Automation of the analysis of insect camera data

Using deep learning to monitor the decline of insects

Tijs van Lieshout

Automation of the analysis of insect camera data

Using deep learning to monitor the decline of insects

Tijs van Lieshout

Abstract

Insects are vital to many ecosystems as pollinators and as a food source for insectivores. Recent studies have indicated a worldwide decline in biodiversity, biomass and occurrence of many different insect species. Traditional methods to measure these insect trends are very time consuming, labour intensive and are lethal to the insects. We propose a novel method of measuring these trends with insect cameras. In the summer of 2019 in total eighty-seven insect cameras have taken over thirteen million pictures, of which eight million pictures were labelled to contain activity. The analysis of all these pictures would take entomologists very long.

In this study, we present a pipeline that automates the analysis of insect camera data. This pipeline detects, tracks (deduplicates), segments, identifies and estimates the biomass of terrestrial arthropods with help of different convolutional neural networks.

Finally, we present a discussion about the possibility of using these cameras as an insect monitoring network for the Netherlands.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The decline of insects and other terrestrial Arthropods

Insects are a vital part of any ecosystem, firstly by acting as a food source for more than half of the bird species (Morse 1971). Secondly, the research done by Ollerton, Winfree, and Tarrant 2011 estimates that between 78% and 94% of all flowering plants are pollinated by animals, of which mainly insects.

Recent studies have reported on a worrying decline in occurrence, biodiversity and biomass of insects globally (Hallmann, Zeegers, Klink, Vermeulen, Wielink, Spijkers, Deijk, et al. 2019; Powney et al. 2019). Specifically for western Europe, the research done by Hallmann, Sorg, et al. 2017 shows a decline of more than 75% in flying insect biomass over twenty-seven years in designated nature reserves in west Germany. By measuring the total insect biomass with Malaise traps in sixty-three nature protection areas in Germany, they conclude that the decline seems to be happening in all tested habitat types and that changes in the habitat characteristics, land use or weather is no simple answer to the decline of flying insects.

A review by Sánchez-Bayo and Wyckhuys 2019 showed seventy-three reports of insect declines world-wide and claimed that this overview of the literature revealed "dramatic rates of decline that may lead to the extinction of 40% of the world's insect species over the next few decades". This claim was questioned by the entomologist community because of their limiting search methods for literature, their data under-representing insect groups (such as *Diptera*, *Hemiptera* en *Orthoptera*) and over-representing the regions of North America and Western Europe (Wagner 2019). Sánchez-Bayo and Wyckhuys 2019 collected publications to review by searching the *Web of Science* with the query *[insect*] AND [declin*] AND [survey]*, resulting in a biased set of studies that focused on declines with none to few results focusing on increases (Simmons et al. 2019).

Multiple drivers of insect decline are hypothesised such as the loss of habitat, intensification of agriculture, the increased use of pesticides, climate change, light pollution and invasive species (Montgomery et al. 2019; Grubisic et al. 2018; Habel et al. 2019; Homburg et al. 2019). Martin et al. 2018 considers the increase in road traffic to be an underestimated driver of the decline in flying insect abundance. There are many different hypotheses, however, more research is certainly needed on testing possible causes for the decline of insects (Hallmann, Zeegers, Klink, Vermeulen, Wielink, Spijkers, and Jongejans 2018; Montgomery et al. 2019).

1.2 The need for a better monitoring network to understand insect loss

Most studies of insect trends are based on capture methods such as pitfall, malaise or sticky traps. Setting up these traps and collecting the insects to measure is very labour intensive and has to be repeated a large number of times because a study can take many years before a conclusion can be reached (Hallmann, Sorg, et al. 2017). Furthermore, Hallmann, Zeegers, Klink, Vermeulen, Wielink, Spijkers, Deijk, et al. 2019 reports on the absence of standardised monitoring networks for the state of insects in the Netherlands and describes the exceeding need for the development of such a monitoring network. A more standardised monitoring network could lead to improved comparisons between data sets and analysis.

1.3 The use of insect cameras for continuous monitoring of insect biodiversity

An insect camera can be defined as a device that makes photographs of insects autonomously. One of the main advantages of an insect camera is the fact that it is non-lethal as opposed to other traditional insect collection methods. Malaise nets and pitfall traps kill insects to collect them which raises an ethical concern (B. Fischer and Larson 2019). Another large improvement is the fact that an insect camera is a lot less labour intensive than previous methods. An insect camera also creates more samples than a Malaise or pitfall trap during the same period since the camera takes a picture every ten seconds for the time it is deployed.

1.4 Project aim

In the summer of 2019, the DIOPSIS project has placed eighty-seven insect cameras as a pilot for a novel insect monitoring network. In this study, we aim to build a pipeline to automate the analysis of over eight million pictures taken by these insect cameras. This analysis consists of the detection, tracking, segmentation, identification and biomass estimation of terrestrial arthropods and other organisms commonly seen as "critters" (such as a subset of *Mollusca*) with the use of multiple different convolutional neural networks. Hereinafter, we will refer to these terrestrial arthropods and other "critters" as insects to simplify the readability.

Lastly, we discuss the possibility of using an insect camera network for the long-term monitoring of insect trends in the Netherlands.

Chapter 2

Methods

This project builds upon the work of the insect cameras created by the DIOPSIS team to further the development of a pipeline to automate the analysis of insect camera data. For this analysis, the pipeline has been developed and used on images taken by the insect cameras to detect, track (deduplicate), segment, identify and estimate the biomass of insects in order to get data about the diversity, occurrence and total biomass of insects in a given area.

2.1 Data collection with insect cameras

In this study, images taken by insect cameras are seen as the main raw data. These insect cameras consist of a Raspberry Pi controlled camera with an LED lamp and a yellow screen on which insects are photographed (**figure 2.1**). The colour yellow was chosen to attract the most diverse group of insects as possible (J. W. Campbell and Hanula 2007). The insect camera has taken pictures with an interval of ten seconds, with each picture being compared to the preceding picture. If more than 3×10^{-5} of the total amount of pixels changed brightness by more than 20% it has gotten the label of containing "*Activity*". In total eighty-seven insect cameras were placed in urban, agricultural and nature areas in the Dutch provinces of North- and South- Holland, Zeeland, Utrecht, North Brabant and Gelderland (**figure 2.2**).

Figure 2.1: One of the cameras used to collect the data. The green box on the left contains the camera, lamp, SD card and the Raspberry Pi. Seen on the right is the yellow screen of which photos are taken every ten seconds.

Figure 2.2: A map showing the locations of the insect cameras The blue markers show where one or multiple cameras have been placed.

2.2 NOUS interactive Artificial Intelligence

To be able to reach a conclusion of these eight million pictures the analysis had to be automated with deep learning. In deep learning, neural networks are used to automate tasks such as detecting an object or classifying what kind of object the neural network is given. Neural networks learn by giving manually annotated examples and improve with the increase in the amount of annotations (Lin et al. 2017). The process of annotation can be very time-consuming and labour intensive. COSMONiO offers a solution for this in with their product called *NOUS*. NOUS is an interactive platform in which the user can train a neural network for different tasks such as classification, detection or segmentation. To reduce annotation effort NOUS selects the most difficult images to be annotated by the user first.

In this project, NOUS has been used to annotate detections (a box around insects) and segments (a polygon hull over only the pixels containing the insects, without background). The user draws these squares and polygons in NOUS over an image after which the positions of the corners get saved in a JSON-formatted file.

For the detection of insects RetinaNet by Lin et al. 2017 was used as a starting point. This state-of-the-art detector has been trained in NOUS on the insect camera data set.

U-Net was the chosen neural network for the segmentation of insects (Ronneberger, P. Fischer, and Brox 2015).

2.3 Nature Identification Application Programming Interface

For the identification of insects the Nature Identification Application Programming Interface (Nature Identification API) has been used (Naturalis Biodiversity Center 2018). The classifier behind this API is the Inception-v4 network (Szegedy et al. 2017).

Observation.org works with citizen science, resulting in the neural network behind the Nature Identification API having been trained with over three million six hundred thousand pictures, of which one and a half million pictures of insects and other terrestrial arthropods, all identified by experts.

2.4 Retrained order identification model

Here we refer to the result of a trained convolutional neural network classifier as the identification model. Because of some middling success with the Nature API, another test was done by retraining the identification model for insects on the taxonomic order level (*e.g. Coleoptera*). This model is not based on observation.org, having been trained from scratch solely with data from the insect camera images. This neural network was annotated by a member of the DIOPSIS team with sufficient entomological knowledge to be able to classify insects on the taxonomic rank 'order'.

