

Development of a multi-sensor data acquisition and computer vision system for honeybee colony health monitoring

Lowie Coussée, Dieter Van Hove

Student numbers: 02109071, 02008145

Supervisors: Prof. dr. ir. Paul Devos, Prof. dr. Dirk de Graaf

Counsellor: Dr. Ellen Danneels

Master's dissertation submitted in order to obtain the degree of Master of Science in Electrical Engineering Technology - Electrical Engineering

Academic year 2024-2025

Acknowledgment

We would like to express our sincere gratitude to our promotor, Prof. dr. ir. Paul Devos and Prof. dr. Dirk de Graaf, for their valuable guidance, constructive feedback and continuous support throughout the development of this thesis. Their expertise and critical insights have been instrumental in shaping the direction and quality of our research. We are also deeply grateful to our counselor, Dr. Ellen Danneels, for her dedicated supervision, thoughtful advice and encouragement during every stage of this work. Her commitment and availability greatly contributed to the progress of our project. Furthermore, we wish to thank the employees at Honeybee Valley for their practical support, willingness to share their knowledge and brainstorming sessions. Their assistance was of great value in the execution of our research activities. We would especially like to thank our families and friends for their unwavering support and encouragement throughout this journey. In particular, we are deeply thankful to our parents for always being there to listen and for their steadfast belief in us. We also thank our significant others for their patience, understanding and continued interest in our work. Without their presence and support, this thesis would not have been possible. Finally, we extend our appreciation to all those who supported us in this important chapter of our lives. Your contributions, in various forms, have been truly appreciated.

"The author(s) gives (give) permission to make this master dissertation available for consultation and to copy parts of this master dissertation for personal use. In all cases of other use, the copyright terms have to be respected, in particular with regard to the obligation to state explicitly the source when quoting results from this master dissertation."

22/05/2025

Disclaimer oral presentation master's thesis

This master's dissertation is part of an exam. Any comments formulated by the assessment committee during the oral presentation of the master's dissertation are not included in this text.

Use of artificial intelligence

Throughout this thesis, AI tools were used to support the development and analysis process. Specifically, AI-assisted platforms were consulted to help write and troubleshoot Python code used for image processing, data handling and statistical analysis. Additionally, AI was used to identify appropriate statistical methods, such as selecting relevant tests (e.g., ANOVA, PCA, MANOVA) based on the nature of the data. These tools accelerated development, reduced debugging time and ensured methodological accuracy, while all final decisions and interpretations were made by the researcher. These tools also supported the writing process, assisting in translating the initial draft from Dutch to English and transferring the text from Word to LaTeX, refining grammar and style, rewriting unclear sections for improved readability and brainstorming off of terms provided by the researcher.

Abstract

This thesis presents an integrated system for automated honeybee (*Apis mellifera* L.) monitoring, combining computer vision, custom hardware and web-based data visualization. It includes a computer vision-based arenatest, a sound stimulus study, a pollen monitoring setup and thermal imaging to track bee behavior inside the hive.

The arenatest, using a Raspberry Pi system, tracks five bees and a Varroa destructor mite. Through thresholding and contour analysis, bee and mite positions are identified and analyzed using behavioral metrics to assess potential Varroa resistance. A custom pollen monitoring setup records returning bees from below. Color-based segmentation and contour analysis detect pollen loads, while color correction improves accuracy. This enables quantification of foraging activity and hive efficiency. A sound stimulus experiment tests whether hive health can be inferred from bee responses to a short sound. Audio features are extracted and analyzed using ANOVA, PCA and MANOVA, revealing significant responses that could correlate with colony condition. Thermal imaging beneath the hive captures bee movement and temperature distribution. Image zones help track activity in specific hive regions, offering insights into colony size and behavior without opening the hive. All data is stored and visualized through a custom website hosted on a Raspberry Pi. Despite some use of simulated data, the complete system architecture was tested. The results show this non-invasive monitoring approach is valuable for research and practical beekeeping.

Keywords: Honeybee monitoring, computer vision, non-invasive beekeeping, sound stimulus response, thermal imaging

Inhoudsopgave

Abstract	v
List of Figures	xxi
List of Tables	xxvi
List of Acronyms	xxvii
1 Introduction	1
2 Objective	4
3 Literature study	6
3.1 The honeybee	6
3.1.1 Bee roles	6
3.1.2 Beehive structure	8
3.1.3 Bees outside the hive	10
3.2 <i>Varroa destructor</i>	11
3.2.1 Impact of the <i>Varroa destructor</i> on the honeybee in Europe	12
3.2.2 Control methods	13
3.2.3 Arenatest	15
3.3 Knocker test	16
3.3.1 Experiment with one hive	17
3.3.2 Difference of signal through the year	19
3.3.3 Multiple colonies	20
3.3.4 Conclusion of the paper	20
3.3.5 Hearing bees	21
3.4 Infrared imaging	21
3.5 Pollen counting	22
3.5.1 Purpose	22
3.5.2 Traditional methods of pollen monitoring	22

3.5.3	Computer vision for pollen detection	23
3.5.4	Impact on bees and hives	25
3.5.5	Conclusion	25
4	Arenatest	26
4.1	Simulation	26
4.1.1	Generating the simulation	26
4.1.2	Simulation analysis and classification	28
4.1.3	Results	31
4.2	Test setup	31
4.2.1	Equipment	31
4.2.2	Test conditions and parameters	35
4.3	Computer vision based monitoring and analysis	36
4.3.1	Detection and classification of bees and mites	36
4.3.2	Statistic and metric extraction	44
4.3.3	Analysis of extracted metrics	50
4.3.4	Data storage and -access	68
4.4	Limitations, challenges and future perspectives	68
4.5	Conclusion	69
5	Pollen monitoring	71
5.1	System setup and data collection	71
5.2	Detection and counting algorithm	75
5.3	Limitations, challenges and future perspectives	81
5.4	Conclusion	82
6	Sound stimulus	83
6.1	Purpose	83
6.2	Test setup	83
6.3	Statistic analysis results	85
6.3.1	Comparison between bee and no bee recordings	85
6.3.2	Bee reaction to the stimulus	93
6.4	Practical implementation	116
6.5	Conclusion	117
7	Infra-red imaging	119
7.1	Purpose	119
7.2	Test setup	119

7.3	Results	120
7.4	Practical implementation	131
7.5	Conclusion	132
8	Prototype bee monitoring system	134
9	Website	136
9.1	Website overview	136
9.1.1	Homepage	137
9.1.2	Sensordata page	138
9.1.3	Pollen monitoring page	139
9.1.4	Arenatest page	140
9.1.5	Knocker test page	142
9.1.6	Infrared imaging page	142
9.1.7	Activity page	143
9.1.8	Logs page	144
9.2	Technical implementation	145
9.2.1	Frontend	145
9.2.2	Backend	147
9.2.3	Data flow	148
9.2.4	Node-RED testing	148
9.3	Deployment and accessibility	149
9.4	Limitations, challenges and future perspectives	150
9.5	Conclusion	150
10	Ethical and social reflection	152

List of Figures

1.1	Summation of major beehive products. (Gladstone, n.d.)	2
3.1	Visual comparison between the different bee types (Hooningkorf, 2020)	8
3.2	Structure of a beehive (Danneels, n.d.)	9
3.3	The <i>Varroa destructor</i> (Texas Apiary Inspection Service, n.d.)	11
3.4	Effects of mite invasion over time (Oliver, 2006)	12
3.5	Bee and mite population growth curves for a temperate climate (Oliver, 2006)	13
3.6	Thermotherapy bee treatment (ThermoMite, n.d.)	14
3.7	Result of the knock on the beehive. (A) A collection of three-second-long accelerometer recordings, with the time of the recording on the horizontal axis, and the time relative to the artificial stimulus on the vertical axis. (B) A plot of a particular recording taking place on the 29th Aug 2021 with a remarkably strong colony response, which is also made. Taken from Bencsik et al. (2024)	18
3.8	A: Magnitude of the signal that reaches the accelerometer. B: Magnitude of the positive response from the bees. C: Magnitude of the negative response from the bees. Taken from Bencsik et al. (2024)	19
3.9	Top: average of transmitted vibration. Bottom: average of vibrational response. Taken from Bencsik et al. (2024)	20
3.10	Front porch type traditional pollen trap (Amazon, n.d.)	23
4.1	Spawn zones for bees and mites in the arenatest simulation	27
4.2	Simulation bee and mite detection	29
4.3	Arenatest setup diagram	31
4.4	Arenatest setup in the Honeybee Valley lab	33
4.5	Raspberry Pi 5 placement in arenatest setup	34
4.6	HSL and HSV explained visually (Wikipedia, 2025)	37
4.7	Arenatest bee (green) and mite (red) classifications, with the circular mask visible	38
4.8	Pseudo-code for the first part of ellipse fitting method	40
4.9	Pseudo-code for the second part of ellipse fitting method	41
4.10	Pseudo-code for the last part of ellipse fitting method	41
4.11	Case where ellipse fitting was used to be able to separate close together bees	42
4.12	Arenatest bee detection process	42

4.13	Arenatest mite detection process	43
4.14	Bee-to-mite distance visualized. Taken from the arenatest performed on 25/04/2025	44
4.15	Intra-bee distance visualized. Taken from the arenatest performed on 25/04/2025	45
4.16	Bee-to-center distance visualized. Taken from the arenatest performed on 25/04/2025	46
4.17	Bee-to-edge distance visualized. Taken from the arenatest performed on 25/04/2025	47
4.18	Mite-to-center distance visualized. Taken from the arenatest performed on 25/04/2025	48
4.19	Heatmap of bee positions. Taken from the arenatest performed 25/04/2025	49
4.20	Heatmap of mite positions. Taken from the arenatest performed 25/04/2025	50
4.21	Bee-to-mite distance metric for timeframe 2 of the arenatest performed on 14/03/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.	52
4.22	Intra-bee distance metric for timeframe 2 of the arenatest performed on 14/03/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.	54
4.23	Heatmap of bee positions of the arenatest performed on 14/03/2025	55
4.24	Bee-to-center distance metric for timeframe 2 of the arenatest performed on 14/03/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.	56
4.25	Amount of detected bees metric for timeframe 2 of the arenatest performed on 14/03/2025	57
4.26	Bee-to-mite distance metric for timeframe 1 of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.	60
4.27	Intra-bee distance metric for timeframe 1 of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.	61
4.28	Heatmap of bee positions of the arenatest performed on 25/04/2025	62
4.29	Bee-to-center distance metric for timeframe 1 of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.	63
4.30	Bee-to-edge distance metric of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.	63
4.31	Mite-to-center distance metric for timeframe 1 of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.	64
4.32	Amount of bees detected & amount of frame where bees are close to mites for timeframe 1 of the arenatest performed on 25/04/2025	65
5.1	Observation tunnel set up in front of beehive	72
5.2	Beehives in the bee hall on which data was collected	74
5.3	Observation tunnel technical drawing	75
5.4	Pseudo-code that illustrates the background subtraction process	76
5.5	Yellow pollen segmentation process	77
5.6	Dark-red pollen segmentation process	78
5.7	Pseudo-code that illustrates the pollen volume calculation process	79

5.8	Color correction process (Xrite, n.d.)	80
5.9	Frame from the extracted pollen monitoring results video	81
6.1	Setup used to sent a stimulus and record the reaction from the bee colony	84
6.2	The microphone and speakers used for the tests.	85
6.3	Box plots of the RMS value of the signal by frequency and bee presence.	88
6.4	Box plots of the RMS value of the signal by volume and bee presence.	88
6.5	Box plots of the spectral centroid of the signal by frequency and bee presence.	89
6.6	Box plots of the spectral centroid of the signal by volume and bee presence.	89
6.7	Box plots of the bandwidth of the signal by frequency and bee presence.	90
6.8	Box plots of the bandwidth of the signal by volume and bee presence.	90
6.9	Box plots of the spectral rolloff of the signal by frequency and bee presence.	91
6.10	Box plots of the spectral rolloff of the signal by volume and bee presence.	91
6.11	Box plots of the zcr value of the signal by frequency and bee presence.	92
6.12	Box plots of the zcr value of the signal by volume and bee presence.	92
6.13	RMS value from a recording at 70% volume and 1500 Hz	93
6.14	RMS value from a recording at 70% volume and 1500 Hz	94
6.15	RMS value from a recording at 70% volume and 1500 Hz	94
6.16	RMS value from a recording at 70% volume and 1500 Hz	95
6.17	RMS value from a recording at 70% volume and 1500 Hz	96
6.18	RMS value from a recording at 70% volume and 1500 Hz	96
6.19	knock on the beehive	97
6.20	Scatter plot that shows the distribution of recordings by PC1 and PC2 components for PCA with 5 features	98
6.21	Scatter plot that shows the distribution of recordings by PC1 and PC2 components for PCA with 31 features	99
6.22	Box plot for t-test on PC1, bee presence	100
6.23	Contribution of the different features to principal component 1	101
6.24	Contribution of the different features to principal component 2	101
6.25	Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 0.01s before and after the stimulus	102
6.26	Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 0.1s before and after the stimulus	103
6.27	Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 0.3s before and after the stimulus	103
6.28	Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 0.5s before and after the stimulus	104

6.29	Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 1s before and after the stimulus	104
6.30	Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 2 s before and after the stimulus	105
6.31	Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 3 s before and after the stimulus	105
6.32	Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 5 s before and after the stimulus	106
6.33	PC1 shift for each frequency	107
6.34	PC1 shift for each frequency at 35% volume	108
6.35	PC1 shift for each frequency at 70% volume	108
6.36	Plots of two recordings at 1500 Hz for both volume levels with bees	110
6.37	Plots of two recordings at 3500 Hz for both volume levels with bees	110
6.38	Plots of two recordings at 1500 Hz for both volume levels without bees	110
6.39	Plots of two recordings at 3500 Hz for both volume levels without bees	111
6.40	The explained variance for the first five principal components	111
6.41	Scatter plot PCA 0,2 seconds after the stimulus	112
6.42	Violin plot for PC1 0,2 seconds after the stimulus	112
6.43	Violin plot for PC2 0,2 seconds after the stimulus	113
6.44	Violin plot for PC3 0,2 seconds after the stimulus	113
6.45	Violin plot for PC4 0,2 seconds after the stimulus	114
6.46	Violin plot for PC5 0,2 seconds after the stimulus	114
6.47	PC1 shift for each frequency 0,2 seconds after the stimulus	115
7.1	MLX90640 thermal camera	120
7.2	Thermal image used in GIF	121
7.3	Hottest location in the hive throughout a day	122
7.4	Hottest location in the hive throughout a day on a thermal image	123
7.5	Standard deviation of the position	124
7.6	Distance from mean thermal centroid	124
7.7	Temperature at the edges of the thermal	125
7.8	Number of clusters in the beehive	126
7.9	Size of the clusters	126
7.10	Thermal image divided in zones	127
7.11	Mean temperature for each zone	127
7.12	Standard deviation for each zone	128
7.13	Movement in y-direction for each zone	129

7.14	Movement in x-direction for each zone	129
7.15	Movement in y-direction for each zone during ten minutes	130
7.16	Mean temperature for each zone during 10 minutes	131
7.17	Standard deviation for each zone during 10 minutes	131
8.1	Setup for all the sensors underneath the beehive	135
9.1	Website homepage	138
9.2	Website sensordata page	139
9.3	Website pollen monitoring page	140
9.4	Website arenatest page	141
9.5	Print screen from the knocker page	142
9.6	Print screen from the infrared page	143
9.7	Website activity page	144
9.8	Website logs page	145
9.9	Node-RED pollen- and sensor dummy data generating flow	149

List of Tables

3.1	Worker bees' tasks at different ages (Van De Putte G., personal communication, August 5 th 2024) . . .	7
4.1	Area thresholds used for bee and mite classification of contours in the arenatest simulation	30
4.2	Equipment used in the arenatest setup	32
4.3	Bee classification thresholds of contours for the arenatest	39
6.1	Three-way ANOVA bee presence	87
6.2	Three-way ANOVA bee presence, interactions	87
6.3	Paired t-test on PC1	102
6.4	Paired t-test results for each frequency	107
6.5	One-way ANOVA on delta PC1	109
6.6	One-way ANOVA on delta PC2	109
6.7	MANOVA Test Results for Segment	116

List of Acronyms

ABPV Acute Bee Paralysis Virus.

AI Artificial Intelligence.

ANOVA Analysis of variance.

API Application Programming Interface.

BGR Blue Green Red.

CCD Colony Collapse Disorder.

CNN Convolutional Neural Network.

CORS Cross-Origin Resource Sharing.

CSI Camera Serial Interface.

CSS Cascading Style Sheet.

CSV Comma Seperated Values.

DOM Document Object Model.

DWV Deformed Wing Virus.

FAO Food and Agriculture Organization.

fps Frames per second.

HSL Hue Saturation Lightness.

HSV Hue Saturation Value.

HTML Hypertext Markup Language.

HTTP Hypertext Transfer Protocol.

HTTPS Hypertext Transfer Protocol Secure.

I2C Inter integrated circuit.

ID Identification.

IP Internet Protocol.

IPBES Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services.

IR Infrared.

ISO International Organization for Standardization.

JSON JavaScript Object Notation.

LED Light Emitting Diode.

MANOVA Multivariate analysis of variance.

mfcc Mel frequency cepstral coefficients.

MIPI Mobile Industry Processor Interface.

PC Principal component.

PCA Principal component analysis.

PNG Portable Network Graphics.

RAM Random Access Memory.

RGB Red Green Blue.

RMS Root mean square.

ROI Region Of Interest.

SDG United Nations Sustainable Development Goals.

TCP Transmission Control Protocol.

TPS Thin Plate Spline.

USB Universal Serial Bus.

VSH Varroa-sensitive hygiene.

YOLO You Only Look Once.

zcr Zero crossing rate.

1

Introduction

Honeybees (*Apis mellifera* L.) play an indispensable role in our ecosystems and food production, making them one of the most important pollinators on the planet. Honeybees play a vital role in maintaining natural ecosystems, ensuring the survival of wild plant species that provide habitat and food for numerous other organisms. Their ability to transfer pollen between flowering plants ensures the reproduction of countless species, supports biodiversity and contributes to global food security. Without their tireless work, many of the fruits, vegetables and nuts we rely on, such as cherries, apples, blackberries, raspberries, mangoes, zucchinis, bell peppers and avocados, would become scarce or even disappear. In addition, essential agricultural products such as sunflower oil and mustard, as well as forage crops that support livestock, depend on bee pollination (Departement Omgeving, 2025). According to the Food and Agriculture Organization (FAO) of the United Nations, a third of the world's food production is dependent on bees. This shows how beekeeping serves as a significant source of income for many individuals and communities. As stated by the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services (IPBES), the honeybee is the most widely managed pollinator worldwide, with over 80 million hives producing approximately 1.6 million tonnes of honey each year (United Nations Environment Programme, 2022).

Although honeybees are best known for their essential role in pollination and honey production, their contributions extend beyond these functions. In addition to honey, honeybees produce several valuable substances, including beeswax, royal jelly, propolis and pollen, each with unique applications in various industries. Beeswax is a natural secretion used by bees to construct their combs and has been utilized for centuries in candle-making, cosmetics and pharmaceuticals. Royal jelly is a nutrient-rich secretion that serves as the exclusive food for queen larvae. It is widely studied for its potential health benefits, including immune support and anti-aging properties. Propolis is a resinous compound collected from plant sources and acts as a natural antimicrobial agent within the hive and has been traditionally used in medicine for its antibacterial and anti-inflammatory effects. Finally, bee pollen is a rich source of proteins, vitamins and antioxidants and is commonly marketed as a dietary supplement (Pasupuleti et al., 2017). A selection of these products is illustrated in Figure 1.1.

1 Introduction



Figure 1.1: Summation of major beehive products. (Gladstone, n.d.)

Beyond their material contributions, honeybees have considerable cultural and historical importance. Across civilizations bees have symbolized industry, cooperation and resilience. In ancient Egypt, bees were associated with royalty and divinity, and honey was considered a sacred substance. Greek and Roman societies revered honey as the "nectar of the gods," incorporating it into medicine and religious rituals. Many indigenous cultures have also recognized the vital role of bees in nature's balance, integrating bee products into traditional healing practices. In contemporary society, honeybees have become a powerful symbol of environmental sustainability and biodiversity, representing the delicate balance necessary to maintain healthy ecosystems (Prendergast et al., 2021).

