The Journal of Open Source Software

EcoAssist: A no-code platform to train and deploy custom YOLOv5 object detection models

Peter van Lunteren [1]

1 Independent Researcher, The Netherlands

DOI: 10.21105/joss.05581

Software

- Review
- Repository
- Archive

Editor: Marcel Stimberg

Reviewers:

- @animikhaich
- @oparisot

Submitted: 11 May 2023
Published: 04 August 2023

Summary

`EcoAssist` is a tkinter based graphical user interface (GUI) which is designed to enable object detection for novices. The package allows the user to annotate images, train and deploy custom models, and post-process imagery without having to write code (Figure 1). Installation is automated and dependencies will be installed in a virtual environment to avoid conflicts. Annotation is done via the open-source package `labelImg` (Heartexlabs, 2022) and post-processing features include folder separation, detection visualization, cropping, label creation, and exporting results to CSV files. It will automatically run on NVIDIA or Apple Silicon GPU and is available for Microsoft Windows, Apple macOS, and Linux. The main target audience is ecologists working with motion triggered camera traps, although it is not limited to detect animals as object detection models are generic. The package is developed to be used in conjunction with Timelapse (Greenberg et al., 2019) (an image analyzer for camera traps) and MegaDetector (Beery et al., 2019) (a trained object detection model to localize animals in camera trap imagery). More features are planned and will be added in the future.

Statement of need

Given the unprecedented global decline of animal diversity (Ceballos et al., 2020), conservationists urgently need tools for accurate and fast assessment of wildlife diversity (Tuia et al., 2022). A commonly used method for this is deploying ecological motion triggered camera traps (Schneider et al., 2018). However, the analysis of such data is often expensive, labor-intensive, and time-consuming (Beery et al., 2019). Fortunately, optimizing this workflow with machine learning techniques has proven to be an effective method (Gomez Villa et al., 2017; Norouzzadeh et al., 2018; Schneider et al., 2018).

One such example is the MegaDetector model (Beery et al., 2019). This model is able to drastically reduce the workload by facilitating the removal of empty camera trap images. Although the model does not identify the species, it offers a simplification of the annotation process for users wanting to train their own project-specific species classifier by providing bounding box coordinates. Still, the only way to interact with the model is exclusively through Python code and command-line interfaces. Users without a programming background might find it difficult to implement this software. These users, thus, might miss out on valuable and open-source techniques without GUIs such as `EcoAssist` - which are designed to overcome this limitation.

Besides `EcoAssist` there are three other GUIs able to deploy the MegaDetector model (i.e., Evans, 2023; Gyurov, 2022; McWilliam, 2021). However, none of these packages offer features to annotate and train custom models, nor do they accept the deployment of custom models. Furthermore, Gyurov (2022) and McWilliam (2021) do not offer functionality on systems other than Microsoft Windows.

Generic platform

Although `EcoAssist` was originally designed for ecologists working with camera trap imagery, it evolved to be a more generic platform to be used by any researcher wanting to work with object detection models. All features are available with any kind of object, which makes it an interesting tool for many academic disciplines. The package has proved its ease of use and ability to efficiently analyze large datasets and is currently used by dozens of research institutions worldwide. Its user-friendly design, simplicity, and support will likely extract further scientific interest.