Chapter 3

Results

3.1 Data collected per camera

The number of pictures on each SD card of the insect cameras has been plotted in **figure 3.1**. In total over thirteen million pictures have been taken, of which nearly eight million were deemed to contain activity. If more than 3×10^{-5} of the total amount of pixels changed brightness by more than 20% the picture is deemed to contain activity, meaning that possibly an insect has moved. The labelling of activity on images is used to filter out the images with no activity so that there are less redundant images. An example of a picture taken by one of the insect cameras is found in **figure 3.2**. An image is on average six hundred kilobytes, resulting in the need for total storage space of almost eight terabytes for all thirteen million pictures.
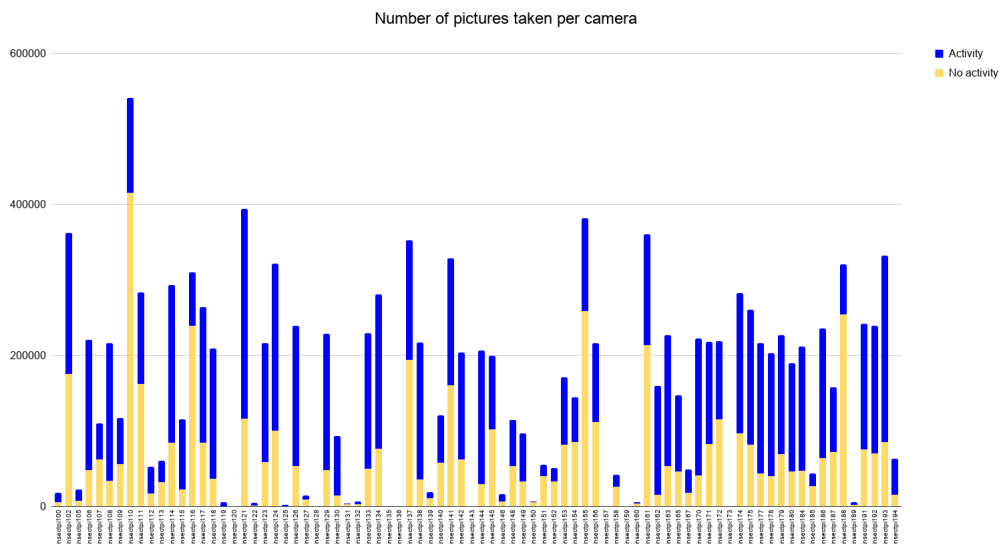
Figure 3.1: The number of pictures taken per camera. The camera identifier is plotted on the x-axis with the number of pictures taken on the y-axis. Data plotted in blue shows the pictures deemed to contain 'activity' and in yellow are pictures without 'activity'. Activity is defined here as the picture being significantly different from the last taken picture, meaning that in theory an insect or other organism has moved compared to the preceding picture.
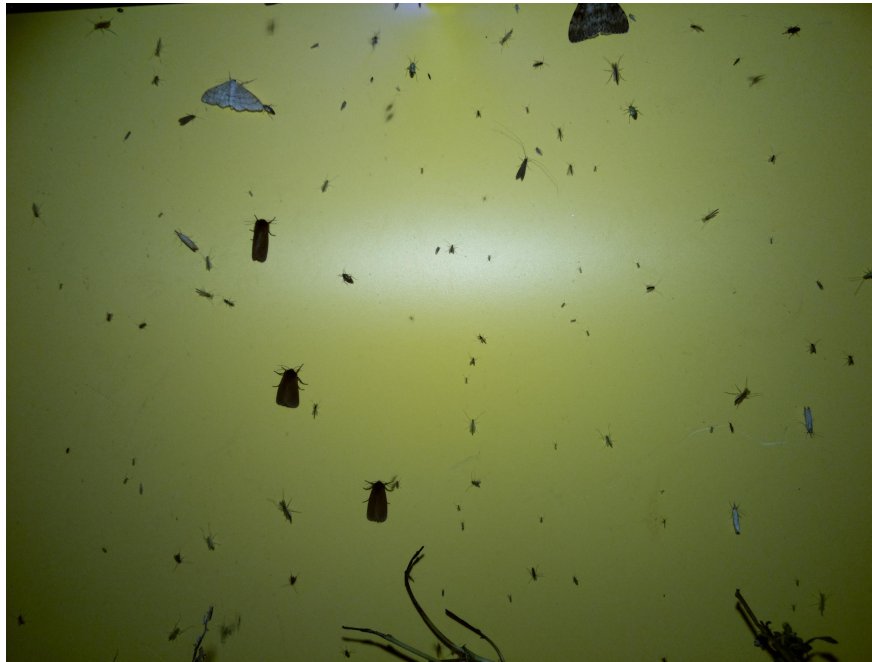
Figure 3.2: An example picture taken by one of the insect cameras.

3.2 Insect taxa to order generator

The Nature Identification Application Programming Interface returns taxa in the form of a binomial name on species-level (Naturalis Biodiversity Center 2018). To be able to analyse the data on the taxonomic order-level a list had to be created linking all species names in the classifier with their respective order. To achieve this a file in JSON-format containing all species in the Nature Identification API was retrieved. Species contained in this file could then be queried against the Netherlands Biodiversity Application Programming Interface (NBA) to get more taxonomic information (Snijders et al. 2014). The order of 15773 of the 16148 species in the Nature Identification API could be retrieved in this way. Species and their respective order were then saved to a list if either their phylum was *Arthropoda*, *Gastropoda* or *Clitellata*. Because not all *Gastropoda* or *Clitellata* are terrestrial or expected to be encountered in an insect camera some classes were not allowed. These classes consisted of *Branchiopoda, Remipedia, Cephalocarida, Maxillopoda, Ostracoda, Malacostraca* and *Pycnogonida*. These steps resulted in a list of 7724 species in 59 different taxonomic orders that have a possibility of being encountered in a picture taken by an insect camera. Contrary to other python scripts made for the pipeline, this script only has been run once to create the list that the pipeline uses. The python code to create this list can be found in **appendix A.1**.

3.3 Pipeline for the automatic analysis of insect camera data

3.3.1 Overview of the pipeline

A pipeline has been made to be able to analyse the nearly eight million pictures taken by the cameras. This pipeline has been written in Python 3.5 (Rossum 1995). The source code of the pipeline can be found in **appendix A.2**. **Figure 3.3 and 3.4** show the individual python scripts and their input and output that make up the pipeline. The pipeline starts with the CSV-formatted files that describe the detections (the positions of rectangles around insects in a taken picture). These files have been generated with NOUS. The pipeline exists of the following seven main steps; segment.py, identify.py, track.py, post_process_identify.py, biomass_estimation.py, report.py and an optional video.py (**Table 3.1**). Between each step, a CSV-formatted file gets saved with the resulting data. It was chosen to create a new CSV-formatted file in each step to allow running only a few updated steps without having to run the entire pipeline again.

python script	short description
segment.py	Calls the convolutional neural network through NOUS to create segments of all detections.
identify.py	Calls the nature identification API to classify all detections. Adds taxonomic order to the species identification.
track.py	Deduplicates the detections between two sequentially taken pictures with the Hungarian algorithm.
post_process_identify.py	Assigns the most likely taxonomic order to each object. An object consists of one or more detections.
biomass_estimation.py	Estimates the biomass of each identified segment through the use of formulas from literature.
report.py	Generates a HTML report containing plots and tables of the given data set.
video.py	Generates an MP4-formatted video showing the detections, identifications and body lengths of insects drawn upon the original images.

