IBIS
BOU international journal of avian science
Ibis (2022) doi: 10.1111/ibi.13099

Semi-automated detection of tagged animals from camera trap images using artificial intelligence

ANDREA SANTANGELI,*[1,2,3] YUXUAN CHEN,[4] MARK BOORMAN,[5] SOFÍA SALES LIGERO[3] & GUILLERMO ALBERT GARCÍA[3]

[1]*Research Centre for Ecological Change, Organismal and Evolutionary Biology Research Programme, University of Helsinki, Helsinki, FI-00014, Finland*

[2]*FitzPatrick Institute of African Ornithology, DST-NRF Centre of Excellence, University of Cape Town, Cape Town, South Africa*

[3]*Finnish Museum of Natural History, University of Helsinki, Helsinki, FI-00014, Finland*

[4]*Department of Electrical and Electronic Engineering, Imperial College London, South Kensington Campus, London, SW7 2AZ, UK*

[5]*Vultures Namibia, Swakopmund, Namibia*

The use of technology in ecology and conservation offers unprecedented opportunities to survey and monitor wildlife remotely, for example by using camera traps. However, such solutions typically cause challenges stemming from the big datasets gathered, such as millions of camera trap images. Artificial intelligence is a proven, powerful tool to automate camera trap image analyses, but this is so far largely been restricted to species identification from images. Here, we develop and test an artificial intelligence algorithm that allows discrimination of individual animals carrying a tag (in this case a patagial yellow tag on vultures) from a large array of camera trap images. Such a tool could assist scientists and practitioners using similar patagial tags on vultures, condors and other large birds worldwide. We show that the overall performance of such an algorithm is relatively good, with 88.9% of all testing images (i.e. those not used for training or validation) correctly classified using a cut-off discrimination of 0.4. Specifically, performance was high for correctly classifying images with a tag (95.2% of all positive images correctly classified), but less so for images without a tag (87.0% of all negative images). The correct classification of images with a tag was, however, significantly higher when the tag code was at least partly readable compared with the other cases. Overall, this study underscores the potential of artificial intelligence for assisting scientists and practitioners in analysing big datasets from camera traps.

Keywords: animal tag detection, camera trapping, capture–mark–recapture, conservation-technology, deep learning, image recognition, resighting, vulture.

INTRODUCTION

The fields of ecology and conservation biology are undergoing an unprecedented revolution in which traditional survey methods are being replaced with passive technological approaches (Pimm *et al.* 2015). One example is the exponential increase in the use of camera traps to survey animals (Wearn & Glover-Kapfer 2019). However, the use of technology, such as camera traps, often results in enormous amounts of material (e.g. millions of images) that then need to be organized, stored and analysed in order to be used in ecological studies.

The use of artificial intelligence to process the large amount of material obtained from passive sensors such as drone-borne images and camera traps is still in its infancy but rapidly developing (Norouzzadeh *et al.* 2018, Miao *et al.* 2019, Tabak *et al.* 2019, Willi *et al.* 2019, Santangeli *et al.* 2020a). Overall, machine-learning approaches

*Corresponding author.
Email: andrea.santangeli@gmail.com

show high potential for automating the identification of animals from camera trap images and thus reducing the burden and errors associated with analysing millions of images manually (Norouzzadeh *et al.* 2018, Tabak *et al.* 2019).

Most applications of machine learning to aid camera trap image analyses so far have focused on identifying wildlife species (Willi *et al.* 2019). Recently, the use of camera traps was shown to be a particularly effective means for identifying and thus resighting individually marked animals, such as wing-tagged vultures in Africa, so allowing survival analyses (Santangeli *et al.* 2020b). However, that study relied exclusively on manual identification of tagged individuals. So far, to the best of our knowledge, no applications of artificial intelligence for analysing camera trap images have focused on the identification of tagged animals. This is unfortunate given the importance of resighting information of tagged individuals to facilitate survival rate estimation, a critical demographic rate in wildlife management decision-making that is unquantified in many taxa (Conde *et al.* 2019). There is a need for an automated, or at least semi-automated, system to reduce human labour and processing time of images. Such an automated system would benefit not only projects focused on vultures or other similarly tagged large birds but would serve as an example for other projects where animals are individually marked and then resighted using camera traps.

