



APPLICATION

AIDE: Accelerating image-based ecological surveys with interactive machine learning

Benjamin Kellenberger^{1,2} | Devis Tuia^{1,3} | Dan Morris²

¹Laboratory of Geo-Information Science and Remote Sensing, Wageningen University & Research, Wageningen, The Netherlands

²Microsoft AI for Earth, Seattle, WA, USA

³Environmental Computational Science and Earth Observation Laboratory, Ecole Polytechnique Fédérale de Lausanne (EPFL), Sion, Switzerland

Correspondence

Benjamin Kellenberger
Email: benjamin.kellenberger@wur.nl

Funding information

NVIDIA Corporation

Handling Editor: Laura Graham

Abstract

1. Ecological surveys increasingly rely on large-scale image datasets, typically terabytes of imagery for a single survey. The ability to collect this volume of data allows surveys of unprecedented scale, at the cost of expansive volumes of photo-interpretation labour.
2. We present *Annotation Interface for Data-driven Ecology* (AIDE), an open-source web framework designed to alleviate the task of image annotation for ecological surveys. AIDE employs an easy-to-use and customisable labelling interface that supports multiple users, database storage and scalability to the cloud and/or multiple machines.
3. Moreover, AIDE closely integrates users and machine learning models into a feedback loop, where user-provided annotations are employed to re-train the model, and the latter is applied over unlabelled images to e.g. identify wildlife. These predictions are then presented to the users in optimised order, according to a customisable active learning criterion. AIDE has a number of deep learning models built-in, but also accepts custom model implementations.
4. Annotation Interface for Data-driven Ecology has the potential to greatly accelerate annotation tasks for a wide range of researches employing image data. AIDE is open-source and can be downloaded for free at https://github.com/microsoft/aerial_wildlife_detection.

KEYWORDS

applied ecology, conservation, monitoring (population ecology), population ecology, statistics, surveys

1 | INTRODUCTION

Ecological research has recently witnessed a tremendous increase in the usage of visual data: motion-triggered camera traps produce hundreds of millions of images worldwide (Swanson et al., 2015; Weinstein, 2015), unmanned aerial vehicles (UAVs) cover large areas with sub-decimeter resolution (Baxter & Hamilton, 2018; Linchant et al., 2015; Nowak et al., 2019), and in-field sound recorders capture spectrograms ('soundscapes') as a visual product in the terabytes for a

single project (Servick, 2014). Such visual data enables non-invasive estimation of different wildlife population characteristics, such as censuses through aerial surveys (Hodgson et al., 2016; Kellenberger et al., 2018; Rey et al., 2017), behaviour analyses (de Kort et al., 2018), and habitat monitoring (Stark et al., 2018). However, this high abundance of visual data may quickly result in large workloads for the photo-interpretation phase following data acquisition: researchers spend weeks manually identifying species in images, or significant amounts of money are invested to have the annotation work outsourced

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2020 The Authors. *Methods in Ecology and Evolution* published by John Wiley & Sons Ltd on behalf of British Ecological Society

(Torney et al., 2019). To this end, software solutions have been proposed, such as Trapper (Bubnicki et al., 2016), Aardwolf (Krishnappa & Turner, 2014) and camtrapR (Niedballa et al., 2016). While these facilitate data management, they lack *labelling assistance* and require users to carry out all annotation work manually. On a different track, some interfaces were designed with explicit focus on annotation, like VATIC (Vondrick et al., 2013), Labellmg,¹ VGG Image Annotator (Dutta & Zisserman, 2019), VIOLA (Bondi et al., 2017), LabelMe (Russell et al., 2008) and commercial tools like LabelBox.² A few of them have some form of simple annotation assistance; for example, both VATIC and VIOLA offer interpolation for video data to reduce the number of annotations required. However, more elaborate labelling assistance is often absent.

Recently, computer vision research has focused on automatically interpreting ecological imagery (Kellenberger et al., 2018; Norouzzadeh et al., 2018; Schneider et al., 2019; Tabak et al., 2019; Willi et al., 2019) through machine learning (ML) *models*, in particular convolutional neural networks (CNNs; LeCun et al., 2015). CNNs are a family of deep learning models designed for recognition tasks in images, such as image classification (Krizhevsky et al., 2012) or object detection (Lin, Goyal, et al., 2017), and have become the most widely used variant of ML models in computer vision tasks.³ However, employing these models requires substantial programming effort, as well as a very large collection of labelled images for training. In ecological applications, data acquisition campaigns often result in large quantities of images, but no annotations, which prevents CNN training. Furthermore, although methodologies like pre-training and transfer learning exist that can reduce the required number of images and annotations (Kornblith et al., 2019), obtaining a model that can generalise across an entire image dataset still requires large amounts of annotated data from the target image campaign. This can be attributed to the visual heterogeneity of the objects of interest in an image, as well as the images themselves: for example, objects (animals, plants, etc.) may exhibit viewpoint or pose variations, they may be of different sizes depending on their age and distance to the camera, or they might have different fur colours and patterns. Similarly, images may be taken with different camera models, resolutions or during the day or at night. ML models need to be exposed to these variations by means of training data, and labels, for them to be able to generalise and yield high-quality predictions throughout the full dataset. These data may not be readily available for image labelling campaigns, which limits the usefulness of CNNs, unless they can be included in the annotation process and incrementally trained on new annotations provided by the users.

In this work we address both problems—the tedium of manual photo-interpretation and the constraints of ML models—by unifying them into one labelling framework, which we denote *Annotation Interface for Data-driven Ecology* (AIDE). AIDE is a web-based, open-source collaboration platform that integrates a versatile labelling tool and ML models for image annotation, without the requirement of

writing code. The incorporation of ML models into annotation platforms has been proposed before, e.g. by the camera trap image tool Timelapse (Greenberg et al., 2019). However, AIDE does so by means of a feedback loop, leveraging a heuristic known as active learning (AL; Settles, 2009). In AIDE, the ML model is repeatedly trained on the latest, user-provided annotations. Once training has finished, the model is used to obtain predictions on (yet) unlabelled images. Critically, the images are further sorted by an AL criterion, which e.g. prioritises images that contain highly unconfident ML model predictions. The promise of using AL then is that a lower number of annotated images are required to train an ML model for the task at hand. AIDE has a number of CNN-based ML models and AL criteria built-in, but also accepts custom, user-provided implementations. The result is a collaborative platform that (a) has the potential to greatly accelerate large-scale image annotation projects and (b) allows training ML models with potentially lower amounts of training data. To the best of our knowledge, AIDE is the first open-source software suite that integrates ML models in an AL manner for image annotation.

