Copy-paste augmentation improves automatic species identification in camera trap images

Cédric S. Mesnage[1,3], Andrew Corbett[1], Jake Curry[2], Sareh Rowlands[3], Anjan Dutta[4], Richard Everson[1,3], Benno I. Simmons[2*]

1. Institute for Data Science and Artificial Intelligence, University of Exeter, Exeter, UK
2. Centre for Ecology and Conservation, University of Exeter, Penryn, UK
3. Department of Computer Science, University of Exeter, Exeter, UK
4. Institute for People-Centred Artificial Intelligence, University of Surrey, Guildford, UK

* Corresponding author: bsimmons.research@gmail.com

Acknowledgements
We are grateful for the assistance of George De Ath, Charlie Kirkwood, Mark Kelson and Finley Gibson. CSM, AC, BIS and were supported by an IDSAI Research award as funded through a Research Grant from the Alan Turing Institute. BIS was also supported by a Royal Commission for the Exhibition of 1851 Research Fellowship. JC was supported by a doctoral training grant awarded as part of the UKRI AI Centre for Doctoral Training in Environmental Intelligence (UKRI grant number EP/S022074/1).

Author contributions
BIS, SR and AD conceived the work. CSM conducted analyses. All authors helped coordinate and design the analyses. CDM and BIS wrote the first draft of the manuscript. All authors contributed to subsequent revisions.

Data Availability Statement
The Serengeti dataset is publicly accessible at https://lila.science/datasets/snapshot-serengeti.

Abstract

1. Effective conservation requires effective biodiversity monitoring. The pace of global biodiversity change far outstrips the ability of manual fieldwork to monitor it. Therefore, technological solutions, like camera traps, have emerged as a crucial way to meet biodiversity monitoring needs. Camera traps produce vast amounts of data and so AI is increasingly used to label images with species identities. However, AI struggles to identify species from new locations that are not part of the training data ('generalisation'). Resolving this is crucial for the promise of automated biodiversity monitoring to be realised.
2. Here we use 'copy-paste' augmentation to help resolve the generalisation challenge. Copy-paste augmentation refers to isolating animal 'segments' from existing images and pasting the segments onto novel backgrounds, to create new, synthetic images that are then used as part of the training data. Theoretically, this could make a model agnostic to backgrounds and therefore more able to generalise to unseen locations. While generation of synthetic images is commonly used as an augmentation method in other fields, such as medicine, it has not been used before in biodiversity science.
3. We found that copy-paste augmentation improved the ability of AI to identify species in new, unseen locations by $8 \pm 2\%$. There was species-level variation in improvement, but the vast majority of species benefited from the approach. We found mixed results when using copy-paste augmentation on models trained with very small numbers of images (1-8 per species).
4. Copy-paste augmentation improves the ability of AI models to generalise to new, unseen locations. Our method also shows promise for resolving the challenge of long-tailed camera trap data. AIs perform poorly on species in the 'long tail' of these distributions because there are very few images to train on. Copy-paste augmentation can help rebalance datasets by adding synthetic images of underrepresented species. Overall, our results suggest a promising role for augmentation methods that generate new, synthetic images in biodiversity science. Ecologists and conservationists must move beyond simple augmentation methods, such as image transformations, if we are to resolve key challenges in species identification AI.

Keywords: camera trap, machine learning, AI, augmentation, computer vision, mammals, serengeti, monitoring

1 Introduction

Monitoring biodiversity is essential to track progress towards policy objectives and to assess the effectiveness of conservation actions. Traditionally,

biodiversity monitoring relies on manual fieldwork, where researchers sample some aspect of biodiversity across space and/or time. However, this kind of fieldwork is both expensive and time consuming and thus scales poorly (Caughlan & Oakley 2001). While traditional fieldwork will always play a role in ecology, it alone cannot meet the growing need for up-to-date information about the state of global biodiversity (Leadley et al. 2022).

Technology offers a possible solution to this problem, through the advent of passive monitoring techniques, such as camera traps. Camera traps – motion- or heat-activated cameras that capture images of wild animals – have great potential to help monitor biodiversity at scale: their low cost allows them to be deployed in vast arrays, collecting data on wildlife locations and behaviour across long time spans and large spatial extents (Swanson et al. 2015). Camera traps are already widely used and demonstrably useful, producing essential insights into population sizes, species richness, animal behaviour, disease spread, migration patterns, movement ecology, predator-prey interactions and conservation management (Delisle et al. 2021).

One of the biggest barriers to harnessing the full potential of camera traps, however, is that processing the large amounts of data they collect remains a manual task: humans must view tens of thousands, or even millions, of images and identify any species that occur in each image. This work is extremely time consuming and it can take multiple person-years to label all images in a single dataset (Norouzzadeh et al. 2018).

To solve this problem, deep learning algorithms have been proposed to automate the identification of animals in camera trap images. These AI approaches have produced impressive results. For example, using the 3.2 million-image 'Snapshot Serengeti' dataset, deep neural networks automatically identified animals correctly in 96.6% of images, representing a saving of 8.4 years of human effort (Norouzzadeh et al. 2018).

While these figures are impressive, and highlight the potential for artificial intelligence to transform conservation biology, they may also be misleading. This is because the majority of camera trap AI studies only evaluate performance on images from locations seen during training (Shahinfar et al. 2020, Schneider et al. 2020, Tabak et al. 2019). Conversely, when algorithms have been tested on their ability to generalise to new, previously unseen locations, they perform significantly worse (Beery et al. 2018, Schneider et al. 2020). The panacea for this field is for biodiversity monitoring to be fully automated, based on AI which can accurately identify all species in any camera trap image from anywhere in the world. Generalisation to new locations is clearly central to this mission and thus it was recently identified as one of the main unsolved problems in the field (Schneider et al. 2020).