Here we evaluate for the first time the performance of a semi-automated system based on camera trap images and a deep-learning algorithm to detect individually marked vultures from the multitude of images captured by camera traps. Specifically, we first assess the overall performance of the system in detecting images with a tagged vulture. Next, we assess how the system performs in detecting tags from images where the individual code is readable, which are therefore images that contribute to quantification of demographic rates based on detecting known individuals (Santangeli *et al.* 2020b). We then compare the time taken for manual screening of images with the time taken to screen only the images with a tag as identified by the system in order to measure the time saved. In this specific case, we consider the identification from camera trap images of Lappet-faced Vultures *Torgos tracheliotus* marked with a yellow plastic cattle ear tag; a very commonly used method for tagging vultures and other large birds across southern Africa and beyond (Birds of Prey Working Group 2006, Martin & Major 2010, Monadjem *et al.* 2012). For all such projects, a semi-automated tag identification system could offer practitioners enormous time savings when processing the camera trap material.

METHODS

Ethics statement

The work was conducted in accordance with all relevant national and international guidelines. The handling and wing tagging of nestlings were carried out by experienced bird ringers holding a valid ringing licence approved by the Namibian Ministry of Environment and Tourism, and following the guidelines for ringing provided by SAFRING (http://safring.adu.org.za/).

Field study design

The study was conducted in south-western Namibia, specifically within and around the Namib-Naukluft Park (24°14′05″S, 15°19′06″E). In this arid savannah landscape, the breeding population of the Lappet-faced Vultures has been monitored since 1991, when efforts to find vulture nests and ring the chicks began. These efforts increased until 2006 and have been largely stable since (Santangeli *et al.* 2018). From 2006, aerial surveys were used to search for active nests, followed by ground visits to fit the nestlings with a stainless-steel ring and to mark them using patagial wing tags with a unique code (Santangeli *et al.* 2018) to aid their subsequent individual identification. The same survey was performed every year from 2006.

Resightings of vultures with a wing tag (hereafter tag) were obtained from two sources: opportunistic observations by citizens, and observations from camera traps (Santangeli *et al.* 2020b). The latter started in 2014. Camera traps were placed around water points, which are regularly visited by vultures, particularly during the breeding season (September to January). During the first year (2014) a single camera trap was placed in one location, but the project was then progressively expanded to nine camera trap locations that have been recording continuously since their installation, taking pictures of any moving animal occurring at and around the water point during daylight. All images from the cameras have to date been manually scanned and inspected to identify tagged vultures, and to read the wing tag number. The information was then directly entered in a database that was recently used to estimate the survival rate of this species for the first time (Santangeli *et al.* 2020b). Overall, processing the hundreds of thousands of images over the years has amounted to an enormous burden of manual work. For this study, we used images collected by cameras at nine different sites located in the north-eastern part of the Namib-Naukluft Park.

Deep-learning algorithm for tag identification

To automatically detect a tagged vulture in an image and enclose the detected tag in a bounding box for better visibility than simply training a classification model, we trained a deep-learning object detection model to detect tags while minimizing the occurrence of false positives (hereafter false presences) and false negatives (hereafter false absences). In doing so, we chose a convolutional neural network approach (Zhang *et al.* 1990) and the YOLOv3 training program (Redmon & Farhadi 2018), as used in similar recent studies (Corcoran *et al.* 2019, Santangeli *et al.* 2020a).