2 | METHODS

2.1 | Overview

Annotation Interface for Data-driven Ecology is a web-based, collaborative annotation platform that includes humans and a prediction model in a loop, with both parties reinforcing each other for accelerated label retrieval. Figure 1 illustrates this loop and the key components of AIDE, including:

- *Labelling interface*, the primary access point for annotators and a window into the dataset to be annotated (Section 2.2).
- *Database*, the storage solution for annotations and metadata (Section 2.3).
- *Integrated model training*, which allows training an ML model on user-provided annotations and obtaining predictions in (yet) unlabelled images (Section 2.4).
- *Active learning (AL) criterion*, responsible for ordering the model predictions, e.g. to maximise model accuracy gain during re-training (Section 2.5).

By default, AIDE iterates this loop until the entire dataset has been annotated. The annotation process can also be terminated earlier, e.g. upon satisfactory prediction quality of the model. The following sections outline this loop and the individual components.

2.2 | Labelling interface

The labelling interface (Figure 2) is written in JavaScript with the jQuery library⁴ and is accessible through any modern web browser.

¹<https://github.com/tzutalin/labellmg>

²<https://labelbox.com>

³For a thorough introduction to CNNs, please refer to Goodfellow et al. (2016).

⁴<https://jquery.com>



FIGURE 1 Overview of the workflow in AIDE

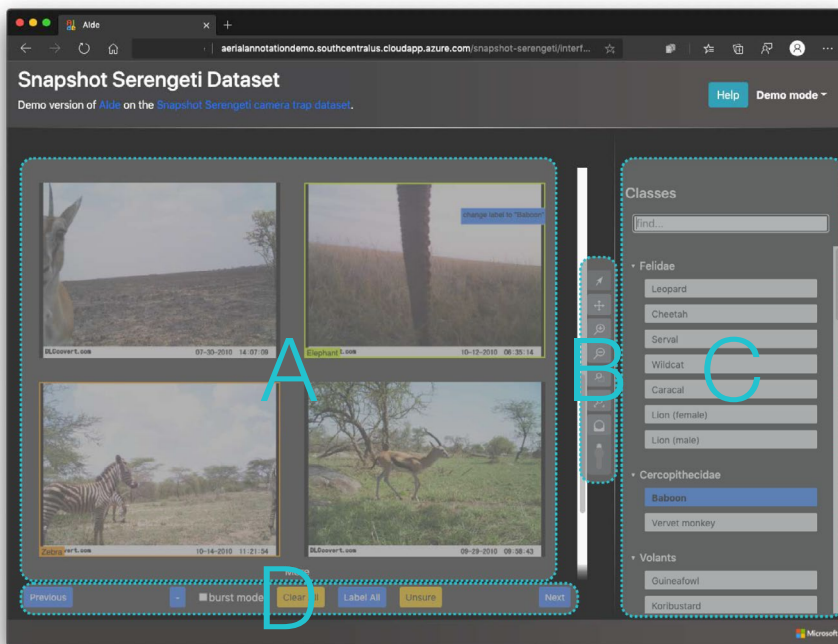


FIGURE 2 User interface of AIDE, with the main image viewer (A), viewing controls (B), the list of label classes (C) and annotation and navigation controls (D)

It is structured into the following parts: the main image viewer (A) with visualisation controls for zooming, panning, a loupe, etc. (B), the list of label classes defined for the current project (C), and controls to navigate through the images and create and modify annotations (D). Since the main target of AIDE is to obtain labels in the most efficient way, multi-step workflows, nested dialogues and pop-up messages have been avoided as much as possible.

2.2.1 | Annotation types

Annotation Interface for Data-driven Ecology supports a number of annotation types, namely image *labels*, *points* (with pixel coordinates), *bounding boxes* and *segmentation maps* (where every pixel

gets assigned a label). The interface and tool set are automatically adjusted depending on the annotation type selected for a project. AIDE has been designed to allow one type of annotation per project, rather than e.g. a fully customisable cascade of dialogues or annotation tags. This allows for a leaner annotation interface and more straightforward integration of the ML model (Section 2.4). Figure 3 illustrates examples of the interface set up for the four currently supported annotation types.

2.2.2 | Annotating images

Users can create, modify and delete annotations; the precise interaction depending on the annotation type. For instance, a click

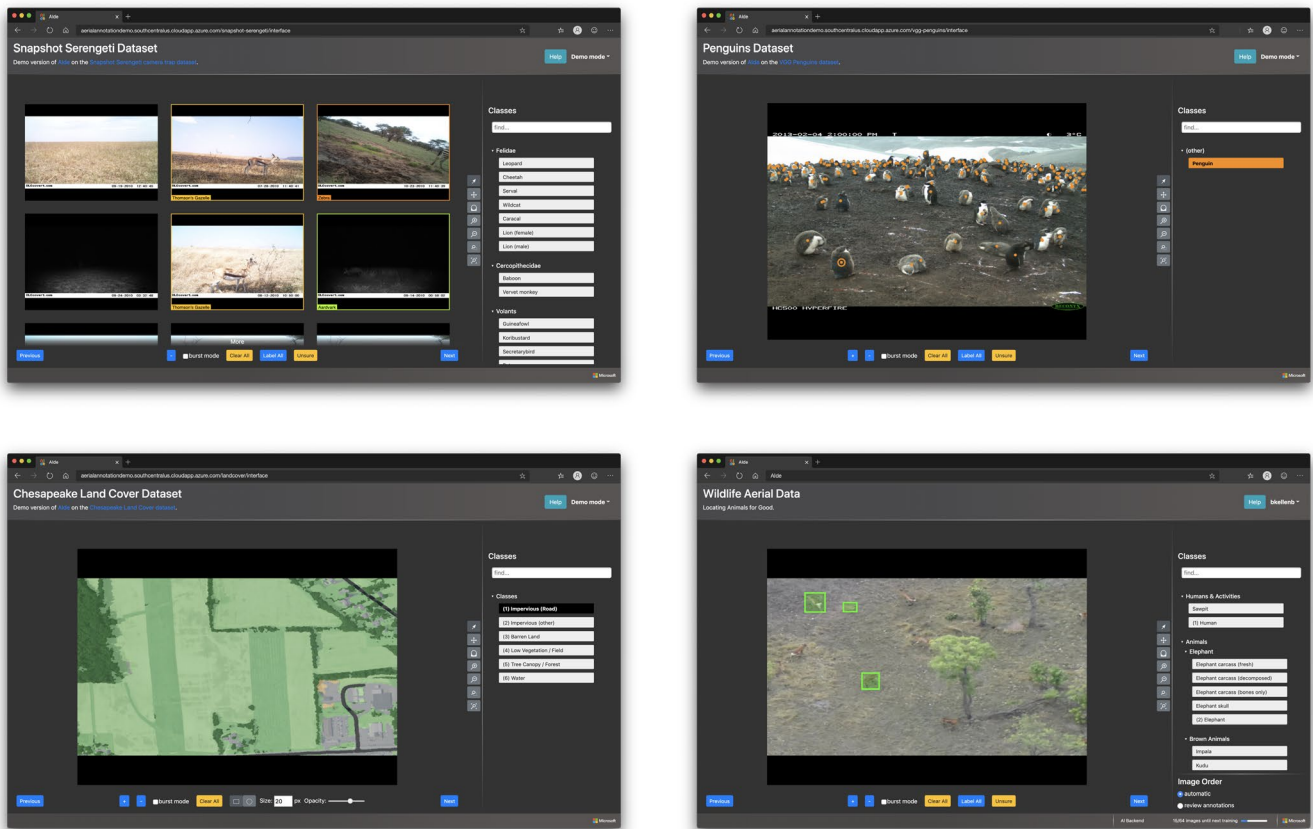


FIGURE 3 AIDE's labelling interface can be customised in many ways and supports multiple annotation types (clockwise, from top left): image labels, points, bounding boxes and segmentation masks

onto an image either assigns it to a label (for whole-image labelling projects), or else sets a point at the specified position (for point annotation projects). Clicking and dragging allows drawing and modifying bounding boxes, or painting or clearing a segmentation map.