Table 3.1: Overview of the main steps of the pipeline.
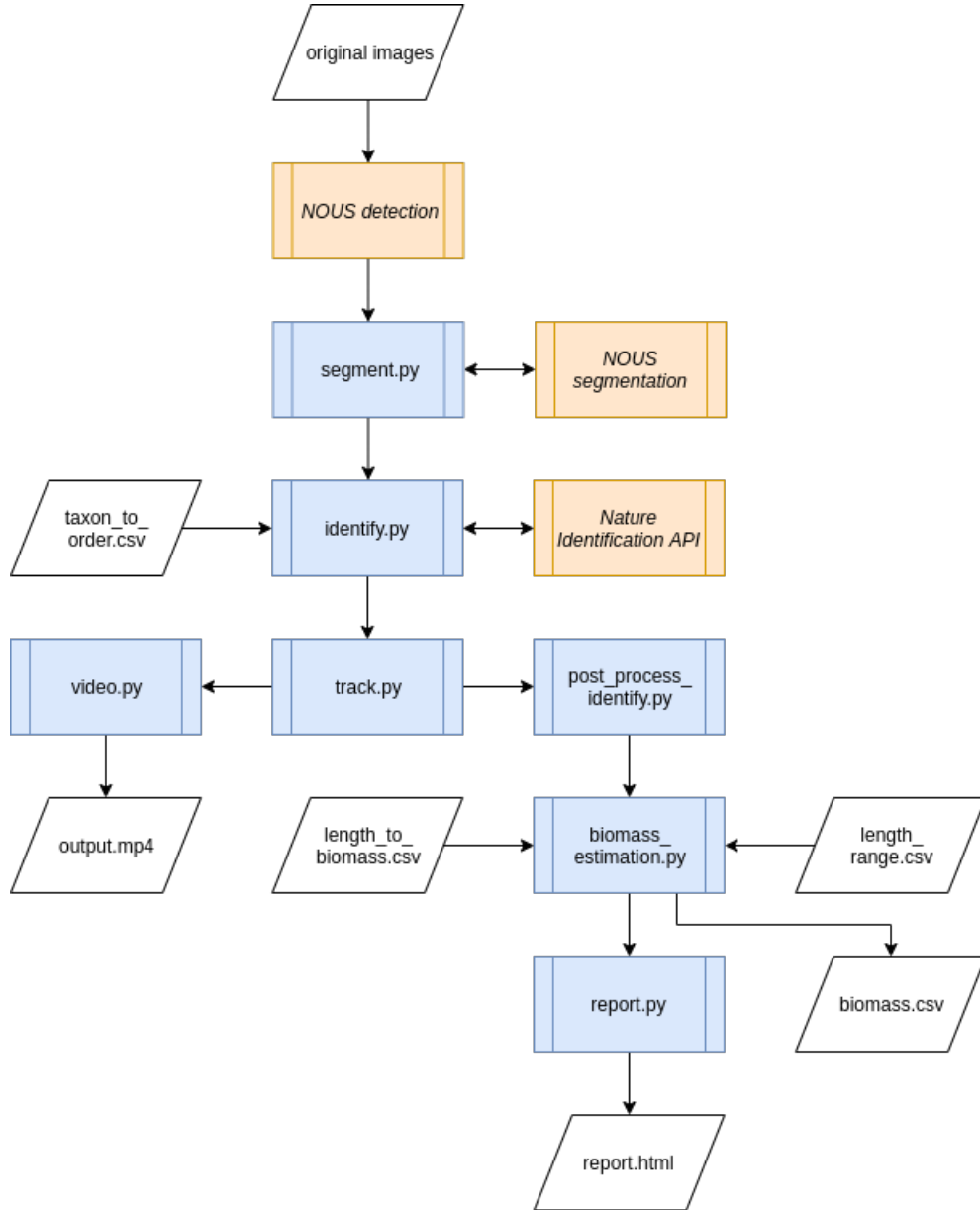
Figure 3.3: A simplified version of the flowchart visualization of the pipeline.
Boxes highlighted in blue are the main python scripts of the pipeline that the user can run. Coloured yellow shows the external Application Programming Interfaces that are required for this pipeline. The full pipeline can be seen in **figure 3.4**.
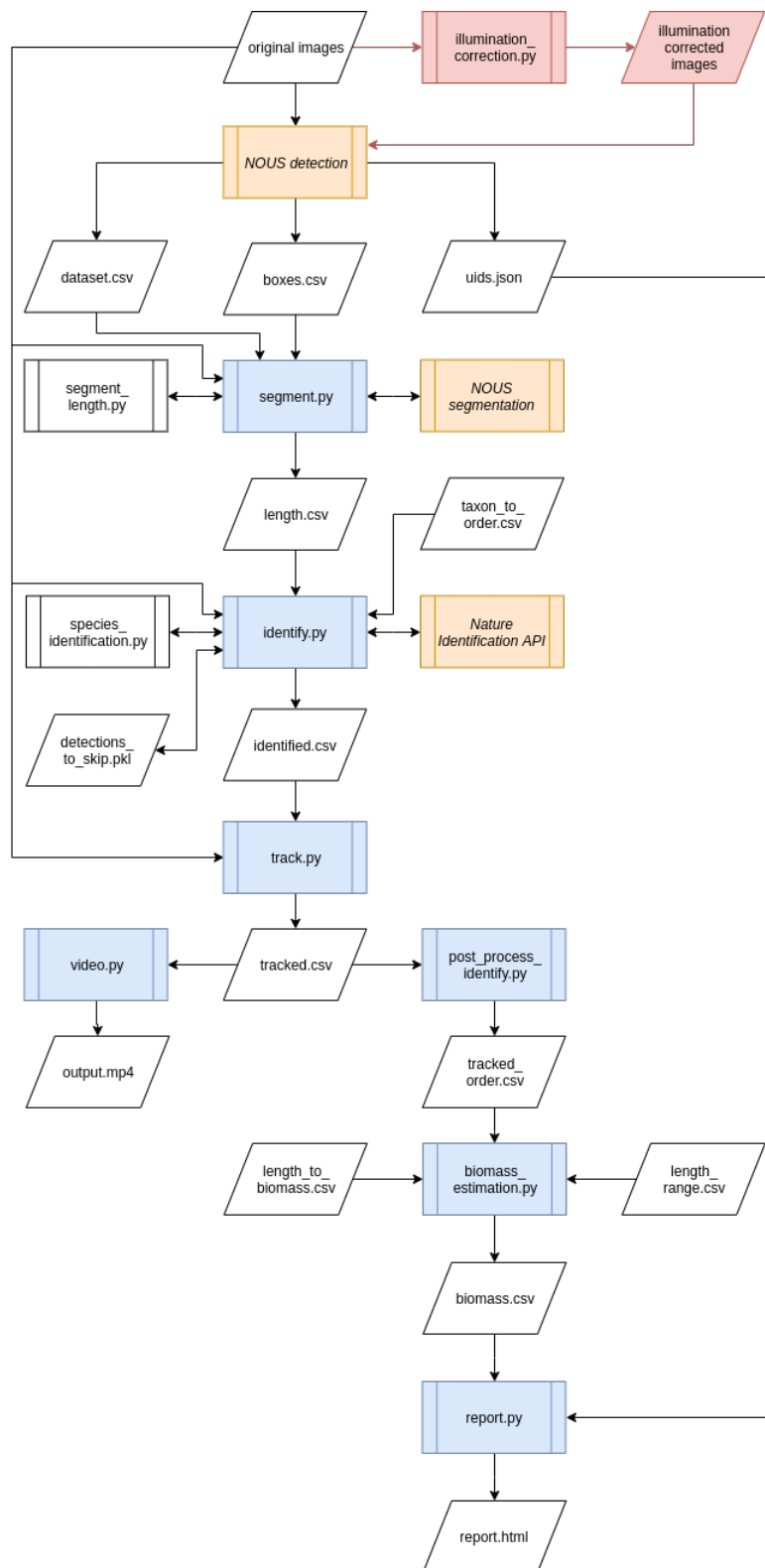
Figure 3.4: Flowchart of the pipeline with all python scripts and their input and output. Boxes highlighted in blue are the main python scripts of the pipeline that the user can run. Coloured yellow shows the external Application Programming Interfaces that are required for this pipeline. Finally, boxes highlighted in red show an optional preprocessing step to correct the images for uneven illumination.

3.3.2   Performance of the pipeline

The time to perform the steps of the pipeline was measured to get an idea of how long it would take to analyse all eight million pictures containing activity (**table 3.2**). These steps were performed on an Amazon Web Services instance with 64 gigabytes of RAM (further specifications can be found in **appendix A.3**). Time to complete can differ between data sets depending on the number of insects per image.

Step	Time to complete	Number of pictures containing activity
Detection	±03:30:00	10000
Segment	±00:50:00	10000
Identify (Nature Identification API)	±03:10:00	10000
Track	±00:25:00	10000
Post processing	±00:12:00	10000
Biomass	±00:05:00	10000
Video	±00:20:00	10000
Report	±00:01:00	10000
Total	±08:30:00	10000

Table 3.2:   Time to complete steps of the pipeline.   Time to complete is in HH:MM:SS format. Time to complete is shown per ten thousand pictures containing activity. Scripts have been run on a g4dn.4xlarge Amazon Web Services instance.

The performance of the detection model has been tested through its F1 score. The F1 score (also referred to as the F-measure) calculates the harmonic mean of the precision and recall (**formula 3.1, 3.2 and 3.3**). As of March 2020, the F1 score of the detection model is 0.741. The precision-recall curve of the detection model has been plotted in **figure 3.5**.

$$F = 2 \times \frac{precision \times recall}{precision + recall} \tag{3.1}$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{3.2}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{3.3}$$
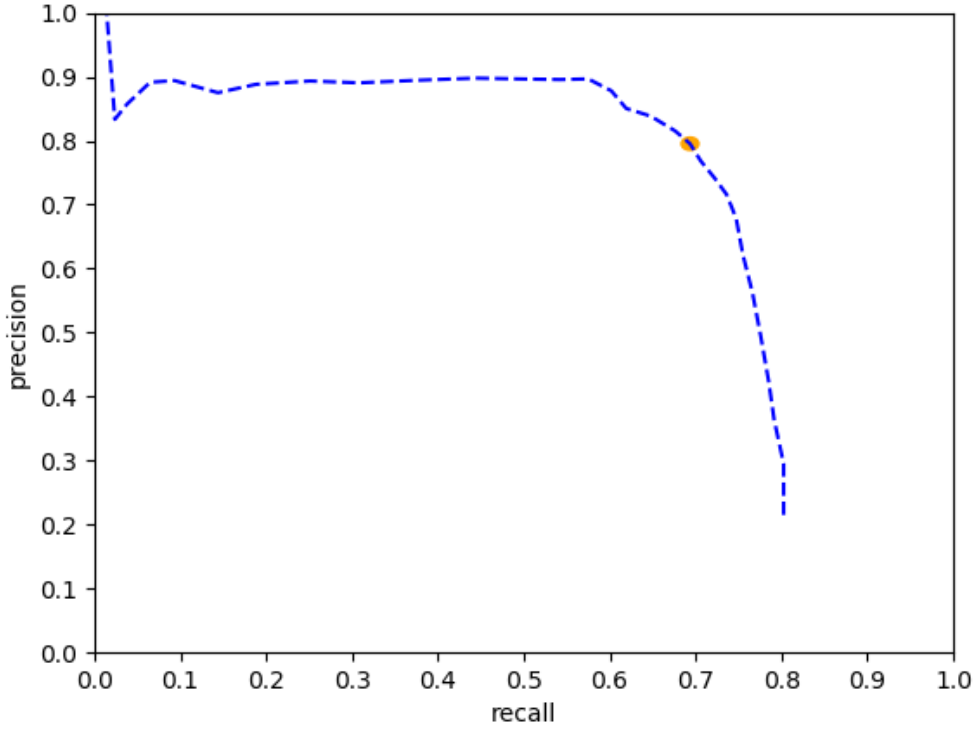
Figure 3.5: Precision recall curve of the detection model. The orange dot shows the chosen precision-recall combination of the detection model. This orange dot is positioned at a precision of 0.795 and a recall of 0.694.