We selected all images with vultures carrying a visible tag (positive images, n = 1379), and a sample of randomly selected images without a tag (negative images, n = 817). We selected more positive than negative images because the former are later split into three sets of images (see below), and because it is always recommended to have more training (positive) images than negative (testing) images for achieving a good performance in these types of models. We then manually labelled all positive images using ImageJ 1.8.0 (Rasband 2018) with bounding boxes (i.e. we drew a rectangle (bounding box) around each visible tag/-tags in each image; see Fig. 1 for examples). The size of the box was not fixed, because the size of the tag in each image differed depending on how close the bird was to the camera, and in what position. The main criterion for drawing the bounding box was that it had to be large enough to include the whole tag while minimizing the inclusion of other unnecessary parts of an image. We next extracted the coordinates of the top left and bottom right corner of each bounding box in each image and added them to a database along with the corresponding image name. This database was then used to inform the algorithm of the position of the object (the tag), in each image. This

Figure 1. Example images representing a vulture carrying a readable tag (left; whereby the tag code identifying the individual bird is readable from this image), and one with unreadable tag (right; whereby it is not possible to read the tag code). The bounding box and the percentage confidence that the object is indeed a tag (60.4% in the left image, 84% in the right image) are also shown as text on top of the bounding box. Images were cropped to facilitate their interpretation.

information was used for training (as well as testing and validating) the YOLOv3 model. Next, we allocated all positive images containing only one tag per image (n = 909 images) to a training set (i.e. used to train the algorithm to identify a tag). The remaining positive images with multiple tags per image were split into a validation set (n = 240 images used to select the best candidate neural network model) and a testing set (n = 230 images used for assessing the performance of the identified best candidate neural network model).

Training was executed for 500 epochs ('iterations'), whereby each training image passes forward and backward through the neural network. By executing multiple epochs (i.e. multiple 'learning sessions'), the algorithm progressively 'learns' how to recognize a tag (training program available at github.com/ultralytics/yolov3). During the training stage, each connection between neurons (i.e. nodes) of the network gets a weight value, which in practice represents the strength of the connection between neurons. After training was completed, a weights file, which included the final weight value stored by each neuron, was generated. We then developed a program that reads the configuration file of the YOLOv3 neural network architecture, which is a default basic architecture that is used as a start and can be modified. The program then reads the trained weights file to configure the neural network architecture for the purpose of this study within the PyTorch implementation that permits building of neural network architectures within a Python environment. Within PyTorch, we developed our own program to read the trained weights file and allow the construction of a deep-learning model specifically for tag detection. This was then validated among all other candidate models and tested against the testing set using various metrics (see Results section).

Through the testing, an output value was produced representing the confidence that an image included a tag. Values close to zero indicate an empty image (i.e. no tag) with a high confidence, whereas values close to one indicate that an image included a tag with a high confidence. In cases where multiple spots appeared as possible tags, the overall confidence for that image was taken from the spot with the highest confidence value.

Performance of the neural network in discriminating images as having a tag or not was assessed during both training and testing stages. In the training stage, we fine-tuned the intersection-over-union threshold (IoU) for each epoch. IoU is the ratio between the overlapping area and the union area of a proposed label (i.e. the candidate object identified by the model as a potential tag) and a ground truth label (the bounding box coordinates that included a tag as identified manually). Higher IoU indicates that the object of interest is more likely to be contained in the proposed detection. Simultaneously, we recorded and evaluated precision (propensity to minimize false positives), recall (propensity to minimize false negatives), mean average precision (the overall discrimination accuracy averaged over different detection threshold levels) and F1 score (the harmonic mean of precision and recall). The last metric was relevant because from the mathematical definitions of precision and recall, there exists a trade-off amid both metrics, resulting in the two metrics often having an inverse relationship (see Fig. 3 in the Results section), thus making both metrics uninformative. The F1 score can be viewed as the optimal balance between precision and recall, that is, the score at which false positives and false negatives are minimal. We also report loss, which shows the overall error value, that is, a measure of the mismatch between outcomes of the neural network (the candidate object identified by the model as a potential tag) and ground truth labels (the bounding box that included a tag as identified manually). The loss value was derived using loss functions predefined by the neural network. As training proceeds through the 500 epochs, the aforementioned four metrics are expected to grow and converge at a relatively high value, whereas loss is expected to decrease and converge to a low value (see Fig. 2).