Most of the labelling tools are assigned keyboard shortcuts, so that the user can keep their focus on the images, without having to look around to find the necessary tool. This also applies to the list of label classes, whose entries can be organised into hierarchical groups, collapsed and searched. For instance, the search field can also be accessed through a keystroke—this way, users can keep the mouse cursor in the image view, and select the desired label class through simple keyboard operations, without having to scroll through the list of classes.

After a user annotates a set of images, clicking 'Next' commits the annotations to the database (see Section 2.3.1 below) and presents a new set of images. Metadata related to the annotation process are stored as well, e.g. annotation author, image view count, date and time of creation, time required, browser agent, window size, number of interactions and more. Clicking 'Previous' re-displays the image (or batch of images, depending on the configuration) the user has seen before and allows modifying annotations therein. Finally, the platform also supports re-visiting existing annotations, filterable by date and annotation presence/absence to skip empty images.

2.3 | User and data management

2.3.1 | Server backend

Annotation Interface for Data-driven Ecology stores annotations and metadata in a relational database (RDB), specifically *Postgres*,⁵ an open-source database system. RDBs enable concurrent (i.e. multi-user) access, scalability and security on the one hand, but also facilitate tabular data download for further analyses on the other. Note that images are only referenced through the database, but stored as files on disk for easier organisation. Images can be uploaded and managed through the web browser; large images can automatically be split into patches on a regular grid during upload, if requested. Data input and output between the RDB and the annotation interface is handled by the server-sided logic of AIDE, which is written in Python and based around *bottle.py*, a lightweight web server engine.⁶

2.3.2 | User performance evaluation

Expertise and diligence of annotators may vary, which might become a challenge in collaborative labelling projects. To assist project

⁵<https://www.postgresql.org>

⁶<https://bottlepy.org>

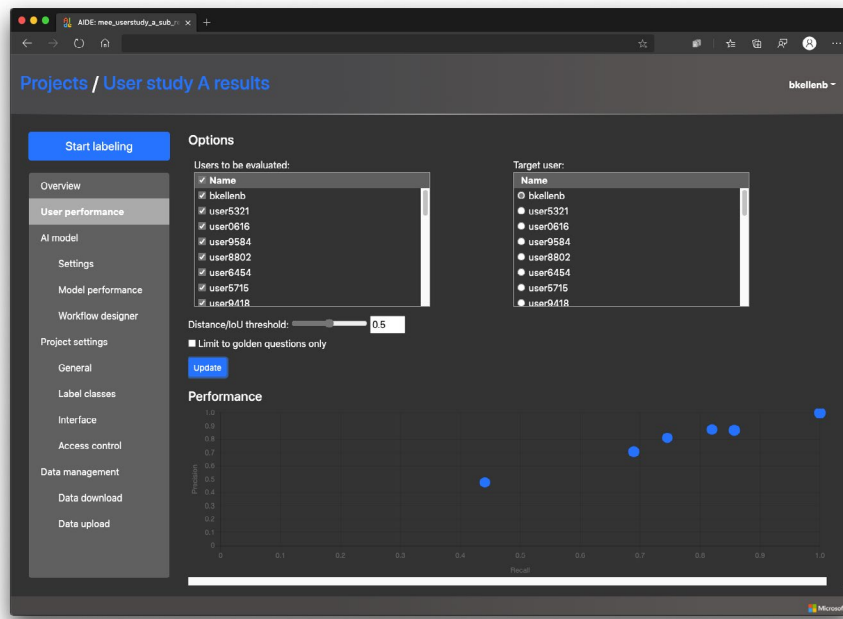


FIGURE 4 AIDE provides project administrators with a graphical interface to evaluate the annotation accuracy of users. In the example shown, bounding boxes drawn by annotators are compared to those of one selected target user and require a minimum IoU of 0.5 to be considered correct. AIDE then reports precision (y axis) and recall (x axis) values in a scatter plot (bottom)

administrators, AIDE offers tools for assessing the performance and annotation accuracy of users. All users' annotations can be compared to each other (including project administrators) through the web interface (Figure 4). The returned statistics are calculated on the server and adjusted to the annotation type: for image labels and segmentation masks, the overall accuracy is returned; for points and bounding boxes, AIDE provides precision and recall scores as well as average spatial point distances, resp. intersection-over-union (IoU) scores. Furthermore, AIDE also allows the specification of 'golden questions' which are images that serve as a reference for evaluation: project administrators can flag an arbitrarily large set of images as 'golden questions'. Every annotator then first sees only the golden question images when they begin with the labelling process in a specific project. The platform can further be configured to only allow new users to continue if they pass a certain accuracy criterion (e.g. a recall of 80% or more) on the golden questions, or after explicit admission by the project administrator.

2.4 | ML backend

At the heart of AIDE lies its capability of training ML models, based on the annotations provided by the users. Including ML models into the labelling process provides a number of potential advantages, such as:

1. Guidance: the model can draw the annotators' attention to parts of an image that look like the objects of interest, which might otherwise have been neglected.
2. Assistance: in the database, user annotations and model predictions are stored in different tables. However, the interface can be configured to automatically convert model predictions into annotations, which means that humans spend less time labelling targets that have already been identified by a sufficiently well-trained

model (Figure 5). User annotations, model predictions and predictions converted into annotations all have different drawing styles that can be customised to maintain transparency of the origin of an annotation. A welcome screen upon first launch of the interface further visualises and explains the different annotation and prediction types. Predictions that have been converted into annotations can be modified, or deleted, by the human; each annotation that originated from a model prediction is flagged as such by means of a metadata field in the RDB. Model predictions that are shown to the user can be filtered by a confidence threshold between zero (all predictions are shown) and 1 (all are hidden). Although we did not observe any speedups or accuracy improvement of annotators when showing predictions in the images in our tests, the option is available.

3. Acceleration: AIDE can alter the order of images based on the model predictions to e.g. prioritise particularly difficult images (i.e. with low-confidence predictions), or images with a high number of predictions (Figure 6).