Despite their ecological, economic, and cultural significance, bee populations worldwide are under severe threat. A combination of natural and human-induced factors has led to alarming declines in bee numbers. Habitat loss due to urbanization and agricultural intensification has drastically reduced the availability of wildflowers and nesting sites. Pesticides, particularly neonicotinoids, have been associated with weakened immune systems and impaired navigation in bees, making them more susceptible to environmental stressors (WildBnB, 2022). Climate change further exacerbates these challenges by disrupting flowering cycles, leading to mismatches between bees and their food sources, while extreme weather events such as droughts and floods further stress already vulnerable colonies (Le Conte & Navajas, 2008). Air pollution is also believed to have a negative impact on bees. Early research suggests that air pollutants alter the scent molecules emitted by plants, which bees rely on to find food. This disruption confuses their foraging behavior, slowing them down and reducing their efficiency as pollinators (Potera, 2008). Additionally, competition with invasive species and increased pathogen prevalence contribute to the complex and multifaceted decline of bee populations worldwide (WildBnB, 2022). However, steps are already being taken in the right direction. In May 2018, the European Union upheld a partial ban on three insecticides known as neonicotinoids to mitigate the lethal threat they pose to bees and their trickle-down effect on pollination as a whole (DW, 2018).

One of the most significant threats to the Western honeybee is the Varroa mite (*Varroa destructor*), a parasitic species that has spread rapidly in recent decades. The danger lies in the fact that honeybees do not perceive the *Varroa destructor* as a threat and therefore exhibit little or no defensive behavior, allowing the infestation to

1 Introduction

spread largely unchecked. This results in entire honeybee colonies disappearing due to the *Varroa destructor*, if not for human intervention (Bayer AG, 2019).

The Varroa mite latches onto adult bees and developing larvae, feeding on their hemolymph (the bee's equivalent of blood) and transmitting deadly viruses, such as Deformed Wing Virus (DWV) and Acute Bee Paralysis Virus (ABPV). They enter a beehive by clinging onto worker bees and hiding in places like under their wings while they enter the beehive. Their goal is to enter the brood cells which contain bee larvae, in which they then lay eggs that create more mites that feed of the bee colony (Honing van de Imker, n.d.). Research also shows that climate change is helping the spread of the Varroa mite. This is because warmer temperatures can speed up their reproductive cycle (Le Conte & Navajas, 2008). Infestations of Varroa mites weaken colonies, reducing their ability to forage and reproduce, often leading to colony collapse if left unmanaged. The widespread prevalence of this parasite has made Varroa control a top priority for beekeepers and researchers alike, as its presence can devastate entire apiaries within a single season. The ability of the Varroa mite to develop resistance to common chemical treatments further complicates management efforts, necessitating the exploration of alternative control strategies, such as selective breeding for mite-resistant bees, biotechnological interventions and integrated pest management approaches. The fight against the Varroa mite involves multiple approaches, each with its own advantages and limitations (Honing van de Imker, n.d.).

To assist in finding these Varroa resistant bee strains, an experiment called "arenatest" can be performed. This allows the behavior of bees and mites to be analyzed, possible grooming behavior (a hygienic action in which bees use their legs or mouthparts to remove parasites such as Varroa mites from their own bodies, or from the bodies of nest mates) to be observed and different selection strategies to be pursued (Bahreini et al., 2021).

The appendices of this thesis are available through the following GitHub repositories. These include all data, code and results for their respective chapters.

- GitHub repository for arenatest, pollen monitoring and website data
- GitHub repository for sound stimulus test and tests with infrared imaging

2

Objective

The objective of this thesis is to develop a comprehensive data acquisition system that integrates mechanical, electronic and software components to monitor key aspects of honeybee colony health and behavior in a non-invasive way. This includes monitoring the bees reaction to external stimuli, pollen foraging patterns, behavior in the presence of the *Varroa destructor* and location in the beehive using multiple types of sensors connected to a Raspberry Pi, which will serve as the central processing unit. The collected data will be stored in a database structure on the Raspberry Pi where it can be processed, analyzed and visualized in real-time via a website. This system will provide insight into the health of beehives for researchers at Honeybee Valley and, ultimately even beekeepers, through a user-friendly interface.

The thesis will go deeper into the results of the knocker test (Bencsik et al., 2024). This is a test where a short pulsed vibration, or a knock, is applied to the beehive. This knock is supposed to provoke a reaction from the bees. This reaction is measured by an accelerometer that will show the strength of the reaction signal. This data can then be used to determine the strength of the beehive. An alternative method will be examined to induce and measure the stimulus. Here, the stimulus induced by the knocker will be replaced by a sound played by speakers. This sound is a short sinusoidal pulse of a certain frequency that is played underneath the beehive. In this adaptation, the accelerometer is replaced by a microphone underneath the beehive. This way, there is no need to open the hive and disturb the bees, which would be necessary to place an accelerometer. This method, if proven successful, can then be used to distinguish a weak and strong bee colony.

In addition, an infrared camera will be placed underneath the beehive. This camera will produce an image of the size and location of the brood, which will give more information about how the bees move through the beehive throughout the winter. The images of the camera can be used to analyze the movement of the brood through the beehive for a certain duration. Multiple graphs can be made showing the temperature and movement of the bees throughout the beehive. This analysis can also be extended to certain parts of the beehive, such as the zones between the honeycombs. This provides a lot of information for the beekeeper without the need to ever open the beehive. The camera can be used on different types of beehives to examine the reaction of bees to the chosen beehive. A different amount of isolation might trigger a certain level of bee movement.

2 Objective

A monitoring setup will be installed at the entrance of the beehive to track the incoming pollen using a Raspberry Pi camera. This setup will feature a narrow corridor with a plexiglass floor that guides incoming bees over the camera, which is positioned beneath the glass. A white background will be placed above the corridor to enhance image contrast. Using computer vision techniques, such as different filters and thresholding, a segmentation of pollen on the bees' legs can be made. The volume of the pollen can be determined, by assuming that they are a perfect sphere. In automating this traditionally hard to quantify process, the system will contribute to a more precise assessment of foraging efficiency and colony health. Ultimately, the goal is to create an efficient, real-time monitoring platform that supports beekeeping research and sustainable hive management that gives the researchers of Honeybee Valley valuable insights in their beehives.

Beyond hive monitoring, this thesis will investigate the behavior of honeybees in response to the *Varroa destructor* through a controlled arena test as mentioned in the introduction. This experiment will involve placing several bees in a circular arena along with a few Varroa mites, aiming to assess how the bees interact with this parasite. Key behavioral questions include whether bees actively approach or avoid mites, whether they exhibit cautious or rapid movement and whether they display defensive or aggressive responses. The findings of this test will aid in the identification of Varroa-resistant colonies, which could support future breeding programs and improve colony survival rates. The goal is to achieve this monitoring system without the use of artificial intelligence (AI), as this would strain the computational resources of the Raspberry Pi. Traditional computer vision techniques, which are more resourceful, will be used.

The website that accompanies this thesis will consist of multiple pages that display live sensor- and pollen data, alongside historical records. It will also provide access to past test results, as well as images captured by the infrared camera. The website will be hosted directly on the Raspberry Pi and will use a local server to retrieve data stored in a folder-based database structure.

3

Literature study

3.1 The honeybee

3.1.1 Bee roles

In a honeybee colony there are three types of bees, that each fulfill specific roles: worker bees, drones and the queen. They differ in appearance and behavior, and their tasks change depending on their age and function within the colony.

Worker bees are female bees that perform most functions within the colony. They go through different task phases during their life cycle, which typically lasts four to six weeks in the active summer months but can be longer in winter. They are twelve to fifteen millimeters long on average and have a striped body with a brown to yellowish hue. Worker bees have wax glands to build combs, feeding glands to care for larvae and a sting to defend the colony. They have relatively short wings compared to drones and strong hind legs with pollen baskets (corbiculae) to collect pollen. Worker bees develop from fertilized eggs and undergo complete metamorphosis over approximately 21 days, transitioning through egg, larva, pupa and adult stages. The differentiation between queens and workers is primarily influenced by larval nutrition; larvae destined to become queens are fed exclusively on royal jelly, while those becoming workers receive a combination of royal jelly and bee bread (Kamakura, 2011).

Worker bees exhibit temporal polyethism, an age-related division of labor where tasks change as the bee ages (Collison, 2018). These tasks by age are shown in Table 3.1.

3 Literature study

Table 3.1: Worker bees' tasks at different ages (Van De Putte G., personal communication, August 5th 2024)

Age	Task
1-3 days	Cleaning cells to prepare them for new eggs or food storage
4-10 days	Consume pollen and honey and begin feeding the larvae
10-20 days	Produce wax and build the combs, and store nectar and pollen in the cells
20-30 days	Patrol and guard the entrance of the hive, to repel intruders such as wasps
30-40 days	Leave the hive to collect nectar, pollen, water and propolis

This division is not rigid, worker bees can revert to previous tasks in response to colony needs. This demonstrates remarkable behavioral plasticity. Hormonal changes, particularly in juvenile hormone levels, regulate these transitions and align individual development with colony requirements (Robinson, 1992).

Worker bees employ sophisticated communication methods to coordinate colony activities. The first is the waggle dance. This is a unique behavior where a forager bee communicates the location of food sources to nest mates through a series of movements that convey direction and distance relative to the sun's position. The duration and angle of the waggle run correlate with the distance and direction of the resource. This dance enables efficient exploitation of food sources and exemplifies complex non-verbal communication in insects (von Frisch, 1967). Pheromones also play a crucial role in maintaining colony cohesion and regulating behavior. Worker bees produce various pheromones, including:

- Nasonov pheromone: Used to recruit and orient foragers to the hive or new food sources.
- Alarm pheromones: Released when the colony is threatened, triggering defensive behaviors.

These chemical signals facilitate complex social interactions and ensure colony survival (Guelph, n.d.).

In recent decades, worker bees have been at the center of concerns about Colony Collapse Disorder (CCD) and global pollinator declines. Multiple stressors, including pesticides, parasites (like the *Varroa destructor*), pathogens and habitat loss, negatively affect worker bee health and foraging behavior (Goulson et al., 2015). Research has shown that sub-lethal pesticide exposure can impair navigation and reduce foraging efficiency. The vulnerability of workers to such stressors reflects the fragility of the colony's structure and resilience (Henry et al., 2012).

Drones are the male bees. They live shorter than worker bees and are usually expelled from the colony in autumn to save resources. They are slightly larger than worker bees, averaging fifteen and seventeen millimeters long. They have a darker and more robust body without the typical yellow stripes of worker bees. They have notably large eyes that cover most of their head, which is essential to find a queen during mating flights. Drones lack sting and pollen baskets, as they have no defensive or foraging roles. Throughout their lives, drones serve a single primary purpose: reproduction. They leave the hive to join mating flights in designated drone congregation

3 Literature study

areas, where they attempt to fertilize a queen. Successful mating is fatal, as the drone's reproductive organs are irreparably damaged and torn away during the process. Those that do not mate typically survive for only a few weeks. By late summer, drones are expelled from the hive and left to starve, a strategy that conserves the colony's food reserves for the winter months.

The queen is the reproductive center of the colony and the only fertile female bee. She can live up to five years, but most colonies replace their queen after one to two years. The queen is the largest bee in the colony, usually eighteen to twenty mm long. She has a long, pointed abdomen, which helps her lay eggs into the hexagonal cells. Her wings are relatively short compared to her long body. Her legs are smooth, without pollen baskets or wax glands. Depending on the subspecies, the queen may be darker or lighter, but she typically has a shinier and smoother appearance compared to worker bees (Danneels E., personal communication, October 14th 2024; Van De Putte G., personal communication, August 5th 2024).

Figure 3.1 shows the visual difference between the different bee types.



Figure 3.1: Visual comparison between the different bee types (Hooningkorf, 2020)

3.1.2 Beehive structure

The structure of a beehive reflects the organized and efficient lifestyle of honeybees. Each section within a beehive has a specific function, and the bees utilize different spaces to efficiently carry out their tasks. Beehives are often based on the Langstroth system, a modular design consisting of multiple stacked sections that can be easily opened and maintained. Each hive consists of various components that create a well-organized living environment for the bees (Danneels E., personal communication, October 14th 2024).

3 Literature study

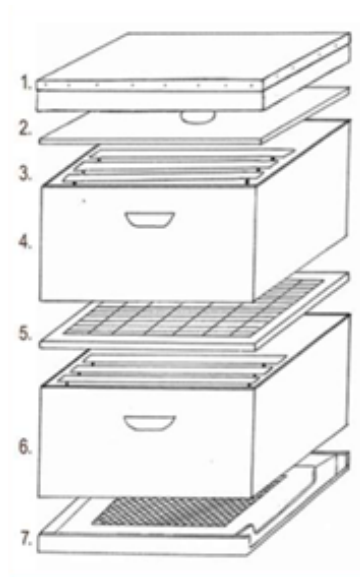


Figure 3.2: Structure of a beehive (Danneels, n.d.)

The key components of a beehive, as shown in Figure 3.2, are listed below.

1. **The lid:** The top box is sealed with a lid to protect the hive from weather conditions, and to provide extra isolation during the winter period if needed. Through the plastic lid the colony can be observed without opening the hive, reducing disturbance to the bees.
2. **The cover board:** The cover board prevents bees from being disturbed when inspecting the hive. If a plastic sheet is placed under the cover board, both can be removed at once. The sheet or board often has a hole, allowing the bees to be fed without removing the cover.
3. **Frames:** Inside the hive boxes beekeepers hang frames making it easier to inspect or extract honeycombs. These frames often contain a sheet of artificial wax. This thin layer of beeswax has cell imprint patterns on both sides, guiding the bees to build straight honeycombs. The wax sheet is reinforced with embedded wires or a wire mesh for extra stability.
4. **Honey super:** The honey super is an additional box placed above the brood chamber where the bees store honey. Beekeepers can harvest honey from these supers without disturbing the main hive.
5. **Queen excluder:** This is made of metal or plastic mesh with openings just large enough for worker bees to pass through but too small for the queen. It is placed between the brood chamber and honey supers to ensure the queen stays in the brood area and does not lay eggs in the honey storage.
6. **Brood chamber:** The brood chamber is where the queen lays her eggs and where young bees develop. It is the core of the colony, containing most of the bees and essential resources like pollen and nectar.

3 Literature study

7. **Bottom board:** Every hive is closed at the bottom with a bottom board. It usually has a ventilated mesh to allow air circulation and to capture fallen debris. A wide opening is used as the hive entrance, and this entrance can be adjusted in height and width.

Proper air circulation is crucial for the colony's health. In summer, bees ventilate the hive to prevent brood overheating, while in winter insulation and airflow management help prevent moisture buildup, which can be harmful to the colony (Danneels E., personal communication, October 14th 2024; Danneels, n.d.).

3.1.3 Bees outside the hive

Outside the beehive, bees engage in a wide range of activities essential for the colony's survival. This primarily involves foraging for food, consisting of nectar, pollen, water and propolis. The daily activities of bees outside the hive are strongly influenced by the time of day, weather conditions and the availability of flowers. Their flight behavior and foraging times play a crucial role in maintaining the colony.

Nectar collection is one of the bees' primary activities outside the hive. Nectar is a sugary liquid produced by flowers that bees convert into honey. Bees collect nectar in their honey stomach and bring it back to the hive, where it is processed into honey. Nectar serves as the primary energy source for bees. The sugar in nectar provides the fuel needed for their daily activities. A single bee can visit hundreds of flowers per day to collect nectar. Another one is pollen collection. Pollen is a source of protein and fats and is used to feed the brood. Worker bees collect pollen on their hind legs in specialized pollen baskets. They use their legs to scrape pollen from flowers and transport it back to the hive. Pollen is an essential part of bee nutrition, especially for developing larvae. The quality of pollen directly impacts the colony's overall health. Water collection is also important for the colony. Bees collect water to cool the hive and to dilute stored nectar or feed larvae. In warm weather they spread water throughout the hive and fan their wings to enhance evaporation, creating a cooling effect. Finally there is propolis collection. Propolis, a resinous substance gathered from plants, is used to repair the hive and seal openings. It has strong antimicrobial properties and is used by bees to improve hive hygiene through the sealing of the gaps and preventing external influences such as moisture and bacteria.

The flight activity of bees is influenced by the time of day, weather conditions and the bloom cycles of plants in their surroundings. In the morning bees typically begin their activities as soon as the sun rises. This timing varies depending on the season and temperature, but around sunrise, bees often leave the hive to forage. They visit flowers that release nectar early in the morning. The peak activity of bees usually occurs around noon when the temperature is optimal and many flowers are fully open. At this time of day, bees travel the furthest and nectar production in flowers is at its highest. As the day progresses and the sun begins to set, bee activity decreases. Bees usually return to the hive before sunset and stop flying when the temperature starts to drop.

Bee flight patterns change with the seasons. In spring and summer, when flowers are abundant, bee activity is at its highest. In autumn and winter, however, their activity declines significantly. In cold weather, typically below

3 Literature study

ten degrees Celcius, they do not leave the hive at all (Danneels E., personal communication, October 14th 2024).

3.2 *Varroa destructor*

The parasitic mite *Varroa destructor* (see Figure 3.3) is one of the most significant threats to honeybee colonies worldwide. The mite transmits harmful viruses and weakens bees by feeding on their hemolymph and fat bodies. It originated in Asia, where it co-evolved with the Eastern honeybee. There, infestations are controlled through natural grooming behaviors and resistance mechanisms. However after being introduced to European honeybee populations through global beekeeping practices, the mite has spread rapidly and has become a major cause of colony loss worldwide. They are small and oval-shaped with a reddish-brown exoskeleton. Adult females measure approximately 1.1 mm in length and 1.5 mm in width, making them visible to the naked eye. Males are smaller and pale in color. The mite has eight legs and a flattened body, allowing it to cling tightly to the host bee and move easily through the hive.

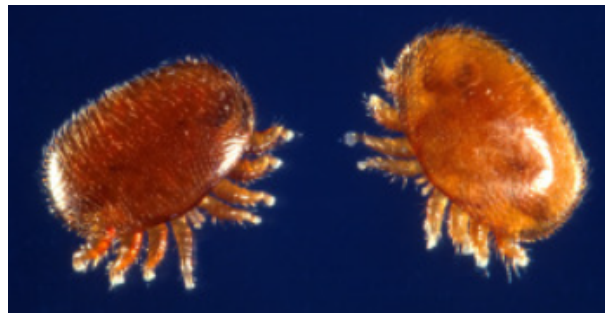


Figure 3.3: The *Varroa destructor* (Texas Apiary Inspection Service, n.d.)

The life cycle of the *Varroa destructor* is closely linked to honeybee brood development. Mites invade brood cells before capping, where they reproduce. The mite lays eggs, with the first developing into a male, followed by several female offspring. These offspring mate within the cell and mated females leave with the emerging bee to infest new brood cells. They have a preference for drone brood due to its longer development time, which allows for higher reproductive rates. The mite's ability to reproduce rapidly and infest new hosts makes it a persistent threat to honeybee colonies. Infestation by the *Varroa destructor* leads to direct and indirect damage to bees. Direct effects include weight loss, reduced lifespan and impaired immunity due to hemolymph depletion. Indirectly, the mite serves as a vector for various viral pathogens, including Deformed Wing Virus (DWV) and Acute Bee Paralysis Virus (ABPV). These infections exacerbate colony decline, particularly in winter when weakened bees struggle to survive. Severe infestations can result in colony collapse, especially in unmanaged hives (Honing van de Imker, n.d.).

3 Literature study

3.2.1 Impact of the *Varroa destructor* on the honeybee in Europe

The relationship between mite infestation levels and colony health is well-documented, with higher mite loads correlating to increased colony losses. A study conducted in Austria showed that when Varroa infestation levels exceeded 40% in adult bees, the probability of winter colony loss rose dramatically to over 80% (Morawetz et al., 2019). Similarly in France widespread Varroa infestations were linked to 40-50% colony mortality, with the mite identified as a major contributing factor. These findings highlight the severity of Varroa infestations and the urgent need for effective management strategies. A study (Oliver, 2006) has shown how mite populations can double every one to two weeks if not controlled, leading to rapidly escalating stress on the colony. Figure 3.4 depicts the devastating effect of unchecked Varroa infestations on colony strength and survival.

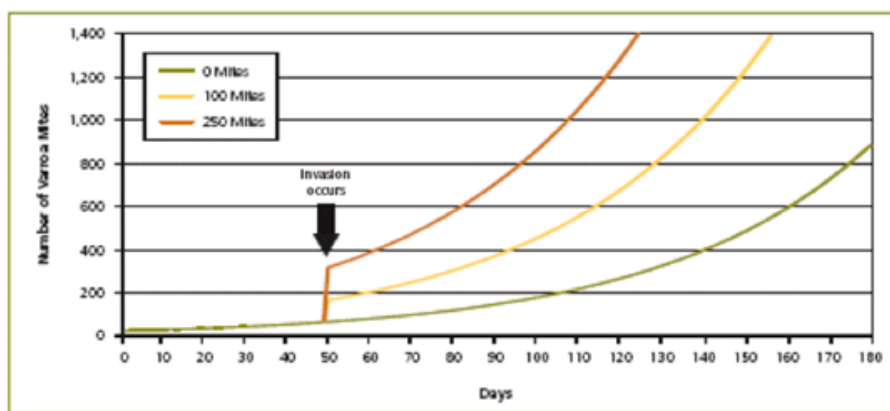


Figure 3.4: Effects of mite invasion over time (Oliver, 2006)

At the individual level Varroa mites parasitize both adult bees and developing larvae, consuming their hemolymph and bodily tissues. This feeding weakens the bees by depleting their energy reserves, disrupting normal development and impairing their immune system. Research found that the Varroa mite does more than simply feed on hemolymph. It actually consumes the fat body tissue of bees which plays a critical role in detoxification, immune function and energy storage. This discovery shows why Varroa infestations are so devastating, as bees with compromised fat bodies struggle to resist infections and environmental stressors. Infected bees often display abnormal behaviors, such as reduced grooming, impaired flight and disorientation (Ramsey et al., 2019). Another study demonstrates that bees parasitized by the Varroa mite have difficulty navigating back to their hive, resulting in increased rates of forager loss. Since foraging bees are responsible for collecting nectar and pollen, their premature loss places additional strain on the colony, accelerating its decline (Kralj & Fuchs, 2006).