Deep learning algorithms may struggle to generalise to new locations because models overfit to particular backgrounds (Schneider et al. 2020). Thus, when new backgrounds are encountered, algorithms are more likely to fail. Some studies have tried to remedy this problem by cropping images, such that they contain fewer background pixels and animals occupy more of the frame (Norouzzadeh et al. 2021). This approach has shown promise, with algorithms trained on cropped images having greater accuracy than those trained on full images (Norouzzadeh et al. 2021, Beery

et al. 2018). However, cropping is not a perfect solution because background pixels still remain in the image, preventing algorithms from being truly decoupled from the environmental contexts on which they were trained.

Recently, it was proposed that segmentation approaches could be used to completely remove the background from camera trap images, leaving just the animal 'segments' (Schneider et al. 2020). Training datasets could then be augmented with generated images, comprising animal segments 'pasted' onto novel backgrounds (Ghiasi et al. 2021*a*). Theoretically this approach could allow models "to become agnostic to backgrounds, and thus able to generalize to any unseen location" (Schneider et al. 2020). However, despite the immense potential of this approach, it has never been attempted.

Here we make the first such attempt, using segmentation to create novel 'copy-paste' images to augment a large dataset of real camera trap images. We assess the ability of copy-paste augmentation to improve the ability of algorithms to generalise to new, unseen locations. We find that this approach improves accuracy and conclude this could have important implications for future work building towards a general AI for global biodiversity monitoring.

2 Materials and Methods

2.1 Data

We analysed the Snapshot Serengeti dataset (Swanson et al. 2015), available at `https://lila.science/datasets/snapshot-serengeti`. Snapshot Serengeti has a number of advantages: (i) it is the largest camera trap image dataset available; (ii) it has a large number of bounding box annotations, which are a relatively uncommon annotation, but which were essential for our study; and (iii) it has been used by other studies in related work, facilitating comparisons between approaches (e.g. Norouzzadeh et al. (2018, 2021)). Of the 7 million images in the dataset, 74616 have bounding box annotations around individual animals, giving their position and species identity. The dataset covers 225 different locations over 6 seasons. Unfortunately, the species identity (hereafter referred to as 'classes') annotations are given for a sequence of 3 images, and not for a particular bounding box, making it impossible to know which class corresponds to which bounding box without further manual inspection. We therefore focused on images where a single class was identified to remove this uncertainty. Another issue with the dataset is that images from some locations have been rescaled, while their bounding boxes have not; images from these locations were removed. The list of removed locations can be found in the supplementary in Table S1).

2.2 Monte Carlo Cross Validation

To test for transferability (the ability of our trained AI to generalise to new, unseen locations), and to estimate the statistical significance of our

results, we apply Monte Carlo cross validation. The following experiment is reproduced k times. We first randomly sample locations, selecting 80% of the locations for training and 20% of the locations for testing. We use a small subset of the test set for validation purposes (10 images per class from the test locations); this is used at each epoch to evaluate the training in terms of accuracy. We evaluate on the test set for each k once the training is finished (note that results shown below are averaged over the k iterations of the Monte Carlo cross validation; the standard error is provided to capture variability between iterations).

We produce multiple training sets per iteration: a 'raw' training set with only real images, and 'augmented' training sets, which contain both real images and generated images. The augmented sets contain varying numbers of generated images, determined by the augmentation factor. An augmentation factor of 1 (indicated by aug_1 in plots) would describe an augmented set with an equal number of raw and generated images; an augmentation factor of 2 (indicated by aug_2 in plots) would describe an augmented set with twice as many generated images as raw images, and so on. Examples of raw and augmented images can be seen in Figure 1.

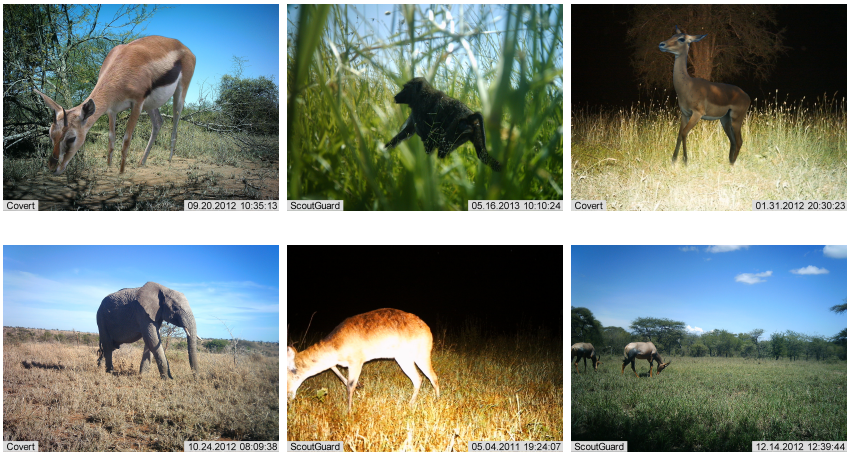


Figure 1: Examples of raw and generated training images. The top row shows generated training images resulting from our automated copy-pasting strategy onto empty backgrounds. The bottom row shows raw, unedited images from the Snapeshot Serengeti dataset.

2.3 Image segmentation

Copy-paste augmentation involves pasting images of animals that have no background, onto backgrounds that contain no animals, to create new images (Figure 1). Images of animals that have had their backgrounds removed are called segments. To automate segmentation, we use U^2-NET, a convolutional neural network for image segmentation and background removal (Qin et al. 2020). To prevent having multiple animals in the

same segment, we focus on images with a single bounding box. For each bounding box, we extend the bounding box by 10%, crop the image to the bounding box and use the pretrained U^2-NET to remove the background. After removing the background, we tighten the bounding box to match the silhouette of the segment. Following segmentation, we exclude segments that contain less than 30% non-transparent pixels. This is because these segments are usually cases where the segmentation algorithm has made errors, such as selecting the background instead of the animal or where animals are missing most of their bodies. We manually filter the resulting 5235 segments to remove any remaining erroneous segments, leaving 3585 usable segments.

2.4 Automated copy pasting of animals

To generate augmented datasets, we use the segmented animals from images randomly sampled from the raw dataset and from the training locations of that iteration. For each class (species), we create as many copy-paste images as required, depending on the number of usable segments available for that class. For each image, the segment is randomly shrunk or expanded by ±5%, rotated by ±5 degrees, and flipped horizontally (mirrored in the y-axis). We choose a random x and y coordinate as the location to paste the segment, such that the bottom of the segment is in the bottom half of the background being pasted onto. Segments are pasted onto empty background images chosen randomly from the test locations of that iteration.