By fine-tuning different sets of the model hyper-parameters (i.e. parameters that can be manually fine-tuned to find the optimal performance), multiple candidate models were generated. These alternative neural network models were then evaluated using the validation set (i.e. the set of 240 images, see above) to select the best model, indicated by F1 score, at a 40% confidence threshold (Fig. 3). This threshold indicates the cut-off point that minimizes the risk of false negatives and false positives. Our proposed model was selected from some 40 candidates and was then progressed into the final testing stage, during which we applied the testing set (consisting of 230 images) and recorded its performance in correctly retrieving relevant detections. Specifically, we recorded precision and recall, calculated the F1 score and plotted a precision–recall curve for confidence threshold across the range from 1% to 99% with 10% intervals.

Statistical analyses

We quantified whether the performance of the deep-learning algorithm differed between images where the tag code was at least partly readable and where it was unreadable. For this we used only the testing set of positive images (n = 230 images), that is, all images including a vulture with a tag that were not used for training the deep-learning algorithm. We ran a beta regression model (with logit link, which is appropriate for proportional data) with the response variable being the confidence probability (the maximum probability of any detection in an image if this had multiple detections) that an image included a tag as obtained from the deep-learning algorithm. As predictor we used a categorical variable with two classes representing whether the tag in the image was readable or not readable. The model was run using the glmmTMB package (Brooks *et al.* 2017) in R version 4.0.3 (R Core Development Team 2019). The least square means of the predictor classes were obtained using the EMMEANS package (Russell 2020).

Quantifying manual time savings using artificial intelligence

We attempted to estimate the time saved by using the deep-learning algorithm versus manually processing the images (the status quo). In doing so, we selected a representative sample of 100 images from across the thousands yielded by the camera traps in our study system. Next, four of the authors (one with multiple years of experience in manually processing the images), two with moderate experience (a few weeks of experience) and one largely unexperienced were asked to scan through the set of 100 images for those having a vulture with a tag, and to record the time that this scanning process lasted. This resulted in four different times, one per observer, which were then converted to a unit of value of time (days) for 100 images per day for one camera trap active through the year. The average (drawn from the values of the four observers) of days per year spent manually scanning the images for tagged vultures represents the net saving in time (days) that the deep-learning framework could yield by allowing practitioners to bypass this manual step.

RESULTS

Model performance, assessed based on metrics with respect to number of epochs trained (Fig. 2), was very good. Metrics were derived using the same training set at the end of each epoch. The model reached 70.2% precision, 98.5% recall (and therefore 82.0% F1 score) and 97.4% mean average precision at the final epoch (Fig. 2). The loss of the model significantly dropped after around the 80th epoch, indicating the point from which the neural network started 'grasping' the key features of tag objects and started to improve in performance, with test loss gradually converging to its minimal value at the last epoch (Fig. 2).

All testing metrics, acquired by repeatedly evaluating the model against the testing set (n = 230)

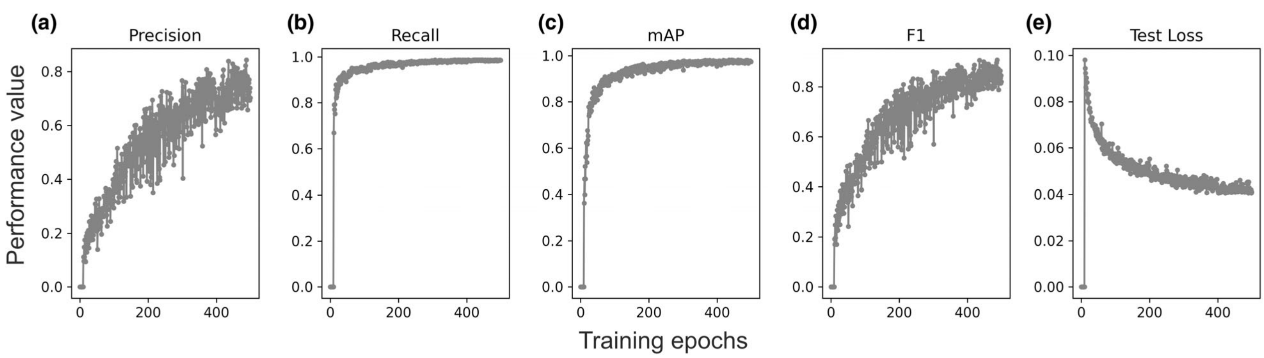

Figure 2. Training performance as measured by the proportion of images correctly identified by the neural network in relation to the number of epochs. Performance is shown based on five metrics: precision (a), recall (b), mean average precision (mAP) (c), F1 (d) and test loss score (e). See Methods for a detailed description of each of the performance metrics.

