The Automation of Camera Trap Distance Sampling with Machine Learning for the Estimation of Population Density and Abundance

Author:
TOM RAYNES

DEPARTMENT OF COMPUTER SCIENCE

UNIVERSITY OF BRISTOL

A thesis submitted to the University of Bristol in accordance with the requirements of the degree of MSc COMPUTER SCIENCE

SEPTEMBER 2025

Abstract

[Abstract]

Acknowledgements

[Acknowledgements]

Author's declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: ..... DATE: .....

Table of Contents

Page

1 Introduction **1**

2 Background **2**

2.1 WCF Dataset . . . 2

2.2 Methods and Models . . . 2

2.2.1 Deep Learning . . . 2

2.2.2 Mega Detector . . . 2

2.2.3 Segment Anything . . . 2

2.2.4 Depth Anything . . . 2

2.2.5 Dense Prediction Transformers . . . 2

2.2.6 Calibrating Distances . . . 2

2.3 Estimating Activity . . . 2

3 Experimantal **3**

3.1 Calibration Frame Preparation . . . 3

3.1.1 Frame Extraction . . . 3

3.1.2 Frame Mask Creation . . . 4

3.2 Detection Frame Preparation . . . 7

3.2.1 Sampling . . . 7

3.2.2 Frame Extraction . . . 8

3.3 Distance Estimation . . . 10

3.3.1 Setup . . . 10

3.3.2 Configurations . . . 10

3.3.3 Testing Calibration Modifications . . . 10

3.4 Activity Estimation . . . 11

3.4.1 Formatting . . . 11

3.4.2 Running Activity Script . . . 12

4 Analysis **13**

4.1 Analysis of Distance Estimates . . . 13

4.1.1 Model / Manual Distance Comparison . . . 13

4.1.2 Error Analysis . . . 13

4.1.3 Qualitative Analysis 13
4.1.4 Effects of Varying Calibration 13
4.2 Analysis of Activity Estimates 14
4.2.1 Manual Sample Activity Analysis 14
4.2.2 Automated Sample Activity Analysis 14

5 Evaluation of Methodology **15**

6 Conclusion **16**
6.1 Further Work 16

List of Tables

TABLE **Page**

List of Figures

FIGURE **Page**

3.1 Frame extractor program showing the drag/drop (top) and save frame (bottom) features 3

3.2 Raw calibration frames 4

3.3 Manual calibration frame masks 5

3.4 Automated calibration frame masks 6

3.5 Mixed subsample of extracted detection frames capturing chimpanzees (taken from both the manual and automated sample) 8

3.6 Mixed subsample of empty detection frame (taken exclusively from the automated sample) . . 9

1 Introduction

Manual distance sampling bottleneck

Aim to automate distance sampling of a large dataset and use estimated distances to achieve accurate estimates for population activity

WCF dataset

How abundance and density is calculated using distances

2 Background

2.1 WCF Dataset

2.2 Methods and Models

2.2.1 Deep Learning

2.2.2 Mega Detector

2.2.3 Segment Anything

2.2.4 Depth Anything

2.2.5 Dense Prediction Transformers

2.2.6 Calibrating Distances

2.3 Estimating Activity

Distance estimation[1]

3 Experimantal

3.1 Calibration Frame Preparation

For each camera location of the dataset, a set of calibration frames (generally fifteen, location dependant) and corresponding binary masks were prepared. The frames were extracted from location-specific reference videos supplied with the dataset consisting of a person standing, facing the camera, at known distance intervals and holding a sheet of A4 paper labelled with the corresponding distance.

3.1.1 Frame Extraction

In an effort to streamline the frame extraction process, an extractor program was created (Figure 3.1). This tool enabled reference videos to be dragged and dropped into a GUI window allowing for the easy navigation between individual frames to extract the optimal one for each distance. With this software, the next/previous frames are accessed with the 'left'/'right' arrow keys, the frame index position is moved ahead/behind 20 places with the 'd'/'a' keys and the frame found at a specific timestamp is accessed with the 's' key followed by entering a time (in seconds) into an input bar. The active frame (at the current index) can be saved to disk with the 'enter' key followed by entering a filename in an input bar.