2.5 Evaluation

The software package we use for object detection , YOLOv5 (Jocher 2020), produces mean average precision metrics (mAP). The mean average precision is the average precision over all classes detected and the average precision is calculated based on precision and recall.

We evaluate the performance of our algorithms using the mean delta mean average precision (mAP):

$$\overline{\Delta mAP} = \frac{1}{k}\sum_{i=1}^{i=k} mAP(i, raw + aug) - mAP(i, raw) \tag{1}$$

We define the $\overline{\Delta mAP}$ as the mean over k Monte Carlo iterations of the difference between the mAP resulting from training on an augmented dataset and the mAP resulting from the training on the raw dataset in the same iteration. mAP is a widely used metric for evaluating object detection algorithms, derived from the confusion matrix.

We carried out two sets of experiments: (i) a traditional experiment using 500 images per class in the raw training set; and (ii) a few-shot learning approach using very small numbers of images per class in the raw training set (between 1 and 8 images).

Figure S2 gives the versions of all software packages used in this analysis to facilitate reproducibility.

3 Results

3.1 500 images per class

When training with 500 raw images and 500 augmented images per class, $\overline{\Delta mAP}$ is positive. This means that the mean average precision is higher when raw and augmented images are used (when copy-paste augmentation was used), compared to when just raw images are used (without copy-paste augmentation). Figure 2 shows the mean mAP for raw and augmented datasets throughout the training, calculated on the validation set. The mean $\overline{\Delta mAP}$ over 10 iterations is 0.0156 ± 0.00496 (SE) when evaluated on the test sets (ranging between 10 and 15,000 images), corresponding to an $8 \pm 2\%$ gain in accuracy.

Figure 3 shows the relationship between the number of usable segments per species and the mAP test results. We find a significant correlation of the form $y \sim log(x)$ (estimate $= 0.0569$, $p < 0.001$, $R^2 = 0.597$).

Figure 4 shows model performance for each class. For the vast majority of classes, model performance was higher with copy-paste augmentation. The jackal, guinea fowl and kori bustard gained the most from copy-paste augmentation. However, for eight (17%) classes (bat-eared fox, civet, eland, elephant, hippopotamus, striped hyenas, rodents and waterbuck) model performance was substantially lower with copy-paste augmentation compared to when only raw images were used.

3.2 Few-shot learning

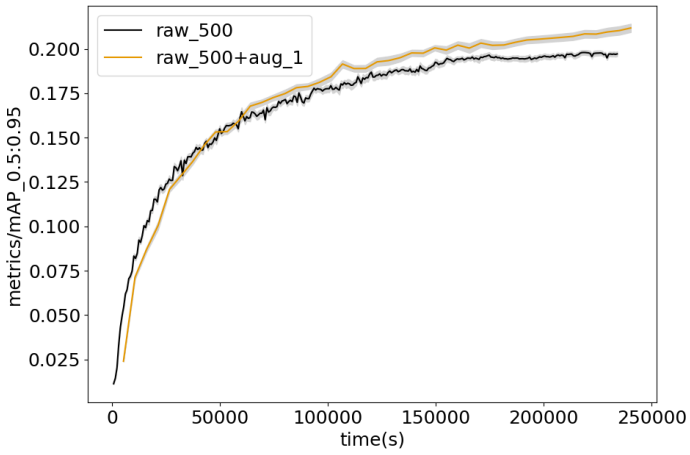
Model performance when using very small amounts of training data was mixed (Figure 5). When using 1 or 2 images per class, copy-paste augmentation improved model performance. When 4 or 8 images per class were used, copy-paste augmentation appeared to worsen model performance (Figure 5). Figure 6 shows $\overline{\Delta mAP}$ and standard errors from the few-shot learning approach, evaluated on each iteration's test sets. Results per class are given in Figure S1.

4 Discussion

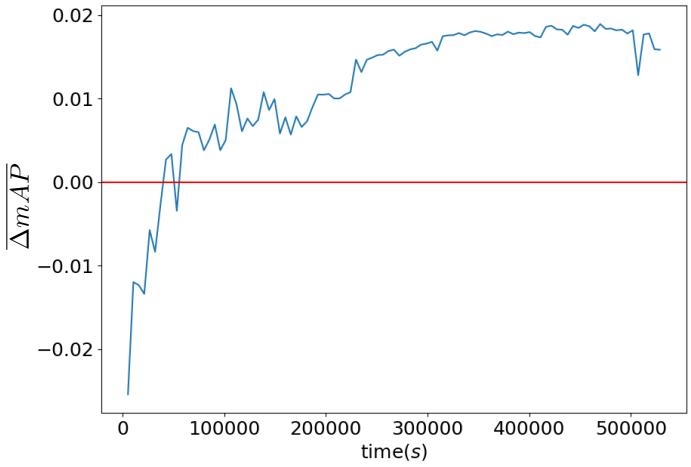
We find that copy-paste augmentation improves the ability of AI models to identify species in camera trap images from unseen locations. Specifically, we found that using copy-paste augmentation to double the number of training images per species improved performance by $8 \pm 2\%$.

There are two main augmentation strategies in computer vision: (1) transforming existing images through processes like flipping, rotating, and cropping (Schneider et al. 2020, Shorten & Khoshgoftaar 2019), and (2) generating new, artificial images (Barile et al. 2021, Garcea et al. 2023, Shorten & Khoshgoftaar 2019). While fields like medicine frequently adopt the latter (Garcea et al. 2023), biodiversity science has lagged behind, and primarily uses basic image transformations (Schneider et al. 2020), or no augmentation at all (Norouzzadeh et al. 2021).