Annotation Interface for Data-driven Ecology is designed to accommodate any ML model, as long as it can be trained in a supervised way on images annotated by the users of the interface. To this end, AIDE comes with a number of ML models built-in (Section 2.4.2), but also accepts third-party models (Section 2.4.3).

2.4.1 | Model training

Upon project creation, or later on, administrators can select one of the available model types that is compatible with their project's selected annotation and prediction types. AIDE has a number of ML models built-in, but those built-in models can be replaced by almost any user-provided ML model.

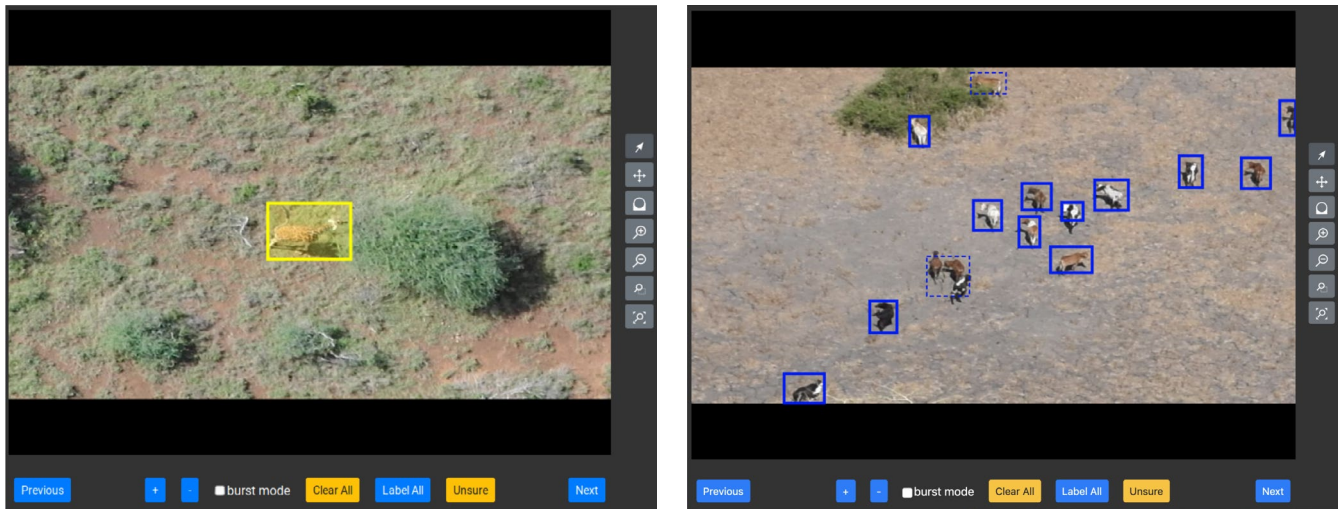
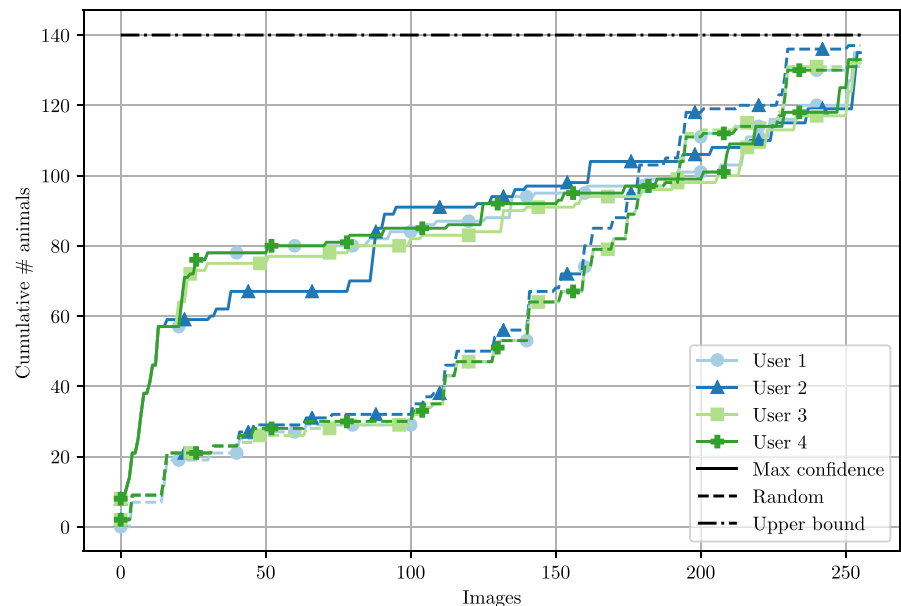


FIGURE 5 If properly trained, a prediction model can help reduce the annotator workload. In the example shown, the model successfully detected the giraffe (left) and most of the cattle (right), which means that no interaction is required for the left image, and only a few missing animals need to be annotated in the right image

FIGURE 6 Number of animals found by four volunteers in 256 images over the annotation course. AIDE allows using AL criteria (see Section 2.5) that prioritise high-confidence model detections. This can lead to faster retrieval of animals, which may be important for certain applications. See Appendix 5.3 for details on the study reported in the figure



By default, AIDE counts the number of images viewed by the annotators and the number of annotations made therein, and compares them to thresholds defined by the project administrator. Once the number of images and/or annotations made reaches the set thresholds, it automatically uses the latest annotations to (re-) train the selected ML model in the background, followed by a prediction pass ('inference') with the model in its latest state over unlabelled images. All parameters for automated training and inference, such as the number of images until the next re-training, can be customised through the interface for a project. To the annotators, the model's status is visible through a small notification panel, but does not interfere with the annotation process in any way. Also, AIDE can outsource the model training and inference process to one, or more, dedicated servers, for scalability. In the case of multiple connected servers, the images will be split and the

training or inference task distributed across all available machines for maximum performance. The labelling interface is available all-time without interruptions.

Annotation Interface for Data-driven Ecology further offers advanced controls for administrators to manually start arbitrarily complex sequences of training and inference through a graph-based interface (Figure 7). Furthermore, all statistical evaluation functionalities described in Section 2.3.2 are also available for evaluating model performance.