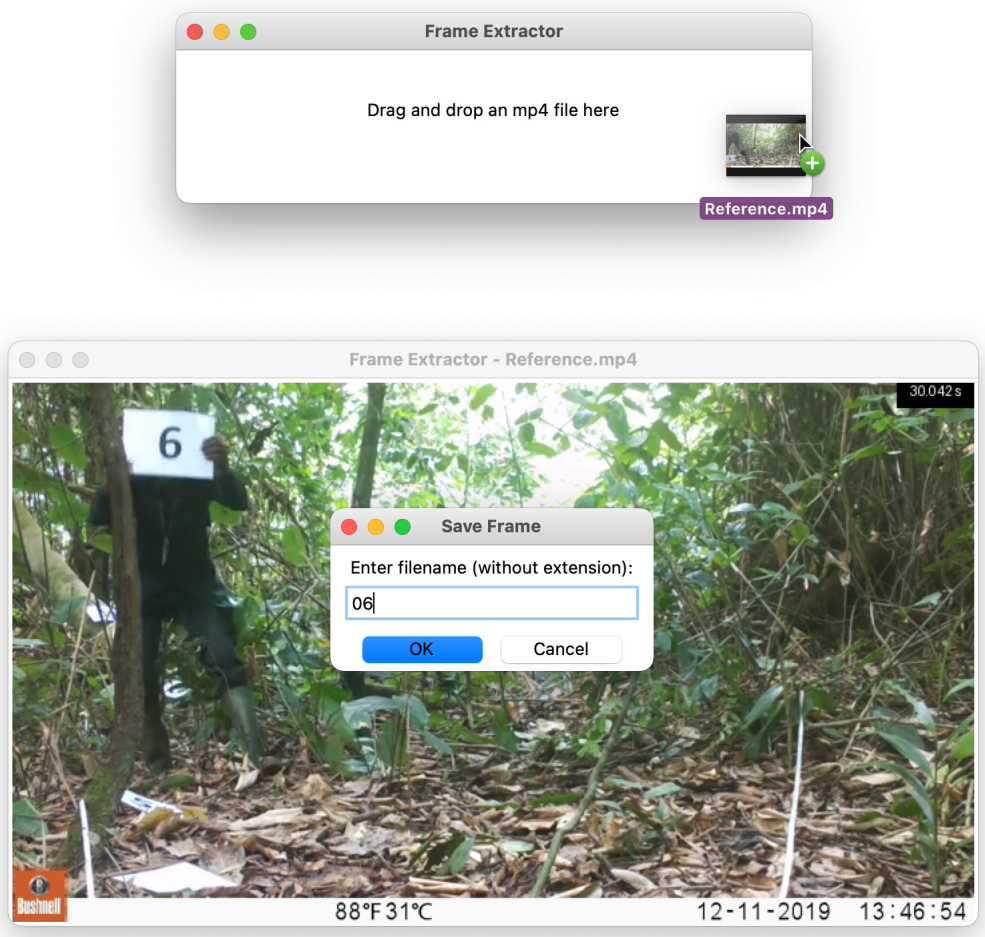

Figure 3.1: Frame extractor program showing the drag/drop (top) and save frame (bottom) features

3.1.2 Frame Mask Creation

For each of the extracted calibration frames, a corresponding binary mask was created. This task can be completed manually using image manipulation software (e.g. Photoshop, GIMP, etc...) where the segmentation boundary is manually traced then filled. This approach, however, is rather labour-intensive, constituting somewhat of a bottleneck, and therefore it is desirable to automate the process.

To assess the feasibility of automation, a two-stage detection/segmentation pipeline was created. Here, raw calibration frames are first processed with YOLOv5 detector[2] to generate bounding boxes enclosing the frame landmarks. The bounding boxes are then passed to Segment Anything Model which predicts segmentation masks for the landmarks.

Two examples of the raw calibration frames (Figure 3.2) along with their corresponding manual (Figure 3.3) and automated (Figure 3.4) masks are shown below.

Figure 3.2: Raw calibration frames

Figure 3.3: Manual calibration frame masks

Figure 3.4: Automated calibration frame masks

3.2 Detection Frame Preparation

Detection frame preparation is the process whereby the raw camera trap video of the dataset is transformed into an array of representative images depicting the chimpanzees whose distances to the camera will be estimated. Two distinct frame sampling techniques were used in this study which will be referred to as 'manual' and 'automated' sampling.

3.2.1 Sampling

3.2.1.1 Manual Sample

Accompanying the raw camera trap video, another component of the dataset is a list of human-annotated distance-to-camera estimates for all observed chimpanzees at a recorded date and time (along with additional metadata). These distance data must be used a benchmark by which the accuracy of any modelled estimates are assessed and therefore the specific frames corresponding to these manual annotations must be sampled.