Our results suggest a promising future for augmentation with artificial



(a)



(b)

Figure 2: (a) The mean mAP for raw and augmented datasets throughout the training, calculated on the validation set, averaged over 10 Monte Carlo samples. The raw dataset contained 500 images per class ('raw_500') and the augmented dataset contained 500 raw and 500 generated images per class (an augmentation factor of 1, 'raw_500+aug_1'). The mAP [0.5:0.95] y-axis label represents the number of true positives over the total number of true positives and false positives with an intersection over union (IoU) between 0.5 and 0.95. The IoU relates to the overlap of the original bounding boxes and the detected ones. Gray bands represent the standard error of the mean. (b) $\overline{\Delta mAP}$ training results on 500 raw images with an augmentation factor of 1 over the $k = 10$ Monte Carlo samples.

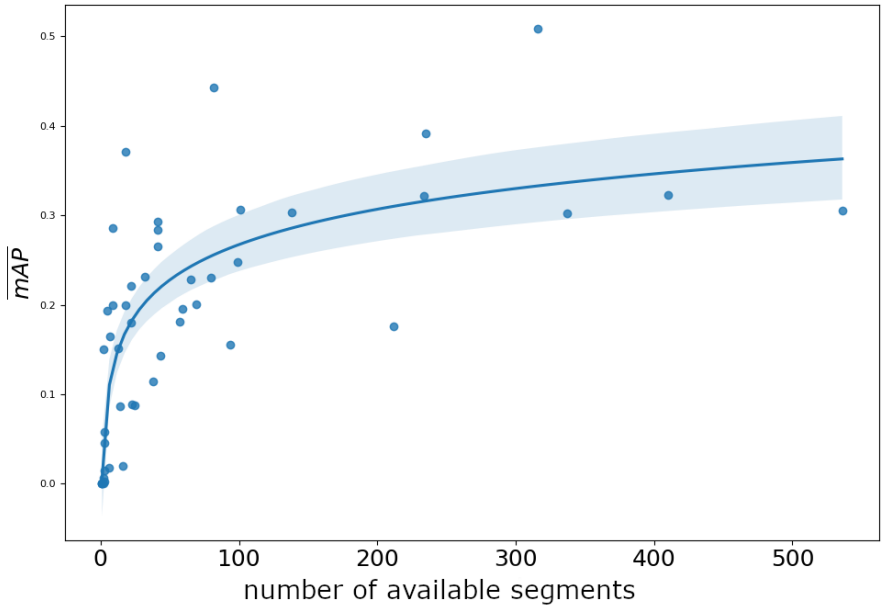


Figure 3: Correlation between the $\overline{mAP}$ and the number of usable segments per class over 10 iterations. Each point represents a class (species).

images in biodiversity monitoring. Specifically, copy-paste augmentation can help address two main challenges (Schneider et al. 2020). First, it helps improve transferability to new locations. This 'domain shift' is a major challenge: in a recent study, the best performing model achieved 95.6% accuracy when tested on locations seen during training, but only 68.7% accuracy when tested on unseen locations. In general, neural networks perform best when the testing and training data are similar (Goodfellow 2016, LeCun et al. 2015). However, this is rarely the case in conservation, where users will often want to identify species in images in new locations that have different backgrounds to those seen in training (Meek et al. 2013). Here we show that augmentation improves model performance on unseen locations for 'free'; that is, higher transferability can be achieved via augmentation without the need for any additional data.

Second, augmentation helps address the issue of unbalanced datasets. Camera trap datasets have highly skewed frequency distributions across species, with a few species having large numbers of images, and many species having few images. Accurately classifying species with few images to train on poses a significant challenge for species identification AIs, as models typically require large amounts of data for training (Norouzzadeh et al. 2018, Tabak et al. 2019, Willi et al. 2019). Here we show that copy-paste augmentation improves performance in classes with only 500 images, suggesting that it is a valid strategy for rebalancing datasets and addressing this problem.