2.4.2 | Built-in models

Annotation Interface for Data-driven Ecology has a number of deep learning models built-in that have been shown to yield high performances on computer vision tasks. These include:

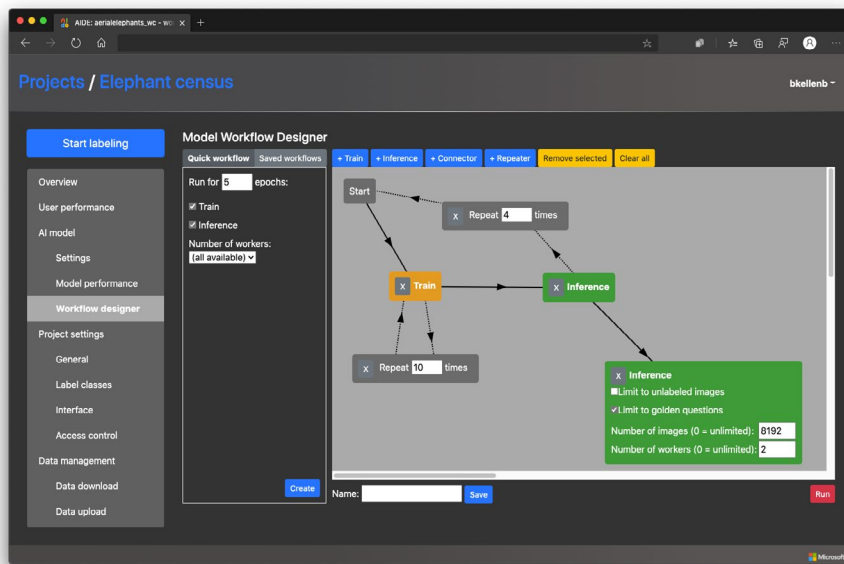


FIGURE 7 AIDE allows scheduling custom, arbitrary ML model training and inference sequences. The example sequence in the figure would train the model 11 times, then predict images, repeat this for four more times and then use the latest model state to infer the labels over all images flagged as 'golden questions' (see above)

- *ResNet*, a CNN for image classification (He et al., 2016). ResNet contains skip connections that can bypass an entire set of layers ('residual blocks'). This enabled training CNNs with more layers and resulted in a significant increase of accuracy in e.g. the ImageNet classification challenge (Deng et al., 2009). As a result, ResNet currently is one of the most popular architectures for image classification, including in ecological applications, where it has been used for species identification in camera trap imagery (Tabak et al., 2019) and bird call classification in spectrograms (Sankupellay & Konovalov, 2018). The implementation built-in to AIDE offers all common variants of ResNet, including ResNet-18, 34, 50, 101 and 152.
- *RetinaNet* for object detection and classification with bounding boxes (Lin, Goyal, et al., 2017). RetinaNet is an evolution of Faster R-CNN (Ren et al., 2015), which is widely used in computer vision research and ecology (Schneider et al., 2018). RetinaNet provides two advantages over Faster R-CNN: the first is a sequence of layers called 'Feature Pyramid Network' (Lin, Dollár, et al., 2017), which enables obtaining both high-resolution and semantically expressive features for each location in the image for object detection with high accuracy. The second is the 'Focal loss', which reduces the penalty for correct predictions whose confidence is not perfect, but is already good enough, allowing the model to become more robust to datasets that exhibit strong class imbalances. RetinaNet has been successfully used for aerial wildlife counting (Eikelboom et al., 2019) and coral detection (Modasshir et al., 2018).
- *U-Net* for semantic segmentation (Ronneberger et al., 2015).⁷ U-Net contains a sequence of encoder and decoder, which map the image to a lower spatial resolution, but high-dimensional features (encoder) and scale them back to high spatial resolution through transposed convolutions or interpolation (decoder). Like

ResNet, U-Net employs a form of skip connections between matching layers of the encoder and decoder for maximising information and gradient flow through the network, and hence performance. In terms of ecological applications, U-Net has been used to map forest types (Wagner et al., 2019) and habitats (Abrams et al., 2019), and to segment plant roots in soil images (Smith et al., 2020).

All models built-in to AIDE are implemented in PyTorch⁸ and are ready to be used with a few clicks through the web interface. All models can be configured to the needs of the project directly through the web browser (Figure 8); configuration parameters include e.g. the learning rate, optimiser type, type of ResNet model and more (see Appendix 5.1 for an example).

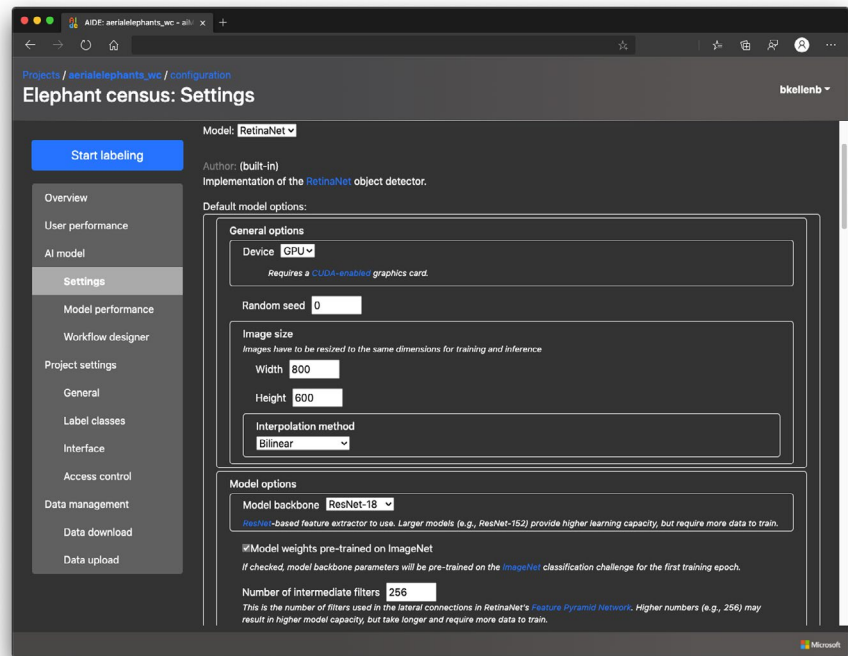
2.4.3 | Custom models

In some cases, the built-in models of AIDE might not be adequate, or else users of the system may already have an ML model available that they would like to use in the annotation process. For these cases, AIDE supports the integration of third-party models. To do so, users have to implement Python functions for training, inference, custom configuration parameters, etc., and make their model accessible within AIDE (see Appendix 5.2 for details). As long as the required Python functions are provided, any libraries, or even programming language, can be used. AIDE will embed the model and automatically handle all data I/O (annotations and predictions to and from the database, images, model states, etc.). Finally, all models, including third-party contributions, directly benefit from the model training and performance evaluation options discussed above.

⁷Semantic segmentation is the assignment of a label class to every pixel in an image.

⁸<https://pytorch.org>

FIGURE 8 AIDE allows configuring model options for each project through the web browser. If model settings are provided in the right format, they will be rendered with graphical elements and can incorporate explanation texts and links for each parameter; this is also available for third-party models (see Appendix 5.2)



2.5 | Active Learning for human-machine collaboration

In most ML workflows, a model is trained once on parts of a dataset and then kept static during a prediction phase on the rest of the images. While this may work if sufficient data have been labelled, it is less than optimal for situations where the initial number of existing annotations is low, or a model is to be re-used over e.g. a new set of images whose visual appearance is very different from the one in the training set. In this case, specific domain adaptation strategies can be devised to compensate for the domain shift (Tuia et al., 2016), but at the cost of custom-built ML models that are difficult to use for non-specialists.