Before sampling, these data were first cleaned by running an automated script to flag any identifiable abnormalities in the data such as duplicate entries and inconsistencies in the date/time formatting. These were then manually corrected.

In order to identify the correct frames to extract, each of the manual annotations was assigned a timestamp corresponding to a time in seconds in which it was recorded in its associated camera trap video. These timestamps then constituted discreet identifiers of sampled frames.

Although the start-date/times of the videos themselves were not explicitly pre-labelled within the dataset, assigning timestamps was still possible since the date/time at zero seconds into a given video can be inferred as equal to that of the earliest of all recorded annotations associated with the video. This was a valid assumption to make as the overwhelming majority of videos begin exactly at the point in which a chimpanzee enters the frame, thus corresponding to the first annotated observation. In the few rare cases where this heuristic did not apply, all relevant timestamps were manually corrected upon being flagged.

3.2.1.2 Automated Sample

The automated sampling process is significantly more straightforward. Here, detection frames are sampled at an interval of two seconds over all camera trap video in the dataset. Resultant model-estimated distances from this sample are intended to be used purely as distance data for the automated modelling of population abundance.

In contrast to the manual sample, this automated sample is not directly associated with any human-annotated distance estimates. It does, however, remain representative of the dataset, capturing images of the same individual chimpanzees over the same time period. This sample also differs due to the sampling of many 'empty' frames where the image captures no individual. The impact of these empty frames on subsequent abundance estimates will be minimal. The empty frames may collectively incur a small number of false positive detections; however, the total will be negligible and the probability of a given empty frame resulting in a false positive detection is no more likely than that of any other frame.

All in all, this approach exemplifies a simple frame sampling method that can be used as part of an automated distance sampling pipeline.

3.2.2 Frame Extraction

The identified sample frames were extracted from the raw camera trap video using an automated script. Examples of extracted detection frames capturing chimpanzees (Figure 3.5) as well as empty detection frames (Figure 3.6) are shown below.

Figure 3.5: Mixed subsample of extracted detection frames capturing chimpanzees (taken from both the manual and automated sample)