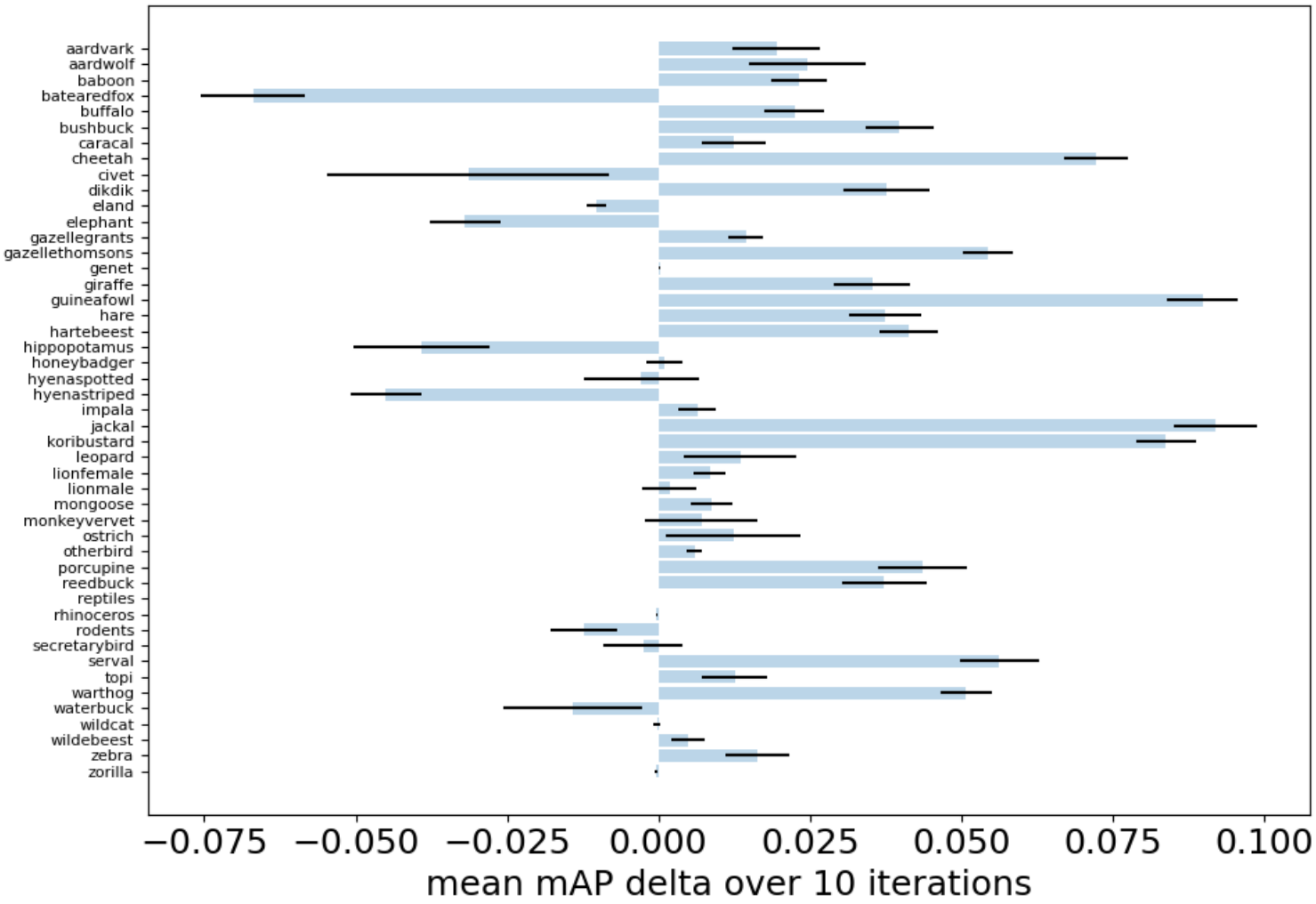
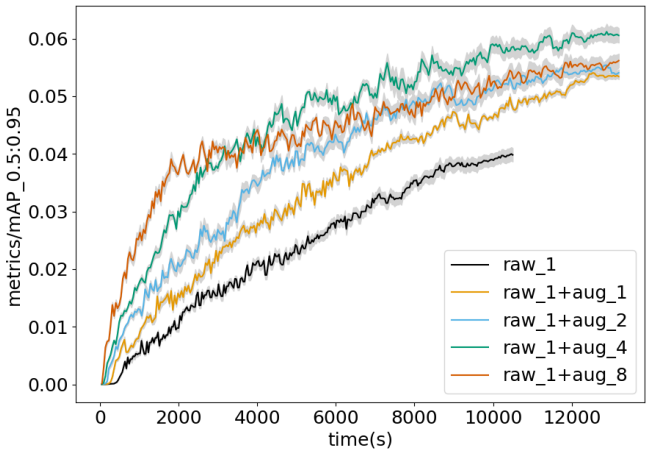
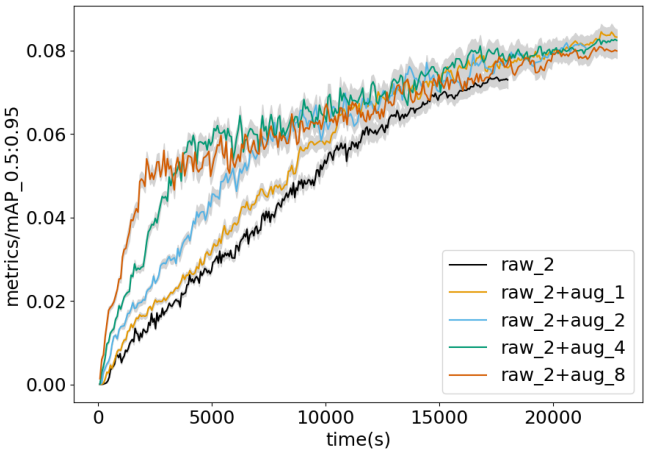


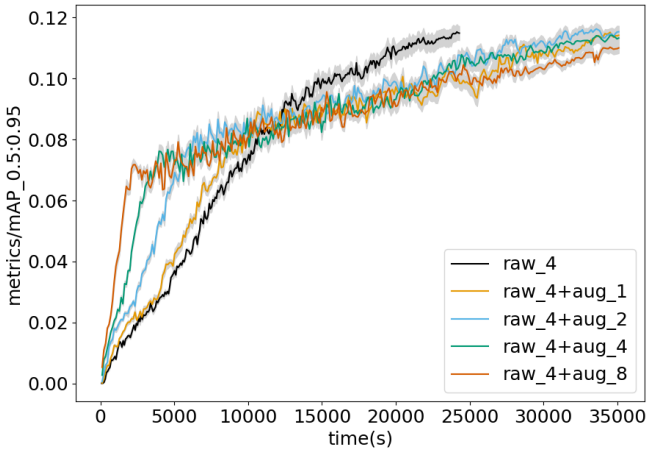
Figure 4: Horizontal bar plot of the $\overline{\Delta mAP}$ per class over 10 iterations when evaluated on the test sets with the best weights at 300 epochs for the raw dataset, and 50 epochs for the raw+aug dataset (an equivalent total training time).



