabling compatibility with both Android and iOS operating systems. This allows offline inference in a setting where users are most likely to make use of the model, i.e. when out and about in natural environments.

The weights of the small model were quantised to int-8 using PyTorch's dynamic quantisation functionality. Quantization reduced the size of the small model from 157.4 MB to 95.4 MB. It was then converted to a TorchScript representation using PyTorch's JIT tracer for production deployment. TorchScript is designed to create model descriptions that can be run independently from Python and is therefore well suited for creating model files that can be run across a variety of platforms.

The app itself provides basic functionality for interacting with the model: users can load images stored on their device or take new ones. Once inference has completed, the app displays an information card that reports the top-5 predictions; for each species, it provides the scientific name and corresponding probability estimate. If the confidence for the top prediction is above a threshold, set to 0.4 in the current version of the app, this species is reported as the result at the top of the information card. The threshold of 0.4 was empirically selected to balance the rates at which predictions below the threshold are vetoed against the rate at which predictions are correct, based on the validation data.

Like the web version, the mobile app keeps a metadata file containing common names in English and Te Reo Maori where these could be found, as well as the introductory paragraph in Wikipedia (in both languages where available) and links to the corresponding Wikipedia pages. If the prediction confidence is above the threshold, this information is also presented. As with the web application, if the species is identified as a pest by MPI

or is likely a plant, the app produces an appropriate notification. Finally, the app also keeps a record of past predictions for the user to assess.

Conclusions

We have trained neural network-based classification models for automatic species classification in Aotearoa and presented empirical results on the accuracy of these models. The models can be downloaded for downstream applications, such as classification tasks in the biosecurity domain, and have also been integrated into a web application and mobile applications for iOS and Android. Of particular note is the ability of the neural network models to provide confidence scores in the form of probability estimates, which enables the user to decide whether a classification provided by the neural networks should be disregarded. For example, a useful rule of thumb when applying our mobile app is to only consider a classification with a probability greater than 90% as a serious candidate for the correct classification. An important feature of our mobile apps is that they do not require an internet connection because the (small) model can be hosted on the device. Future work could involve retraining the models on newer snapshots of the iNaturalist data or AI-generated synthetic images of rare species (Dasgupta et al. 2024). A limitation of the models presented here is that they do not accurately classify species that are poorly represented in the training data available to us.

Notes

1. Note that in this paper, the term 'class' refers to the category to be predicted for an instance, i.e. the species name associated with an 'instance', i.e. an image.
2. NatureWatch is the former name of Mātaki Taiao – iNaturalist NZ.
3. https://what-is-this.cms.waikato.ac.nz/.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by Ministry for Business Innovation and Employment [CONT-64517-SSIFDS-UOW].

ORCID

Eibe Frank http://orcid.org/0000-0001-6152-7111
Bernhard Pfahringer http://orcid.org/0000-0002-3732-5787
Nick Lim http://orcid.org/0000-0003-4690-5780

References

Chopra S, Hadsell R, LeCun Y. 2005. Learning a similarity metric discriminatively, with application to face verification. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego, CA: IEEE Computer Society. p. 539–546.

Cubuk ED, Zoph B, Mané D, Vasudevan V, Le QV. 2019. Autoaugment: learning augmentation strategies from data. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, CA: Computer Vision Foundation / IEEE. p. 113–123.

Dasgupta D, Mondal A, Chakraborty PP. 2024. Can synthetic plant images from generative models facilitate rare species identification and classification? In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA: IEEE. p. 7530–7540.

Dymond JR, editor. 2013. Ecosystem services in New Zealand: conditions and trends. Manaaki Whenua Press.

Flutter. 2024. Flutter; [accessed 2024 Oct 16]. https://flutter.dev/.

GBIForg. 2021. Free and open access to biodiversity data; [accessed 2021 Feb 20]. https://www.gbif.org/.

GBIForg. 2023. Occurrence download. https://www.gbif.org/occurrence/download/0002035-230828120925497.

Glorot X, Bengio Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the 13th International Conference on Artificial Intelligence and Statistics. Sardinia: PMLR. p. 249–256.

Grinberg M. 2018. Flask web development: developing web applications with Python. O'Reilly Media, Inc.