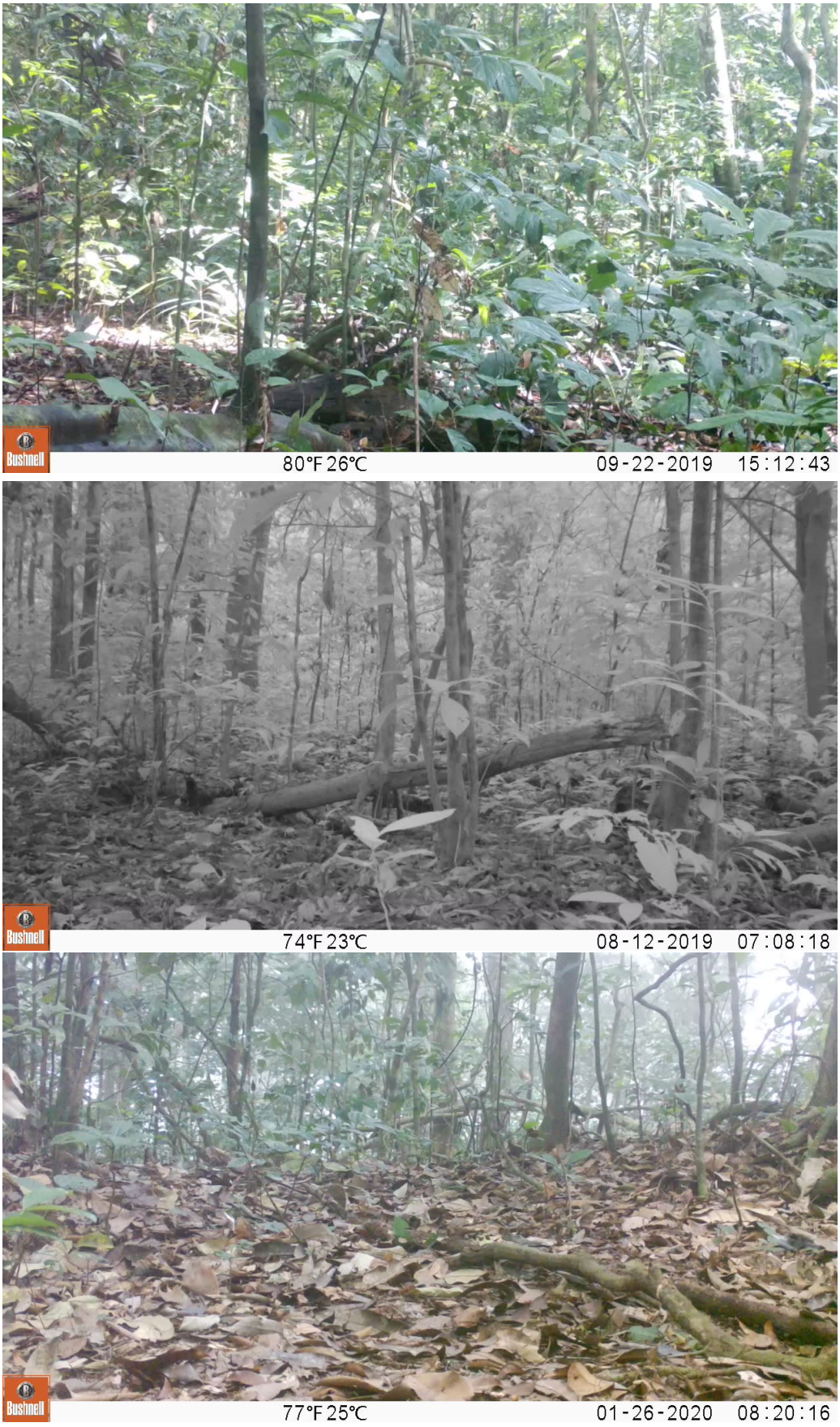

Figure 3.6: Mixed subsample of empty detection frame
(taken exclusively from the automated sample)

3.3 Distance Estimation

3.3.1 Setup

BlueCrystal4 was used to run the distance estimation pipeline across the dataset. Minor adjustments to the load_model() methods of the DPT and DepthAnything classes were made in order to configure ONNX Runtime to utilise all CPU cores allocated by SLURM (28 in this case).

It is also worth noting that for these distance estimation runs, the manual calibration frame masks were used. The reason for this was to maximise the consistency of the masks due to a drop in the quality of the automated masks at longer distances (see analysis section).

3.3.2 Configurations

Four distinct configurations of the distance estimation pipeline were applied to the dataset. Each configuration combined one of two detection methods (bounding box, segmentation) with one of two distance estimation models (DPT, Depth Anything). Using each of these configurations, distance estimates for both the manually and automatically sampled detection frames were computed.

3.3.2.1 Detection Methods

Both detection methods use Mega Detector as a starting point to generate bounding boxes enclosing the animal.

For the plain bounding box detection method, all pixels enclosed in the bounding boxes are used as candidates during the later-stage calculation of detection distance from the depth maps generated by either DPT or DepthAnything. Specifically, once the depth estimation model has predicted a distance for each pixel, the distance estimate for the entire detection is calculated to be equal to that of the pixel corresponding to the 20th percentile depth of all pixels within the bounding box.

The segmentation method, however, introduces an additional step in which the generated bounding boxes are passed to Segment Anything Model which subsequently predicts segmentation masks for the detection. It is these segmented pixels exclusively that are used in the later detection distance calculation. In contrast to the plain BBOX detection method where distance is calculated based on percentile, the segmentation method detection distance is calculated as equal to the distance estimate of the pixel corresponding to the centre-most point of the segmentation mask (i.e., the pixel that is furthest from the segmentation boundary).

Additionally, all runs using the segmentation detection method were parameterised with the 'mask animals' flag. This has the effect of, during the calibration stage, using only pixels defined with the calibration frame masks to fit the regression curve and calculate the calibration correction function.

3.3.3 Testing Calibration Modifications

The effects altering the calibration stage of the distance estimation pipeline were also tested. Calibration frame preparation comprises significant manual annotation resulting in a bottleneck, therefore it would be a positive outcome if this stage could be streamlined with a reduction in calibration requirements.

Firstly, a modification was tested over the whole dataset with DPT distance estimation using both bounding box and segmentation detection methods. Here, the program's run() method was altered to ignore all calibration frames except those corresponding to closest and furthest landmarks from the camera.