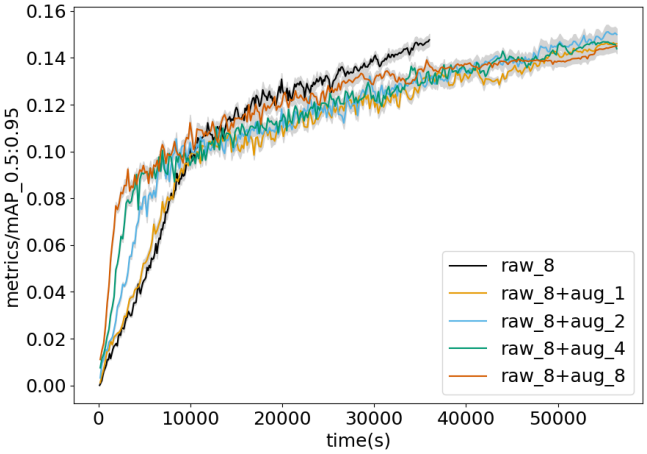
(a) 1 image per class



(b) 2 images per class

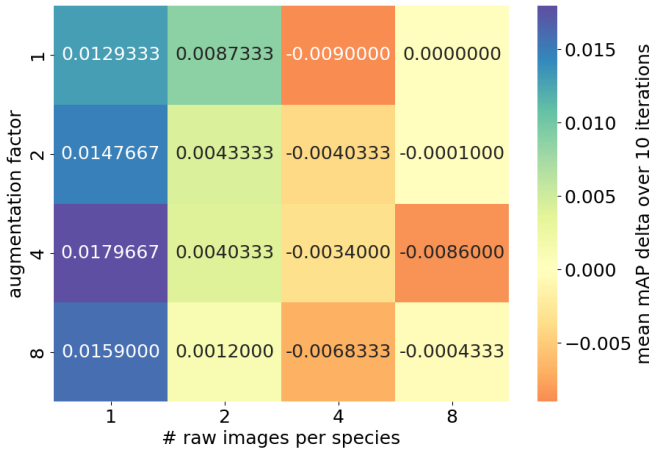


(c) 4 images per class

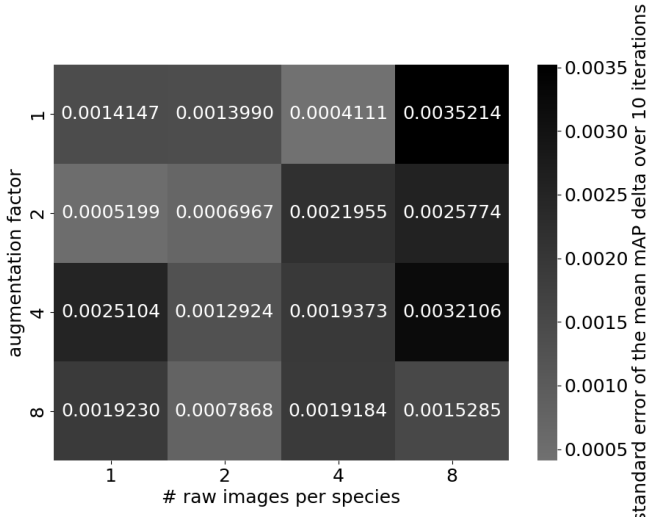


(d) 8 images per class

Figure 5: Results of few-shot learning validated on a 10 images per class validation set at each epoch, up to 300 epochs. Copy-paste augmentation improves accuracy when 1 or 2 images per class are used, but worsens accuracy when 4 or 8 images per class are used. Results are projected onto training time. Results are shown for 1, 2, 4, 8 images per class and augmentation factors of 1, 2, 4, 8. 'raw_n' by itself shows the performance of the model when trained on just raw images. 'raw_n+aug_n' shows the performance of models trained on augmented datasets containing raw and generated images. 'raw_n' indicates n raw images per class in a dataset. 'aug_m' indicates an augmentation factor of m. An augmentation factor of 1 means the augmented dataset contains an equal number of raw and generated images; an augmentation factor of 2, means the augmented set contains twice as many generated images as raw images, and so on. For example, 'raw_8+aug_8' shows the performance of a model trained on 72 images, comprising 8 raw images and 64 generated images. The mAP [0.5:0.95] y-axis label represents the number of true positives over the total number of true positives and false positives with an intersection over union (IoU) between 0.5 and 0.95. The IoU relates to the overlap of the original bounding boxes and the detected ones.



(a) $\overline{\Delta mAP}$ over the iterations



(b) Standard error of the mean

Figure 6: Aggregated results of the few-shot learning experiment, showing (a) mean change in mAP and (b) standard error of those means, for different numbers of raw images and augmentation factors.

However, augmentation approaches are not immune from data demands. As Figure 3 shows, model performance was higher for species with a larger number of available segments. This could result in a 'rich get richer' effect, where performance is higher for frequently occurring species that have many available segments to paste, compared to data-sparse species for which a less diverse set of augmented images can be created. Thus, while copy-paste augmentation can improve performance for species that have a low numbers of available images from which to source segments, it may still not solve the issue of unbalanced data for species with extremely low numbers images (e.g. one); a minimum number of available segments is needed to produce an augmented dataset with sufficient diversity to improve performance. Encouragingly, however, the non-linear relationship between number of segments and $\overline{mAP}$ (Figure 3) suggests that, initially, large increases in performance are achieved for small increases in number of segments, with relatively few segments needed to approach peak performance ($\sim$50 segments). Further research across datasets is needed to confirm if this pattern is general.

While, for the vast majority of species, copy-paste augmentation improved performance, for eight species (17%) it substantially decreased performance. One possible reason for this is that all but one (elephant) of these negative-performing species are to some degree nocturnal or crepuscular: the bat-eared fox, civet, striped hyena and many rodents are nocturnal, the common eland is crepuscular, and hippopotamuses and waterbucks are often active at night. This means it is likely that the segments of these animals are from images captured at night. We did not synchronise times between segments and backgrounds, thus night segments of these species could have been pasted on to day backgrounds, resulting in unusual images that the model performed worse on. This could have been compounded by AI models' generally lower performance on night camera trap images, due to greyscale, grain, glare from flash and shorter viewing distance (Mitterwallner et al. 2024). However, given that our augmentation approach improved performance substantially for other nocturnal species, such as aardvark and porcupine, this cannot fully explain our results. Another possible explanation is that these species have some morphological or behavioural traits that are not captured through simple duplication; these may be species that require a greater diversity of images for improved model performance. Alternatively, these species' traits might mean the segmentation algorithm performed worse, perhaps removing too much or too little of the source image, resulting in subtly erroneous segments that lose necessary detail. Further research is needed to fully understand species-level variation in improvement by augmentation.

We found that augmentation had a mixed impact in a few-shot learning context. Augmentation substantially improved model performance, when there were only one or two raw images per class. However, when four or eight images per class were used, augmentation reduced performance. The augmentation factor (number of augmented images) seemed to have little impact on performance, with the exception of when only one raw image per class was used: in this analysis, augmentation factor four performed best, followed by eight, two, then one. Taken together, these results are hard to fully explain. Data augmentation is an established

few-shot method (Liu et al. 2022, Tian et al. 2024), and has previously shown consistent improvements in model performance across a range of dataset sizes (Ghiasi et al. 2021*b*), and thus we would expect consistent improvement. Notably, in all cases, augmented models perform better than raw-only models in the early parts of training (Figure 5). Simply increasing computational resources could therefore produce improved results; for example, training the models for a longer time period, or running more iterations of each analysis to reduce stochastic effects. Alternatively, drastically increasing the number of augmented images could produce improvements — our analysis includes a maximum of 64 augmented images (eight raw images with an augmentation factor of 8). Testing, for example, eight raw images with thousands of augmented images could be an interesting future direction to establish the limits of this approach. Further research is needed into the value of copy-paste augmentation in a few-shot context before its utility can be fully assessed. However, the results shown here for one or two images per class, show there is significant promise of the approach.