Varroa infestations typically become problematic in the fall due to the shifting balance between bee and mite populations, as shown in Figure 3.5.

3 Literature study

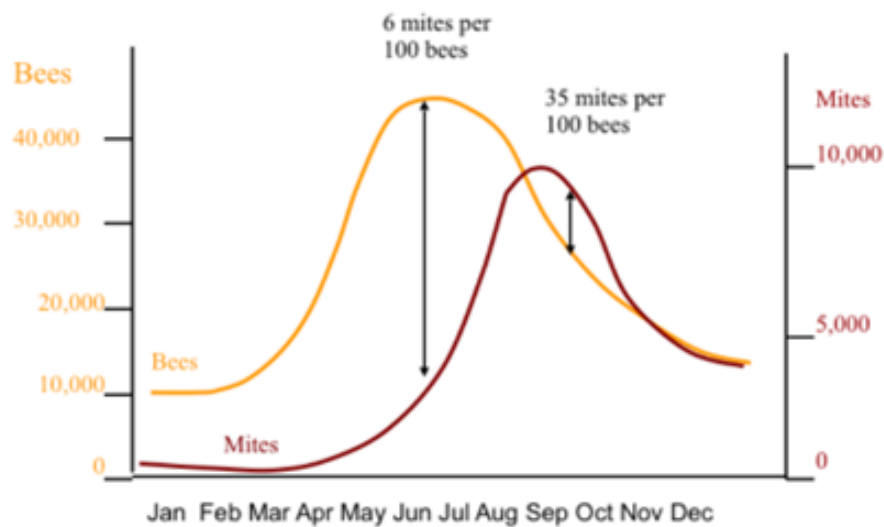


Figure 3.5: Bee and mite population growth curves for a temperate climate (Oliver, 2006)

In early spring, both populations are at their lowest, but while the bee population grows rapidly until midsummer, the mite population lags before surging during the drone-rearing season. By early September, the amount of mites climbs a lot while the amount of bees falls leading to stressed or dying brood, increased viral transmission and a weakened winter bee generation (Kralj & Fuchs, 2006). The relative infestation rate is more critical than absolute mite numbers: honey production drops above 2% in spring and winter survival is at risk above 5% in fall (Currie & Gatien, 2012). Unchecked, Varroa mites can multiply rapidly, increasing up to twelvefold in a short brood-rearing season (Martin, 1998) and even 100- to 300-fold with continuous brood rearing (Martin & Kemp, 1997).

This is why it is important to protect honeybees from these parasites, which can be done through several control methods.

3.2.2 Control methods

One of the most widely used methods is biotechnical control, specifically through drone brood removal. The Varroa mite has a strong preference for drone brood (male bee brood) over worker brood, as drone cells are larger and take three days longer to hatch, providing more space and time for the mites to reproduce. Beekeepers exploit this behavior by systematically removing drone brood, along with the mites inside, which helps slow down the exponential growth of the mite population. However, this method does not completely eliminate the mites and is used primarily as a population management strategy in both conventional and organic beekeeping (Bayer AG, 2019).

A more radical method involves the use of organic acids, such as oxalic acid and formic acid. These naturally occurring substances are applied to the hive at specific times to kill the mites while leaving the bees unharmed.

3 Literature study

Although this treatment is highly effective in reducing Varroa infestations, it is not without drawbacks. The treatment process is likely stressful for the bees, but it remains the most reliable method to ensure colony survival. Without intervention, the infestation would inevitably lead to colony collapse. Organic acid treatments are widely used in both conventional and organic beekeeping (Bayer AG, 2019).

Next, pharmaceutical treatments were once a common approach in the early days of Varroa control. However, resistance to these treatments developed quickly, reducing their effectiveness. Additionally, incorrect use of these products poses risks of residue contamination in honey and beeswax. While some pharmaceutical products are still legally permitted, they are rarely used by beekeepers in the Benelux region today (although illegal products are still used in abundance) (E. Danneels, personal communication, May 19th, 2025).

Furthermore, heat treatments are available. These target the Varroa mite's lower heat tolerance compared to bees. By exposing the hive to controlled high temperatures, mites can be killed without harming the bees. However, practical challenges make this method difficult to implement on a large scale (ThermoMite, n.d.). This system is displayed in Figure 3.6.

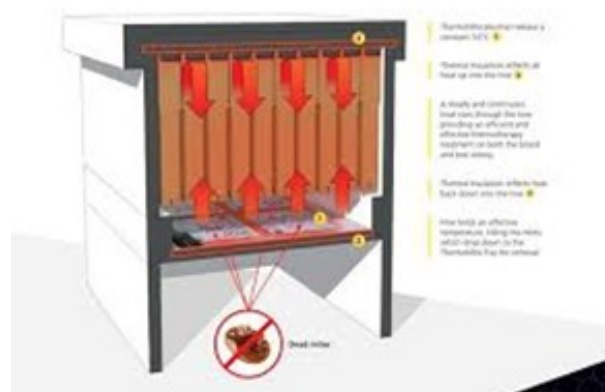


Figure 3.6: Thermotherapy bee treatment (ThermoMite, n.d.)

Smaller cell size is another proposed strategy. Beekeepers can influence cell size by providing bees with wax foundations of a specific dimension. Reducing the cell size from 5.4 mm to 4.9 mm decreases the space between the larva and the cell wall, theoretically limiting Varroa development. However, results from this method remain inconclusive (WildBnB, 2022).

An alternative method for controlling Varroa mites at the hive entrance involves the Varroa Gate technology. This system, developed through collaboration between Bayer researchers and bee research institutes, consists of a plastic strip with evenly spaced holes containing a varroacide. Varroacides are chemical or organic substances specifically developed to kill or control Varroa destructor mites in honeybee colonies. When installed at the hive entrance, the bees come into contact with the strip as they enter or exit, causing small amounts of the active substance to adhere to their bodies. Through natural social behaviors, such as grooming and contact, the substance is

3 Literature study

distributed throughout the colony. The strip is designed to maintain an effective level of the substance over time, offering ongoing protection against Varroa mites. Since it remains effective for several weeks, the Varroa Gate not only safeguards bees within the hive but also helps reduce the risk of reinfestation from external sources (Bayer AG, 2019).

While the Varroa gate appears to be an effective solution with minimal drawbacks, an even better approach exists. All of the previously mentioned methods rely on human intervention to manage the Varroa mite. The most promising and sustainable strategy, however, is enabling the bees to defend themselves. Bees capable of doing so are known as Varroa-resistant bees. Researchers and beekeepers worldwide are searching for Varroa-resistant bee strains. Some honeybee colonies have been observed surviving without treatment, similar to their Asian counterparts that naturally groom and remove mites. Identifying and breeding such bees could provide a long-term solution, that's why it's this solution the researchers at Honeybee Valley are interested in. Varroa-resistant bee strains show grooming behavior, which means they are able to remove mites from their own and other honeybees' bodies before the mites can reproduce. Some of these strains also have the ability to detect which brood cells are infested with Varroa mites, and can thus uncap these cells and remove the mite larvae. This is a behavioral trait called Varroa-sensitive hygiene (VSH) (Bayer AG, 2019). And finally some resistant bee strains build smaller brood cells to limit the ability of Varroa mites to reproduce successfully. The test that helps to identify these bee strains is called the arenatest. It must be noted that this test does not give a definitive conclusion about possible Varroa-resistant behavior, but only about avoidance behavior. Other aspects of Varroa resistance, such as reproducing less, not reproducing or the uncapping of cells, are not tested in the arenatest (Danneels E., personal communication, October 14th 2024).

3.2.3 Arenatest

The arenatest provided a controlled environment to observe the interactions between the *Varroa destructor* and honeybees. This experiment is essential for understanding bee and mite movement patterns, host selection and response to environmental stimuli. The effectiveness of the arenatest depends on multiple factors, including the choice of substrate, the dimensions of the test area and the conditions under which mites and bees are observed. Research has shown that certain surfaces influence mite activity, with standard plastic Petri dishes often causing distress and increased mortality among mites. In contrast, gelatine-coated Petri dishes create a more suitable environment, reducing stress and improving the reliability of behavioral assessments (LeDoux et al., 2000).

Grooming behavior in honeybees is a key natural defense mechanism against the *Varroa destructor*. The arenatest has been used to compare the grooming responses of different bee populations. Research has shown that Africanized honeybees exhibit more frequent and intense grooming compared to Carnolian bees. However, despite the increased grooming activity, the effectiveness of mite removal remains limited as mites are often dislodged but not killed. This finding suggests that while grooming is an important defense mechanism, it is not sufficient as a standalone strategy for controlling mite populations (Aumeier, 2001).

3 Literature study

One of the objectives of the arenatest is to analyze the orientation behavior of the *Varroa destructor*. The ability of mites to locate and attach to a suitable host is a crucial factor in their reproductive success. Studies have demonstrated that mites exhibit specific directional preferences when exposed to different chemical and physical cues. A bioassay can be developed to assess the mite's response to host odors and movement, providing valuable insights into host recognition mechanisms (LeDoux et al., 2000). Such studies are vital for designing targeted control strategies that interfere with mite orientation and reduce infestation rates.

In addition to a laboratory-based arenatest, field bioassays are employed to assess the resistance of *Varroa destructor* to acaricides. These bioassays involve exposing mites to treated bees or surfaces and monitoring their survival and reproductive success. A practical field bioassay was developed to detect resistance within apiaries, allowing beekeepers to make informed decisions about treatment protocols (Veto-pharma, 2023). The widespread use of acaricides has led to the development of resistant mite populations, making routine resistance testing an essential component of integrated pest management strategies.

In conclusion, arena tests play a pivotal role in advancing our understanding of *Varroa destructor* behavior. Their application in both laboratory and field settings continues to contribute to the development of innovative control measures. As resistance to chemical treatments increases, integrating behavioral insights from arenatests into broader pest management approaches will be essential for sustaining healthy honeybee colonies.

3.3 Knocker test

In this section, the paper about the knocker test (Bencsik et al., 2024) will be discussed. The goal of this paper is to assess the physiological status of bees by applying a shock to the hive. This shock is generated by an electromagnetic shaker attached to the outside of the cabinet. The shaker delivers an 0.1-second signal at a frequency of 340 Hz. The signal is provided hourly, but not always at the exact same time to prevent the bees from becoming accustomed to it.

The experiment is first conducted on a single colony, and later expanded to eight colonies. The study shows that bees become immobilized when a short-band vibrational waveform is applied near the hive. This effect can last up to four hours, and stops immediately after the signal is removed (Frings & Little, 1957). A long-lived broadband signal similarly causes immobilization in bees (Stefanec et al., 2021). Additionally, a short knock on the hive also immobilizes the bees for a brief period. Immediately after this immobilization, the bees display an increase in vibration. This is the buzzing noise beekeepers hear when they knock on the hive.

This knocker test is described because of its similarities to the test with a sound stimulus, which is conducted in this thesis. To gain a better understanding of the results of the test with a sound stimulus, the results of the knocker test are shown in this section. The difference between both tests is that the accelerometer is replaced by a microphone just outside the beehive, and the stimulus is a sound instead of a knock.

3 Literature study

3.3.1 Experiment with one hive

First, the results of the pilot study with one beehive will be discussed. To provide the signal, an electromagnetic shaker with 1 W power is used. This shaker is placed in a weatherproof box, which is then attached to the cabinet. To record the reaction of the bees, an accelerometer is placed in the middle of a honeycomb. The accelerometer is positioned as close as possible to the center of the brood.

When applying a short 'beep' waveform (a short pulsed monochromatic waveform) with a frequency of 300 Hz, the bees begin to react. Initially, the bees go quiet for a very short time. After this, they start vibrating more. These vibrations are also known as whooping signals. The signal is presented to the bees approximately every hour, during daytime, using the shaker over the course of a few days. The bees' reactions are recorded with the accelerometer. The recordings show consistent results: the noise produced by the bees decreases by about 20% immediately after the knock signal. Following this decrease, there is an increase in vibrations within the hive. This increase lasts much longer than the decrease, but is less consistent.

The decrease in the signal is examined in more detail using a different setup. The cabinet features a transparent stage that allows cameras to film the honeycomb with the accelerometer. Various signals, ranging from 0 Hz to 2000 Hz in steps of 50 Hz, are presented to the bees. These signals are applied every 2.2 seconds using the electromagnetic shaker. To quantify the amount of immobilization, the mean difference between two consecutive frames is calculated. This data is then combined with the accelerometer data.

The results show that the maximum decrease in vibration occurs at 500 Hz. After this brief decrease, the bees start moving again. The time constant for the curve at 500 Hz is 1 second. For other frequencies, the signal-to-noise ratio is too small to obtain a reliable time constant.

3 Literature study

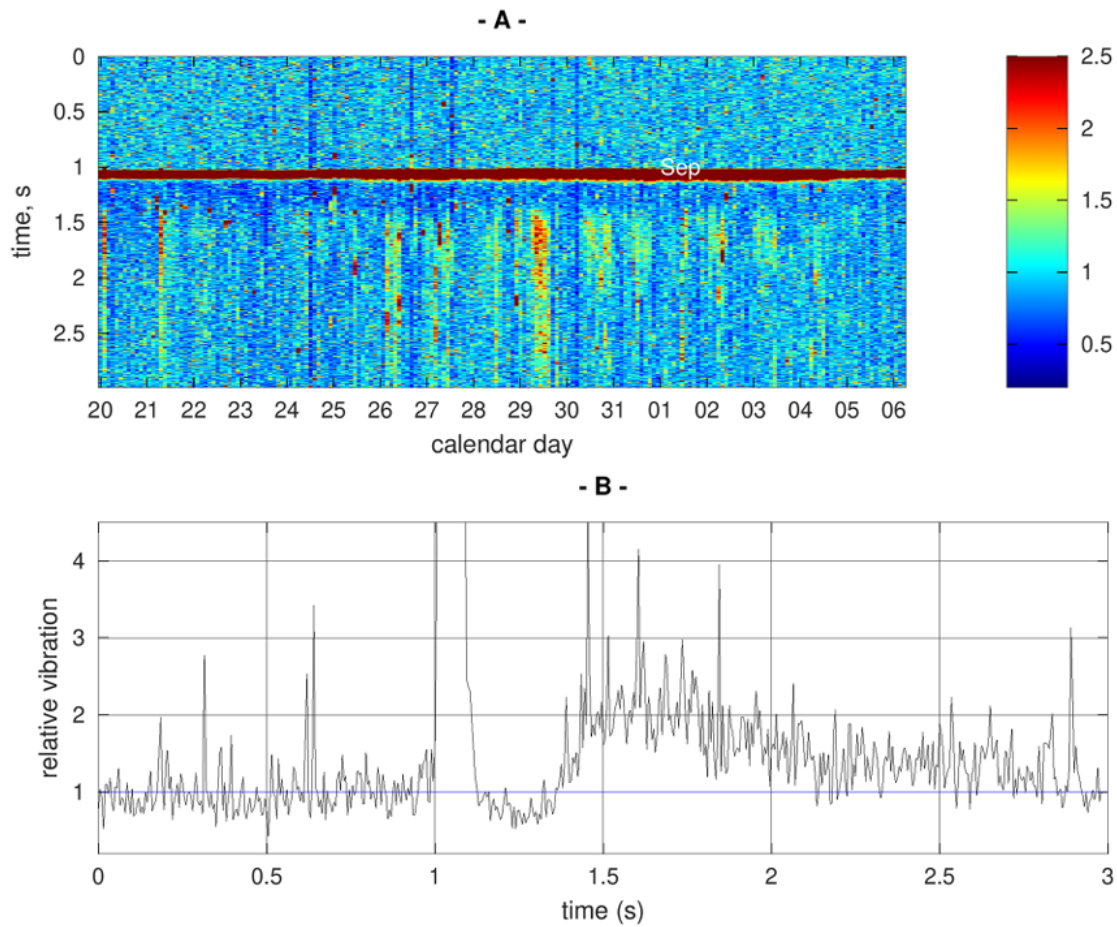


Figure 3.7: Result of the knock on the beehive. (A) A collection of three-second-long accelerometer recordings, with the time of the recording on the horizontal axis, and the time relative to the artificial stimulus on the vertical axis. (B) A plot of a particular recording taking place on the 29th Aug 2021 with a remarkably strong colony response, which is also made . Taken from Bencsik et al. (2024)

Figure 3.7 shows the beep at 1 second and the subsequent decrease in the signal (Bencsik et al., 2024). After this short decrease, the signal starts to increase rapidly. The signal reaches an amplitude higher than the original amplitude, and this buzzing is audible to the listener. This is the sound beekeepers hear when they knock on their beehive.

The increase in the signal is then analyzed. The experiments described in this paper ran from 20th August 2021 to 10th December 2022, resulting in 5,595 recordings of the stimuli. This data is analyzed using principal component analysis, which allows for the distinction between components that exhibit whooping signals and those that do not. Whooping signals are produced by individual bees and are not of interest for this experiment.

The principal component analysis reveals that components one, two and three do not display whooping signals,

3 Literature study

while all higher components do. By focusing solely on principal components one and two, only signals originating from the entire colony are considered.

3.3.2 Difference of signal through the year

First of all, the signal that the bees receive is not always the same throughout the year. This has a simple explanation: the size of the colony changes substantially over the course of the year. In the summer, the colony is much larger than in the winter. This variation affects how the signal propagates through the hive and reaches the accelerometer. When the colony is at its smallest, the signal that reaches the accelerometer has the highest amplitude. In contrast, during the summer, when the colony is at its largest, the amplitude can be significantly smaller.

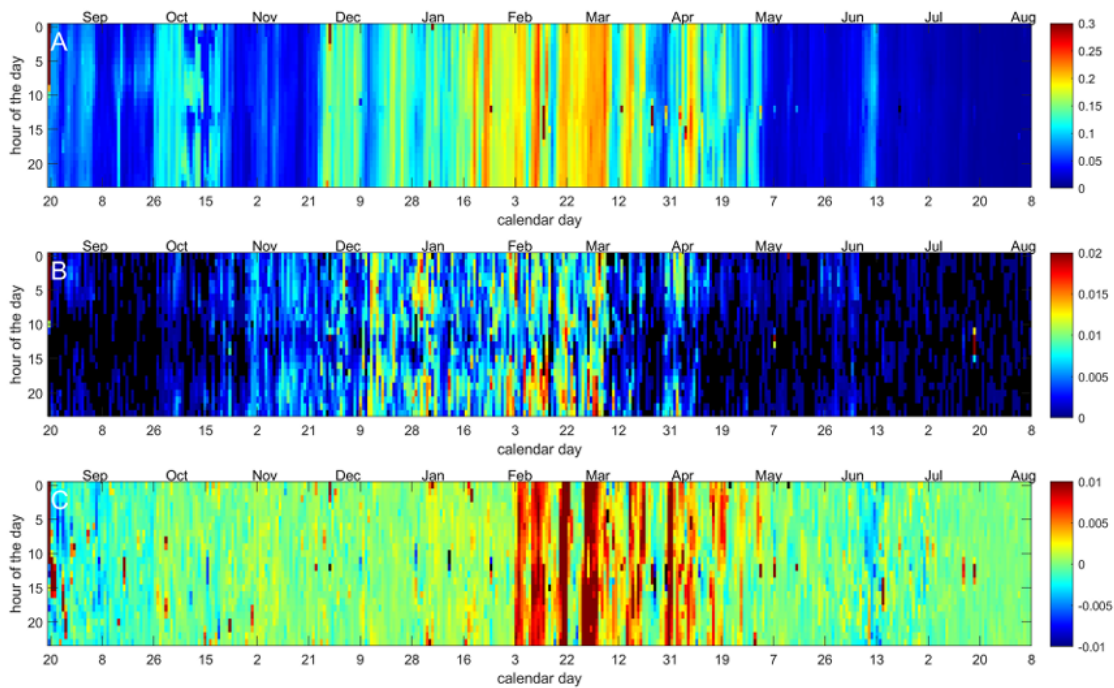


Figure 3.8: A: Magnitude of the signal that reaches the accelerometer. B: Magnitude of the positive response from the bees. C: Magnitude of the negative response from the bees. Taken from Bencsik et al. (2024)

Figure 3.8 (Bencsik et al., 2024) shows the magnitude of the response from the bees throughout the year. This shows that bees show a larger response to the stimulus in the winter months than throughout the rest of the year.