To measure the quality of the segmentation model the Sørensen–Dice coefficient (DSC) was calculated (**formula 3.4**).The Sørensen–Dice coefficient represents the similarity between predicted (pixels in set X) and actual (pixels in set Y) segments. As of March 2020, The segmentation model used in this pipeline has reached a Sørensen–Dice coefficient of 0.765.

$$DSC = \frac{2|X \cap Y|}{|X| + |Y|} \tag{3.4}$$

The identification model annotated on order without the use the Nature Identification API has been tested (**table 3.3 and 3.4**). The identification model was trained with ±9 times as many detections than number of validated detections. The overall accuracy of the identification model was found to be 0.916 (**formula 3.5**).

$$ACC = \frac{True\ Positives + True\ Negatives}{True\ Positives + True\ Negatives + False\ Positives + False\ Negatives} \tag{3.5}$$

actual class

| predicted class	diptera	multiple organisms	araneae	lepidoptera	odonata	trichoptera	coleoptera	hymenoptera	unidentified	opiliones	orthoptera	hemiptera
diptera	*13*	0	0	0	0	0	0	0	2	0	0	0
multiple organisms	0	*27*	0	3	0	1	0	0	0	0	0	0
araneae	0	0	*1*	0	0	0	0	0	0	0	0	0
lepidoptera	0	2	0	*141*	0	1	0	0	0	0	0	0
odonata	0	0	0	0	*0*	0	0	1	0	0	0	0
trichoptera	0	0	0	0	0	*34*	0	0	0	0	0	0
coleoptera	0	0	0	0	0	0	*1*	0	0	0	0	0
hymenoptera	1	0	0	0	0	0	0	*7*	0	0	0	0
unidentified	0	1	0	2	0	1	0	0	*8*	0	0	2
opiliones	0	0	0	0	0	0	0	0	0	*3*	0	0
orthoptera	0	0	0	0	0	0	0	0	0	0	*1*	0
hemiptera	0	0	0	0	0	0	0	0	5	0	0	*2*
total for class	14	30	1	146	0	37	1	8	15	3	1	4

Table 3.3: Confusion Matrix for the order identification model (without the use of the Nature identification API). Multiple organisms are cases in which more than one other class is found in a detection.

class	n (truth)	n (classified)	accuracy in %	precision	recall	F1-score
diptera	14	15	98.85	0.87	0.93	0.90
multiple organisms	30	31	97.31	0.87	0.90	0.89
araneae	1	1	100	1.0	1.0	1.0
lepidoptera	146	144	96.92	0.98	0.97	0.97
odonata	0	1	99.62	0.0	0.0	0.0
trichoptera	37	34	98.85	1.0	0.92	0.96
coleoptera	1	1	100	1.0	1.0	1.0
hymenoptera	8	8	99.23	0.88	0.88	0.88
unidentified	15	14	95	0.57	0.53	0.55
opiliones	3	3	100	1	1	1
orthoptera	1	1	100	1	1	1
hemiptera	4	7	97.31	0.29	0.50	0.36

Table 3.4: Derivations of the Confusion Matrix for the order identification model (without the use of the Nature identification API). n (truth) refers to the number of actual cases in a class. n (classified) refers to the number of cases classified by the model to a class.

3.3.3 Correcting uneven illumination through image processing

At night the brightness is not constant over the entire picture causing insects at the edge of the picture to be underlit and more difficult to detect. An algorithm has been developed to compensate for the uneven illumination caused by the LED lamp by averaging the brightness over the entire picture (**figure 3.6**). This LED lamp is placed just above the camera shining directly upon the yellow screen as seen in **figure 2.1**. The illumination correction algorithm consists of the following two steps: **(1)** taking the average of the Gaussian blur (a low-pass filter used to smooth an image) of the lightness channel of all images in CIELAB colour space. **(2)** applying the computed average Gaussian blur to all images. A more in-depth explanation in the form of pseudocode can be found in **algorithm 1**.
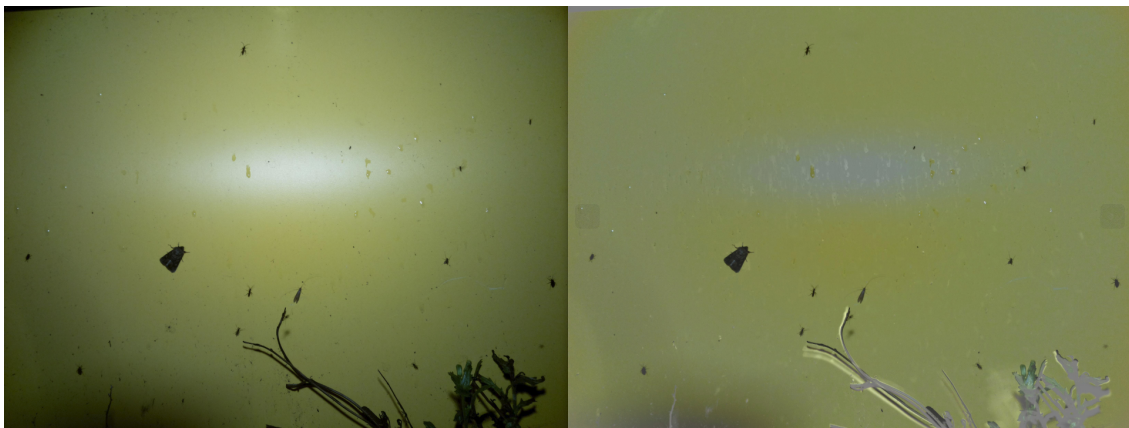
Figure 3.6: The result of applying the illumination correction algorithm to an example image. The original picture is on the left and the result of the image processing is seen on the right.

Algorithm 1: Illumination correction. This pseudocode shows how to implement the algorithm to correct the uneven illumination in insect camera images.

Data: A directory containing n original pictures taken by a single insect camera

Result: A directory with illumination corrected images

begin;

setup input and output directories;

for $i = 0 \rightarrow n$ **do**

    read current image;

    convert image to CIELAB color space;

    apply Gaussian blur with a 3 by 3 kernel to the lightness channel of the CIELAB image;

    if *first image* **then**

        | $totalGaussian \longleftarrow Gaussian$;

    else

        | $totalGaussian \longleftarrow totalGaussian + Gaussian$;

    end

    $i \longleftarrow i + 1$

end

$meanGaussian \longleftarrow \frac{Gaussian}{i}$;

for $i = 0 \rightarrow n$ **do**

    read current image;

    convert image to CIELAB colour space;

    substract the *meanGaussian* from the lightness channel of the CIELAB image;

    linearly interpolate the min and max values $\rightarrow$ 0 to 255;

    convert image back to original BGR colour space;

end

3.3.4 Tracking insects between sequentially taken pictures

Some insects (*e.g.* moths) will rest on the yellow screen for longer than the ten-second interval in between taken pictures. These individual insects will then possibly be detected more than once. To be able to have an accurate count of insects, deduplication has to be performed on detections. Here we define an object as an individual insect or another organism that is related to one or more detections.

The tracking algorithm uses an implementation based on the Hungarian algorithm (Munkres 1957). The specific implementation used in the pipeline is called "Munkres" and was written in python by Clapper 2019. The Hungarian algorithm solves the assignment problem of a two-dimensional matrix to get the lowest total cost. In this case, the rows of the distance matrix are represented by detections of the current image and the columns are represented by the detections of the previous image. A combination of the distance between relative positions in the picture and (mis-)identification score is used for the final score in each position of the matrix. The *Position only cut-off* methods match two detections if their value in the distance matrix is under the cut-off. The *identification penalty* adds the amount to the value in the distance matrix before matching if the identification is not the same on the taxonomic order level. the Different parameters have been tested and results are shown in **table 3.5 and 3.6**. No method is significantly better in both tests and at the time of writing, no method has been chosen as the definitive yet.

tracking method used	number of insects correctly tracked	number of insects total	percentage correctly tracked
Position only cut-off 0.01	131	153	85.62
Position only cut-off 0.1	133	153	86.93
Position only cut-off 0.25	117	153	76.47
Position only cut-off 1	110	153	71.9
Position cut-off 0.1, identification penalty 1	112	153	73.2
Position cut-off 0.1, identification penalty 0.1	115	153	75.16
Position cut-off 0.25, identification penalty 0.1	118	153	77.12
Position cut-off 1, identification penalty 1	110	153	71.9

Table 3.5: Results of tracking test A, the first test set. In this table, results of tuning different parameters are shown for the same implementation of a tracking algorithm.

tracking method used	number of insects correctly tracked	number of insects total	percentage correctly tracked
Position only cut-off 0.01	65	73	89.04
Position only cut-off 0.1	67	73	91.78
Position only cut-off 0.25	69	73	94.52
Position only cut-off 1	67	73	91.78
Position cut-off 0.1, identification penalty 1	57	73	78.08
Position cut-off 0.1, identification penalty 0.1	57	73	78.08
Position cut-off 0.25, identification penalty 0.1	69	73	94.52
Position cut-off 1, identification penalty 1	67	73	91.78

Table 3.6: Results of tracking test B, the second test set. In this table, results of tuning different parameters are shown for the same implementation of a tracking algorithm.

3.3.5   Estimating biomass from pictures of insects

Outside of diversity and abundance, insect trends are often measured in biomass (Hallmann, Sorg, et al. 2017; Hallmann, Zeegers, Klink, Vermeulen, Wielink, Spijkers, Deijk, et al. 2019; Homburg et al. 2019; Seibold et al. 2019; Cardoso and Leather 2019; Habel et al. 2019; Montgomery et al. 2019). The insect camera did not have a mass scale or other method to measure this biomass. However, since the distance from the camera to the screen on which insects landed was constant between all cameras, the length of insects was available to be calculated. The length in millimetres of insects could then be used together with an identification on order level to convert to biomass in milligram with formulas from Sample et al. 1993 and Sabo, Bastow, and Power 2002 as seen in **table 3.7**. Getting the length of insects was done by segmenting the insect, taking the longest axis and then counting the number of pixels from the head to the end of the body, excluding wings beyond this point. More precisely said in entomological terms: from the frons to the tip of the abdomen, excluding any cerci (**figure 3.7**). The segment can be defined as the polygon inside of a detection containing only the pixels related to the insect without pixels that are related to the background.