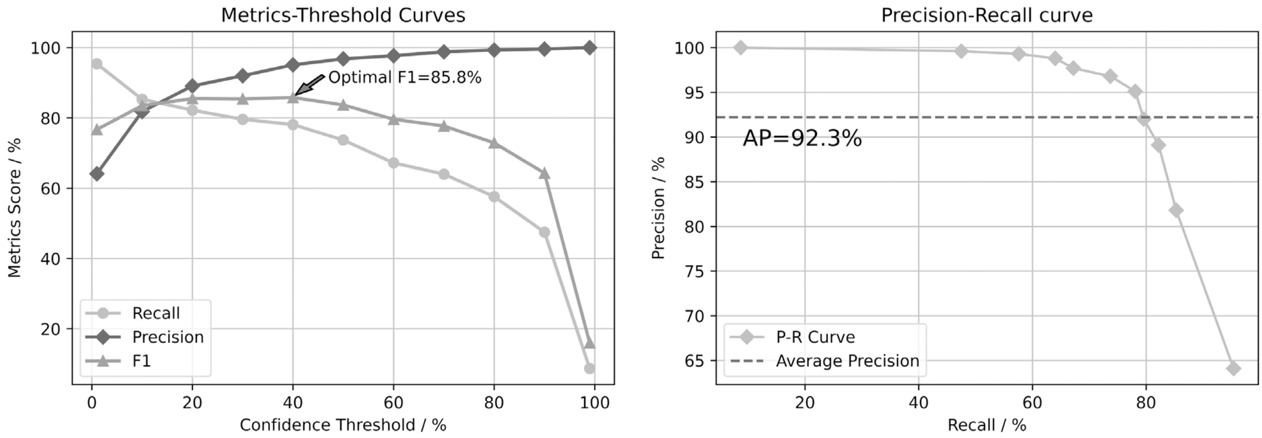

Figure 3. Testing performance as indicated by recall, precision and F1 scores on the left panel. The inverse relationship between recall and precision can be observed from this left panel, and therefore F1 is used to evaluate the overall model performance at each iteration. Optimal performance based on highest F1 score is reached at a 40% threshold, as indicated. The right panel shows the overall model performance curve (the precision–recall curve), where the average model precision value is shown by the dashed horizontal line.

from 1% to 99% for every 10% interval, suggest a very good model performance. Numbers of true positives, false positives and false negatives were recorded for each resultant image and summed to calculate recall, precision and F1 at each iteration, and the optimal performance, indicated by F1 score, was found at a 40% threshold (Fig. 3), reaching 85.8%, backed by a recall at 78.1% and a precision at 95.1%. Finally, by averaging the precision values over all iterations, we calculated that the mean average precision of this model was 92.3% (Fig. 3).

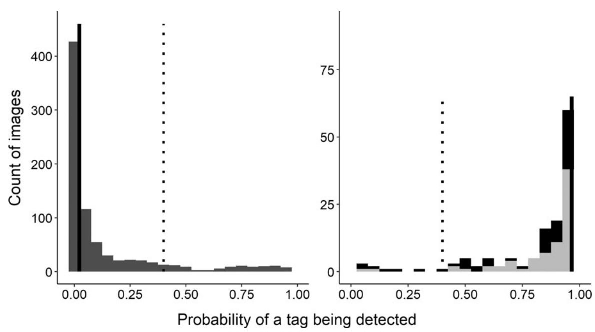

Figure 4. The frequency distribution of the probability of an image having a vulture with a tag in it as assigned by the deep-learning algorithm for images that do not include a tag (left panel, $n = 817$ negative images), and images that do include a tag (right panel, $n = 230$ positive images). These images, used for testing, were never seen by the algorithm for learning. The right panel shows the overall frequency of images with a tag (black bars) and the frequency of images in which the tag code is at least partly readable (light grey bars; see Methods for more details). The black, vertical, continuous line depicts the median value of the overall frequency in each plot, while the dotted vertical line denotes the cut-off value of 0.4, which is suggested here to automatically separate images as having or not having a tag.