3 Literature study

3.3.3 Multiple colonies

In the first case with one beehive, it seems that the reaction of the bees is stronger in the winter months and much harder to measure in the summer months. The signal is also weaker in the middle of the day. The same experiment is now repeated, but with eight beehives. Again, a signal from the shaker is given approximately every hour at a randomized time. One of the eight colonies lost its queen during the summer. After this, the colony continued to deteriorate. The colony did not recover and was eventually replaced with a new one. The results show that this colony (colony 3), which lost its queen, exhibited a stronger response to the stimulus than all the other colonies. Even though the results of the received signal magnitudes from the accelerometer indicate that colony 3 did not receive higher amplitudes. These results are shown in Figure 3.9 (Bencsik et al., 2024).

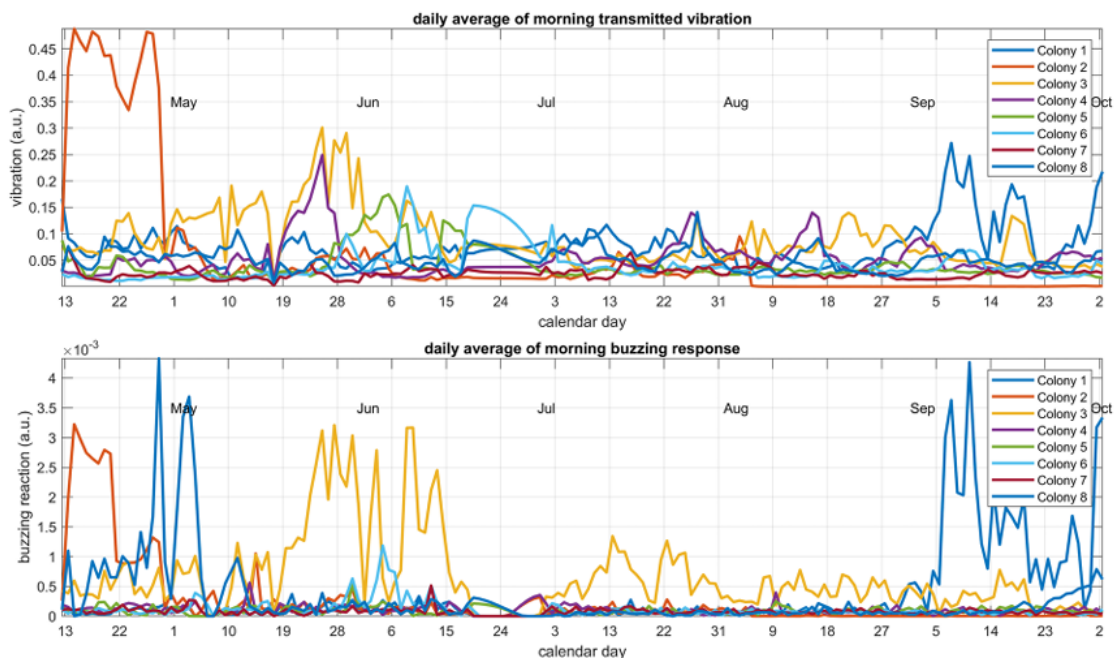


Figure 3.9: Top: average of transmitted vibration. Bottom: average of vibrational response. Taken from Bencsik et al. (2024)

3.3.4 Conclusion of the paper

Applying a stimulus every hour at randomized times, results in a long-lasting increase in the vibrations picked up by the accelerometer. This increase also causes whooping sounds from a few individuals. The long-lived buzzing response is almost absent in healthy colonies during the active season. Only in the morning, a response from the bees is measured. When winter begins, the response increases and decreases again when summer begins. The positive response from the bees is of the same magnitude in winter. The paper proposes that the strength of the signal depends on the restfulness of the bees. When bees are calm, like in winter or early morning, their response

3 Literature study

seems to be the strongest. During the active season, the bees are busy until late at night. Only one of the eight colonies lost its queen and experienced deterioration in health. This is also the only colony that showed a strong response to the shaker during the summer months. This may indicate that the magnitude of the bee response during the active season could be used as an indicator of the colony's health. However, this cannot be generalized because the sample size is too small.

The decrease in the signal immediately after the knock is applied to the beehive, is very repeatable. This can be done multiple times in one minute, and does not seem to fade due to habituation. This decrease appears to be independent of the longer increase following immobilization. The paper suggests that the higher the mobility of the bee population, the higher the magnitude of the vibrations before the stimulus, and also the greater the decrease in the signal after the stimulus. The drop in the signal is small and cannot be physically heard. It can be measured by an accelerometer because bees vibrate much more than they make actual noise. This suggests that the drop probably cannot be measured with microphones. The increase, on the other hand, might be recorded just as well with a microphone if it is placed in the same area of the hive. Recordings from a microphone are also less dependent on changes in the density of the hive, which might result in a more consistent signal.

3.3.5 Hearing bees

To see if the test with a sound stimulus is useful, it is necessary to look a little closer at the hearing capabilities of honeybees. Bees were long thought to be deaf to airborne sounds (Connor, 2017). However, bees have been shown to detect airborne sounds by detecting air-particle movements (Kirchner, 1989). The hearing of bees seems to be sensitive enough to detect the vibrations of air particles. This means that they can, in fact, detect the sound stimulus, so they might react to it as well. Another experiment also says that bees show a reaction to a pure tone sound stimulus (Micinski et al., 1972). There was a reaction measured for tones between 100 Hz and 6000 Hz. This information shows that the test with the sound stimulus might trigger a response in bees as well.

3.4 Infrared imaging

Infrared imaging can be used to assess the population size of a bee colony in a non-invasive manner (Brogioli et al., 2011). It forms a way to rapidly get an image of the bee population without opening the beehive, which would disturb the bees. The method presented (Brogioli et al., 2011) is different from the strategy used in this paper. There, the thermal camera is set up 3 to 5 meters from the beehive and films from the side. Here, the thermal camera is positioned underneath the beehive at a distance of 35 cm. In this way, it is possible to obtain information about the separate honeycombs. The camera used here is also not strong enough to capture the heat

through the wood of the beehive.

3.5 Pollen counting

3.5.1 Purpose

Monitoring pollen collection by honeybees is a valuable tool in both scientific research and practical beekeeping. Pollen is a primary source of protein for the hive and a key indicator of foraging activity, floral availability and overall colony health. Variations in pollen intake can signal environmental stressors, habitat loss, pesticide exposure, or disease. For researchers, tracking pollen flow provides insights into pollinator-plant interactions and ecosystem dynamics. For beekeepers, it serves as a non-invasive way to assess colony strength, productivity and food reserves and to make informed management decisions (Leponiemi et al., 2023; McKinnon et al., 2023; Woodcock et al., 2022).

Regularly monitoring pollen intake can help: identify floral resource availability throughout the foraging season, predict colony development and brood rearing (which are tightly linked to protein availability), gauge environmental health (including pesticide exposure and habitat quality) and improve crop pollination strategies by identifying which plants are being visited and when (Leponiemi et al., 2023; McKinnon et al., 2023; Woodcock et al., 2022).

3.5.2 Traditional methods of pollen monitoring

Traditionally, beekeepers have employed mechanical pollen traps to collect and analyze pollen brought back by foraging bees. These traps are typically installed at the hive entrance and function by forcing returning foragers to pass through a mesh or screen that dislodges a portion of the pollen pellets from their hind legs. The dislodged pollen falls into a collection tray beneath the trap, allowing beekeepers to harvest it for various purposes, such as supplementing bee diets or selling it as a health product. Such a trap is shown in Figure 3.10 (McKinnon et al., 2023).



Figure 3.10: Front porch type traditional pollen trap (Amazon, n.d.)

There are several designs of pollen traps, including bottom-mounted, top-mounted and front porch styles. Each design has its own advantages and considerations. For instance, bottom-mounted traps are integrated into the hive's bottom board while top-mounted traps are placed above the brood boxes. Front porch traps attach to the hive entrance and are often easier to install and monitor (Betterbee, n.d. Company, n.d. Rusty, n.d.).

While effective, traditional pollen trapping methods have certain drawbacks:

- **Invasiveness:** Pollen traps can stress colonies by reducing the amount of pollen entering the hive, potentially impacting brood rearing (Smith, 2022).
- **Labor-Intensive:** Regular maintenance is required to prevent issues such as mold growth in collected pollen, especially in humid conditions (Smith, 2022).
- **Timing Sensitivity:** It's crucial to install and remove pollen traps at appropriate times to avoid disrupting the colony's natural cycles. For example, traps should not be used during peak brood-rearing periods or swarm season, as they can hinder the queen's movement and overall colony dynamics (Betterbee, n.d.).

Despite these challenges, traditional pollen traps remain a valuable tool for beekeepers seeking to monitor and collect pollen, provided they are used judiciously and with proper management.

3.5.3 Computer vision for pollen detection

Recent advancements in computer vision and deep learning have significantly enhanced the ability to monitor pollen-carrying honeybees in a non-invasive manner. By installing cameras at hive entrances and employing image or video analysis, researchers can detect individual bees, classify them based on pollen presence, count returning foragers and analyze temporal trends in pollen flow.

Several studies have demonstrated the efficacy of combining computer vision with deep learning techniques

3 Literature study

for this purpose. For instance, in a study published in *Computers and Electronics in Agriculture*, an automated system for monitoring honeybee pollen foraging behavior was introduced using a deep learning-based imaging approach. The system employs an off-the-shelf camera installed at the hive entrance to capture video streams, which are processed by a lightweight real-time object detection and classification model supported by an object tracking algorithm. This model distinguishes between pollen-bearing and non-pollen-bearing honeybees with an F1-score of 0.94, achieving precision and recall values of 0.91 and 0.99, respectively. The system also integrates environmental sensors to assess the impact of weather conditions on foraging activity. Experimental results demonstrated that daily pollen foraging trip ratios averaged 24.5%, with a single hive collecting approximately 49.1 grams of pollen per day. The study found that pollen foraging activity increased with higher temperatures and light intensity, while it decreased with rising relative humidity, rainfall and wind speed. This research highlights the potential of combining embedded imaging systems with deep learning techniques to provide continuous, non-invasive monitoring of honeybee foraging behavior, offering valuable insights for both ecological research and practical beekeeping applications (Ngo et al., 2021).

Another study introduced an innovative system for the automated video monitoring of honeybee foraging activity at the hive entrance. This system utilizes convolutional neural networks to perform multi-animal pose estimation, enabling precise detection of honeybee body parts without the need for physical markers. Such detailed detection facilitates accurate identification of entrance and exit events, as well as the presence of pollen loads on returning foragers. Additionally, the system incorporates the capability to identify bees marked with barcodes, allowing for both aggregate and individual-level monitoring. The authors conducted extensive evaluations of the system's detection quality and parameter effects, demonstrating its applicability for large-scale deployment (Rodriguez et al., 2022).

Research explores the integration of artificial intelligence and computer vision technologies into beekeeping practices. By leveraging models like YOLO (You Only Look Once), the article discusses how these advancements can automate hive monitoring, enhance disease detection and optimize pollination tracking, thereby supporting healthier bee colonies and improving honey production. Among others it lists the following challenges in modern beekeeping: tracking bee populations, monitoring foraging behavior and identifying pollen sources. Key advantages of the use of computer vision in beekeeping are listed as scalability, non-invasive monitoring and improved pollination tracking. Although this integration also comes with limitations in the form of environmental constraints, data collection challenges and integration costs (Abdelrahman, 2025).

However, it is important to note that these methodologies mostly rely on the integration of machine learning with computer vision. There is a lack of documented instances where traditional computer vision techniques, devoid of machine learning components, have been solely utilized for detecting and counting pollen-carrying bees. This suggests that the complexity of accurately identifying pollen presence on bees necessitates the advanced pattern recognition capabilities afforded by machine learning algorithms.

In summary, while the fusion of computer vision and deep learning has proven effective in monitoring pollen-

3 Literature study

carrying bees, the exclusive use of traditional computer vision techniques for this task remains unexplored in the current literature.

3.5.4 Impact on bees and hives

One of the significant advantages of computer vision-based pollen detection is its non-intrusive nature. Unlike traditional pollen traps, it does not disturb foraging bees, allowing them to enter the hive unimpeded. It also enables continuous, high-frequency data collection, which is ideal for studying daily and seasonal patterns. Finally, it is scalable and requires minimal maintenance once deployed (Abdelrahman, 2025).

However, it's worth noting that continuous video monitoring may require significant data storage (depending on the strategy) and processing resources. Cameras must also be strategically placed to ensure a clear view of bees and minimize occlusions. Environmental factors like lighting, rain and temperature can also affect image quality and thus classification accuracy (Ngo et al., 2021).

Despite these challenges, most implementations of vision-based systems have shown no measurable negative effects on bee behavior or hive dynamics, provided the system is properly installed and unobtrusive.

3.5.5 Conclusion

Computer vision has transformed the way researchers and beekeepers monitor pollen collection in honeybee colonies. While traditional methods provide reliable, direct sampling, they are labor-intensive and disruptive. Vision-based techniques offer a scalable, automated and non-invasive alternative, capable of delivering high-resolution data in real-time. These advances open up new possibilities for large-scale ecological studies, precision apiculture and long-term monitoring of pollinator health.

4

Arenatest

The process of performing the arenatest was divided into a couple of clear steps. A Python-based simulation of the test was initially developed, in which bees and mites moved in a pseudo-random manner. This simulation allowed the creation of a first version of the computer vision detection system and the statistical data extraction program. This was useful due to it being winter when the simulation was developed, and working with live bees was subsequently not possible. Next the setup for the real test was put together and first tests with dead bees and mites were performed, so the first version of the Python program could be adapted to real bees and mites and the test conditions. Finally the real arenatest was performed and statistics were extracted. A goal for this test is that the detection could be done using only computer vision algorithms and to avoid the use of artificial intelligence. This is because the strain of running a machine learning model on a Raspberry Pi is much larger than running the computer vision algorithm, and this way computational power is saved.

All data, code and results for this chapter can be found in Appendix A¹.

4.1 Simulation

4.1.1 Generating the simulation

The simulation is made through Python code in the Google Colab cloud environment. It is developed to be a visual simulation of an arenatest, in which the behavior of honeybees and Varroa mites is simulated within a confined environment. Bees and mites are visualized as images moving over a background.

When the script is ran, the necessary libraries are installed and imported, as well as the images that are used for the bees, mites and background. The environment in which the simulation takes place is defined as a rectangular arena with boundaries in both the x and y directions. This differs from the eventual circular shape of the real arena because, at this point in the process, the ideal shape of the real arena was not yet determined. Since this simulation is primarily used to be able to create a first version of the bee and mite detection program and the

¹GitHub repository for arenatest, pollen monitoring and website data

4 Arenatest

statistics extraction, this has no effect on its goal. The number of bees and Varroa mites is predefined at ten and three. This again differs from the eventual size of the test, due to the parameters of the test being changed after the size of the physical arena differed from the original proposed size. However this doesn't impact the purpose of the simulation either as this is just a scaling. The mites are randomly placed in the center of the arena, corresponding to the setup of a typical arenatest where mites are placed in the middle. The bees are positioned around the outer edge of the arena at a sufficient distance from the center, and a radius is defined which bees are not allowed to spawn within to realistically mimic the initial setup of the real experiment. Their spawn zones are shown in Figure 4.1.

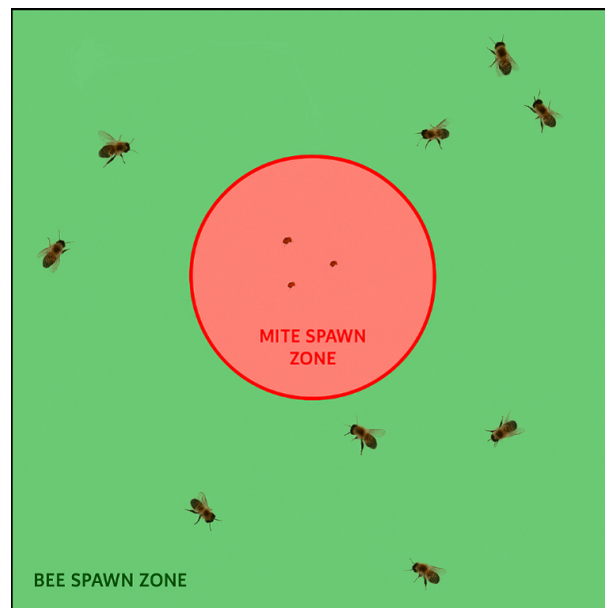


Figure 4.1: Spawn zones for bees and mites in the arenatest simulation

The movement of the bees and mites was implemented in different ways. Initially random number generation was used for both, via the "Random" module from Python. A small random change in x and y coordinate was generated for each frame and the images that represented the bees and mites, were offset by that amount, with the bees having a greater random range than the mites allowing them to move more and faster. This obviously resulted in non-realistic movement pattern for the bees in particular, because their random range was greater the randomness in their movement was more apparent. For the mites, this effect was not that apparent, so this random motion approach was kept. For the bees, however, another approach based on pre-determined trajectories was adopted. Before the creation of the simulation begins, the trajectories are generated for each bee. A trajectory consists of a series of random waypoints within the arena boundaries. The bees move step by step toward their next waypoint. While moving, they are visually rotated to face the direction they are heading, which is important to keep the animation realistic. This is done by calculating the angle between their current orientation and their movement vector, and then rotating the bee image accordingly. A reference vector is defined as the direction that

4 Arenatest

means facing forward for the bee, in this case [1 0] or pointing directly right but this depends on the chosen image of the bee. Then the movement vector is defined as the direction from their current position to the next point on their trajectory. Finally, the angle between these two vectors is calculated, and the image of the bee is rotated by the calculated angle. The movement of both bees and mites is updated in each new frame and rendered on top of a white background image of the arena. This background is chosen to have a stark contrast between the bees and mites and the background, to make it easier for the computer vision techniques to recognize them.

The simulation uses Matplotlib's animation functions to generate each frame individually. Each frame is saved as an image, and these images are later stitched together into an MP4 video file using ffmpeg. A smooth visualization of the entire experiment is now created, which can now be used to create a detection and behavioral analysis algorithm. A simulation that was created this way can be found in Appendix A ².

4.1.2 Simulation analysis and classification

For the bee and mite classification algorithm, only traditional computer vision techniques, such as edge detection, thresholding and contour analysis, were used and no machine learning models. This is firstly because these traditional techniques are lightweight and computationally efficient, making them well-suited for real-time processing on the limited hardware resources of the Raspberry Pi. Unlike machine learning models, especially deep learning architectures, which require large labeled datasets and significant computational power for training and inference. Traditional approaches can be implemented and fine-tuned without extensive data or hardware. Additionally, the visual difference between bees and mites in the arenatest, with clear size and shape contrasts, can be exploited effectively by the traditional techniques.

The classification process again begins with the installation of the necessary libraries and packages to handle image processing, visualizations and calculations, as well as the generated video of the simulated arenatest is uploaded. The used paths for the input and output of each part of the code are provided at the beginning of each block. The first block of the script is the extraction of the simulation video frames into a directory within the Google Colab environment. This is done using OpenCV's VideoCapture function, the video is read frame-by-frame and each frame is saved as a PNG image ready for further processing.

The next part of the script handles the detection and classification of the bees and mites, and extracts the statistics used for analyzing their behavior. A for-loop is used to analyze all the extracted frames one by one. A frame is first loaded in grayscale since techniques such as thresholding require this, a color (in Blue Green Red, or BGR) copy is made to draw the detected bounding boxes on after the processing is done. The first step in frame processing is the use of adaptive Gaussian thresholding. Adaptive thresholding is a technique used to convert a grayscale image into a binary image by determining the threshold value for each pixel based on the local neighborhood. Instead of using one fixed value, adaptive thresholding computes the threshold for a pixel by averaging the intensities of the

²GitHub repository for arenatest, pollen monitoring and website data

4 Arenatest

surrounding pixels within a defined window and subtracting a constant. This allows it to preserve details in darker or lighter regions and effectively segment objects from the background even under non-uniform illumination. This is particularly useful because of the white background. In the result, shown in Figure 4.2, the outlines of the bees and mites become visible instantly. A next step could be cleaning up this thresholding image to get rid of white “spots” that are the result of e.g. noise, motion blur caused by bees moving rapidly, shadows, ... This is however not necessary in the case of this simulation, since the images are noise free and the thresholding thus near-perfect. For the real test this step will be important and crucial to ensure correct classification of bees and mites.