Our research demonstrates the potential for artificial image augmentation in biodiversity monitoring, and thus opens promising avenues for future research. First, it is important to validate our approach in other datasets that span a wide range of species, habitat types and locations. Second, while we manually removed erroneous segments, future research could automate this step, perhaps using an animal detection model, such as MegaDetector (Beery et al. 2019). Third, there are several ways the copy-paste approach could be improved to potentially achieve higher performance. For example, time could be synchronised between segments and backgrounds so that segments are pasted onto backgrounds of an approximately similar time of day, creating a better match between segment and background lighting conditions. Methods for 'smart' pasting could also be developed, to ensure that segments are pasted onto backgrounds in a sensible way that results in realistic images; for example, ensuring land animals are not pasted onto the sky. A more complex solution could ensure animals are pasted at locations on the background such that the resulting images look natural. Currently our approach can result in nonsensical images: for example, pasting elephant segments onto backgrounds with blades of grass in the foreground, resulting in images where elephants appear smaller than blades of grass. Improving pasting methods and assessing whether this increases performance is an important direction for future research.

Overall, we show that copy-paste augmentation shows significant promise as a way to address key challenges in biodiversity monitoring AI. Specifically, it improves transferability to unseen locations and can help balance typical long-tailed ecological camera trap data. Ecologists and conservationists must move beyond just simple image transformations and embrace artificial images as another tool for augmentation.

References

Balestriero, R., Bottou, L. & LeCun, Y. (2022), 'The effects of regularization and data augmentation are class dependent', *arXiv preprint arXiv:2204.03632* .

Balestriero, R., Misra, I. & LeCun, Y. (2022), 'A data-augmentation is worth a thousand samples: Exact quantification from analytical augmented sample moments', *arXiv preprint arXiv:2202.08325* .

Barile, B., Marzullo, A., Stamile, C., Durand-Dubief, F. & Sappey-Marinier, D. (2021), 'Data augmentation using generative adversarial neural networks on brain structural connectivity in multiple sclerosis', *Computer methods and programs in biomedicine* **206**, 106113.

Beery, S., Liu, Y., Morris, D., Piavis, J., Kapoor, A., Joshi, N., Meister, M. & Perona, P. (2020), Synthetic examples improve generalization for rare classes, *in* 'Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision', pp. 863–873.

Beery, S., Morris, D. & Yang, S. (2019), 'Efficient pipeline for camera trap image review', *arXiv preprint arXiv:1907.06772* .

Beery, S., Van Horn, G. & Perona, P. (2018), Recognition in terra incognita, *in* 'Proceedings of the European conference on computer vision (ECCV)', pp. 456–473.

Caughlan, L. & Oakley, K. L. (2001), 'Cost considerations for long-term ecological monitoring', *Ecological indicators* **1**(2), 123–134.

Delisle, Z. J., Flaherty, E. A., Nobbe, M. R., Wzientek, C. M. & Swihart, R. K. (2021), 'Next-generation camera trapping: systematic review of historic trends suggests keys to expanded research applications in ecology and conservation', *Frontiers in Ecology and Evolution* **9**, 617996.

Garcea, F., Serra, A., Lamberti, F. & Morra, L. (2023), 'Data augmentation for medical imaging: A systematic literature review', *Computers in Biology and Medicine* **152**, 106391.

Geiping, J., Goldblum, M., Somepalli, G., Shwartz-Ziv, R., Goldstein, T. & Wilson, A. G. (2022), 'How much data are augmentations worth? an investigation into scaling laws, invariance, and implicit regularization', *arXiv preprint arXiv:2210.06441* .

Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.-Y., Cubuk, E. D., Le, Q. V. & Zoph, B. (2021*a*), Simple copy-paste is a strong data augmentation method for instance segmentation, *in* 'Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition', pp. 2918–2928.

Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.-Y., Cubuk, E. D., Le, Q. V. & Zoph, B. (2021*b*), 'Simple copy-paste is a strong data augmentation method for instance segmentation'.
URL: *https://arxiv.org/abs/2012.07177*

Goodfellow, I. (2016), 'Deep learning'.

Jocher, G. (2020), 'Yolov5 by ultralytics'.
URL: *https://github.com/ultralytics/yolov5*

Leadley, P., Gonzalez, A., Obura, D., Krug, C. B., Londoño-Murcia, M. C., Millette, K. L., Radulovici, A., Rankovic, A., Shannon, L. J., Archer, E. et al. (2022), 'Achieving global biodiversity goals by 2050 requires urgent and integrated actions', *One earth* **5**(6), 597–603.

LeCun, Y., Bengio, Y. & Hinton, G. (2015), 'Deep learning', *nature* **521**(7553), 436–444.

Liu, Y., Zhang, H., Zhang, W., Lu, G., Tian, Q. & Ling, N. (2022), 'Few-shot image classification: Current status and research trends', *Electronics* **11**(11).
URL: *https://www.mdpi.com/2079-9292/11/11/1752*

Meek, P. D., Vernes, K. & Falzon, G. (2013), 'On the reliability of expert identification of small-medium sized mammals from camera trap photos', *Wildlife Biology in Practice* **9**(2), 1–19.

Mesnage, C. (2024), 'Project software repository for the "Copy-paste augmentation improves automatic species identification in camera trap images" paper.'.
URL: *https://doi.org/10.5281/zenodo.12773166*

Mitterwallner, V., Peters, A., Edelhoff, H., Mathes, G., Nguyen, H., Peters, W., Heurich, M. & Steinbauer, M. J. (2024), 'Automated visitor and wildlife monitoring with camera traps and machine learning', *Remote Sensing in Ecology and Conservation* **10**(2), 236–247.