Instead, AIDE integrates prediction models in an active learning (AL) loop (Kellenberger et al., 2019; Settles, 2009), also known as a 'human-in-the-loop' system (Brodley, 2017): humans begin labelling images, and after a (customisable) number of annotations have been made, these are automatically used to re-train the model. Model training is performed in the background, optionally on separate machines. This does not interfere with the annotation process; i.e., users can continue labelling while the model is updated with the images and annotations that were available when the re-training process was started. Once the training session has finished, the latest model state is committed to the RDB and employed to predict images that have not yet been reviewed by the annotators. The newly predicted images are then directly considered through an active learning (AL) criterion: AIDE can be configured to prioritise the order in which the images are presented to the user, e.g. by how many predictions, or by the confidence of such predictions. To this end, a number of AL criteria are built-in, including Breaking Ties (Luo et al., 2005), and sorting

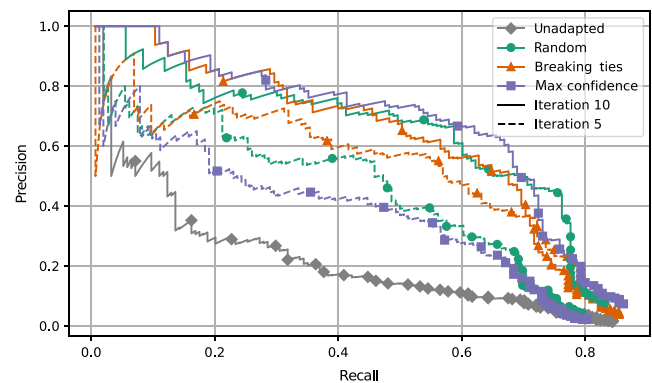


FIGURE 9 Precision-recall curves of an object detector CNN with initial performance (grey) and after five (dashed) and ten (solid) AL iterations with three different AL criteria

after maximum confidence. Like for ML models, AIDE also supports custom AL criteria. Once the chain of model training, prediction and ranking through the AL criterion is completed, the image entries in the database are updated with the priority score provided by the AL criterion. Then, as soon as the annotator(s) click 'Next' they are automatically presented with the newly predicted images, sorted by the priority score. In the end, this means that more relevant images are shown to the user with higher priority throughout the entire labelling process, with the notion of relevance depending on the task.

As an example, AL can be used to improve model performance after a given number of annotated images. Figure 9 shows precision-recall curves of CNNs on large mammal detection in aerial images, before fine-tuning (grey) and after five (dashed) and ten (solid) iterations with different AL criteria (see Appendix 5.4 for details). Note that the prediction quality of the CNN improves with all tested

criteria, including simple random image ordering (i.e. no AL) over the original base model (grey), but the improvement after only five AL iterations is the highest with a dedicated AL criterion (Breaking Ties; Luo et al., 2005).

3 | AIDE FOR COMMUNITY DEVELOPMENT

Integrating ML models into the labelling process eventually results in model states that are highly optimised for the data at hand. This is particularly helpful for large-scale image campaigns, where a well-trained model may result in reduced annotation efforts. However, the benefits of ML reach further: once trained, models can be used *across individual projects*. Oftentimes, ecologists conduct image campaigns with similar targets in mind, e.g. with images containing the same species, comparable types of background, or from the same viewpoint (ground-based, airborne, etc.). In these cases, re-using ML model states from other, similar projects provides a starting point that has the potential to accelerate labelling campaigns even further.

To this end, an upcoming release of AIDE will include a 'model marketplace' where users will be able to share trained ML model states across projects. At the start of each annotation project, users will be able to browse through a catalogue of available model states. Each state is accompanied with a description, a list of label classes the model supports and other related metadata. This way, users can select the most appropriate model state as a starting point and obtain even higher quality predictions straight from the start of the annotation process. Likewise, once a user decides that the model in their own project is sufficiently trained, they can decide to share its state with others by providing the mentioned metadata (name, description, etc.) and sharing it on the marketplace. For privacy reasons, only the aforementioned metadata and model parameters will be shared, which sufficiently prevents conclusions about the images of the originating project. Also, model states have to be shared explicitly by a project administrator and will be shareable either only across the administrator's own projects, or globally. Owners of the model states can further discard any information about the origin, such as their AIDE account name.

Eventually, we foresee AIDE and the model marketplace as a platform to enhance ecological image analysis in a collaborative way, beyond the individual project. Once a sufficient number of applications and image types have been covered by shared model states, labelling efforts will be reduced to a minimum for any new image campaign. This will enable ecologists to allot more time for the data interpretation, rather than the annotation process.

4 | LIMITATIONS OF AIDE

Annotation Interface for Data-driven Ecology was designed to enable large-scale, collaborative annotation projects for ecological

applications by means of interactive integration of ML models in an easy-to-use manner. Effectively, AIDE does not require users to write a single line of code, if they decide to use one of the built-in or contributed third-party models. However, AIDE is still a growing project, and as such has a number of limitations, including the following:

- Annotation Interface for Data-driven Ecology currently only supports RGB images and is not compatible with multi-band images, georeferenced data or other media types like videos.
- Only the four annotation types mentioned are supported at this moment. We plan to add compatibility for other types, such as more complex polygons or instance segmentation maps, in upcoming releases, and will also include appropriate ML models for them.
- Annotation Interface for Data-driven Ecology does not offer 'instantaneous' updates or predictions, i.e. live updates in an image on the screen after every click of the user. Rather, it is designed for projects with a high number of images where model updates are to be carried out after a number of images have been annotated.
- Annotation Interface for Data-driven Ecology relieves the user from having to write code, if they select one of the built-in models. However, training ML models still requires a certain degree of expert knowledge. AIDE does not offer any automation (e.g. hyperparameter search) or suggestions to this end, as many model training details depend on the data and objective at hand. However, users can evaluate different models through the built-in tools for model training and performance assessment.
- Models need to be trained to a certain degree on the data to be useful for interactive setups. In the case of deep learning models, this requires a comparably large set of existing labels, limiting their use at the start of annotation projects. If a new project is started with a completely untrained deep learning model, the latter will usually provide random labels per image, resp. per pixel in the case of image classification and semantic segmentation, or predictions in all possible locations of the image for points and bounding boxes. We intend to address this obstacle in a future release of AIDE through the 'model marketplace' as highlighted in Section 3.
- While AIDE offers tools to train ML models and evaluate model prediction and user performances (cf. Section 2.3.2), it does not guarantee high-quality annotations or well-performing ML models by itself. Eventually, it will always be the project administrators' responsibility to verify the accuracy of provided annotations, and to ensure that ML models are trained to the degree required for the individual annotation project.

Finally, we would like to note that AIDE is still work in progress and will grow in functionality over time. We hope to be able to deliver a solution that facilitates using ML models in as many ecological applications as possible.