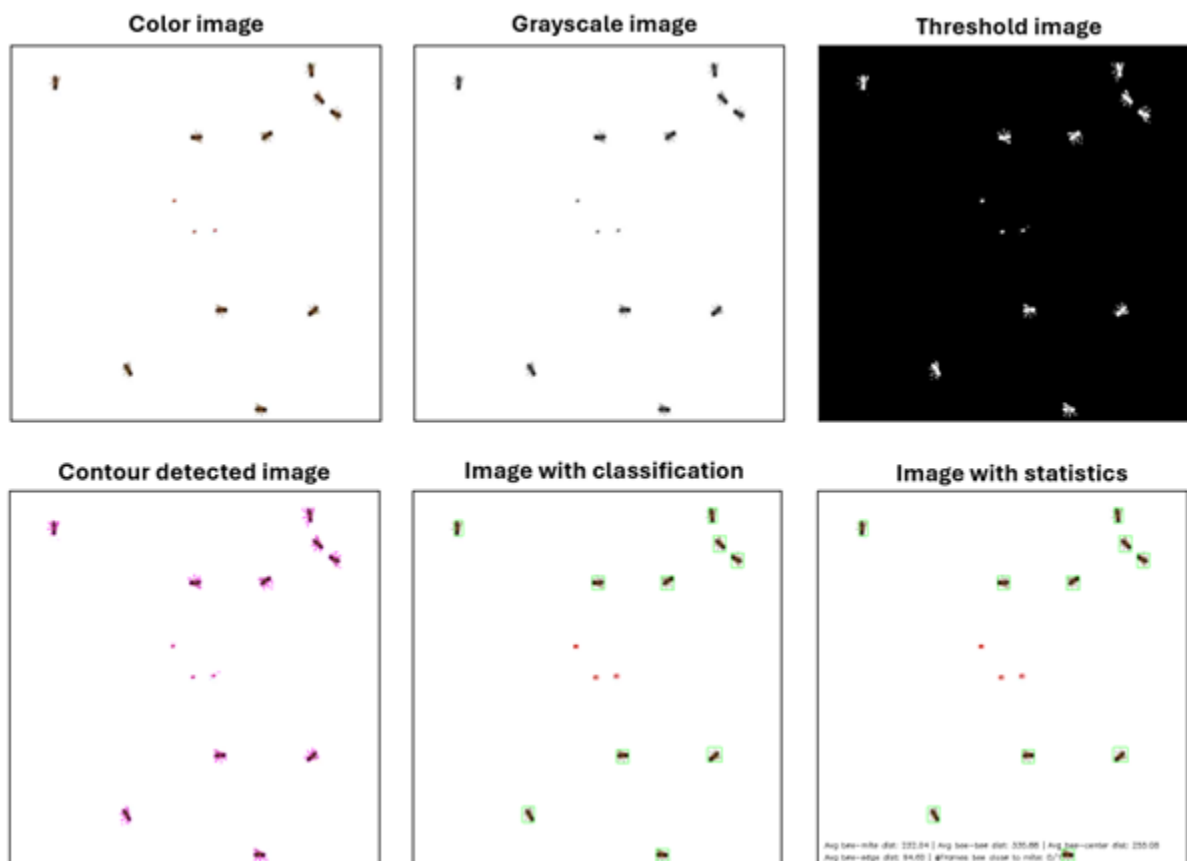


Figure 4.2: Simulation bee and mite detection

After this, contour detection is done using OpenCV's "findContours". This function detects the contours, or the outlines of shapes, in a binary image, the threshold image in this case. This way the outlines of the bees and mites can be found and returned as a list of coordinates. OpenCV also defines functions that allows the computation of values like the inner area and perimeter from the contours, and allows bounding boxes or minimum enclosing circles to be drawn around them. Using a for-loop, the contours are processed and the bees and mites can be classified based on their size difference. Three classes are defined for the contours: "bee", "mite" and "overlapping bees". This last class is necessary because if the bees are too close together, they will be seen as one bee in the

4 Arenatest

thresholding and will thus form one contour. The implication of this for the calculated metrics will be addressed later in 4.3.2: Statistic and metric extraction.

The classification is done using a value threshold on the contours' inner area, since smaller objects such as mites will logically have a smaller contour and thus a lower contour inner area and bees will have a larger inner area. These values are in pixels squared and were obtained by first creating a video that displays the inner area's next to the contours to get an idea of what good threshold candidates could be, and then choosing values and tuning them to get a good classification. The final values are displayed in Table 4.1.

Table 4.1: Area thresholds used for bee and mite classification of contours in the arenatest simulation

Class	Lower threshold	Upper threshold
Mite	13	27
Bee	80	450
Overlapping bees	450	/

The lower threshold for the mite class is important, as to filter out possible noise or small contours detected on or near bees (as a consequence of small blobs on bees, or thin bee legs) and not detect mites where there are none. False positives form an issue in the statistic extraction, because this influences the “Amount of frames close with bees close to mites” metric which gives important insights into the behavior of the bees in the arenatest. An upper threshold is not necessary here since the simulation is free of noise and shadows. This means all the detected contours are bees or mites. An upper, as well as a lower, threshold will be important for the real arenatest, to reduce the amount of false positives due to the effects of e.g. noise in classification. For each class a bounding box is drawn on the color image, with a fitting color (green for “bee”, red for “mite” and blue for “overlapping bees”) and label. Furthermore, the center coordinates of the contour are obtained and added to a separate list, the coordinates of a mite contour to a “mite position” list and the coordinates of the bee and overlapping bees contours both to a “bee position” list.

Using these lists of bee- and mite positions, different metrics are calculated. These include the distance to the arena center, to the nearest edge of the frame and to the closest mite. These distances and a heatmap of bee positions help analyze spatial behavior. A proximity threshold is used to determine whether any bee is near a mite; if so, a counter tracking such frames is incremented. This quantitative information is overlaid on each frame using text annotations for easy interpretation. Since these metrics were expanded upon for the real arenatest analysis, these will be addressed later (see 4.3.3: Analysis of extracted metrics). Finally the frame is saved in a directory of analyzed frames. The first processed frame is displayed for verification using Matplotlib. The frames are compiled into a video using OpenCV's VideoWriter. The output video preserves the resolution and frame rate of the original video, displaying all overlays and annotations frame-by-frame.

4.1.3 Results

The simulation proved to be a useful asset to get a basic outline of an arenatest classification and analysis script. Enough realism was achieved to get a good idea of the challenges and issues for this computer vision problem. The classification accuracy was very high, with bees and mites being correctly recognized in nearly all cases. However, it is important to note that this performance was achieved under ideal conditions. The most challenging differences between the simulation and the real test are the image quality and the noise in the images as a result of e.g. lighting conditions.

4.2 Test setup

To perform the arenatest and monitor it, a setup was designed. In this chapter the used equipment and the test conditions and parameters will be listed and discussed.

4.2.1 Equipment

The equipment used to monitor the arenatest is listed up in Table 4.2 below. The numbering used in this Table is consistent with the numbering in Figure 4.3, which gives a visual idea of the setup through a schematic drawing. Data sheets for the Raspberry Pi 5 and the Raspberry Pi camera module V2.1 can be found in Appendix D³.

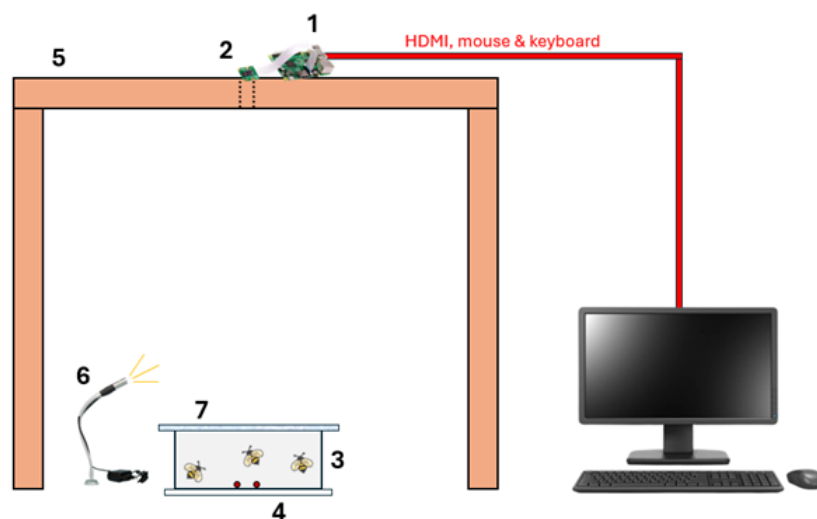


Figure 4.3: Arenatest setup diagram

³GitHub repository for arenatest, pollen monitoring and website data

4 Arenatest

Table 4.2: Equipment used in the arenatest setup

Nr	Equipment name	Function description	Main characteristics
1	Raspberry Pi 5	The core of the setup. Handles data acquisition and video processing.	5 V USB-C power supply 4 GB RAM 16 GB microSD Two 4-lane MIPI connectors Dimensions: 85.6 mm x 56.5 mm
2	Raspberry Pi Camera Module v2.1	Used to capture the experiment. Connected to the Raspberry Pi through the Camera Serial Interface (CSI) port.	Sony IMX219 sensor Up to 1080p/30 fps Resolution 8MP (3280 × 2464) Fixed focus lens Objects are sharp from about 0.5 m.
3	Glass dish	Used as the arena.	Made of glass Ø19 cm 8 cm high Round shape
4	White paper	Used to provide a white background for the captured images.	A4 paper
5	Camera mount	Used to be able to capture images from above the arena, and to provide a platform for the Raspberry Pi.	Made of cardboard 52 cm high
6	Light source	Used to provide even lighting in the captured images.	LED light 1 W 110-230 V
7	Plexi glass	Put on top of the glass dish to keep the bees in, but allow the images to be captured.	25.5 cm x 25.5 cm Transparent

4 Arenatest

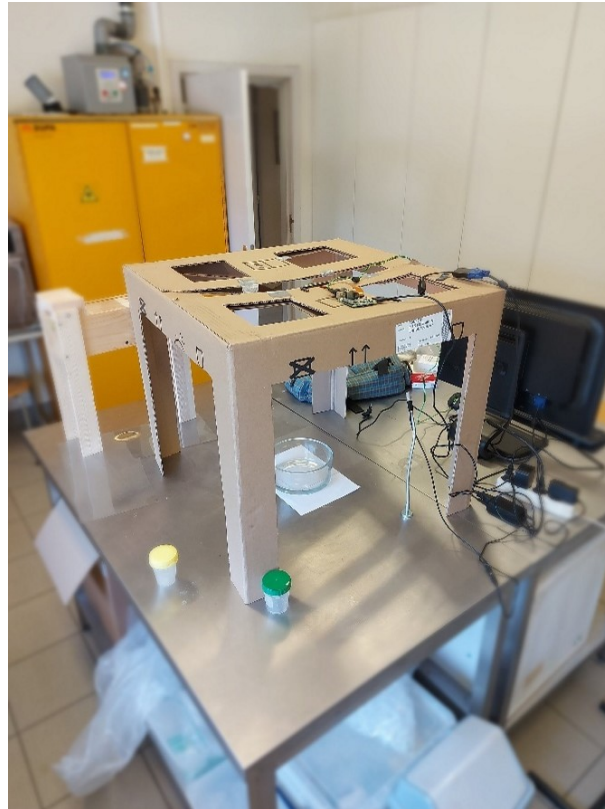


Figure 4.4: Arenatest setup in the Honeybee Valley lab

The experiment is performed in the lab of the Honeybee Valley research facility. The setup (see Figure 4.4) consists of a round glass dish which is the arena. This dish provides a clean, transparent surface for bee and mite interaction while allowing unobstructed imaging from above. The round shape was chosen over a square shape to ensure that the bees don't stay and hide in the corners of the arena. This was predicted to be their natural reaction to a new environment with corners, and the round shape thus eliminates this behavior from happening and enables them to be more active during the test. The dish is placed on a white A4 paper that provides a white background for the captured images. This makes sure the bees and mites are clearly visible, are in contrast to the background and also eliminate any possible reflections from the metal table the test was performed on (which would distort the detected contours). On top of the glass dish a sheet of plexiglass was placed to keep the bees in, but allow them to be observed (to monitor the experiment) and captured by the camera (E. Danneels, personal communication, December 17, 2024).

To orient the camera downwards to obtain top-down view a custom mount was constructed out of cardboard. The height of this mount is crucial to ensure the camera captures sharp images. Looking at the camera characteristics, the distance at which the sharpest images can be captured is about 50 cm. Through a test done to confirm the best camera distance, a height of 50 cm was determined. To add some margin for the thickness of the glass dish and the height of the bees, the mount was made 52 cm in height. The width and length of the mount were

4 Arenatest

made in such a way that the glass dish can fit under it easily, but are otherwise of no crucial importance. Because sufficient (natural) light and no shadows on the dish bottom are important to capture well processable images, big squares of material were cut out of the mount. All material that could be taken away, while making sure the mount still had the strength to hold up the Raspberry Pi, was cut out. In the center of the top plane a hole was cut out that allows the camera to capture images (E. Danneels, personal communication, December 17, 2024).

On this mount the Raspberry Pi (see Figure 4.5) was placed in such a way that when the camera module was connected to the CSI port and the camera was put down on the top surface of the mount, it would line up to the opening provided to capture images through. To ensure that the camera stays stable, and doesn't shake when for example the mouse connected to the Raspberry Pi's USB port was moved, a small (relatively) heavy object was placed on the camera cable. For this, a hex key was used. Finally a LED light was placed next to this setup and pointed up to the top of the cardboard mount. This way the light could bounce off the material and indirectly shine on the arena, to reduce shadows and provide additional lighting without producing a glare in the glass. The indirect light also makes sure that bee behavior is not disturbed, as bees naturally tend to go towards light sources as a result of their tendency of going towards the sun (E. Danneels, personal communication, December 17, 2024).

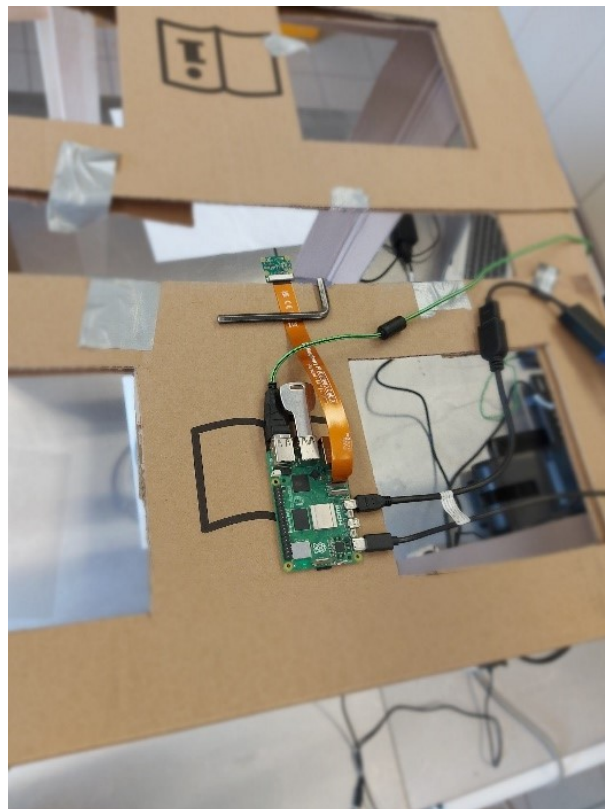


Figure 4.5: Raspberry Pi 5 placement in arenatest setup

4 Arenatest

The placement of the arena is important for the monitoring algorithm and needs to be exact, because the algorithm precisely segments out the arena floor through cropping and masking. This can be done through visually placing the arena in the center of the camera, then confirming by showing the camera images on the screen and adjusting the placement of the arena if necessary. The choice to only monitor the arena floor (and not the walls) was because of several reasons. First, it was not expected that the bees would walk up the walls or plexiglass ceiling. However, upon conducting the test some bees would walk across, stay or group up on the walls and/or ceiling. Still, the walls were chosen to be ignored because on the captured images the bees there would be seen from a whole other perspective than the bees below. When grouped up, occlusions also appear more frequently. Furthermore, when this behavior occurs, it will be visible in the extracted metrics and statistics. In the analysis it can thus immediately be concluded that these bees are walking on the walls and are not engaging with the mites. This will be expanded upon in 4.3.3: Analysis of extracted metrics (E. Danneels, personal communication, October 14, 2024).

4.2.2 Test conditions and parameters

Before performing the test itself, the conditions and parameters need to be determined. Firstly, in choosing the amount of subjects, the size of the arena is the most important factor. The bees need to be given plenty, but not too much, space to allow them to move around comfortably but still be close enough to the mites at all times. Because of this, the number of bees was chosen to be five. This number is an ideal number, tests were also performed with more than five bees due to the variability of the bee gathering method. This method uses a handheld vacuum cleaner that was converted to be able to suck bees into a small container. The bees can be captured from in front of an active beehive while they are leaving or entering the hive. For the experiment only worker bees were used, since these are used to being in natural light as opposed to drones and the queen bee which spend almost their entire life in the hive. Although another test could be done where the arena is placed in a closed box, with only an opening to allow the camera to capture images. Artificial red LED lights in combination with a regular camera would be necessary since mites won't show up on thermal- or night vision cameras. The captured bees are sedated with CO₂ gas, and then put in the arena to wake up. It is important that they aren't sedated too much, since this can result in them being out for a long time (> 30 minutes) or in the worst case death. Too little was also not good, since then the bees would not be asleep and would immediately start moving around. Bees were way more active and flew around in the arena, which was not expected behavior. When the bees are sedated the right amount, they would wake up after about one minute and would not fly around too much (E. Danneels, personal communication, February 24, 2024).

After the bees have woken up, the mites are added to the arena. The mites are captured by catching bees from a suspected mite-infested beehive in a container. These bees are then covered in powdered sugar, and the container is shaken. In this process, potential mites that would've been attached to the bees would fall off together with the powdered sugar. Afterwards, the bees are released and what rests is a container with potential live mites.

4 Arenatest

The sugar is poured onto a white paper and is then carefully searched for mites. When a mite is found, they are carefully put in the middle of the arena. For this test, based on the arena size an amount of bees, two mites would be ideal. However one mite would suffice if no more were found, since the process of capturing them is unpredictable and disturbs the beehive. Depending on the time of year this process is easier or harder. This is directly linked to the amount of Varroa mites present in the hive. In spring this is more time-consuming but in summer, when there are many mites present, it will be faster (E. Danneels, personal communication, February 24, 2024).

When the mites are added to the arena, the experiment is started. The Python detection algorithm is launched on the Raspberry Pi, and the monitoring can be followed in real-time on the screen. The arenatest stops when the mites have all jumped on a bee, or when a time-limit of about fifteen minutes is exceeded. Since all the captured images are saved with and without detection, they quickly take up a lot of space in the storage of the Raspberry Pi. This is why a time limit is needed as to not take up too much space.

Finally, to provide enough and even lighting to capture clear images, the test is done next to a window, to provide natural light. The laboratory lighting is used to provide additional illumination, as well as a LED light.

4.3 Computer vision based monitoring and analysis

In this next chapter, the computer vision based arenatest monitoring algorithm is developed. As a base, the algorithm that was created through the simulation was used, and the processing for the real arenatest was built from there. First, tests were done with paper cutouts of bees and mites (to scale) to be able to build the setup. Next tests with only dead bees were done, then dead mites were added and finally live bees and mites were used. This way, an algorithm was developed step by step. With the detected bee and mite positions, several metrics were calculated, that can be plotted and allow the arenatest to be analyzed.

4.3.1 Detection and classification of bees and mites

The first approach was based on contour detection and inner area thresholding for both bees and mites. For bees this worked fine, with good pre-processing of the images, but the mites contours were simply too small. This resulted in many false positive mite detections when bees had small spots on them, or even on the ends of bees legs. Shadows or glare in the arena bottom by sunlight also often resulted in false detections. The setup used to test this approach was based on paper cutouts of bees and mites, in a non-ideal space with too much light coming from one place which caused shadows. This is another reason why this method failed. A very sensitive detection method, in combination with an environment which produced a lot of low quality images with noise. Because false positives are something that was to be avoided to allow proper analysis of the extracted metrics, and especially in the case of mites, this had to be done another way.

4 Arenatest

The next approach used color-based segmentation in the HSV (Hue, Saturation, Value) color space. HSV, alongside HSL (Hue, Saturation, Lightness), are cylindrical interpretations of the RGB (Red, Green, Blue) color space as illustrated in Figure 4.6. When an image of the bees and mites in the arena was transformed into this color space using Open CV's libraries, the bees and mites were clearly distinguishable from the background. The bees appeared in a bright purple color, while the mites turned into a cyan color. The background was mostly white in this color space, except for shadows or other darker spots which turned into a gray/blue color. This allowed them to be segmented out easily using a simple color threshold after basic pre-processing and filtering, to rid of the shadows or other noise that might affect particularly the mite segmentation. The colors, however, slightly differed depending on the amount of light in the environment which made static color thresholds difficult to determine, and moreover not robust. To address this problem, the color thresholds were not made static but dynamic. Before the segmentation program is launched, a picture is taken of the arena with bees and mites. On this picture ten pixels of bees and ten pixels of mites were selected, based on the HSV values of the selected pixels a proper range was determined. Then the detection program was launched using these threshold values. This approach worked well, but didn't translate to the real test. The colors of the bee cutouts were chosen as a bright yellow and black, and the mite cutout color was chosen as a dark red. These, and specifically the bee colors, turned out to differ significantly from the real colors. Bees had a darker color than expected and also appeared more brown. This made it so there was almost no difference between bees and mites in the HSV image, which resulted in this detection method failing for the real test.