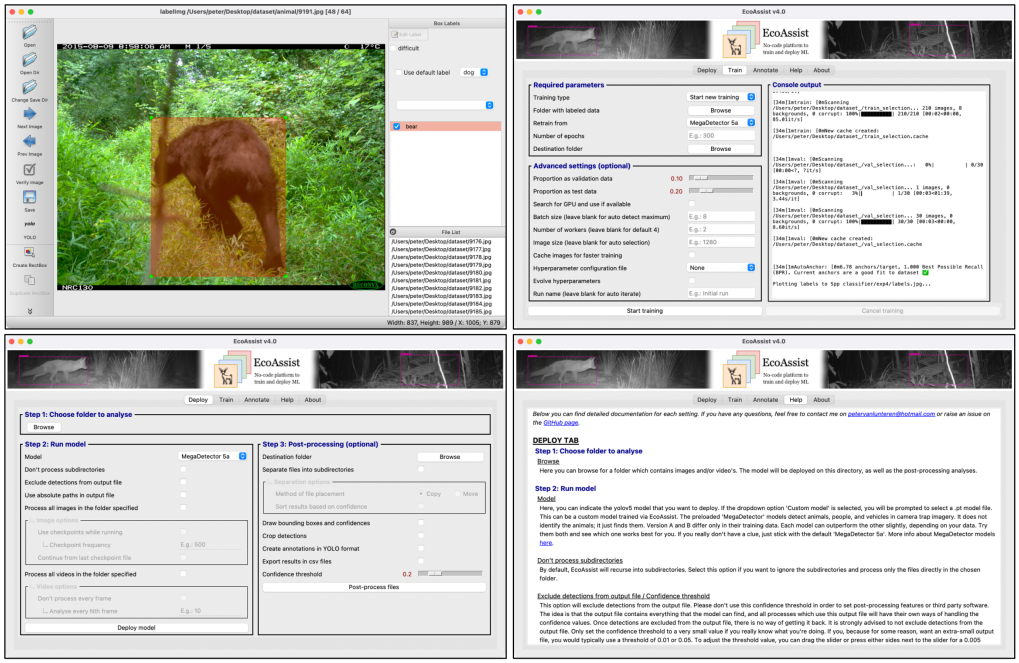

Figure 1: The annotation (top left), training (top right), deployment (bottom left), and documentation window (bottom right).

References

Beery, S., Morris, D., & Yang, S. (2019). *Efficient pipeline for camera trap image review*. https://arxiv.org/abs/1907.06772

Ceballos, G., Ehrlich, P. R., & Raven, P. H. (2020). Vertebrates on the brink as indicators of biological annihilation and the sixth mass extinction. *Proceedings of the National Academy of Sciences*, *117*(24), 13596–13602. https://doi.org/10.1073/pnas.1922686117

Evans, B. C. (2023). CamTrap-detector: Detect animals, humans and vehicles in camera trap imagery. In *GitHub repository*. GitHub. https://github.com/bencevans/camtrap-detector

Gomez Villa, A., Salazar, A., & Vargas, F. (2017). Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks. *Ecological Informatics*, *41*, 24–32. https://doi.org/10.1016/j.ecoinf.2017.07.004

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

Gyurov, P. (2022). MegaDetector-GUI: A desktop application that makes using MegaDetector's model easier. In *GitHub repository*. GitHub. https://github.com/petargyurov/

megadetector-gui

Heartexlabs. (2022). labelImg: Label studio is a modern, multi-modal data annotation tool. In *GitHub repository*. GitHub. https://github.com/heartexlabs/labelImg

McWilliam, N. (2021). MegaDetector-interface: GUI created for the microsoft MegaDetector. In *GitHub repository*. GitHub. https://github.com/NaomiMcWilliam/Megadetector-Interface

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.

Schneider, S., Taylor, G. W., & Kremer, S. (2018, May). Deep learning object detection methods for ecological camera trap data. *2018 15th Conference on Computer and Robot Vision (CRV)*. https://doi.org/10.1109/crv.2018.00052

Tuia, D., Kellenberger, B., Beery, S., Costelloe, B. R., Zuffi, S., Risse, B., Mathis, A., Mathis, M. W., Langevelde, F. van, Burghardt, T., Kays, R., Klinck, H., Wikelski, M., Couzin, I. D., Horn, G. van, Crofoot, M. C., Stewart, C. V., & Berger-Wolf, T. (2022). Perspectives in machine learning for wildlife conservation. *Nature Communications*, *13*(1), 792. https://doi.org/10.1038/s41467-022-27980-y