As a means to get the length in millimetres the number of pixels was then converted with the help of the resolution and distance of the camera. The length in millimetres was then also checked with the literature of the identified taxonomic order to be able to filter out nonsense (Gruner and Richter 1996).

paper	taxon	a	b	R^2	n
sabo et al.	Araneae	0.05	2.74	0.98	23
sabo et al.	Brachycera	0.006	3.05	0.85	30
sabo et al.	Coleoptera	0.04	2.64	0.95	119
sample et al.	Diptera	2.213	-3.184	0.7225	257
sabo et al.	Ephemeroptera	0.014	2.49	0.89	45
sample et al.	Hemiptera	3.075	-4.784	0.8649	70
sample et al.	Hymenoptera	2.696	-4.284	0.7921	274
sample et al.	Lepidoptera	3.122	-5.036	0.8649	384
sample et al.	Mecoptera	1.919	-2.912	0.6724	21
sabo et al.	Nematocera	0.1	1.57	0.9	33
sample et al.	Neuroptera	2.57	-4.483	0.9409	70
sabo et al.	Odonata	0.14	2.27	0.9	45
sabo et al.	Orthoptera	0.03	2.55	0.95	42
sabo et al.	Plecoptera	0.26	1.69	0.95	45
sample et al.	Trichoptera	3.044	-4.61	0.9216	19
sample et al.	Insecta	2.494	-3.628	0.7744	1673
sabo et al.	Pooled	0.03	2.63	0.81	541

Table 3.7: Parameters of the biomass estimation formulas. The formula for Sabo, Bastow, and Power 2002 is defined as $W = aL^b$ and Sample et al. 1993 defines their formula as $W = e^b(L)^a$. In both of these formulas, W is the dry mass in milligram and L is the length in millimetres measured from the frons to the tip of the abdomen. a and b are both constants for their respective order that have been determined by the work of Sabo and Sample. The last row shows the taxon "Pooled", the parameters of this row are used if an insect segmentation does not fall under any of the above taxa.
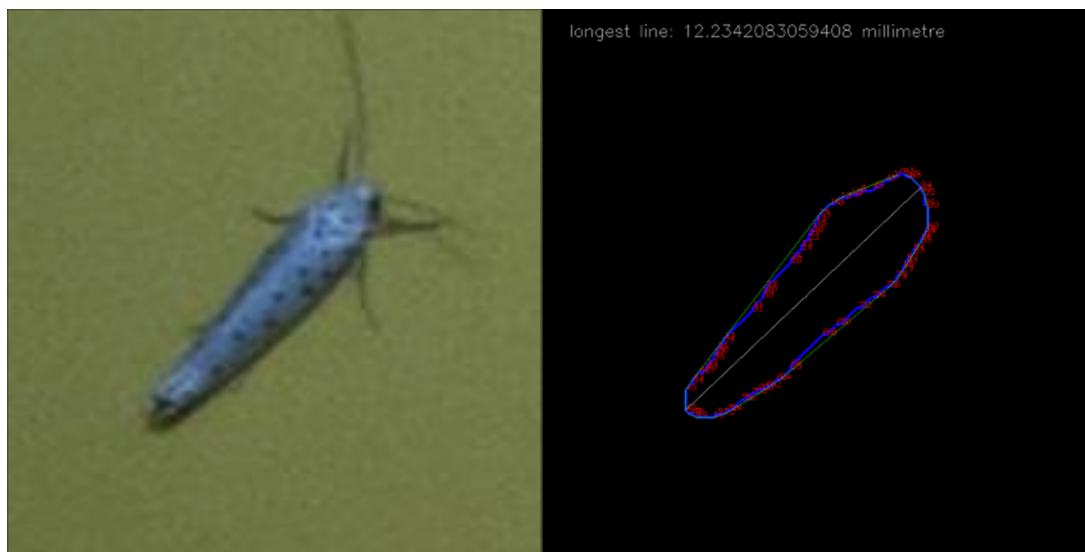
Figure 3.7: Calculating the body length of insects from the longest axis of a segment. On the left, an example crop (detection) can be seen. The right shows the segmented result in blue with corners labelled as numbers in red. The thin grey line around the segment shows the convex hull with the straight line in the segment representing the longest axis inside of the convex hull.
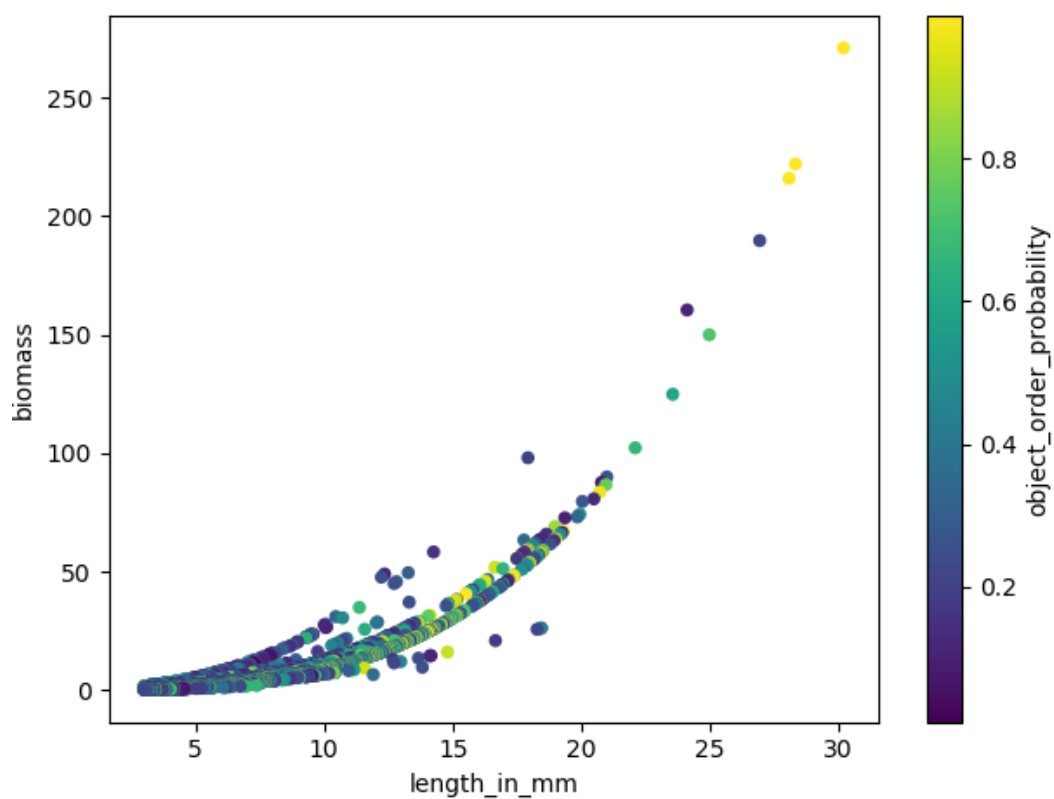
Figure 3.8: The distribution of the relationship between length and estimated biomass of a data set. The colour of the dots range from blue to yellow with blue points having a less certain identification on order level than yellow points.

### 3.3.6 Generating a HTML report of results

Each insect camera can be seen as an individual data set. The data sets generated by the pipeline can contain tens of thousands of entries. To be able to give a better overview of the data set of a camera a HyperText Markup Language document is generated with python. This document acts as a report to summarise the occurrence and biomass of insects of a given camera and mainly consists of plots and tables. Examples of what kind of plots are shown in these reports can be seen in **figure 3.9, 3.10, 3.11, 3.12 and 3.13**. At the top of the document, a summary is given of the entire data set of a camera. Further down the document, the user can find plots and tables visualizing the data per day.
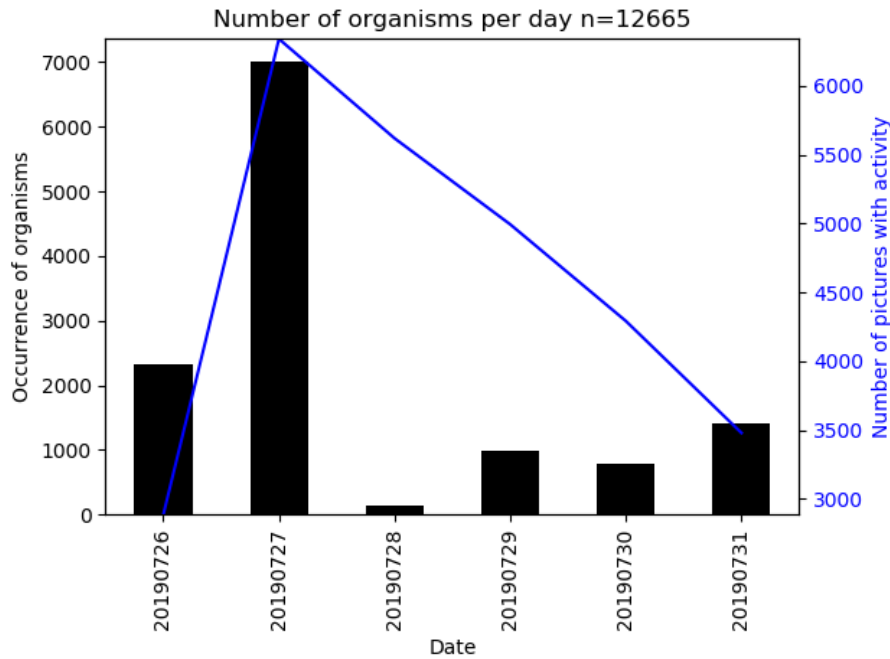