5 | CONCLUSION

Ecological research increasingly relies on large-scale visual datasets, which can dramatically scale the spatial coverage of wildlife surveys, but requires tedious and expensive photo-interpretation of the images acquired. ML models, in particular convolutional neural networks (CNNs), have demonstrated high potential for accelerating this manual work. However, they often require involved coding efforts, which likely prevented broad adoption in many ecology projects.

In this study we presented *Annotation Interface for Data-driven Ecology* (AIDE), an open-source web framework that integrates a flexible and easy-to-use annotation platform with CNN-based prediction models. AIDE is a versatile labelling tool that offers a high degree of customisability, support for various annotation types and support for multiple users. It is also one of the first annotation platforms that employs ML models to assist annotators in their task. Critically, AIDE employs these models through active learning, where humans and the machine work hand-in-hand: humans provide annotations the model can learn from, and the model returns suggested predictions and prioritises images with respect to their relevance.

Annotation Interface for Data-driven Ecology is under active development, and will be expanded in functionalities in upcoming releases. This includes addressing the shortcomings mentioned like support for more annotation types, the ability to share pre-trained models across projects, as well as implementing new functionalities that have the potential to enhance image labelling projects for ecology.

Annotation Interface for Data-driven Ecology is an open-source platform that is free to use. The source code is available at https://github.com/microsoft/aerial_wildlife_detection.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the SAVMAP consortium (in particular Dr Friedrich Reinhard of Kuzikus Wildlife Reserve, Namibia) and the QCRI and Micromappers (in particular Dr Ferda Ofli and Ji Kim Lucas) for the support in the collection of ground truth data for the user studies. We would also like to thank Dr Howard Frederick for providing the image data used in parts of the screenshots shown in the manuscript. Finally, we gratefully acknowledge the support of the NVIDIA Corporation with the donation of a Titan V GPU used for this research.

AUTHORS' CONTRIBUTIONS

B.K. developed the principal framework, conducted the user study and led the writing of the manuscript; D.T. conceptualised and framed the principal idea of using active learning for wildlife conservation; D.M. supervised the software development and hosted B.K. during the principal development phase. All authors contributed critically to the drafts and gave final approval for publication.



PEER REVIEW

The peer review history for this article is available at <https://publons.com/publon/10.1111/2041-210X.13489>.

DATA AVAILABILITY STATEMENT

The proposed platform (AIDE) is open source and available for download at https://github.com/microsoft/aerial_wildlife_detection. The version of AIDE used in this manuscript (Microsoft, 2020) can be obtained at <https://doi.org/10.5281/zenodo.4028309>. Note that this is a frozen code that will not contain the latest updates and developments beyond the state at publication of this manuscript. For the official and latest release, please refer to the official GitHub link. The images used for the studies behind Figures 6 and 9 are available at <https://doi.org/10.5281/zenodo.1204408> (Reinhard et al., 2015). Labels are available from the authors upon request.

ORCID

Benjamin Kellenberger  <https://orcid.org/0000-0002-2902-2014>
Devis Tuia  <https://orcid.org/0000-0003-0374-2459>

REFERENCES

- Abrams, J. F., Vashishtha, A., Wong, S. T., Nguyen, A., Mohamed, A., Wieser, S., Kuijper, A., Wilting, A., & Mukhopadhyay, A. (2019). Habitat-net: Segmentation of habitat images using deep learning. *Ecological Informatics*, 51, 121–128. <https://doi.org/10.1016/j.ecoinf.2019.01.009>
- Baxter, P. W., & Hamilton, G. (2018). Learning to fly: Integrating spatial ecology with unmanned aerial vehicle surveys. *Ecosphere*, 9(4). <https://doi.org/10.1002/ecs2.2194>
- Bondi, E., Fang, F., Kar, D., Noronha, V., Dmello, D., Tambe, M., Iyer, A., & Hannaford, R. (2017). VIOLA: Video labeling application for security domains. In S. Rass, B. An, C. Kiekintveld, F. Fang, & S. Schauer (Eds.), *International conference on decision and game theory for security* (pp. 377–396). Springer.
- Brodley, C. E. (2017). Human-in-the-loop applied machine learning. In J. Y. Nie, Z. Obradovic, T. Suzumura, R. Ghosh, R. Nambiar, C. Wang, H. Zang, R. Baeza-Yates, X. Hu, J. Kepner, A. Cuzzocrea, J. Tang, & M. Toyoda (Eds.), *2017 IEEE International Conference on Big Data (Big Data)* (p. 1). IEEE.
- Bubnicki, J. W., Churski, M., & Kuijper, D. P. (2016). Trapper: An open source web-based application to manage camera trapping projects. *Methods in Ecology and Evolution*, 7(10), 1209–1216.
- de Kort, D., Altrichter, M., Cortez, S., & Camino, M. (2018). Collared peccary (*Pecari tajacu*) behavioral reactions toward a dead member of the herd. *Ethology*, 124(2), 131–134.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In D. Huttenlocher, G. Medioni, & J. Rehg (Eds.), *2009 IEEE conference on computer vision and pattern recognition* (pp. 248–255). IEEE.
- Dutta, A., & Zisserman, A. (2019). The VIA annotation software for images, audio and video. In L. Amsaleg, B. Huet, & M. Larson (Eds.), *Proceedings of the 27th ACM International Conference on Multimedia*. MM '19. ACM.
- Eikelboom, J. A., Wind, J., van de Ven, E., Kenana, L. M., Schroder, B., de Knegt, H. J., van Langevelde, F., & Prins, H. H. (2019). Improving the precision and accuracy of animal population estimates with aerial image object detection. *Methods in Ecology and Evolution*, 10(11), 1875–1887. <https://doi.org/10.1111/2041-210X.13277>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