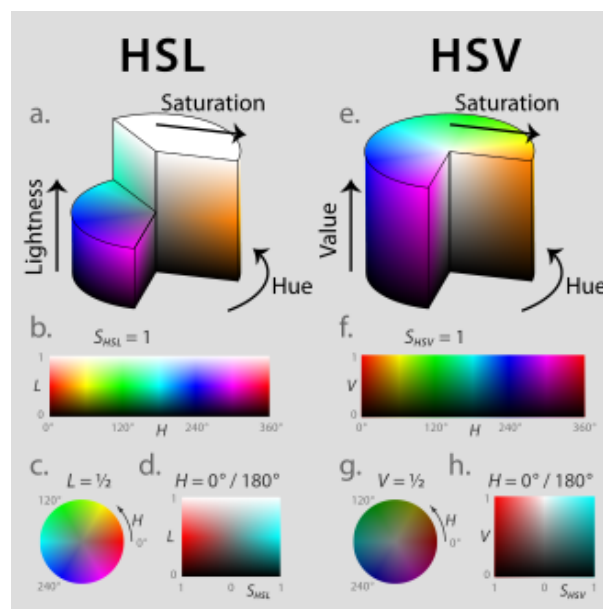


Figure 4.6: HSL and HSV explained visually (Wikipedia, 2025)

For the final approach contour detection was again used for the bee segmentation, and mite detection. The difference from the first approach is the better test setup, mainly more and diffuse lighting conditions and better

4 Arenatest

image processing. These resulted in less noise and thus better contour detection. Again, an inner area threshold-based approach was used, with more filtering for complex situations in which false positives were more likely to appear.

First, a circular mask was used to only keep the bottom of the arena, then the image was cropped to fit this circle exactly (as shown in Figure 4.7). This was done to avoid having bees crawling on the walls. If this happened, these would be seen from another perspective than would be expected by the segmentation code, and occlusions could appear when the bees huddle together. This does mean that these bees will not be visible in the frame, this will be dealt with using the extracted metrics.

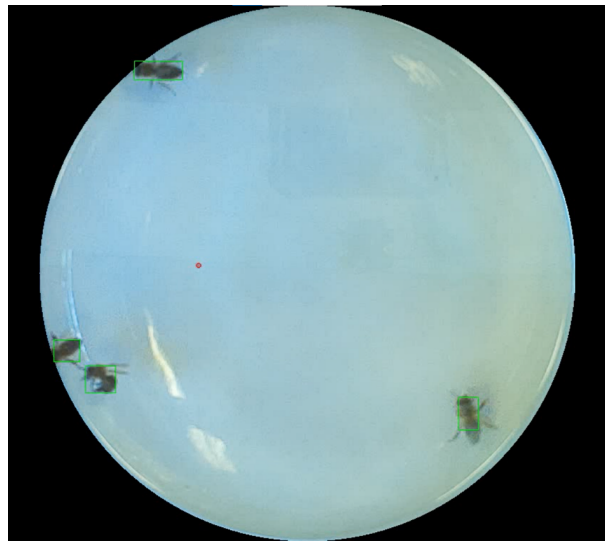


Figure 4.7: Arenatest bee (green) and mite (red) classifications, with the circular mask visible

Next, the image is turned to grayscale for further processing and the bee segmentation is done based on contour analysis. First, Gaussian blur is applied to the image. Gaussian blur is a low-pass filter that smooths an image by averaging nearby pixels, with weights given by a Gaussian (bell curve) function. This is done to reduce noise and minor variations in intensity, and makes thresholding and contour detection more robust by eliminating small artifacts that could otherwise be interpreted as objects. Next, the white background underneath the arena makes it so a binary inverse threshold can be used to highlight darker objects, like bees. This turns the image into a binary format, resulting in only black and white pixels. For binary inverse thresholding, pixels below the threshold (darker) become white (255) and those above it become black (0). A circular mask with a radius two pixels smaller than the one used to mask out the bottom of the arena is applied. This was done because the pixels outside the masked out arena bottom in the rectangular crop affected the bee detection when they clipped the edge of this mask. Then contour detection was used to detect the resulting outlines, which now could be filtered as to only segment the bees.

For the bee segmentation, an inner area based threshold is used, like in the simulation classification, but with

4 Arenatest

extra filters applied. Firstly, for each detected contour, the inner area is measured. An important distinction must be made here between single bees and multiple bees close together. This was achieved through a well-chosen threshold value. This was determined through visualizing the contour inner area's on the image and finding the lower and upper threshold in the case of a single bee, and multiple bees close together. The resulting thresholds (in pixels squared) were as shown in Table 4.3.

Table 4.3: Bee classification thresholds of contours for the arenatest

Class	Lower threshold	Upper threshold
Singular bee	350	1900
Overlapping bees	1900	5300

The lower threshold value for single bee detection was crucial for filtering out noise contours caused by e.g. glare on the arena floor, but needed to be large enough as to still be able to detect bees that are only partially in frame because they are on the edge of the arena floor mask. The large difference between the lower and upper bee inner area thresholds was necessary, because bees would sometimes walk upside down across the plexiglass on top of the arena. This would make them appear larger in the image, and thus have a larger contour inner area.

The singular bees were processed as follows. First, their center coordinates are added to the list that keeps track of the bee positions in the frame. Then, a green bounding box is drawn to visualize the detected bee. Finally, an expanded version of this bounding box is kept in a separate list. This will be used for the mite detection filtering. The rectangles in this last list are 20 pixels higher and 20 pixels wider than the original bounding box. For the overlapping bee detection a similar process was used. Here, an additional filter was also added to be able to further refine the bee detection (if possible). This filter is based on ellipse fitting. Because bees naturally have an elliptical shape, a contour can be classified as bee contour if an ellipse can be fitted to (or recognized in) that contour. This results in a finer bee detection, without too much further processing and extra computational resources needed. The ellipse fitting algorithm used is OpenCV's "fitEllipse", which only takes the contour being classified (bee or not a bee) as a parameter. It uses the Direct Least Squares Fitting of Ellipses method developed by A. Fitzgibbon to approximate an ellipse that best matches that contour, and returns the ellipse center, axes lengths and rotation angle of its major axis (Fitzgibbon et al., 1996). The requirements and assumptions of this algorithm are that:

- Minimum 5 points are available,
- The shape is at least approximately elliptical and
- The points given lie in the same plane (2D).

This method is illustrated through pseudo-code in the added Figures 4.8, 4.9 and 4.10.

4 Arenatest

First, a region of interest (ROI) is cropped out of the original frame and processed again, in an attempt to be able to see more details in this area of the frame. It again is turned into a grayscale image and blurred using Gaussian blur. Next, Otsu's thresholding is performed to separate the image into foreground and background. Otsu's method analyzes the histogram of a grayscale image and chooses the threshold that minimizes the intra-class variance (or equivalently maximizes the inter-class variance) between the two groups of pixels: pixels below the threshold (background) and pixels above the threshold (foreground). This results in a finer thresholding. Next, morphological opening is used to clean up noise in the image. This operation removes small blobs or clutter from images, while preserving larger (connected) shapes. Then, on this image contour detection is performed and these contours are analyzed.

```
# Step 1: Crop the region of interest from the frame
crop = extract region from frame at coordinates (x, y, w_b, h_b)

# Step 2: Convert the cropped region to grayscale
gray_crop = convert crop to grayscale

# Step 3: Apply Gaussian blur to the grayscale image
blur_crop = apply Gaussian blur with kernel size (7, 7) to gray_crop

# Step 4: Apply thresholding to segment the objects
thr_crop = apply Otsus thresholding with binary inverse on gray_crop

# Step 5: Perform morphological opening to remove noise
morph_crop = apply morphological opening using predefined kernel on thr_crop

# Step 6: Find contours in the morphologically processed image
cnts2 = find external contours in morph_crop

# Step 7: Initialize detection counter
detected = 0
```

Figure 4.8: Pseudo-code for the first part of ellipse fitting method

If an ellipse can be fitted, its major and minor axes are thresholded as to be sure that the ellipse was fit on a bee. These threshold values were again determined through visualizing the major and minor axes lengths on the image. When the values fall between the thresholds, and a bee is thus detected, a similar process to singular bee detection is used. Additionally, a variable that keeps track of the bees that were detected through this method, is incremented.


```

# Step 8: Iterate over all found contours
for each contour c2 in cnts2:
    if the contour has at least 5 points:
        ellipse = fit an ellipse to c2

        # Extract ellipse parameters
        center = (xe, ye)
        axes = (MA, ma)
        angle = rotation angle of the ellipse

# Step 9: Check if ellipse matches expected bee size
if MA is between 8 and 25 AND ma is between 35 and 60:

    # Map the center of the ellipse back to the original frame coordinates
    bx = x + xe
    by = y + ye

    # Store detected bee position
    append (bx, by) to bee_positions

    # Draw rectangle around detected bee
    draw rectangle on frame at (bx, by) with size (MA, ma)

    # Create and store expanded bounding box polygon
    rectangle_points = create rectangle around (bx, by) expanded by 10 pixels
    reshape rectangle_points for OpenCV format
    append rectangle_points to bee_bounding_boxes

    # Increment detected bee counter
    detected += 1

```

Figure 4.9: Pseudo-code for the second part of ellipse fitting method

This variable can then be used to tell if the method worked correctly or not. If the bees are so close together that their contours overlap significantly, even ellipse fitting cannot tell them apart. In this case, the code falls back on the original detected contour of the overlapping bees, seen as one, and they are again processed similarly to the singular bees.

```

# Step 10: If no bees were detected using ellipse fitting
if detected == 0:
    # Fallback: use original bounding box center
    append (cx_b, cy_b) to bee_positions

    # Draw original bounding box on frame
    draw rectangle on frame at (x, y) with size (w_b, h_b)

    # Create and store expanded bounding box polygon
    rectangle_points = create rectangle around (x, y) expanded by 10 pixels
    reshape rectangle_points for OpenCV format
    append rectangle_points to bee_bounding_boxes

```

Figure 4.10: Pseudo-code for the last part of ellipse fitting method

A case where ellipse fitting was necessary, because bees were overlapping, is visible in Figure 4.11. In Figure 4.12

4 Arenatest

an overview is given of the complete bee detection process.

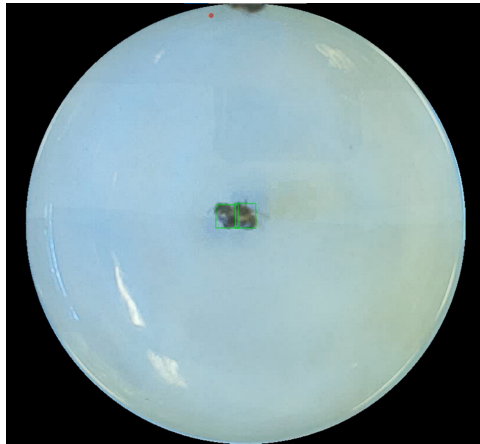


Figure 4.11: Case where ellipse fitting was used to be able to separate close together bees

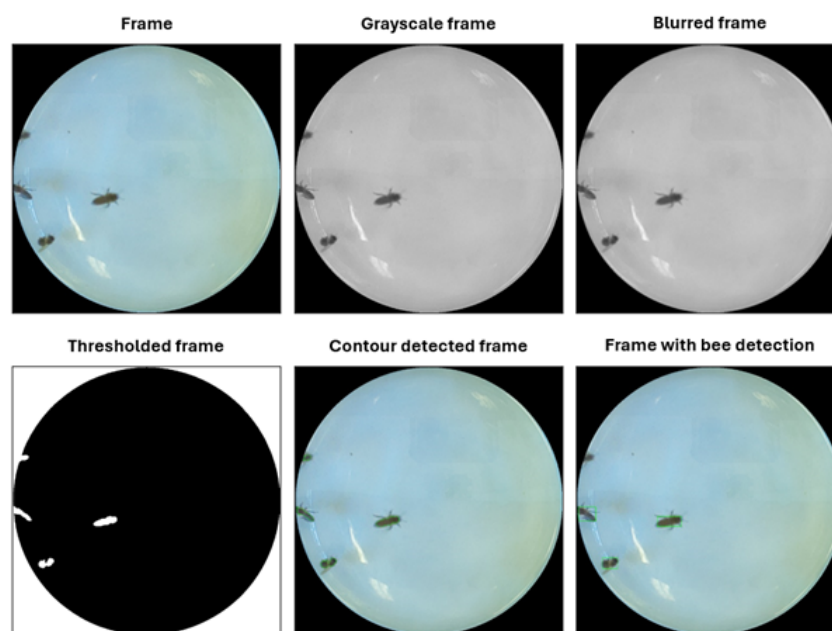


Figure 4.12: Arenatest bee detection process

The mite detection also uses contour analysis, but does this in a different way. First, a circular mask is applied to the frame, to crop out a band of two pixels from the outer edge of the arena. This is done because of issues with glare on the edge of the arena, which resulted in a high number of false positives in the mite detection. Then, the image is blurred using a median blur. Median blur is used for mite detection because it effectively removes salt-and-pepper noise and small intensity variations from the image while preserving edges, which is crucial for accurate detection of small objects like Varroa mites. The blurred image is passed to the Canny edge detector

4 Arenatest

which highlights strong intensity gradients, the edges in the image, by computing first and second derivatives. This allows for precise boundary detection and helps isolate potential mite contours. A circle is drawn on the frame, matching the circular mask that was applied earlier. This is because the edge detection detects the border of the mask as an edge, which this new circle prevents.

At the core of the mite detection is Open CV's SimpleBlobDetector. This is a feature detection algorithm designed to find 'blobs', (connected) regions in an image that are uniform in brightness and differ from their surroundings. The algorithm uses several parameters to tune what blobs are detected. For the mite detection, first, an inner area filter was used, with a lower and upper threshold, to make sure no noise or too large objects were detected. Additionally, a circularity filter was applied to further refine the detection, as mites have a circular shape. Using these parameters, the algorithm returns a list of keypoints, each representing a blob's center and size. These often contained false positives in small circular spots on the bodies of bees, ends of their legs or their eyes. To further filter these out, the expanded bounding boxes which were added to a list during the bee detection can be used. By checking if a keypoint is within this rectangular area or not, these false positives can be removed easily. The bounding boxes are expanded to make sure that the legs of the bees are also captured in this filter.

If a keypoint passes this filter, it's center coordinates are added to the list of mite positions and they are added to a list of filtered keypoints. Using the built-in drawing function, these can then be visualized on the frame in a red color. This process is shown in Figure 4.13.

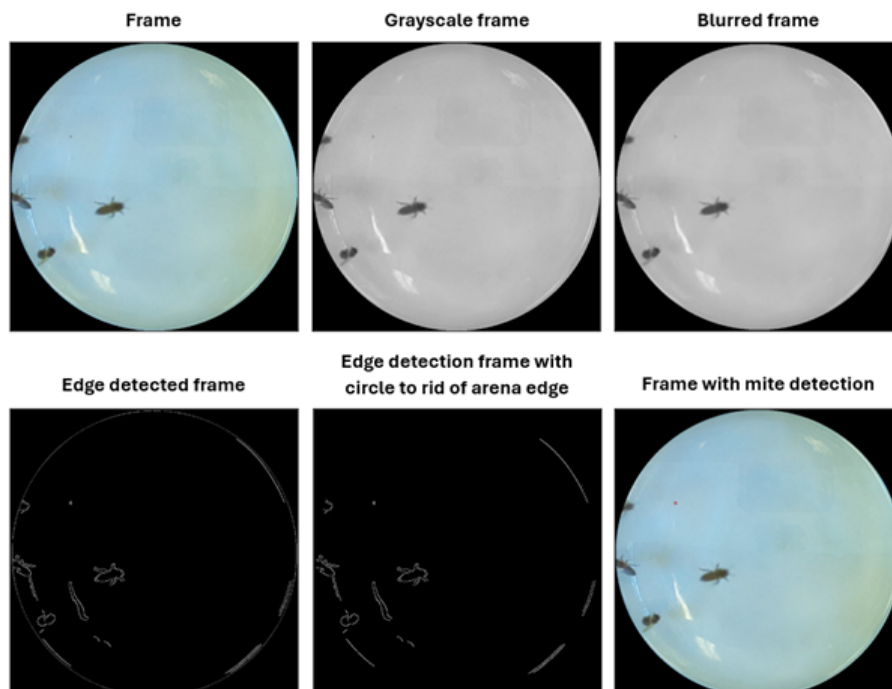


Figure 4.13: Arenatest mite detection process

4.3.2 Statistic and metric extraction

To be able to quantify and analyze the arenatest, metrics can be extracted based on the detected bee and mite positions. Heatmaps of their positions are kept to be able to analyze where they've been over time. From these positions, several distance-based metrics are computed to quantify the spatial relationships among bees, between bees and mites, and relative to the arena geometry. Each metric is defined by standard geometric calculations using Euclidean distances. These metrics make it possible to summarize their interaction and movement patterns. In the following section, the extracted metrics are discussed: how each metric is calculated, what it represents and how it informs analysis of spatial dynamics for the arenatest.

Bee-to-mite distance

This metric is computed as the Euclidean distance between every bee and every mite in the frame. The Euclidean distance between a bee at (x_i, y_i) and a mite at (x_j, y_j) is calculated using Equation 4.1.

$$d_{(i,j)} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (4.1)$$

After calculating these distances for each bee, only the distance to the mite that is closest to them in the frame is kept. Once all bee-to-mite distances are computed, the code calculates the mean, standard deviation, maximum and minimum of this list of distances across all the bees that are detected in this frame. Only the detected bees are considered. The amount of bees is this not a constant, but is allowed to differ between frames. This metric is visualized in Figure 4.14.

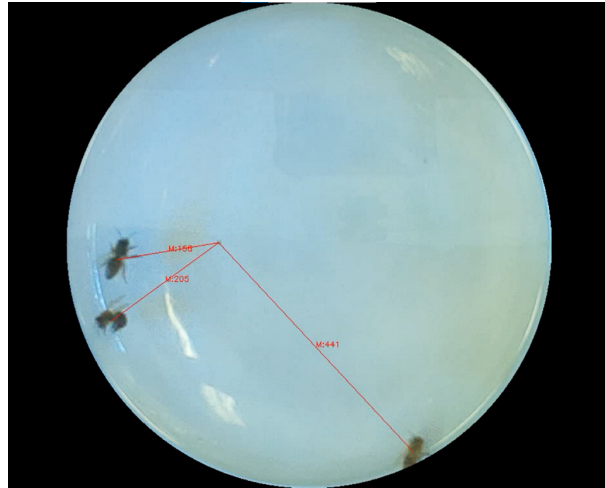


Figure 4.14: Bee-to-mite distance visualized. Taken from the arenatest performed on 25/04/2025

The bee to (closest) mite distance metric captures how close bees and mites are to each other in the arena space. A small average bee-to-mite distance means that, averaged across all subjects, bees are near mites (for example, mites may be trying to jump on bees or bees are engaging in contact to defend themselves). A large average

4 Arenatest

implies that bees tend to be far from mites (bees could be avoiding the mites). The minimum distance often reflects the closest encounter (e.g. a bee directly adjacent to a mite), while the maximum reflects the farthest separation. The standard deviation indicates how variable the distances are. A high variance means some bees are very close while others are far. This may indicate a lack of coordination among the bees. Bee-to-mite distance thus quantify potential interaction events. For instance, a sudden drop in this metric could signal a grooming attempt. By summarizing these distances statistically, the overall level of bee and mite spatial relations are objectively captured.

Intra-bee distance

This metric is computed as the pairwise distance between all distinct pairs of bees using the same Euclidean distance formula as above. This metric is displayed in Figure 4.15. The amount of pairs can be calculated using binary coefficients, which can be written using Equation 4.2 in the case of N bees.

$$\text{Amount of intra-bee pairs} = \frac{N!}{2!(N-2)!} \quad (4.2)$$

Or in the case of 5 bees through Equation 4.3.

$$\text{Amount of intra-bee pairs} = \frac{5!}{2! \cdot 3!} = 10 \quad (4.3)$$

From this set of intra-bee distances, once again, the mean, standard deviation, maximum and minimum across all the bees pairs are calculated.