Figure 3.9: An example of the number of organisms per day. The x-axis is binned for each day labelled in ISO 8601 format (YYYYMMDD). The bars in this plot represent the occurrence of organisms counted. The blue line shows the number of pictures labelled with "activity" of each day. This plot is generated and shown for each camera in their respective HyperText Markup Language report.
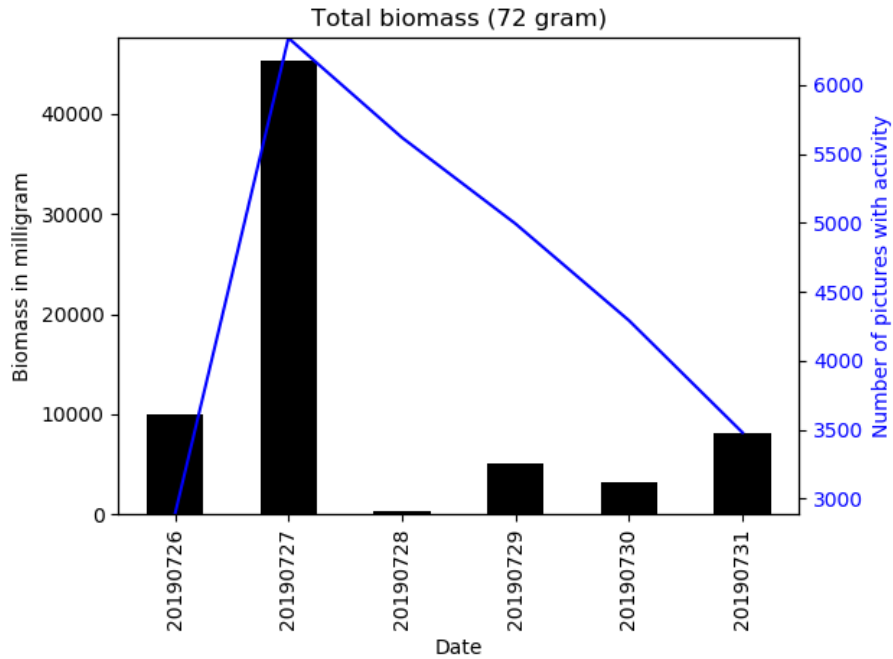
Figure 3.10: An example of the total calculated biomass of a data set. The x-axis is binned for each day labelled in ISO 8601 format (YYYYMMDD). The bars in this plot represent the biomass in milligram. The blue line shows the number of pictures labelled with "activity" of each day. This plot is generated and shown for each camera in their respective HyperText Markup Language report.
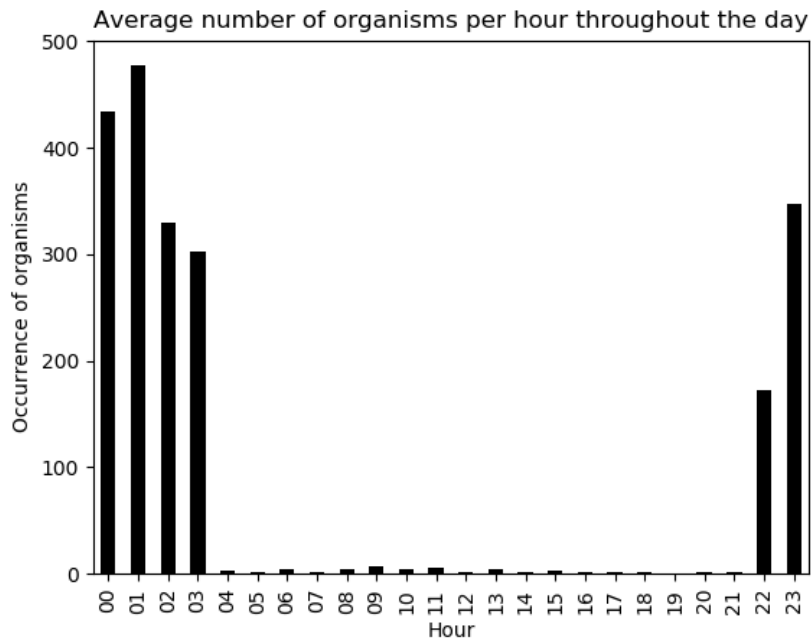
Figure 3.11: An example of insect occurrence within pictures taken by the insect camera throughout a day. On the x-axis the hour of the day is plotted in bins of sixty minutes. This plot is generated and shown for each camera in their respective HyperText Markup Language report.
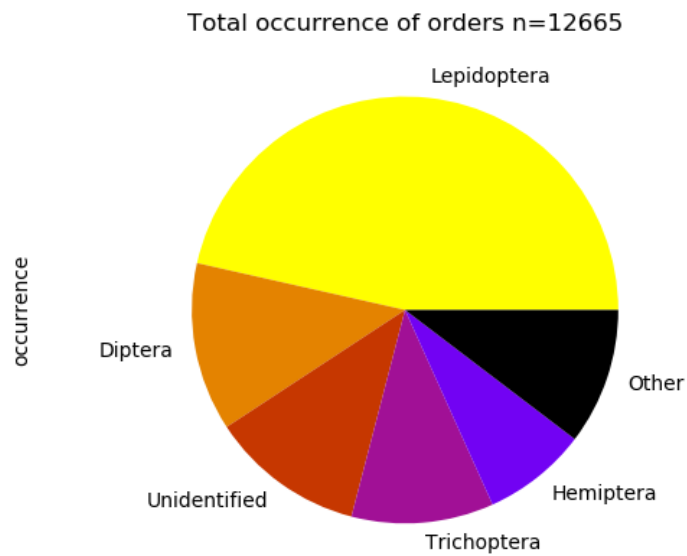
Figure 3.12: Occurrence of orders. This plot is generated and shown for each camera in their respective HyperText Markup Language report.
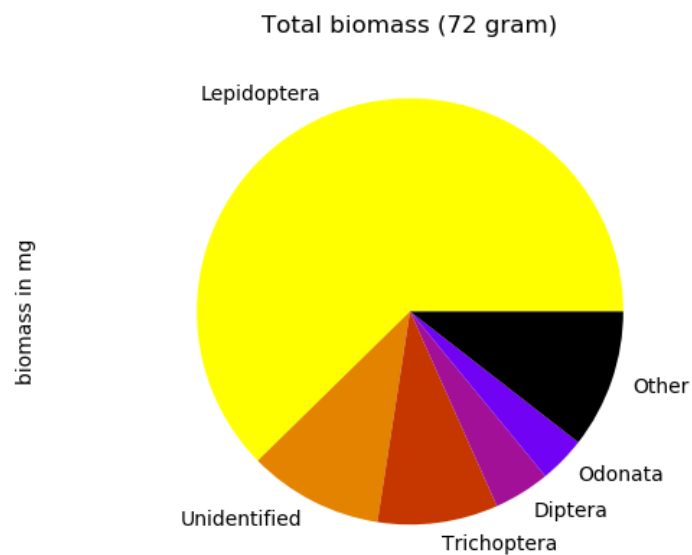
Figure 3.13: Biomass of orders. This plot is generated and shown for each camera in their respective HyperText Markup Language report.

Chapter 4

Discussion

Measuring if a picture contains activity

To remove redundant pictures and speed-up the pipeline each picture is compared to the previously taken picture and labelled if it is significantly different. In theory, this would mean that if insects have not moved positions between the interval of taken pictures, or there are no insects in the picture, a label of 'no activity' would be given to the picture. In practice, multiple different events can falsely cause the picture to be labelled with activity. Firstly, the impact of weather (such as wind or rain) can cause the picture to be significantly different from the last because of raindrops and dirt. Another problem is the vegetation being moved by the wind, causing a plant to be in a different position compared to the last picture and getting the label of 'contains activity'. Because of these reasons, it is believed that a significant part of the eight million pictures containing activity are duplicate or near-identical to each other.

Interpretation of the performance of the pipeline

Based on the precision-recall curve (**figure 3.5**) it has been found that the main problem of the detection model is false negatives. In the context of image detection, this means that the model does not detect all actual insects in the pictures.

The Sørensen–Dice coefficient of the segmentation model is mainly lowered by a few cases in which the segment prediction completely fails. The main case in which this happens is detections of *Opiliones* (colloquially known as harvesters or daddy longlegs) because of their tiny body and long legs.

The identification model on taxonomic order level based on manual annotations of insect camera data seems reliable, although more tests have to be done on classes (taxa) of low sample size.

Comparing biomass with literature

To be able to calculate the biomass the pipeline first has to calculate the body length of the insect. The body length can only be calculated with accurate segmentation of the insect. This is where the trouble of annotating these segments begin. The formulas from Sabo, Bastow, and Power 2002 and Sample et al. 1993 are based on the body length from the frons to the tip of the abdomen with cerci (*e.g.* wings or antennae) extending behind these points not included in the measurement (figure **4.1**). The annotations of these segments then had to be carefully drawn and estimated where the abdomen ends for insects that have opaque wings fully covering their abdomen. This could lead to mistakes in overestimating measured body length. however, with enough entomological knowledge this could still be estimated correctly.
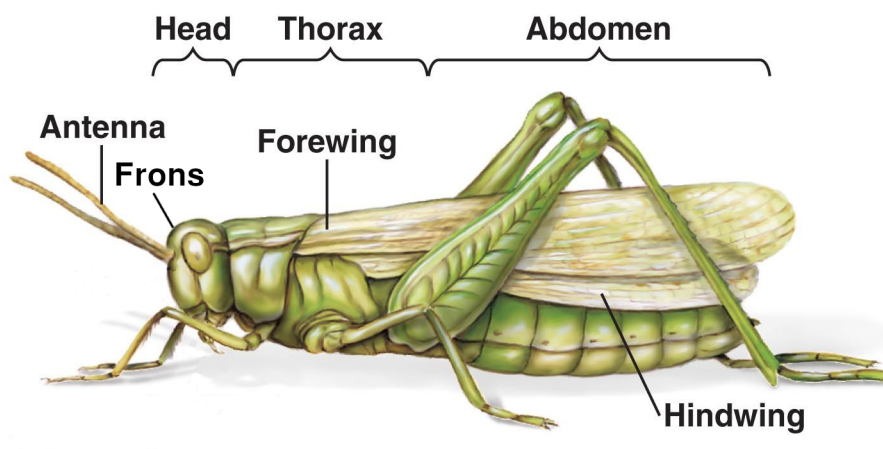
Figure 4.1: Simplified morphology of a grasshopper The body length is defined from the frons to the tip of the abdomen, with cerci such as the antennae or the wings extending outside of these ranges. Figure adapted from N. A. Campbell et al. 2017.