- Greenberg, S., Godin, T., & Whittington, J. (2019). Design patterns for wildlife-related camera trap image analysis. *Ecology and Evolution*, 9(24), 13706–13730. <https://doi.org/10.1002/ece3.5767>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In R. Bajcsy, F. F. Li, & T. Tuytelaars (Eds.), *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778). IEEE.
- Hodgson, J. C., Baylis, S. M., Mott, R., Herrod, A., & Clarke, R. H. (2016). Precision wildlife monitoring using unmanned aerial vehicles. *Scientific Reports*, 6, 22574. <https://doi.org/10.1038/srep22574>
- Kellenberger, B., Marcos, D., Lobry, S., & Tuia, D. (2019). Half a percent of labels is enough: Efficient animal detection in UAV imagery using deep CNNs and active learning. *IEEE Transactions on Geoscience and Remote Sensing*, 57(12), 9524–9533. <https://doi.org/10.1109/TGRS.2019.2927393>
- Kellenberger, B., Marcos, D., & Tuia, D. (2018). Detecting mammals in UAV images: Best practices to address a substantially imbalanced dataset with deep learning. *Remote Sensing of Environment*, 216, 139–153. <https://doi.org/10.1016/j.rse.2018.06.028>
- Kornblith, S., Shlens, J., & Le, Q. V. (2019). Do better imagenet models transfer better? In L. Davis, P. Torr, & S. C. Zhu (Eds.), *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2661–2671). IEEE.
- Krishnappa, Y. S., & Turner, W. C. (2014). Software for minimalistic data management in large camera trap studies. *Ecological Informatics*, 24, 11–16. <https://doi.org/10.1016/j.ecoinf.2014.06.004>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In P. Bartlett & F. Pereira (Eds.), *Advances in neural information processing systems* (pp. 1097–1105). Curran Associates, Inc.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539>
- Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In R. Chellappa, Z. Zhang, & A. Hoogs (Eds.), *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117–2125). IEEE.
- Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In R. Chellappa, Z. Zhang, & A. Hoogs (Eds.), *Proceedings of the IEEE International Conference on computer vision* (pp. 2980–2988). IEEE.
- Linchant, J., Lisein, J., Semeki, J., Lejeune, P., & Vermeulen, C. (2015). Are unmanned aircraft systems the future of wildlife monitoring? A review of accomplishments and challenges. *Mammal Review*, 45(4), 239–252.
- Luo, T., Kramer, K., Goldgof, D. B., Hall, L. O., Samson, S., Remsen, A., & Hopkins, T. (2005). Active learning to recognize multiple types of plankton. *Journal of Machine Learning Research*, 6(April), 589–613.
- Microsoft. (2020). AIDE: Annotation interface for data-driven ecology. Release for Manuscript. <https://doi.org/10.5281/zenodo.4028309>
- Modasshir, M., Rahman, S., Youngquist, O., & Rekleitis, I. (2018). Coral identification and counting with an autonomous underwater vehicle. In H. Zhang (Ed.), *2018 IEEE International Conference on robotics and biomimetics (ROBIO)*. IEEE.
- Niedballa, J., Sollmann, R., Courtiol, A., & Wilting, A. (2016). camtrapR: An R package for efficient camera trap data management. *Methods in Ecology and Evolution*, 7(12), 1457–1462.
- 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 of the United States of America*, 115(25), E5716–E5725.
- Nowak, M. M., Dziób, K., & Bogawski, P. (2019). Unmanned aerial vehicles (UAVs) in environmental biology: A review. *European Journal of Ecology*, 4(2), 56–74.
- Reinhard, F., Parkan, M., Produit, T., Betschart, S., Bacchilega, B., Hauptfleisch, M. L., Meier, P., SAVMAP Consortium, & Joost, S. (2015). Near real-time ultrahigh-resolution imaging from unmanned aerial vehicles for sustainable land use management and biodiversity conservation in semi-arid savanna under regional and global change (SAVMAP) (Version 2.0). *Zenodo*, <https://doi.org/10.5281/zenodo.1204408>
- Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes & N. D. Lawrence (Eds.), *Advances in neural information processing systems* (pp. 91–99). Curran Associates, Inc.
- Rey, N., Volpi, M., Joost, S., & Tuia, D. (2017). Detecting animals in African savanna with UAVs and the crowds. *Remote Sensing of Environment*, 200, 341–351.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *International Conference on Medical image computing and computer-assisted intervention* (pp. 234–241). Springer.
- Russell, B. C., Torralba, A., Murphy, K. P., & Freeman, W. T. (2008). LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1–3), 157–173. <https://doi.org/10.1007/s11263-007-0090-8>
- Sankupellay, M., & Konovalov, D. (2018). Bird call recognition using deep convolutional neural network, ResNet-50. In *Proceedings of the AAS2018 Acoustics Conference* (Vol. 7). Retrieved from https://acoustics.asn.au/conference_proceedings/AAS2018/
- Schneider, S., Taylor, G. W., & Kremer, S. (2018). Deep learning object detection methods for ecological camera trap data. In J. Elder & A. Xu (Eds.), *2018 15th Conference on computer and robot vision (CRV)* (pp. 321–328). IEEE.
- Schneider, S., Taylor, G. W., Linquist, S., & Kremer, S. C. (2019). Past, present and future approaches using computer vision for animal re-identification from camera trap data. *Methods in Ecology and Evolution*, 10(4), 461–470. <https://doi.org/10.1111/2041-210X.13133>
- Servick, K. (2014). Eavesdropping on ecosystems. *Science*, 343(6173), 834–837. <https://doi.org/10.1126/science.343.6173.834>
- Settles, B. (2009). *Active learning literature survey*. Technical report. University of Wisconsin-Madison Department of Computer Sciences.
- Smith, A. G., Petersen, J., Selvan, R., & Rasmussen, C. R. (2020). Segmentation of roots in soil with u-net. *Plant Methods*, 16(1), 1–15. <https://doi.org/10.1186/s13007-020-0563-0>
- Stark, D. J., Vaughan, I. P., Evans, L. J., Kler, H., & Goossens, B. (2018). Combining drones and satellite tracking as an effective tool for informing policy change in riparian habitats: A proboscis monkey case study. *Remote Sensing in Ecology and Conservation*, 4(1), 44–52. <https://doi.org/10.1002/rse2.51>
- 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, 150026. <https://doi.org/10.1038/sdata.2015.26>
- 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., & Teton, B. (2019). Machine learning to classify animal species in camera trap images: Applications in ecology. *Methods in Ecology and Evolution*, 10(4), 585–590.
- Torney, C. J., Lloyd-Jones, D. J., Chevallier, M., Moyer, D. C., Maliti, H. T., Mwita, M., Kohi, E. M., & Hopcraft, G. C. (2019). A comparison of deep learning and citizen science techniques for counting wildlife in aerial survey images. *Methods in Ecology and Evolution*, 10(6), 779–787.
- Tuia, D., Persello, C., & Bruzzone, L. (2016). Recent advances in domain adaptation for the classification of remote sensing data. *IEEE Geoscience and Remote Sensing Magazine*, 4(2), 41–57.
- Vondrick, C., Patterson, D., & Ramanan, D. (2013). Efficiently scaling up crowdsourced video annotation. *International Journal of Computer Vision*, 101, 184–204.

- Wagner, F. H., Sanchez, A., Tarabalka, Y., Lotte, R. G., Ferreira, M. P., Aïdar, M. P., Gloor, E., Phillips, O. L., & Aragão, L. E. (2019). Using the u-net convolutional network to map forest types and disturbance in the atlantic rainforest with very high resolution images. *Remote Sensing in Ecology and Conservation*, 5, 360–375.
- Weinstein, B. G. (2015). Motionmeerkat: Integrating motion video detection and ecological monitoring. *Methods in Ecology and Evolution*, 6(3), 357–362.
- Willi, M., Pitman, R. T., Cardoso, A. W., Locke, C., Swanson, A., Boyer, A., Veldhuis, 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.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section.

How to cite this article: Kellenberger B, Tuia D, Morris D. AIDE: Accelerating image-based ecological surveys with interactive machine learning. *Methods Ecol Evol*. 2020;11:1716–1727. <https://doi.org/10.1111/2041-210X.13489>