Results of a beta regression model using positive images only (i.e. those with a tag) indicate that images with a readable tag were assigned a significantly higher ($z = 2.91$, $p = 0.004$) confidence value of containing a vulture with a tag compared with images with an unreadable tag (least squares mean ± se: 0.75 ± 0.02 and 0.83 ± 0.02, respectively; see examples in Fig. 4).

The estimated time saving per year by using the deep-learning algorithm through the web application could range between 254 and 578 h (equivalent to 11–24 full days) from processing the images of a single camera trap operating 365 days and yielding 100 images per day.

DISCUSSION

For the first time we show that a semi-automated system based on deep learning can accurately identify vultures with a tag, with performance being higher for readable tags compared with unreadable tags. A classification cut-off threshold of 0.4 would allow for the correct classification of almost all images with a tag, but a slightly lower classification accuracy is achieved for images without a tag. We show that time savings by using this semi-automated system could be substantial when cumulated over a year and across multiple camera traps.

Overall, the performance (precision and recall on both learning and testing images) reported for this system is comparable to that of other studies using deep-learning methods to efficiently detect wildlife from remotely sensed camera images (Norouzzadeh *et al.* 2018, Tabak *et al.* 2019, Pucci *et al.* 2020, Schneider *et al.* 2020, Santangeli *et al.* 2020a). The system presented here is based on a limited number of classified images used for training, and this is most likely the factor explaining the slightly lower performance of our system in correctly classifying negative images (i.e. those without a tag) than demonstrated by Schneider *et al.* (2020). However, we deem this level of performance to be acceptable for its purpose. We suggest that practitioners aiming to use the software based on this system should think critically about the trade-off between minimizing false negatives (i.e. classifying an image as not having a tag when it does have a tag), and minimizing false positives (i.e. classifying an image as having a tag when it does not). While we suggest using a default cut-off of 0.4 for discriminating images, this decision should be made in each case based on specific conditions. For example, a project may collect many camera trap images with a high number of tagged animals in them, but there may be a limited workforce to analyse the material. In this case, one may opt to minimize the occurrence of false positives, at the expense of having more false negatives (missing a tag when it was there). This would lower the number of images without a tag that are identified as having a tag, thereby reducing the manual workload for analysing them (i.e. wasting time reviewing many empty images), but it also means that a few more images with a tag would be missed.

We acknowledge that the system we have devised is semi-automated, as it only allows identification of tagged animals. Some manual work is still required to analyse those images identified by the deep-learning algorithm in order to read the tag code and build a database for future ecological analyses. However, even with the above limits, the system is a major step forward towards full automation of camera trap image processing. Overall, the system already saves large amounts of manual labour time and speeds up processing, ultimately reducing the time taken to complete ecological studies. In estimating the time saved by using the software, we did not account for the initial installation and learning to operate the software. We believe this should not take more than half a day, and this is a one-off procedure. Therefore, even if this initial time investment in the software set up and running is factored in, the net time savings from using it would still be close to the values we have reported.

Further research and development should go into making the system fully automated after the training phase, so that it will be possible not only to identify images with a tagged vulture, but also to read the tag code and build a database. This could be achieved, for example, by automating the code reading system in a similar manner as has been recently developed for the digitization of labelled museum specimens (Allan *et al.* 2019). Moreover, we emphasize that the deep-learning system presented here has been trained on the specific yellow cattle ear tag applied to vultures in Namibia, although this is very commonly used in several other projects on large birds across southern Africa in particular (Birds of Prey Working Group 2006). We welcome practitioners and users of this specific tag type to use the software we developed. We also encourage users with similar tag types (e.g. of a different colour or shape) to test the application and classification accuracy of the outcome, and judge for themselves if the outcome is satisfactory for their images.