A comparison of measurements of terrestrial and aquatic invertebrates with the estimated biomass generated by the pipeline was made (**Figure 4.2**). As seen in this figure the estimated data points seem to closely follow the trend of measurements, being mainly centred towards the lower values.
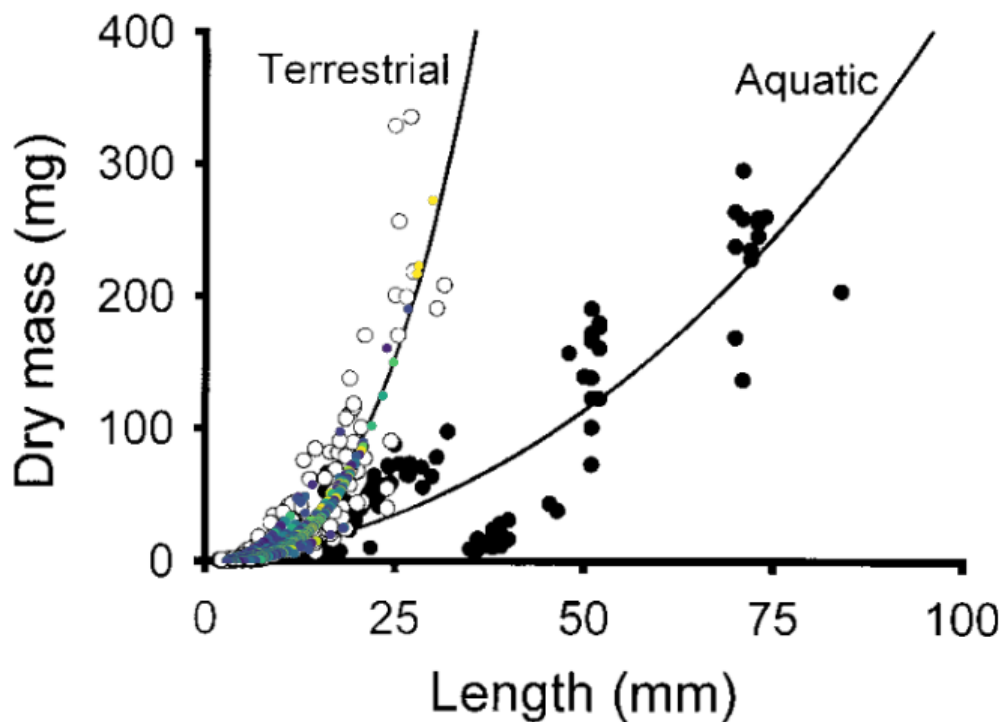
Figure 4.2: Comparison of estimated biomass with literature. Points coloured from yellow to blue represent data estimated with the pipeline. With blue points having a less certain identification on order level than yellow points. Points in white and black are measurements of terrestrial and aquatic invertebrates from Sabo, Bastow, and Power 2002 respectively. Figure adapted from Sabo, Bastow, and Power 2002.

The effects of having a LED lamp at night in the insect camera

The insect camera has a LED lamp that turns on every day from 22:00 until 04:00. It can be seen from **figure 3.11** that more insects are detected during the time that the LED lamp is on. Previous research shows that artificial light at night has a strong impact on insects such as moths, and could also be a possible driver for insect declines (Owens et al. 2019; Grubisic et al. 2018). As such, one could argue that the LED lamp of the insect camera could be an ethical concern and be possibly damaging for an insect population, especially if an insect camera is placed in nature reserves with no other artificial light at night in the surrounding area.

Training the identification based on insect camera pictures

An experiment was done to see if the identification of insects was possible to automate with the Nature Identification API (Naturalis Biodiversity Center 2018). With this API the same neural network model used by waarneming.nl was utilized on cropped images of insects. At first, this was expected as a good starting point to train the identification neural network since the model from Nature Identification API contains over one million five hundred thousand pictures of insects.

This model is not limited to insects and included over sixteen thousand different taxa. Considering the insect cameras only expect insects (and a subset of terrestrial arthropods, *Mollusca* and *Clitella*) the inclusion of other species is a disadvantage causing misidentifications. An example of the wide range of different pictures in the model causing misidentifications can be seen in **figure 4.3**.

After noticing this erroneous behaviour the classifier (the neural network that identifies insects) was manually retrained with only insect camera pictures.

(a) Picture of a *Cyprinus carpio* from waarneming.nl taken by Bex, WJ on 2018-05-06.

(b) Picture of a *Cyprinus carpio* from waarneming.nl taken by van der Knaap, T on 2018-07-05.

(c) An example cropped image of a detection taken by the insect camera during the night.

(d) An example cropped image of a detection taken by the insect camera during the day.

Figure 4.3: Comparison between insects detected in the insect camera data and pictures of *Cyprinus carpio* from waarneming.nl. The pictures above of *Cyprinus carpio* are contained in the training data set for the Nature Identification Application Programming Interface. Notice the very similar yellow shade in the background of both the pictures of *Cyprinus carpio* and the crops of insects hypothetically causing the misidentification by the Nature Identification Application Programming Interface.

The problems behind the illumination correction algorithm

An algorithm was developed to improve the identification by compensating for the uneven lighting caused by the lamp on the insect camera. However, testing showed no significant increase in identification accuracy with the Nature Identification API.

A disadvantage of using the algorithm is the fact that twice the amount of storage space is needed to save the new images if the user wants to keep the original pictures.

A positive side-effect of the illumination correction is that constant noise on the image (for example dirt and plants) are "erased". These parts get more bright than the surrounding and could lead to less false positive in the detection step. An example of this can be seen in the bottom right of **figure 3.6**. It could, however, be argued that the detection of dirt as an insect is not critical since a tracking algorithm of satisfactory quality would always deduplicate this false positive because a speck of dirt will not move between the interval of pictures taken.

Alternative tracking strategies

The major limitation of the current Hungarian tracking algorithm is the fact that it compares a picture (N) only with the one picture taken before ($N-1$). If an individual organism is seen in two non-sequential pictures it will never be tracked or deduplicated with the current method. If an insect is detected in a frame, erroneously not detected in the next frame, and detected again in the frame afterwards, the insect will be counted twice and thus misrepresent the occurrence and total biomass in a data set (**figure 4.4**).

It could be interesting to compare more than one frame backwards. This would, however, be more complex with the Hungarian tracking algorithm as the distance matrix would have to be a size of $n \times n \times n$, adding an extra dimension.
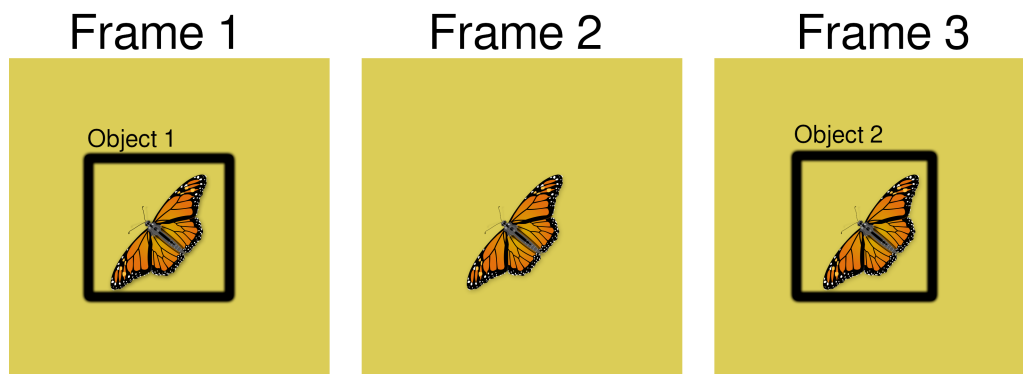
Figure 4.4: The problem of missing detections for the tracking algorithm. In this figure, Three sequential frames are schematically shown. In frame 1 and 3, the butterfly is detected, with the detection missing in frame 2. Because our implementation of a tracking algorithm only compares two sequential frames the butterfly will be counted twice.

Traditional multi-object tracking algorithms work by predicting where an object will be in a next frame based on its trajectory (Milan et al. 2016). The pictures in our data set have an interval of 10 seconds, and if insects or other terrestrial arthropods moved within those 10 seconds the path they took will not be trivial (T. Zeegers, personal communication, 2019).

The state of the art multi-object tracking algorithms are also depended on a far higher number of frames per seconds (FPS) than the 0.1 FPS in our data. These multi-object tracking algorithms are more commonly used in high frame rate videos of car and pedestrian traffic.

Another possible strategy is to greatly increase the interval between taking pictures. The hypothesis being that with an interval of thirty minutes, most insects on the first picture will not be encountered in the next picture. However, many moths (*Lepidoptera*) can be seen in the data set sitting in the same spot in many pictures over the span of hours. Furthermore, it could be argued that this sampling method will miss a lot of insects, although an insect camera photographing every thirty minutes for a month is still a significantly large sample size.

The constraint of not being able to analyse small insects

Currently, a subset of the insects is too small to be able to detect, segment and identify accurately. A rough cut-off of what makes an insect too small has been set at 3 millimetres in body length, measured from the frons to the tip of the abdomen. The cropped images of these small insects consist of too few pixels for the neural networks to give a meaningful result. It could be argued that these small insects are not as important to detect since they contain very little energy as a food source for insectivores. Yet these insects still contribute to the biodiversity of an ecosystem, and not counting these small insects would not be representative of the biodiversity of an ecosystem.

A solution to fix this problem could be to increase the resolution of the images by taking pictures with a better camera, this would, however, increase the file size of all images and could cause storage space issues. The need for more storage space would then increase the cost of the cameras.

Improvements to the hardware

The insect cameras used in this project were still early prototypes. From these first tests, it was found that there is still room for improvement.

Interesting possible future work could contain the use of edge processing with a device such as the *Nano Jetson Development Kit* from *NVIDIA*, the *Coral* by *Google* or the *Intel Movidius*. With these devices (parts of) the analysis pipeline could be performed on the insect camera itself. This has the advantage of a lot less time needed for the analysis since the device could calculate in the ten seconds interval between taking pictures. A possible problem with this could, however, be that the devices could overheat quicker in the Summer due to increased energy consumption and that the increased cost of the camera results in a larger loss when stolen or vandalized.