Guo C, Pleiss G, Sun Y, Weinberger KQ. 2017. On calibration of modern neural networks. In: Proceedings of the 34th International Conference on Machine Learning. Sydney, NSW: PMLR. p. 1321–1330.

He K, Zhang X, Ren S, Sun J. 2015. Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: IEEE International Conference on Computer Vision. Santiago: IEEE Computer Society. p. 1026–1034.

He K, Zhang X, Ren S, Sun J. 2016. Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV: IEEE Computer Society. p. 770–778.

iNaturalistorg. 2021. A community for naturalists. iNaturalist; [accessed 2021 Feb 20]. https://www.inaturalist.org.

Ioffe S, Szegedy C. 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on Machine Learning. Lille: PMLR. p. 448–456.

Kokhlikyan N, Miglani V, Martin M, Wang E, Alsallakh B, Reynolds J, Melnikov A, Kliushkina N, Araya C, Yan S, et al. 2020. Captum: a unified and generic model interpretability library for PyTorch.

Krizhevsky A, Sutskever I, Hinton GE. 2012. Imagenet classification with deep convolutional neural networks. In: Proceedings of the 26th Annual Conference on Neural Information Processing Systems. Lake Tahoe, NV. Curran Associates, Inc. p. 1106–1114.

Mo J, Frank E, Vetrova V. 2017. Large-scale automatic species identification. In: Proceedings of the 30th Australasian Joint Conference on Artificial Intelligence. Melbourne: Springer. p. 301–312.

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, et al. 2019. PyTorch: an imperative style, high-performance deep learning library. In: Proceedings of the Annual Conference on Neural Information Processing Systems. Vancouver, BC, Canada. p. 8024–8035.

Reid J, Rout M. 2020. The implementation of ecosystem-based management in New Zealand – a Māori perspective. Marine Policy. 117:103889. doi: 10.1016/j.marpol.2020.103889

Rogers M, Thompson T, Duporge I, Fischer J, Pütz K, Mattern T, Xue B, Zhang M. 2025. Automated detection of Salvin's albatrosses: improving deep learning tools for aerial wildlife surveys. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop on CV4Animals: Computer Vision for Animal Behavior Tracking and Modeling. Nashville, TN, USA.

Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein MS, et al. 2015. Imagenet large scale visual recognition challenge. Int J Comput Vis. 115(3):211–252. doi: 10.1007/s11263-015-0816-y

Sandler M, Howard AG, Zhu M, Zhmoginov A, Chen L. 2018. Mobilenetv2: inverted residuals and linear bottlenecks. In: IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT: Computer Vision Foundation / IEEE Computer Society. p. 4510–4520.

Szegedy C, Ioffe S, Vanhoucke V, Alemi AA. 2017. Inception-v4, Inception-ResNet and the impact of residual connections on learning. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence. San Francisco, CA: AAAI Press. p. 4278–4284.

Tan M, Chen B, Pang R, Vasudevan V, Sandler M, Howard A, Le QV. 2019. Mnasnet: platform-aware neural architecture search for mobile. In: IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, CA: Computer Vision Foundation / IEEE. p. 2820–2828.

Tan M, Le QV. 2019. EfficientNet: rethinking model scaling for convolutional neural networks. In: Proceedings of the 36th International Conference on Machine Learning. Long Beach, CA: PMLR. p. 6105–6114.

Tan M, Le QV. 2021. EfficientNetV2: smaller models and faster training. In: Proceedings of the 38th International Conference on Machine Learning. Virtual Event: PMLR. p. 10096–10106.

Vetrova V, Coup S, Frank E, Cree MJ. 2018. Hidden features: experiments with feature transfer for fine-grained multi-class and one-class image categorization. In: International Conference on Image and Vision Computing New Zealand. Auckland: IEEE. p. 1–6.

Wightman R. 2019. PyTorch image models. https://github.com/rwightman/pytorch-image-models.

Xiong Y, Liu H, Gupta S, Akin B, Bender G, Wang Y, Kindermans P, Tan M, Singh V, Chen B. 2021. MobileDets: searching for object detection architectures for mobile accelerators. In: IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA. Computer Vision Foundation / IEEE. p. 3825–3834.

Zoph B, Le QV. 2017. Neural architecture search with reinforcement learning. In: Proceedings of the 5th International Conference on Learning Representations. Toulon. OpenReview.net. https://openreview.net/forum?id=r1Ue8Hcxg.