CONCLUSIONS

The use of technology in ecology and conservation offers unprecedented opportunities, but also challenges (Pimm *et al.* 2015, Lahoz-Monfort *et al.* 2019). One such challenge stems from the processing of the 'big data' that technology can yield. Here we present a potential solution to one such challenge, a semi-automated system for the identification of marked animals from camera trap images. Although the performance of this system shows promising results, we encourage other researchers to build and expand on this example for other types of tags, and to potentially go further in automating this system. Such a development is much needed and urgent given the growing number of camera trap and wildlife marking programs (Murray & Sandercock 2020, Trefry *et al.* 2013, Wearn & Glover-Kapfer 2019), and the enormous amount of time currently required for manual processing of thousands or millions of camera trap images. We believe studies like ours represent key steps towards fully automating the collection and usage of large amounts of remotely collected ecological data.

We thank all the sponsors and volunteers assisting the programme run by Vultures Namibia for tagging vultures and for resighting them with camera traps, and particularly to Holger Kolberg and Peter Bridgeford for leading the programme. Three anonymous reviewers are also acknowledged for their constructive comments that helped to improve this work.

AUTHOR CONTRIBUTIONS

Andrea Santangeli: Conceptualization (lead); data curation (equal); formal analysis (equal); investigation (lead); methodology (equal); project administration (lead); supervision (lead); visualization (equal); writing – original draft (lead); writing – review and editing (lead). **Yuxuan Chen:** Formal analysis (equal); investigation (equal); methodology (equal); validation (equal); writing – review and editing (equal). **Mark Boorman:** Conceptualization (supporting); data curation (supporting); methodology (supporting); writing – review and editing (supporting). **Sofía Sales Ligero:** Data curation (supporting); methodology (equal); writing – review and editing (supporting). **Guillermo Albert García:** Data curation (supporting); methodology (equal); writing – review and editing (supporting).

FUNDING

None.

CONFLICT OF INTEREST

All of the authors declare that they have no competing financial or non-financial interests

ETHICAL NOTE

None.

Data Availability Statement

The data collected and analysed in the current study, including the camera trap images, as well as a user-friendly version of the software, are available from the corresponding author upon request.

REFERENCES

Allan, E.L., Livermore, L., Price, B.W., Shchedrina, O. & Smith, V.S. 2019. A novel automated mass digitisation workflow for natural history microscope slides. *Biodivers. Data J.* **7**: e32342.

Birds of Prey Working Group 2006. *Policy and Protocol for the Colour-Marking of Southern African Vultures*. Johannesburg, South Africa: Sasol & Endangered Wildlife Trust, Birds of Prey Working Group.

Brooks, M.E., Kristensen, K., van Benthem, K.J., Magnusson, A., Berg, C.W., Nielsen, A., Skaug, H.J., Machler, M. & Bolker, B.M. 2017. glmmTMB balances speed and flexibility among packages for zero-inflated generalized linear mixed modeling. *R J.* **9**: 378–400.

Conde, D.A., Staerk, J., Colchero, F., da Silva, R., Schöley, J., Baden, H.M., Jouvet, L., Fa, J.E., Syed, H., Jongejans, E., Meiri, S., Gaillard, J.-M., Chamberlain, S., Wilcken, J., Jones, O.R., Dahlgren, J.P., Steiner, U.K., Bland, L.M., Gomez-Mestre, I., Lebreton, J.-D., González Vargas, J., Flesness, N., Canudas-Romo, V., Salguero-Gómez, R., Byers, O., Berg, T.B., Scheuerlein, A., Devillard, S., Schigel, D.S., Ryder, O.A., Possingham, H.P., Baudisch, A. & Vaupel, J.W. 2019. Data gaps and opportunities for comparative and conservation biology. *Proc. Natl Acad. Sci.* **116**: 9658–9664.

Corcoran, E., Denman, S., Hanger, J., Wilson, B. & Hamilton, G. 2019. Automated detection of koalas using low-level aerial surveillance and machine learning. *Sci. Rep.* **9**: 3208.

Lahoz-Monfort, J.J., Chades, I., Davies, A., Fegraus, E., Game, E., Guillera-Arroita, G., Harcourt, R., Indraswari, K., McGowan, J., Oliver, J.L., Refisch, J., Rhodes, J., Roe, P., Rogers, A., Ward, A., Watson, D.M., Watson, J.E.M., Wintle, B.A. & Joppa, L. 2019. A call for international leadership and coordination to realize the potential of conservation technology. *Bioscience* **69**: 823–832.