The need for more data for biological analysis

Insect trends can only be analysed with data points spanning multiple years. Because of this, there is currently insufficient data generated by the insect camera to perform a biological analysis for estimating the decline of insects in the Netherlands. However, once the pipeline has been used on the data of all eighty-seven cameras a comparison could be made between urban, agricultural and nature areas. It would be best to compare data of the same week to minimize the impact of weather. At the time of writing, twenty of the eighty-seven cameras have been automatically analysed by the pipeline.

Chapter 5

Conclusion

Insect cameras have great potential for monitoring trends of the entomofauna in the Netherlands.

Although the identification is currently done on order level, improving this step to family or even species level would be significantly more interesting for the analysis of the biodiversity in the Netherlands. With only thirty different insect orders the analysis of biodiversity is limited (Ishiwata et al. 2011).

From a software standpoint, the main problems to solve in future work is the deduplication (tracking) of insects and the runtime of the calculation. The deduplication is currently diminished by the fact that it can only compare two sequentially taken pictures at a time. If an insect is missed in the detection step the deduplication will not be able to assign the detection to the same object in the next picture. The performance of all steps that involve a neural network can be annotated more to increase the performance.

From a hardware standpoint, the main focus should be to make the insect camera more stable and reliable with a better way to save or even compute the data directly on the device itself. Once these cameras are of satisfactory quality they should be used for a more long-term experiment in order to get a view into the trends of insect diversity, biomass and occurrence of the Netherlands.

Appendix A

Appendix

A.1 NBA taxon to order

The code can be found under: **https://github.com/Magiduck/nba_getter**.

A.2 Source code for the pipeline

Once made public, the code can be found under: **https://github.com/naturalis/diopsis**.

A.3 AWS instance specifications

- 2nd Generation Intel Xeon Scalable (Cascade Lake) processors

- NVIDIA T4 Tensor Core GPUs

- Up to 100 Gbps of networking throughput

- Up to 1.8 TB of local NVMe storage

- Instance: g4dn.4xlarge

- GPUs: 1

- vCPU: 16

- Mem (GB): 64

- GPU Memory (GB): 16

- Instance Storage (GB): 225

- Network Performance (Gbps): Up to 25

Bibliography

Campbell, Joshua W. and J. L. Hanula (2007). "Efficiency of Malaise traps and colored pan traps for collecting flower visiting insects from three forested ecosystems". In: *Journal of Insect Conservation* 11.4, pp. 399–408. ISSN: 1366638X. DOI: `10.1007/s10841-006-9055-4`.

Campbell, Neil A et al. (2017). *Biology: a global approach*. Pearson Higher Ed.

Cardoso, Pedro and Simon R. Leather (2019). "Predicting a global insect apocalypse". In: *Insect Conservation and Diversity* 12.4, pp. 263–267. ISSN: 17524598. DOI: `10.1111/icad.12367`.

Clapper, Brian M. (2019). *Munkres implementation for Python*. `https://github.com/bmc/munkres`.

Fischer, Bob and Brendon M.H. Larson (2019). "Collecting insects to conserve them: a call for ethical caution". In: *Insect Conservation and Diversity* 12.3, pp. 173–182. ISSN: 17524598. DOI: `10.1111/icad.12344`.

Grubisic, M. et al. (2018). "Insect declines and agroecosystems: does light pollution matter?" In: *Annals of Applied Biology* 173.2, pp. 180–189. ISSN: 17447348. DOI: `10.1111/aab.12440`.

Gruner, H E and Stefan Richter (1996). "Lehrbuch der Speziellen Zoologie begrundet von A. Kaestner, 4. Auflage Bd. 1: Wirbellose Tiere. 4. Teil: Arthropoda (ohne Insecta)". In: *Systematic Biology* 45.1, pp. 119–120. ISSN: 1063-5157.

Habel, Jan Christian et al. (2019). "Agricultural intensification drives butterfly decline". In: *Insect Conservation and Diversity* 12.4, pp. 289–295. ISSN: 17524598. DOI: `10.1111/icad.12343`.

Hallmann, Caspar A., Martin Sorg, et al. (2017). "More than 75 percent decline over 27 years in total flying insect biomass in protected areas". In: *PLoS ONE* 12.10. ISSN: 19326203. DOI: `10.1371/journal.pone.0185809`.

Hallmann, Caspar A., Theo Zeegers, Roel van Klink, Rikjan Vermeulen, Paul van Wielink, Henk Spijkers, Jurriën van Deijk, et al. (2019). "Declining abundance of beetles, moths and caddisflies in the Netherlands". In: *Insect Conservation and Diversity* 6. ISSN: 17524598. DOI: `10.1111/icad.12377`.

Hallmann, Caspar A., Theo Zeegers, Roel van Klink, Rikjan Vermeulen, Paul van Wielink, Henk Spijkers, and Eelke Jongejans (2018). "Analysis of insect monitoring data from De Kaaistoep and Drenthe. Reports Animal Ecology and Physiology 2018-2". In: May.

Homburg, Katharina et al. (2019). "Where have all the beetles gone? Long-term study reveals carabid species decline in a nature reserve in Northern Germany". In: *Insect Conservation and Diversity* 12.4, pp. 268–277. ISSN: 17524598. DOI: `10.1111/icad.12348`.

Ishiwata, Keisuke et al. (2011). "Phylogenetic relationships among insect orders based on three nuclear protein-coding gene sequences". In: *Molecular phylogenetics and evolution* 58.2, pp. 169–180.

Lin, Tsung Yi et al. (2017). "Focal Loss for Dense Object Detection". In: *Proceedings of the IEEE International Conference on Computer Vision* 2017-Octob, pp. 2999–3007. ISSN: 15505499. DOI: 10.1109/ICCV.2017.324. arXiv: 1708.02002.

Martin, Amanda E. et al. (2018). "Flying insect abundance declines with increasing road traffic". In: *Insect Conservation and Diversity* 11.6, pp. 608–613. ISSN: 17524598. DOI: 10.1111/icad.12300.

Milan, Anton et al. (2016). "MOT16: A Benchmark for Multi-Object Tracking". In: pp. 1–12. arXiv: 1603.00831. URL: http://arxiv.org/abs/1603.00831.

Montgomery, Graham A. et al. (2019). "Is the insect apocalypse upon us? How to find out". In: *Biological Conservation* August, p. 108327. ISSN: 0006-3207. DOI: 10.1016/J.BIOCON.2019.108327. URL: https://www.sciencedirect.com/science/article/pii/S0006320719313497.

Morse, Douglass H (1971). "The insectivorous bird as an adaptive strategy". In: *Annual Review of Ecology and Systematics* 2.1, pp. 177–200.

Munkres, James (1957). *On the assignment and transportation problems (abstract)*. DOI: 10.1002/nav.3800040112.

Naturalis Biodiversity Center (2018). "User manual Nature Identification 1 API". In:

Ollerton, Jeff, Rachael Winfree, and Sam Tarrant (2011). "How many flowering plants are pollinated by animals?" In: *Oikos* 120.3, pp. 321–326.

Owens, Avalon et al. (2019). "Light Pollution Is a Driver of Insect Declines". In: *SSRN Electronic Journal*. ISSN: 00063207. DOI: 10.2139/ssrn.3378835.

Powney, Gary D. et al. (2019). "Widespread losses of pollinating insects in Britain". In: *Nature Communications* 10.1, pp. 1–6. ISSN: 20411723. DOI: 10.1038/s41467-019-08974-9. URL: http://dx.doi.org/10.1038/s41467-019-08974-9.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-net: Convolutional networks for biomedical image segmentation". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9351, pp. 234–241. ISSN: 16113349. DOI: 10.1007/978-3-319-24574-4_28. arXiv: 1505.04597.

Rossum, Guido (1995). "Python reference manual". In:

Sabo, John L., Justin L. Bastow, and Mary E. Power (2002). "Length – mass relationships for adult aquatic and terrestrial invertebrates in a California watershed Author (s): John L . Sabo , Justin L . Bastow and Mary E . Power Published by : The University of Chicago Press on behalf of the Society for Freshwater". In: *Journal of the North American Benthological Society* 21.2, pp. 336–343. URL: http://www.jstor.org/stable/1468420.

Sample, Bradley E et al. (1993). "The University of Notre Dame Estimation of Insect Biomass by Length and Width Source : The American Midland Naturalist , Vol . 129 , No . 2 (Apr ., 1993), pp . 234-240 Published by : The University of Notre Dame Stable URL : http://www.jstor.org/stable/". In: *The America Midland Naturalist* 129.2, pp. 234–240.

Sánchez-Bayo, Francisco and Kris A.G. Wyckhuys (2019). "Worldwide decline of the entomofauna: A review of its drivers". In: *Biological Conservation* 232.September 2018, pp. 8–27. ISSN: 00063207. DOI: `10.1016/j.biocon.2019.01.020`.

Seibold, Sebastian et al. (2019). "Arthropod decline in grasslands and forests is associated with landscape-level drivers". In: *Nature* 574.7780, pp. 671–674. ISSN: 14764687. DOI: `10.1038/s41586-019-1684-3`. URL: `http://dx.doi.org/10.1038/s41586-019-1684-3`.

Simmons, Benno I. et al. (2019). "Worldwide insect declines: An important message, but interpret with caution". In: *Ecology and Evolution* 9.7, pp. 3678–3680. ISSN: 20457758. DOI: `10.1002/ece3.5153`.

Snijders, Jeroen et al. (2014). "Project Bio-API". In: *TDWG 2014 ANNUAL CONFERENCE*.

Szegedy, Christian et al. (2017). "Inception-v4, inception-ResNet and the impact of residual connections on learning". In: *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, pp. 4278–4284. arXiv: `1602.07261`.

Wagner, David L (2019). "Global insect decline: comments on Sánchez-Bayo and Wyckhuys (2019)". In: *Biological Conservation* 233, pp. 332–333.