Secondly, the effects of varying the number of calibration frames was tested on a single camera location. A total of fifteen runs were carried out also using DPT distance estimation with both bounding box and segmentation detection methods. The first of these used a single (closest) calibration frame with each subsequent run then including the next calibration frame in the sequence (i.e., 1, 1–2, 1–3, . . . 1–15). For the runs using at least two calibration frames (i.e., runs two through fifteen) a regression function could be fitted as normal; however, this was not possible for the first run where a single calibration frame was used. Here, the calibrate() method was modified to return a simple scale factor transforming the single modelled distance to its ground truth.

3.4 Activity Estimation

Using the computed distance estimates from the previous section, chimpanzee population density and abundance was estimated. Estimates were calculated for each of the detector/depth model combinations using a script adapted from one supplied with the dataset.

3.4.1 Formatting

For the distance data to be used for density and abundance estimation, it first must be formatted and joined with the relevant location and camera trap video metadata as to satisfy the script's input requirements. The final must then be identical in structure to the manually annotated detection data supplied with the dataset in CSV formatting containing the columns 'video_name', 'time_in_place_(days)', 'effort', 'starting_date', 'date_observation', 'time_observation', 'year', 'month', 'day', 'hour', 'minute', 'second', 'distance', 'Sample.Label', 'Effort', 'Region.Label' and 'Area'.

3.4.1.1 Manual Sample

When formatting the distance data generated from the manual sample, the goal was to overwrite each of the manually annotated distances (in original supplied dataframe) with the corresponding model estimated distances, leaving all other data unchanged.

For reasons detailed in Section 4.1.1, a decision was made to format the data such as to only overwrite manual distances associated with frames capturing a single chimpanzee. This gave a 'supplemented' dataframe, where distance estimates originating from single-chimp frames were model-estimated while those from multi-chimp frames were estimated manually. Nevertheless, density and abundance estimates based on supplemented data are still informative and give insight into the effectiveness and accuracy of using model estimated distance data in this context.

3.4.1.2 Automated Sample

When formatting the distance data generated from the automated sample however, a different method was used. First, a lookup table was created to hold the metadata associated with each specific camera trap video. A script was then run to join each of the modelled distances with the video metadata on video name which gave a formatted dataframe holding entirely model estimated distances.

3.4.2 Running Activity Script

With the distance data and associated metadata now appropriately formatted, density and abundance estimates were then developed using the activity script. For each dataframe, the script was adapted to apply appropriate truncation and binning to the distance data in order to generate a smooth probability density histogram. Preliminary runs of the script were then used to fit several detection functions to this histogram, with the best fitting (lowest AIC value) being selected.

4 Analysis

Analysis of automated calibration frames, overlays, failure cases, etc

4.1 Analysis of Distance Estimates

4.1.1 Model / Manual Distance Comparison

model vs manual distance graphs

How does parameterisation affect the estimates

Single chimps distance comparison:
Ideally, each and every manual distance should be joined to its corresponding modelled distance estimate; however, due to the absence of any frame-position data associated with the manual annotations, automating this using traditional algorithms is impossible in circumstances where multiple chimps are captured in a single frame. This is because when multiple individuals are detected, there is ambiguity in regard to which distance a given modelled distance should be joined to. Moreover, approaching this task manually is extremely labour-intensive and was therefore outside the scope of this project. As a result, this analysis focuses on distance comparisons associated with frames capturing only a single individual.

4.1.2 Error Analysis

4.1.3 Qualitative Analysis

depth map diagrams close/far failure cases, sweet spot

4.1.4 Effects of Varying Calibration

4.2 Analysis of Activity Estimates

4.2.1 Manual Sample Activity Analysis

Single chimp frame distances supplemented with manual distances

4.2.2 Automated Sample Activity Analysis

5 Evaluation of Methodology

Improvement over fully manual approach?

Bottlenecks, calibration frame preparation, available compute

Time saved using frame extractor, preserves exact pixels

Limitations of the environment

Calibration and detection frame image resolution

Depth estimation point consistency (the animal itself has depth)

Error in reference videos (person has depth, not always standing up straight)

Better calibration (polynomial/not just linear, more calibration frames)

6 Conclusion

6.1 Further Work

References

[1] T. Haucke, H. S. Kühl, J. Hoyer, and V. Steinhage, "Overcoming the distance estimation bottleneck in estimating animal abundance with camera traps," *Ecological Informatics*, vol. 68, p. 101536, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574954121003277

[2] R. Khanam and M. Hussain, "What is yolov5: A deep look into the internal features of the popular object detector," 07 2024.