APPENDIX S1: Technical details

Manuscript Title: Trapper Citizen Science: an open-source camera trap platform for citizen science in wildlife research and management

Authors: Magali Frauendorf, Jakub W. Bubnicki, Filip Ånöstam, Piotr Tynecki, Łukasz Wałejko, Joris P. G. M. Cromsigt, Fredrik Widemo, Tim R. Hofmeester

Text S1: Trapper backend

Trapper CS is an extension of the open-source Trapper platform, originally developed by (Bubnicki et al. 2016) as an end-to-end database system for managing camera trap data. Built with Python, Django and Docker, Trapper provides a robust and scalable multi-platform (supporting AMD64 and ARM64 architectures) web-based system for managing the large-scale data generated by camera traps. These datasets can be voluminous and complex, even for smaller projects, due to the multimedia nature of the collected data. Without a comprehensive data management solution, the potential for delays and long-term data inaccessibility is high. Trapper supports collaborative research projects with the option of various access levels and encourages data sharing among users, which enhances data accessibility and project outcomes. By leveraging widely adopted open-source software components, Trapper offers a flexible and extensible data management framework for camera trap projects.

Currently, Trapper 2.0 is available as an open beta version. The platform is actively developed with regular updates for modules such as Citizen Science (frontend part), Trapper AI (Trapper AI Manager and Trapper AI Worker) and Trapper Expert (core backend module with its own expert interface). Its core features include:

- Open-source accessibility under GPLv3, available for research, academic and wildlife conservation (NGOs) use
- A spatially-enabled database backend capable of managing and processing both images and videos
- A flexible classification model that supports both standardized and custom attributes (defined per project) as well as multiple AI-based and expert-driven classifications
- Compliance with the Camtrap DP standard, promoting data reuse and interoperability and feeding automatic data analysis pipelines
- Support for collaborative project work, access rights and data sharing
- Availability of Jupyter Notebooks and an API for enhanced analytical and data processing capabilities

Trapper can be installed locally, on a Virtual Private Server (VPS) or in academic or commercial cloud environments such as Azure or AWS and is compatible with all major cloud storage providers.

Trapper has been adopted across various major institutions in Europe, including the Mammal Research Institute of the Polish Academy of Sciences, the Swedish University of Agricultural Sciences, KORA - Carnivore Ecology and Wildlife Management, Bavarian Forest National

Park as well as numerous research groups and individual ecologists. All Trapper modules are maintained by the Open Science Conservation Fund with financial and coding support from many international partners.

Text S2: AI model

Trapper AI provides a sophisticated but robust architecture designed to integrate artificial intelligence into wildlife conservation projects, specifically tailored for tasks such as detection and tracking of individual animals and classification of species in camera trap images and videos. The Trapper AI system consists of two main components: the Trapper AI Manager and the Trapper AI Worker (Appendix S1, Fig. S1). Importantly, both AI elements of architecture can work independently from the original Trapper core module, assuming that another custom core data management module implements the same API endpoints as those available in Trapper and sends requests for predictions in the same format.

The Trapper AI Manager functions as an advanced queue system and a web-based application that manages task execution. It organizes tasks received from the Trapper Expert module (or any other compatible software) and communicates with the Trapper AI Workers to facilitate detection and classification processes. The Manager uses RabbitMQ and Redis for the queueing system, Celery for asynchronous task processing and Flower for monitoring, storing results in a PostgreSQL database. An API enables communication between Trapper AI Manager and Trapper Expert (core module).