Martin, J.M. & Major, R.E. 2010. The use of cattle ear-tags as patagial markers for large birds—a field assessment on adult and nestling Australian White ibis. *Waterbirds* **33**: 264–268.

Miao, Z., Gaynor, K.M., Wang, J., Liu, Z., Muellerklein, O., Norouzzadeh, M.S., McInturff, A., Bowie, R.C.K., Nathan, R., Yu, S.X. & Getz, W.M. 2019. Insights and approaches using deep learning to classify wildlife. *Sci. Rep.* **9**: 8137.

Monadjem, A., Kane, A., Botha, A., Dalton, D. & Kotze, A. 2012. Survival and population dynamics of the marabou stork in an isolated population, Swaziland. *PLOS One* **7**(9): e46434.

Murray, D.L. & Sandercock, B.K. 2020. *Population Ecology in Practice*. New York, USA: John Wiley & Sons.

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. *Proc. Natl Acad. Sci.* **115**: E5716–E5725.

Pimm, S.L., Alibhai, S., Bergl, R., Dehgan, A., Giri, C., Jewell, Z., Joppa, L., Kays, R. & Loarie, S. 2015. Emerging technologies to conserve biodiversity. *Trends Ecol. Evol.* **30**: 685–696.

Pucci, R., Shankaraiah, J., Jathanna, D., Karanth, U. & K. Subr. 2020. WhoAmI: An automatic tool for visual recognition of tiger and leopard individuals in the wild. arXiv preprint arXiv:.09962.

R Core Development Team 2019. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/

Rasband, W. 2018. *ImageJ*. Bethesda, MD, USA: National Institutes of Health.

Redmon, J. &A. Farhadi. 2018. YOLO v3: An Incremental Improvement. arXiv preprint arXiv:.08242.

Russell, L. 2020. Emmeans: Estimated marginal means, aka least-squares means. R package version 1.5.2-1. https://CRAN.R-project.org/package=emmeans.

Santangeli, A., Chen, Y., Kluen, E., Chirumamilla, R., Tiainen, J. & Loehr, J. 2020a. Integrating drone-borne thermal imaging with artificial intelligence to locate bird nests on agricultural land. *Sci. Rep.* **10**: 10993.

Santangeli, A., Pakanen, V.-M., Bridgeford, P., Boorman, M., Kolberg, H. & Sanz-Aguilar, A. 2020b. The relative contribution of camera trap technology and citizen science for estimating survival of an endangered African vulture. *Biol. Conserv.* **246**: 108593.

Santangeli, A., Spiegel, O., Bridgeford, P. & Girardello, M. 2018. Synergistic effect of land-use and vegetation greenness on vulture nestling body condition in arid ecosystems. *Sci. Rep.* **8**: 13027.

Schneider, S., Greenberg, S., Taylor, G.W. & Kremer, S.C. 2020. Three critical factors affecting automated image species recognition performance for camera traps. *Ecol. Evol.* **10**: 3503–3517.

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. 2019. Machine learning to classify animal species in camera trap images: Applications in ecology. *Methods Ecol. Evol.* **10**: 585–590.

Trefry, S.A., Diamond, A.W. & Jesson, L.K. 2013. Wing marker woes: a case study and meta-analysis of the impacts of wing and patagial tags. *J. Ornithol.* **154**: 1–11.

Wearn, O.R. & Glover-Kapfer, P. 2019. Snap happy: Camera traps are an effective sampling tool when compared with alternative methods. *R. Soc. Open Sci.* **6**: 181748.

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 Ecol. Evol.* **10**: 80–91.

Zhang, W., Itoh, K., Tanida, J. & Ichioka, Y. 1990. Parallel distributed processing model with local space-invariant interconnections and its optical architecture. *J. Appl. Opt.* **29**: 4790–4797.

Received 19 January 2021;
revision accepted 26 June 2022.
Associate Editor: Beatriz Arroyo