Norouzzadeh, M. S., Morris, D., Beery, S., Joshi, N., Jojic, N. & Clune, J. (2021), 'A deep active learning system for species identification and counting in camera trap images', *Methods in ecology and evolution* **12**(1), 150–161.

Norouzzadeh, M. S., Nguyen, A., Kosmala, M., Swanson, A., Palmer, M. S., Packer, C. & Clune, J. (2018), 'Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning', *Proceedings of the National Academy of Sciences* **115**(25), E5716–E5725.

Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O. R. & Jagersand, M. (2020), 'u^2-net: Going deeper with nested u-structure for salient object detection', *Pattern recognition* **106**, 107404.

Schneider, S., Greenberg, S., Taylor, G. W. & Kremer, S. C. (2020), 'Three critical factors affecting automated image species recognition performance for camera traps', *Ecology and Evolution* **10**(7), 3503–3517.

Shahinfar, S., Meek, P. & Falzon, G. (2020), '"how many images do i need?" understanding how sample size per class affects deep learning model performance metrics for balanced designs in autonomous wildlife monitoring', *Ecological Informatics* **57**, 101085.

Shorten, C. & Khoshgoftaar, T. M. (2019), 'A survey on image data augmentation for deep learning', *Journal of big data* **6**(1), 1–48.

Swanson, A., Kosmala, M., Lintott, C., Simpson, R., Smith, A. & Packer, C. (2015), 'Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna', *Scientific data* **2**(1), 1–14.

Tabak, M. A., Norouzzadeh, M. S., Wolfson, D. W., Sweeney, S. J., VerCauteren, K. C., Snow, N. P., Halseth, J. M., Di Salvo, P. A., Lewis, J. S., White, M. D. et al. (2019), 'Machine learning to classify animal species in camera trap images: Applications in ecology', *Methods in Ecology and Evolution* **10**(4), 585–590.

Tian, S., Li, L., Li, W., Ran, H., Ning, X. & Tiwari, P. (2024), 'A survey on few-shot class-incremental learning', *Neural Networks* **169**, 307–324. **URL:** *https://www.sciencedirect.com/science/article/pii/S0893608023006019*

Willi, M., Pitman, R. T., Cardoso, A. W., Locke, C., Swanson, A., Boyer, A., Veldthuis, M. & Fortson, L. (2019), 'Identifying animal species in camera trap images using deep learning and citizen science', *Methods in Ecology and Evolution* **10**(1), 80–91.

Supplementary Material for

Copy-paste augmentation improves the accuracy of automated species identification in camera trap images

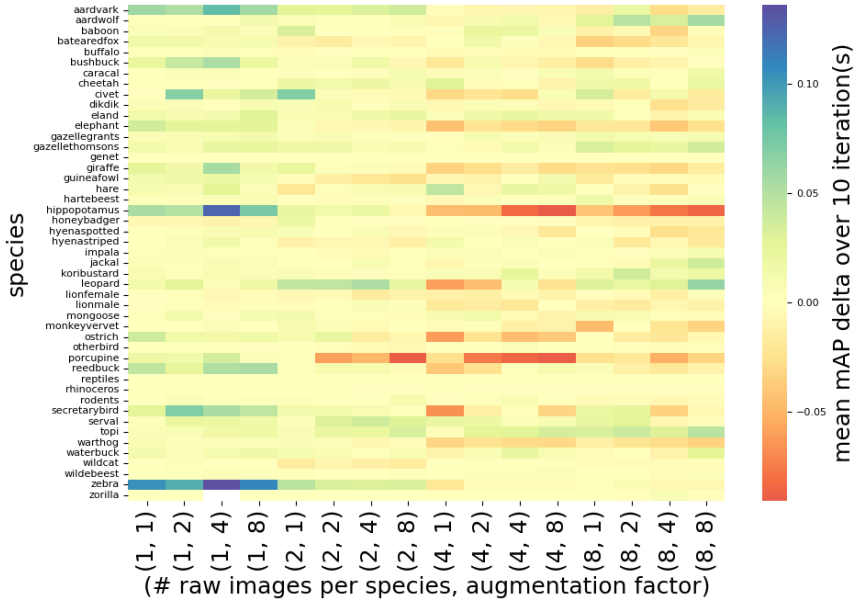


Figure S1: Heatmap of the $\overline{\Delta mAP}$ per species at 300 epochs

Season	Location	Season	Location
S2	E01	S6	J09
S3	E11	S6	D08
S3	J08	S6	U13
S4	R11	S6	S12
S4	E11	S6	P10
S4	E01	S6	L13
S5	G07	S6	S11
S5	L04	S6	F10
S5	E09	S6	J13
S5	E12	S6	E12
S5	C11	S6	E02
S5	B09	S6	C07
S5	D10	S6	C11
S5	I05	S6	K09
S5	J08	S6	E03
S6	G07	S6	R13
S6	G10	S6	B09
S6	E10	S6	Q10
S6	H10	S6	O11
S6	D05	S6	D10
S6	O13	S6	O12
S6	C10	S6	I05
S6	L04	S6	J08
S6	E09	S6	R12

Table S1: List of removed locations per season due to a resizing issue in the original dataset.

```
yolov5==6.2.0
# Base ----------------------------------------
matplotlib>=3.2.2
numpy>=1.18.5
opencv-python>=4.1.1
Pillow>=7.1.2
PyYAML>=5.3.1
requests>=2.23.0
scipy>=1.4.1
torch>=1.7.0  # see https://pytorch.org/get-started/locally/ (recommended)
torchvision>=0.8.1
tqdm>=4.64.0
# protobuf<=3.20.1  # https://github.com/ultralytics/yolov5/issues/8012

# Logging -------------------------------------
tensorboard>=2.4.1
# clearml>=1.2.0
# comet

# Plotting ------------------------------------
pandas>=1.1.4
seaborn>=0.11.0
```

Figure S2: Software packages versions