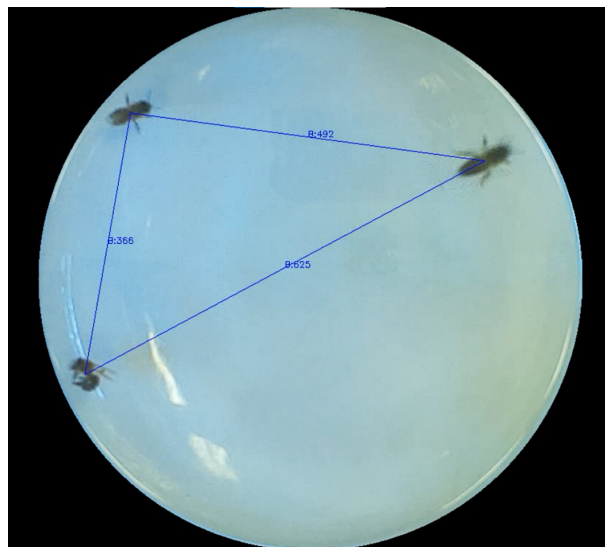


Figure 4.15: Intra-bee distance visualized. Taken from the arenatest performed on 25/04/2025

The intra-bee distances describe the internal spatial relations of the bee group. A small average or minimum distance indicates that bees are clustering tightly (possibly touching or engaging in social interaction or grooming),

4 Arenatest

while a large average or maximum suggests they are well-dispersed across the arena. A low standard deviation means bees are roughly evenly spaced; a high standard deviation indicates some are clustered and others isolated. These distances are thus a measure of aggregation vs. dispersion.

In the arenatest, aggregation could mean bees feel threatened by the mites and thus group together. For instance if mites trigger an alarm pheromone, bees might cluster together (decreasing mean bee-bee distance). On the other hand, they could just be engaging in social interaction, like the expected first behavior was for the bees that would wake up in the arena. Dispersion could indicate that the bees feel safe and are randomly moving within the arena, not noticing the threat of the mites. Thus, the intra-bee distance metrics help quantify collective movement patterns and social spacing among the bees.

Bee-to-center and bee-edge distances

For the calculation of the bee-to-center distance metric, the arena center (e.g., coordinate origin) is first defined. For each bee at position (x, y) , the Euclidean distance to the center coordinate of the arena (c_x, c_y) is computed, using Equation 4.1. From the list of bee-to-center distances of all the bees, the mean, standard deviation, maximum and minimum are calculated. This metric is visualized in Figure 4.16.

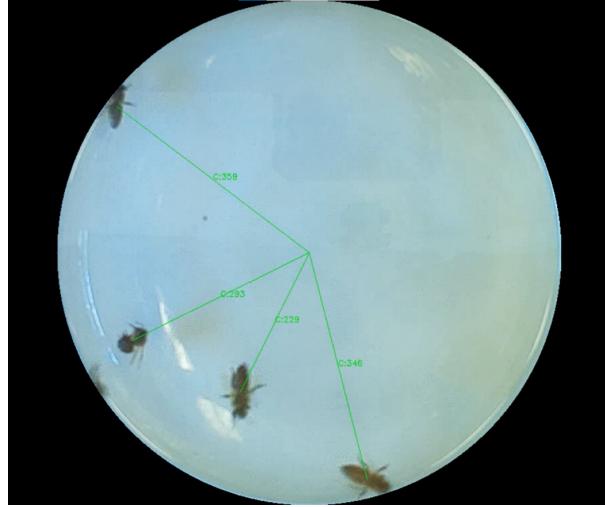


Figure 4.16: Bee-to-center distance visualized. Taken from the arenatest performed on 25/04/2025

For the calculation of the bee-to-edge distance, the edge of the arena is first defined as the boundary of the circular mask that segments out the arena floor. For each bee at position (x, y) , this distance is computed as the circle's radius R minus the already calculated bee-to-center distance via Equation 4.4.

$$d_{edge(i,j)} = d_{center(i,j)} - R \quad (4.4)$$

This metric is visualized in Figure 4.17.

4 Arenatest

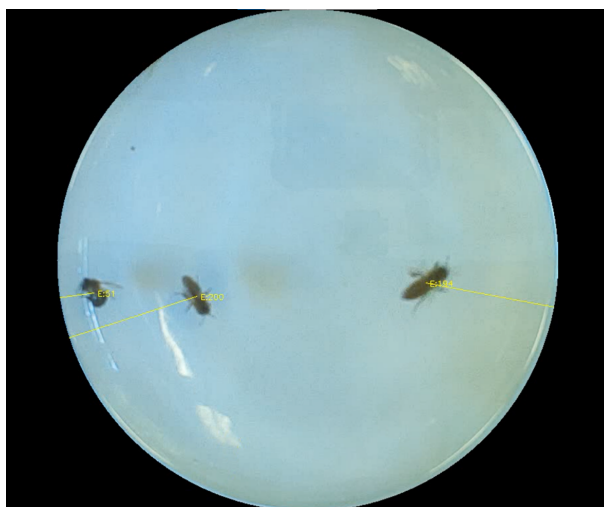


Figure 4.17: Bee-to-edge distance visualized. Taken from the arenatest performed on 25/04/2025

These metrics locate where bees prefer to be in the arena. The bee-to-center distance indicates how far bees typically venture from the center: a small mean indicates bees are clustered near the center; a large mean indicates they are near the arena border. The bee-to-edge distance is essentially a flipped scale: a small mean indicates bees are clinging to the arena walls (equal to a large mean bee-to-center distance), while a large mean indicates they are concentrated centrally (equal to a small mean bee-to-center distance). The minimum and maximum values of these metrics indicate the bee that is closest to the arena center and the bee that is farthest from the center. The standard deviation shows whether some bees hug the wall while others stay near the center of the arena, or whether all bees show similar spatial usage.

Center vs. edge occupancy measures the spatial bias of the bees. Avoidance of the central regions and staying close to walls (small bee-edge distances, or large bee-center distances) could be an indicator of mite avoiding behavior or just the opposite, depending on the location of the mites in the arena. Similarly a bias for a certain area of the arena could prove both mite-avoiding and mite confronting behavior. If no bias is measurable, this could indicate that the bees are oblivious to the mites and show no potential Varroa-resistant behavior (E. Danneels, personal communication, February 24, 2024).

Mite-to-center distance

The calculation of the mite-to-center distance is done in a similar way as the bee-to-center distance. This metric is displayed in Figure 4.18.

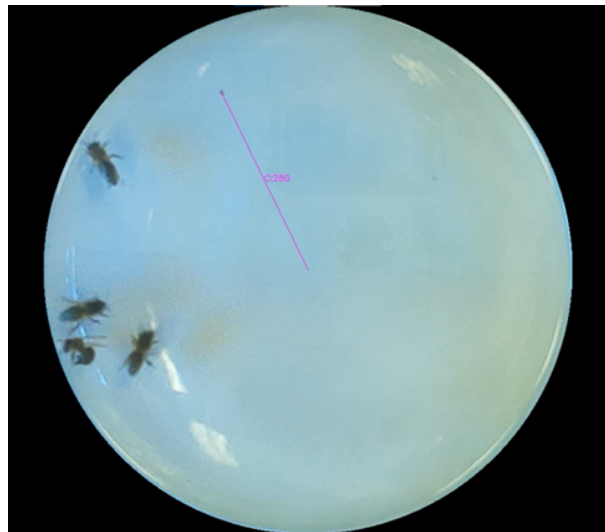


Figure 4.18: Mite-to-center distance visualized. Taken from the arenatest performed on 25/04/2025

These metrics show where mites tend to reside within the arena. A low average distance means mites stay near the center, while a high average means they tend to linger near the walls. The expected behavior before performing the test is that the mites would stay in the center of the arena where they were put down, and would move minimally towards nearby bees. Since the mite's goal is to jump onto the bees, their spatial distribution can reflect that of the bees. If mites are found far from the center while bees are centered, it might indicate that this mite had hitched onto a bee and has fallen off or that it was moved there (unknowingly) by a passing bee.

Comparison of the mite-to-center with bee-to-center can indicate a similarity in movement. For example if both bees and mites have similar distance profiles, mites may be following bees to hitch onto them or alternatively a bee may be attacking a mite. Divergences (e.g. mites hugging walls when bees do not) could mean that there is no mutual interest.

Bee count and proximity metrics

The bee count metric is the number of bees detected in each frame. It is computed by counting tracked bee objects by taking the length of the bee positions list. This metric was added to have an idea of how many bees are detected in the arena in that frame, and subsequently to know how many bees are outside of the frame. This way, it is also known how many bees are included in the calculated statistics.

To quantify direct encounters, proximity metric is calculated. A threshold was used that determines when a bee is close to a mite. Based on the bee and mite positions, and the resulting bee-to-mite distance, a counter that keeps track of how many bees are close to a mite in that frame can be incremented. This metric indicates the frequency and duration of close encounters (potential grooming or interactions) between bees and mites. It essentially discretizes the bee-to-mite distance data into "close" vs. "not close". By quantifying how often the bees come

4 Arenatest

within this interaction radius of a mite, it can be tracked how quickly bees find and respond to mites, or how persistent mites are at staying in close proximity to bees.

Heatmap

Finally, the bee and mite positions are visualized in their own heatmap. Area with warmer colors are spots where bees or mites spend more time and vice versa. This gives an idea of areas in the arena the bees and mites had a bias for, and can help determine if their movement is random, or perhaps stimulus-driven. Areas of considerable overlap in these heatmaps indicate places where interactions might have happened. Additionally the heatmaps from the tests with and without mites can be compared, and possible differences can provide insights into the bee movements patterns. On the bee heatmap, the initial positions of the bees in the arena are shown as a red circle. These are displayed in Figures 4.19 and 4.20, for bees and mites respectively.

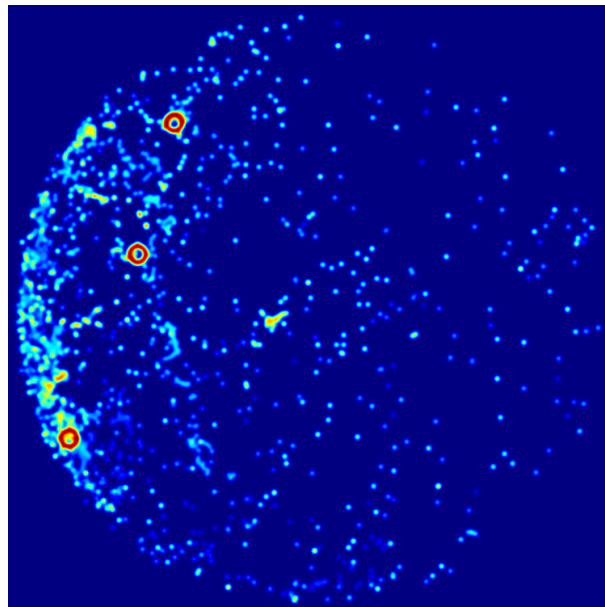


Figure 4.19: Heatmap of bee positions. Taken from the arenatest performed 25/04/2025

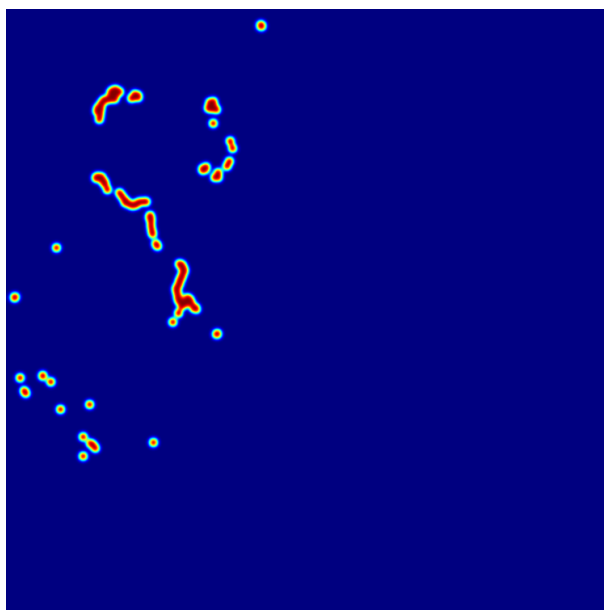


Figure 4.20: Heatmap of mite positions. Taken from the arenatest performed 25/04/2025

Each of these metrics plays a complementary role. Together they quantify the behavior of the arenatest subjects, and allow the arenatest to be analyzed: bee-to-mite distances directly quantify bee-mite proximity, intra-bee distances quantify social spacing, center- and edge distances quantify spatial occupancy patterns, and count/proximity metrics capture event-level information. By examining trends in means and variances of these metrics patterns, such as clustering, avoidance, or attraction, can be detected.

4.3.3 Analysis of extracted metrics

These metrics are now going to be used to analyze the arenatest performed the 25th of April 2025. This test consisted of five bees and one *Varroa destructor*, that were gathered like described in 4.2.2 Test conditions and parameters. As a control, the arenatest performed on the 14th of March 2025 will be used, since this test contained dead mites. This test consisted of five bees and two mites. This test will be analyzed first and will form a basis of comparison for the analysis of the test with live mites. The goal is to be able to quantify the arenatest, and take a first step towards proving possible Varroa resistance (or lack of) in the bees. The hypothesis of this analysis is that the bees are not Varroa resistant, this is based on knowledge of the bee populations and previous observations by the researchers of Honeybee Valley (E. Danneels, personal communication, February 24, 2024).

Arenatest with live bees and dead mites (performed 14/03/2025)

Looking at the test data, it can be separated into three different timeframes, based on regions of similar trends and/or values, that can be analyzed:

4 Arenatest

1. Frames 0–1000
2. Frames 1000–1700
3. Frames 1700–3000

Because the full analysis of all three timeframes for both arenatests is substantial, it was added in Appendix E⁴. A selection of metrics will be analyzed for one timeframe of both arenatests, as to give an idea of what this analysis includes and how the final conclusions are drawn.

For the arenatest performed on the 14th of March 2025 this will be the second timeframe, since this is the most interesting. A conclusion of the findings of timeframes one and three was provided, as to give the necessary context.

Timeframe 1: frames 0 - 1000

The combined analysis of bee-to-mite, intra-bee and bee-to-center distance metrics during timeframe 1, supports the hypothesis that the bees were waking up from anesthesia, and thus remained largely stationary. Across all metrics, values remained approximately constant, with minor fluctuations attributable to tracking noise, temporary overlaps, or bees momentarily moving in and out of frame. The mean distances did not indicate directional trends from bees toward mites, other bees, or the center of the arena, and standard deviations remained stable further reinforcing the absence of coordinated movement or exploratory behavior. Heatmap visualizations corroborated the metric findings, linking starting positions with minimum and maximum distance values. Altogether, the evidence strongly suggests that this timeframe primarily captures the bees' recovery phase, not active interaction or exploration.

Timeframe 2: frames 1000 - 1700

The bee-to-mite distance metric for this timeframe (Figure 4.21) shows roughly constant values for its minimum and maximum graphs. The mean distance shows a downwards trend, while the standard deviation of the bee-to-mite distances shows an upwards trend. The hypothesis is that the bees are starting to explore the arena environment during this timeframe, after waking up from the anesthesia in the previous one. Potential contact with the *Varroa destructor* could occur, no signs of interest in the mites may be a first sign of potential non-Varroa resistant behavior.

- The mean bee-to-mite distance shows an ascending and descending pattern in this timeframe. This may reflect on exploratory behavior by the bees, as they repeatedly move closer to and further from the mites.
- The standard deviation trend aligns with the mean distance graph and further supports the hypothesis of exploratory movement. Toward the end of the timeframe, the standard deviation rises rapidly over a short

⁴GitHub repository for arenatest, pollen monitoring and website data

4 Arenatest

period, which warrants further investigation.

- The minimum distance remains mostly constant at $35 \text{ px} \pm 5 \text{ px}$. Toward the end of the timeframe, during the same period as the rising standard deviation, changes occur. This suggests a likely link between the two statistics. This relationship can be better understood by examining this period in conjunction with other statistics.
- The maximum distance remains generally steady at $225 \text{ px} \pm 5 \text{ px}$, apart from a few brief peaks. This suggests that the same bee consistently stays furthest from the mites, possibly remaining static or moving in a circular path around them (though such behavior is unlikely). The observed peaks may be caused by other bees temporarily moving farther from the mites than the static bee. This hypothesis can be examined using the bee-to-center distance metric.

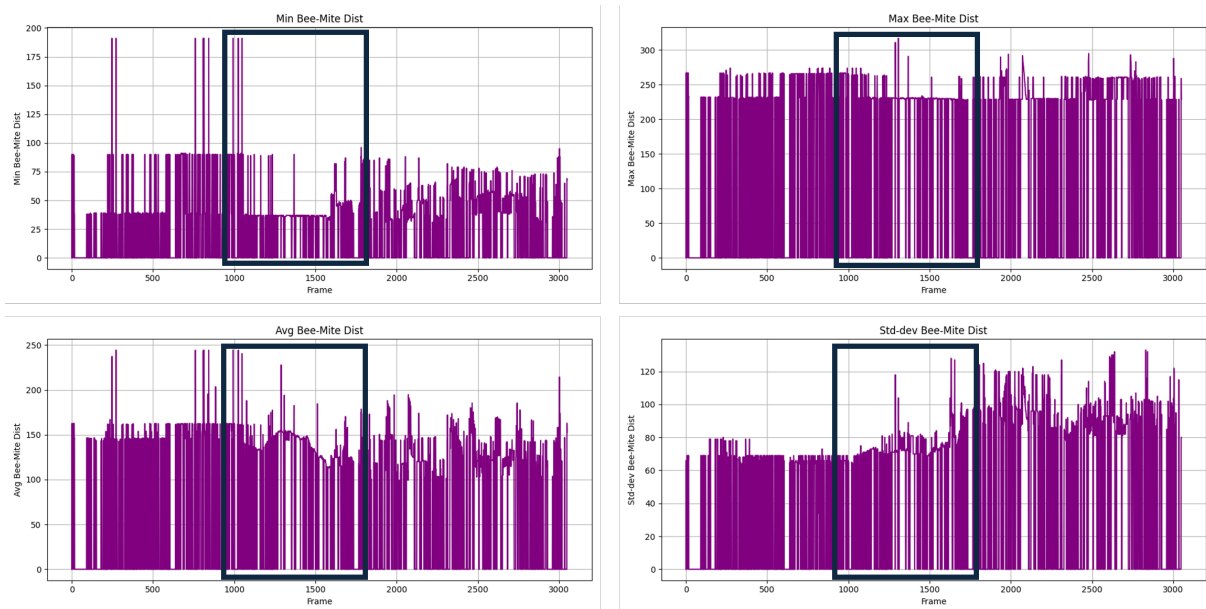


Figure 4.21: Bee-to-mite distance metric for timeframe 2 of the arenatest performed on 14/03/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.

The findings from the mean bee-to-mite distance, and standard deviation during this timeframe support the hypothesis that the bees are showing exploratory behavior. Not all bees are doing this however, as shown by the mostly steady minimum and maximum distances. Furthermore, the minimum distance staying constant throughout the entire timeframe, and even rising near the end, are an indication that there were no interactions with the mites and no interest was taken into them by the bees. Towards the end of the timeframe, the standard deviation and minimum values show a possibly interesting period. This could be exploratory behavior, social interaction between bees or a detection fault and will need to be looked into further to determine the exact cause. It must be noted that the values for these statistics often fall down to zero, and then go back to their previous value. This is

4 Arenatest

because for those frames the mite detection was lost, causing the computed metrics to fall back to zero. Through this 'noise' trends can still be seen in the data however.

Before discussing the intra-bee distance statistics, a note must be made concerning the erratic shape of these and other graphs in the arenatest analysis. The quick peaks and drops visible in most graphs in the analysis, are a direct result of the amount of bees that are detected rising or falling. This can be due to bees overlapping and being detected as one, but by looking at the captured footage this is only the case in a small part of the data. The vast majority of these cases are due to bees going in and out frame in exploration of the arena. This can be identified when comparing the graphs of the amount of bees detected per frame and, for example, the minimum intra-bee distance.

The intra-bee distance statistics for this timeframe (Figure 4.22) all show a peaking trend. This makes the possibility of the time period at the end of this timeframe being social interaction unlikely (if no detection faults occurred). This could however likely support the exploratory behavior hypothesis.

- The mean intra-bee distance shows a quick rising and falling pattern. This supports the exploratory hypothesis, as it indicates that bees move away from each other and later return.
- The standard deviation follows a similar fast rise and fall, followed by a slower rise. This trend reinforces the interpretation of exploratory behavior: as bees spread out, the variability in their pairwise distances increases, and as they regroup, it decreases.
- Both the minimum and maximum distances also exhibit a peaking trend. The minimum distance shows a shorter peak, while the maximum distance shows a more pronounced peak from around 200 px to 390 px and back. These patterns further support the idea that bees disperse across the arena and then return. While the closest bees stay relatively close together, the maximum distance indicates at least one bee straying far and then returning. Numerous short, high peaks are visible in the minimum distance graph. These often follow a slight drop in the minimum distance, suggesting that two bees come so close together that they temporarily overlap and are detected as one. As a result, the minimum distance temporarily jumps to the next closest pair. This effect can be explored further using the graph showing the number of detected bees.