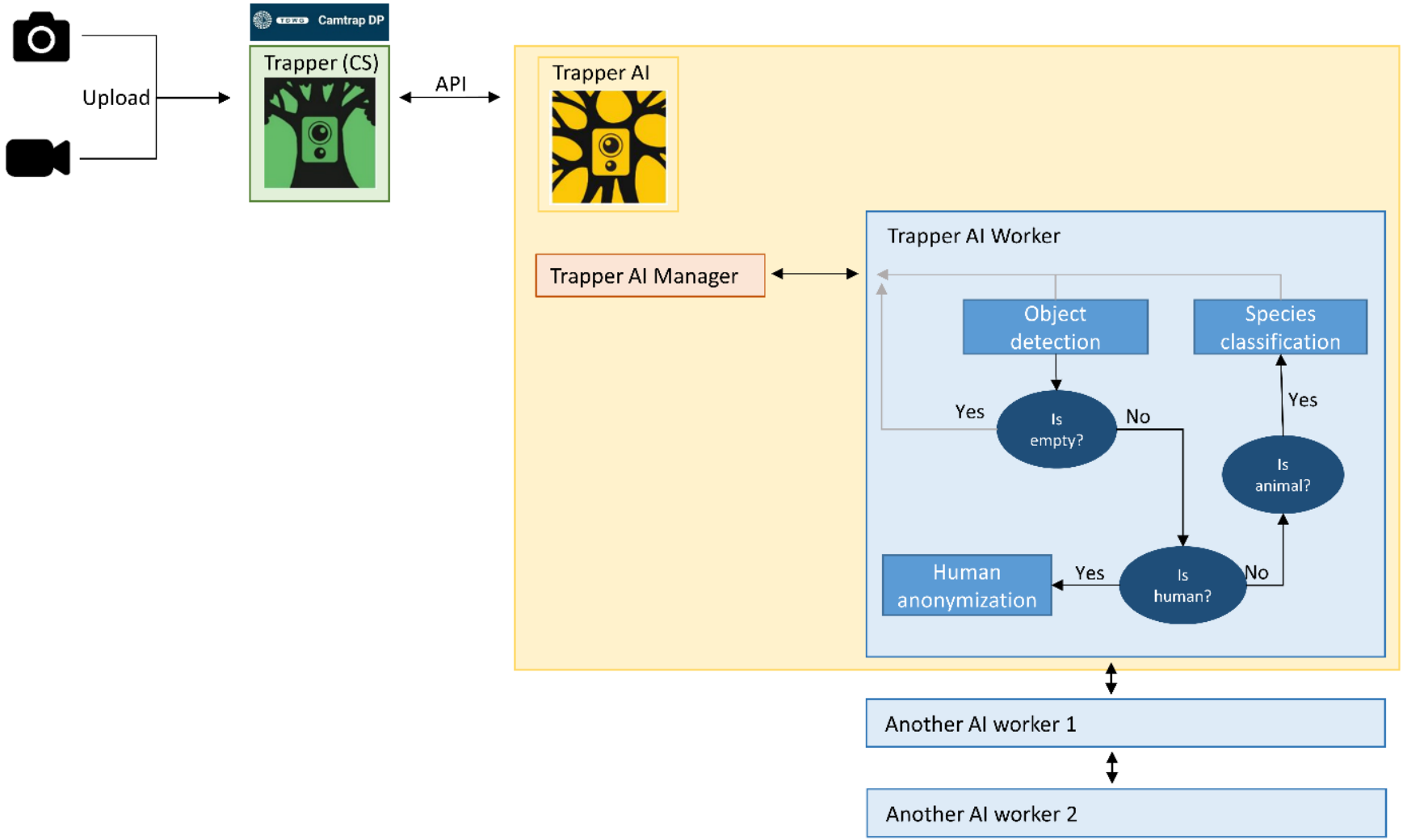
The Trapper AI Worker is responsible for loading and running AI models, with the flexibility to use either GPU via the NVIDIA CUDA driver or CPU when GPU resources are unavailable. This worker component can be deployed in various environments, including Kubernetes, cloud platforms, VPS or even local lab PCs, making it adaptable for different research setups. Moreover, multiple workers can be deployed to various environments simultaneously and connected to the same Trapper AI Manager (Appendix S1, Fig. S1). The system will automatically distribute the workload, allowing for the effective processing of large streams of camera trap data using a distributed network of computing resources. This way the system can be easily scaled from small-scale and low-cost studies to large-scale national monitoring programs.

Trapper AI currently supports multiple deep learning architectures (including Darknet, YOLOR, YOLOv5, YOLOv8, YOLOv9, YOLOv10, YOLOv11, RT-DETR, and ViT) and comes equipped with pre-configured models such as the object detectors MegaDetector v5a and

MegaDetector v6 for detecting empty images, humans, vehicles and animals (Beery et al. 2019; Hernandez et al. 2024), as well as custom species classifiers like Trapper AI Species Classifier (Choiński et al. 2021) (re-trained in 2024; (Trapper 2025). Trapper AI is a versatile tool that supports PyTorch-based AI models and allows users to extend its functionality with custom models tailored for specific wildlife monitoring and management needs. Examples of recently integrated models are DeerAI and DeepFaune v1.2 (Mustafic et al. 2024; Rigoudy et al. 2023), the latter being a state-of-the-art AI model for species-level classification of European mammals, supporting 30 species with high accuracy. Administrators can easily configure new AI model profiles by uploading weight files (e.g., .pt files), setting parameters like image and batch sizes and thresholds and managing model availability, enabling researchers and conservationists to implement and customize deep learning models effectively. If there is no support for a given model architecture, the integration usually involves extending the Trapper AI Worker with a new model adapter (i.e., adding a new Python class and updating the list of requirements), which is a straightforward process due to the platform's modular design.

The Trapper AI Species Classifier itself is highly effective, achieving a 95% F1-score and a mean Average Precision (mAP) of 93% across multiple species. Based on a fine-tuned YOLOv8-m model, it can be used within the Trapper ecosystem or through the Ultralytics package. The training and validation of the model were conducted on a substantial dataset comprising 401 458 images from 5 680 camera deployments across Poland, Germany, Sweden, Austria and Switzerland, covering 2 944 unique locations. This dataset, managed by the Open Science Conservation Fund (OSCF) and available through the Trapper AI repository on Hugging Face, allows the model to identify 18 European mammal species with high accuracy (Trapper 2024; 2025).

Figure S1: The relationship and function of the two main components of the Trapper AI system: the Trapper AI Manager and the Trapper AI Worker.



References

Beery, Sara, Dan Morris, and Siyu Yang. 2019. 'Efficient Pipeline for Camera Trap Image Review'. arXiv:1907.06772. Preprint, arXiv, July 15. http://arxiv.org/abs/1907.06772.

Bubnicki, Jakub W., Marcin Churski, and Dries P. J. Kuijper. 2016. 'Trapper: An Open Source Web-Based Application to Manage Camera Trapping Projects'. *Methods Ecol Evol* 7 (10): 1209–16. https://doi.org/10.1111/2041-210X.12571.

Choiński, Mateusz, Mateusz Rogowski, Piotr Tynecki, Dries P. J. Kuijper, Marcin Churski, and Jakub W. Bubnicki. 2021. 'A First Step Towards Automated Species Recognition from Camera Trap Images of Mammals Using AI in a European Temperate Forest.' In *Computer Information Systems and Industrial Management*, edited by Khalid Saeed and Jiří Dvorský, vol. 12883. Lecture Notes in Computer Science. Springer International Publishing. https://doi.org/10.1007/978-3-030-84340-3_24.

Hernandez, Andres, Zhongqi Miao, Luisa Vargas, Sara Beery, Rahul Dodhia, and Juan Lavista. 2024. 'Pytorch-Wildlife: A Collaborative Deep Learning Framework for Conservation'. *CV4Animals Workshop: Computer Vision for Animal Behavior Tracking and Modeling* 2405.12930. https://doi.org/10.48550/arXiv.2405.12930.

Mustafic, Sead, Dominik Dachs, Rainer Prüller, Florian Schöggl, and Roland Perko. 2024. 'DeerAI: AI Based Game Monitoring'. https://www.ffg.at/sites/default/files/allgemeine_downloads/thematische%20programme/IKT/2024_06_Poster_DeerAI_BMK.pdf.

Rigoudy, Noa, Gaspard Dussert, Abdelbaki Benyoub, et al. 2023. 'The DeepFaune Initiative: A Collaborative Effort towards the Automatic Identification of European Fauna in Camera Trap Images'. *Eur J Wildl Res* 69 (6): 113. https://doi.org/10.1007/s10344-023-01742-7.

Trapper. 2024. 'Trapper-Ai-Worker'. GitLab. May 14. https://gitlab.com/trapper-project/trapper-ai-worker.

Trapper. 2025. 'TrapperAI-V02.2024'. Hugging Face. https://huggingface.co/OSCF/TrapperAI-v02.2024.