4 Arenatest

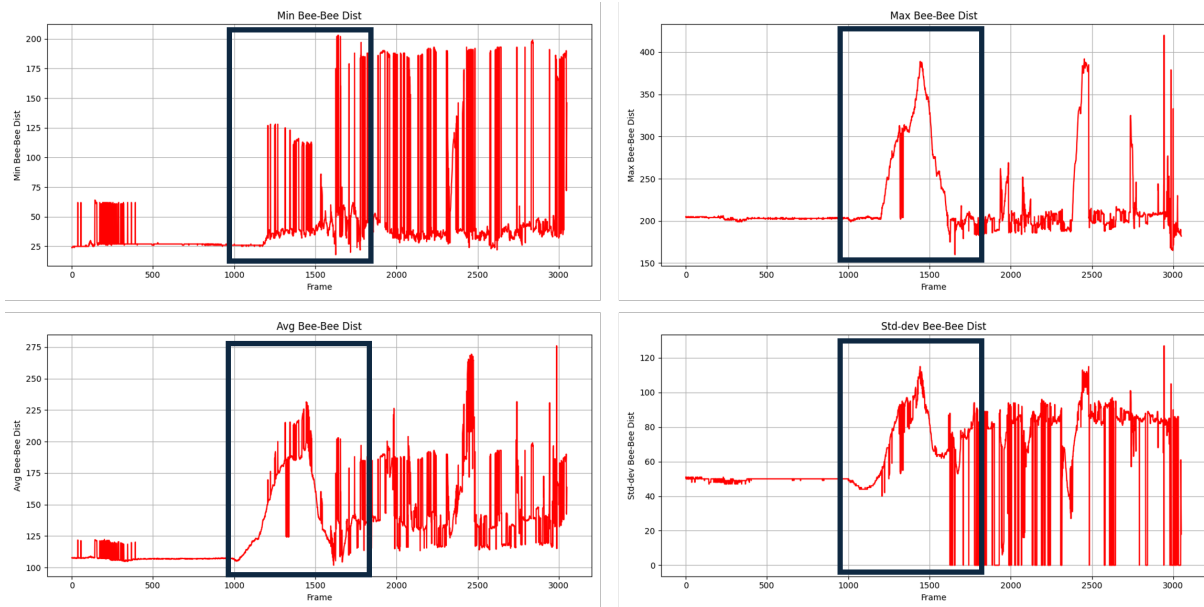


Figure 4.22: Intra-bee distance metric for timeframe 2 of the arenatest performed on 14/03/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.

The peaks observed in the intra-bee distance statistics align with patterns of exploratory behavior, which supports the hypothesis. To further look into the exploratory behavior of the bees, the heatmap of bee positions can be examined.

The heatmap (Figure 4.23) shows that a bee indeed ventured from the top left quarter of the arena, to the bottom and then back to the center. This is indicative of the already suspected exploratory behavior. The other bees near the center can also be observed to have moved from their starting positions, although not to the magnitude of the bee mentioned before. These appeared to have moved towards the center of the arena, which could indicate social interaction between these bees. Finally, the bee in the top left of the arena shows left movement, which could be the static bee that was noticed during the analysis of the maximum bee-to-mite distance. The bees do not seem to exhibit any spatial biases.

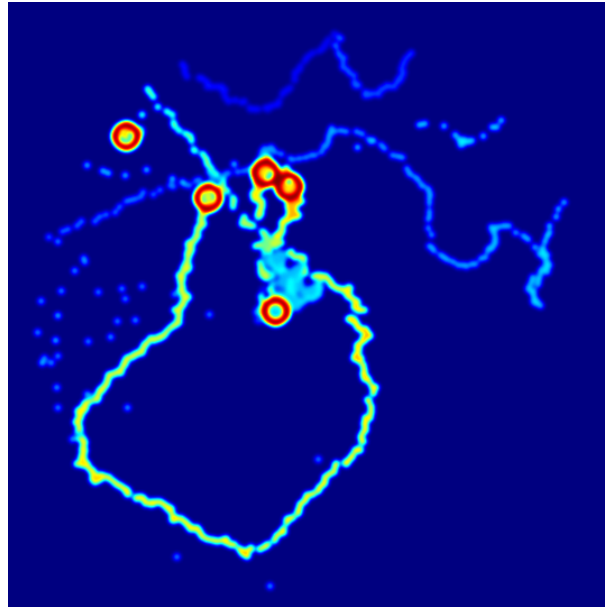


Figure 4.23: Heatmap of bee positions of the arenatest performed on 14/03/2025

The bee-to-center distance statistics for this timeframe (Figure 4.24) show the same trend of peaks values for each of the statistics except for the minimum distance, which stays mostly steady but shows erratic behavior at the end of the timeframe.

- The mean bee-to-center distance increasing and decreasing pattern supports the exploration hypothesis, as it suggests that bees are moving away from, and then back towards the center of the arena.
- The standard deviation rises, then decreases before rising sharply to a peak of $140 \text{ px} \pm 5 \text{ px}$. This indicates that individual bee-to-center distances vary more during exploration and fluctuate over time. The final rise occurs during the same period of interest identified in earlier metrics, reinforcing the exploratory interpretation.
- The minimum distance remains mostly stable at $33 \text{ px} \pm 2 \text{ px}$, but shows erratic behavior toward the end of the timeframe. This suggests that the bee closest to the center remains largely static. As observed in the heatmap, three bees move toward the arena center. This could explain the erratic and downward trending behavior in the minimum distance, possibly due to overlapping detections. Further insight can be gained by examining the number of detected bees.
- The maximum distance peaks quickly, before returning to approximately $218 \text{ px} \pm 5 \text{ px}$, where it remains stable except for a short peak occurring during the previously identified period of interest.

4 Arenatest

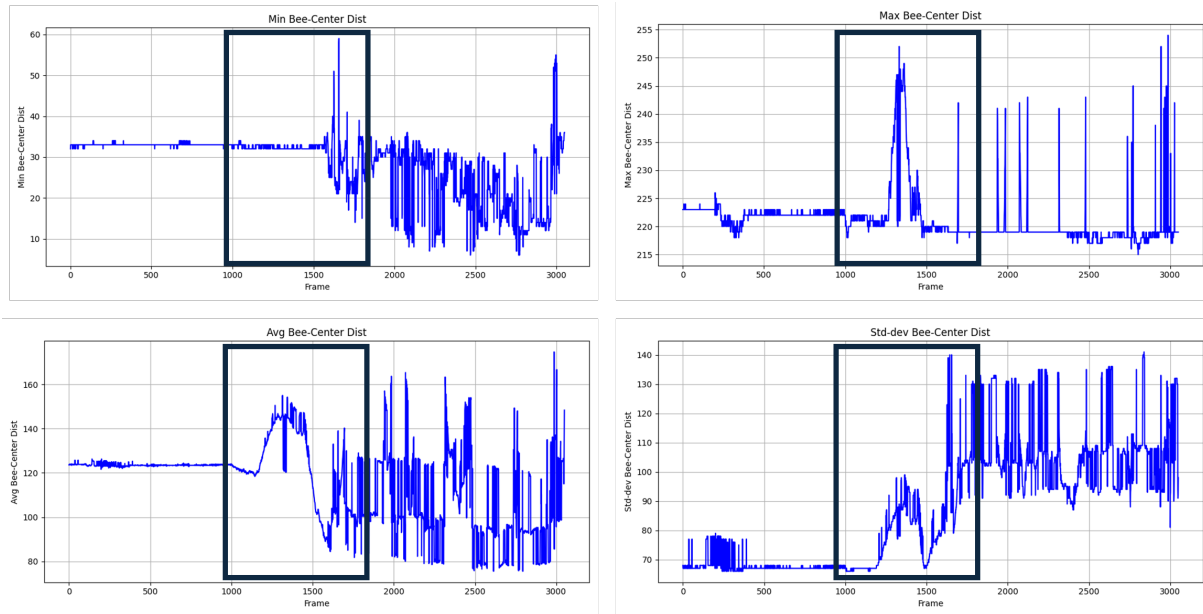


Figure 4.24: Bee-to-center distance metric for timeframe 2 of the arenatest performed on 14/03/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.

The bee-to-center distance statistics confirm the bee behavior and trends from the previous metrics, and again strengthen the exploratory hypothesis. Finally, this metric points towards the need to look into the amount of bees detected metric to be able to explain the erratic behavior in this time period.

The amount of bees detected metric shows a clear downwards trend in the region of interest. This could be an indication that bees are getting close together, and are overlapping. This is supported by the bees showing a trend of moving towards the center in the heatmap and the minimum bee-to-center distance, and the bees getting closer together as observed in the minimum intra-bee distance statistic. When these bees cluster together, and are detected as one, the erratic behavior occurs because of the sudden change in amount of detected bees. This will impact the standard deviation, minimum and maximum values heavily and explains the values in the time period at the end of the frame.

Another possibility is that a bee could be going out of frame, which could be explained by the sudden peak in the in the maximum bee-to-center distance that happens at the same time as the amount of bees detected (Figure 4.25) dropping towards two.

4 Arenatest

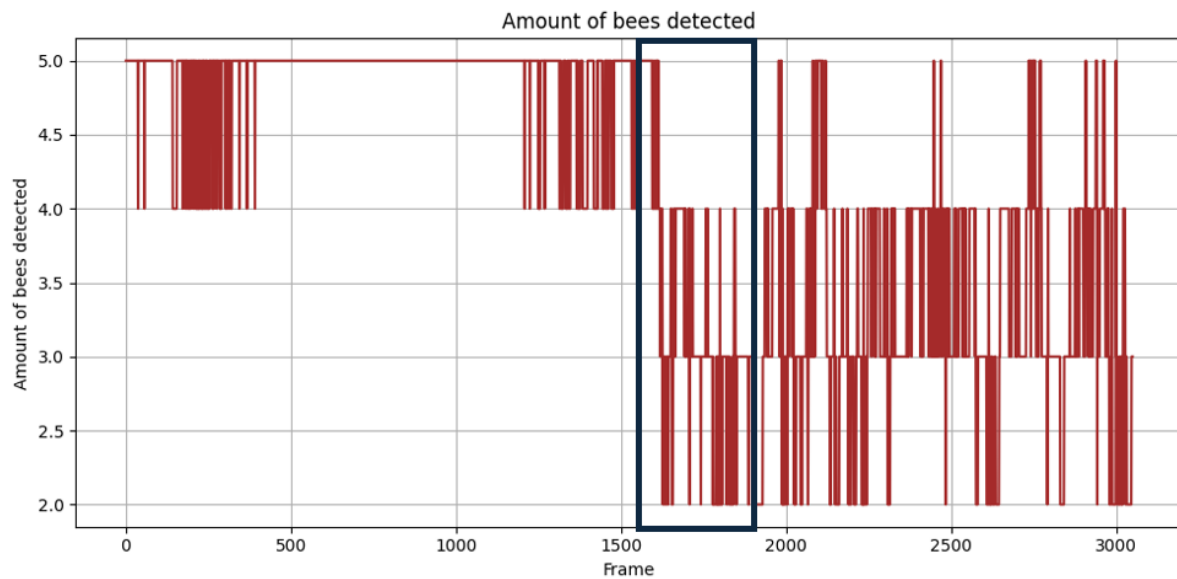


Figure 4.25: Amount of detected bees metric for timeframe 2 of the arenatest performed on 14/03/2025

The hypothesis of the bees showing exploratory behavior is supported through the analysis of these metrics. The average and maximum intra-bee, bee-to-mite and bee-to-center are the biggest indicators of this behavior. Furthermore a timeframe of interest was established at the end of this timeframe, and could be explained by bees clustering in the center. This caused them to overlap and be detected as one, which impacted the standard deviations, minimum and maximum values heavily.

Now the third and final timeframe of this arenatest will be analyzed.

Timeframe 3: frames 1700 - 3400

The analysis of timeframe 3 supports both aspects of the initial hypothesis: ongoing social interaction and continued exploratory behavior among the bees. The bee-to-mite distance statistics remain largely unchanged, indicating sustained disinterest in the mites, while fluctuations are attributable to overlapping bees in the arena center. The intra-bee and bee-to-center distance metrics show occasional sharp peaks, which could be explained by clustering effects and temporary detection errors, rather than actual behavioral shifts. Heatmaps and bee detection counts further support this interpretation, highlighting periods of close proximity and occasional fast movement which is evidence of both social cohesion and exploration. Thus, the behaviors observed at the end of the previous timeframe are seemingly sustained throughout this one.

Conclusions

This arenatest with dead Varroa mites can be divided into three distinct timeframes: 0 - 1000, 1000 - 1700, and 1700 - 3000. The first timeframe (0 -1000 frames) shows almost perfectly stable bee-to-mite, intra-bee and bee-to-center distance metrics. These steady values, corroborated by the heatmap and overlapping bees, support

4 Arenatest

the hypothesis that the bees were recovering from anesthesia and remained nearly perfectly still at their initial positions. No exploratory behavior or interaction with mites is observed during this period.

In the second timeframe (1000 - 1700 frames), changes in distance metrics could be observed. These suggested the beginning of exploratory behavior. The bee-to-mite distance mean and standard deviation begin to fluctuate, while intra-bee distances show peaks. This indicates that bees began moving around the arena, going towards and away from each other and the center. However, the constant minimum bee-to-mite distance and the lack of a significant drop in mean distance suggest no real interest or interaction in the dead mites, which was a first indication of possible non-Varroa resistant behavior. Heatmaps and movement trends strengthen this conclusion, showing most bees exploring their surroundings rather than taking an interest in the mites specifically.

The trends of social interaction in the center of the arena of the bees as well as exploratory behavior, that were shown at the end of the second time frame, continued throughout the third and final timeframe (1700 - 3000 frames).

These findings were supported by looking at the footage of the experiment, the extracted metrics proved able to quantify the arenatest. Overall this test shows a clear change in behavior from a static recovery from the anesthesia to mild exploratory activity, without signs of directed attention towards the dead mites. These findings form a control for comparison of the results of the arenatest with live mites, to assess potential Varroa resistance.

Arenatest with live bees and live mites (performed 25/04/2025)

Looking at the test data, it can again be separated into three different timeframes, based on regions of similar trends or values, to be analyzed:

1. Frames 0–400
2. Frames 400–800
3. Frames 800–1000

It should also be noted that before performing this test, there were issues with the Raspberry Pi blacking out. Due to this reason the beginning of this test was not recorded. This is why no awakening phase will be noticeable during this analysis, which did of course occur.

For the arenatest performed on the 25th of April 2025 it is the first timeframe that will be analyzed, since this is the most interesting. A conclusion about timeframes two and three will again be given, as to provide the necessary context.

Timeframe 1: frames 0 - 400

The bee-to-mite distance statistics for this timeframe (Figure 4.26) show a steady rise with a couple of peaks,

4 Arenatest

followed by a plateau for every statistic except for the standard deviation. This last one drops to zero, and then rises again after about 100 frames. The hypothesis is that the bees are in their exploratory phase during this timeframe. The plateau, or zero values in the standard deviation, indicate a significant loss in the detection of most bees. This could be caused by bees simply going out of frame, or alternatively by a fault in bee detection. The fast peaks during the rise are indicative of fast moving bees or mite, and could support the case of the bees going out of frame in their exploration.

- The mean bee-to-mite distance shows a rise, along with fast peaks, indicating exploratory behavior. This it could also suggest that some interactions occurred at the start of the timeframe, with bees moving away from the mite.
- The standard deviation shows erratic behavior with many quick peaks, until it drops to zero and then rises again around frame 100. The high variability suggests fast exploratory behavior, with bees approaching and retreating from the mite. The drops to zero indicate loss of detection—only one or no bees were detected—possibly due to bees moving in and out of the frame frequently. After the major dropout around frame 200, the standard deviation increases again, suggesting the bees transitioned from being tightly clustered around the mite to a more spread-out, individual behavior pattern, increasing variability in distances.
- The minimum distance trend indicates that bees were initially close to the mite but gradually moved away. The quick drops to zero align with rapid exploratory behavior, where bees pass close to the mite, briefly triggering the minimum value. The plateau, at the same value as the mean distance, suggests that only one bee is being detected during this period. This is something that can be supported by the detected-bees graph.
- The maximum distance rises with several peaks, and a plateau. This again suggests that only one bee was consistently in the frame during this period. The overall rising trend indicates that bees started close to the mite and moved further away over time. The fast peaks and dropouts reflect bees exploring the arena and moving in and out of the frame.

4 Arenatest

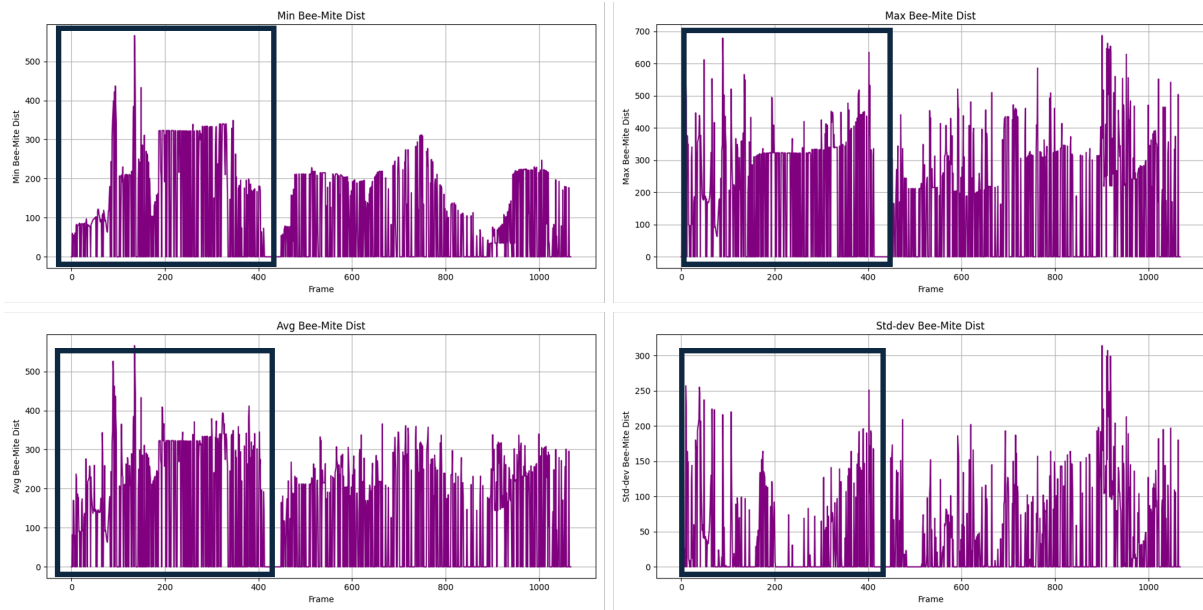


Figure 4.26: Bee-to-mite distance metric for timeframe 1 of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.

In this timeframe the bee-to-mite distance statistics show patterns of exploration, during which detection dropouts occur, likely caused by bees moving out of frame. The upwards trends, together with fast peaks, indicates that bees are moving away from the mite as they explore the arena. It could also show avoidance after interactions early in the timeframe, but the sustained rise supports exploratory behavior. The dropouts can be confirmed through looking at the amount of bees detected graph.

The intra-bee distance statistics for this timeframe (Figure 4.27) show erratic peaks and dropouts, and no constant values for each of the statistics at first glance. The exception is a period where the standard deviation drops to zero for an extended time, indicating that two or less bees were in the frame during this time period. If confirmed, this would support the hypothesis that bees are exploring, and are moving in and out of frame regularly in the process.

- The mean intra-bee distance shows a downward trend, followed by a short span of zero values, before rising again toward the end of the timeframe. This further supports the exploratory behavior hypothesis: the bees are initially clustered closely together, after which all but one exit the frame (resulting in a mean intra-bee distance of zero, since no pairs can be formed). Toward the end, the rising values reflect increasingly individualistic behavior. The dropouts and peaks observed during this period could be attributed to bees moving in and out of frame, rather than faults in mite detection.
- The standard deviation exhibits many peaks and dropouts, similar to the pattern observed in the bee-to-mite distance standard deviation. However, more zero values appear here, since the intra-bee standard

4 Arenatest

deviation drops to zero when two or fewer bees are present in the frame whereas the bee-to-mite standard deviation only does so when one or no bees are detected. This explanation can be verified using the bee detection count graph.

- The minimum and maximum intra-bee distances also show a similar pattern of erratic peaks and dropouts, with an extended period of zero values around frame 200. Afterward, both distances rise quickly toward a peak. This is consistent with previous findings and further supports the idea that bees are moving rapidly toward and away from each other as they explore the arena.

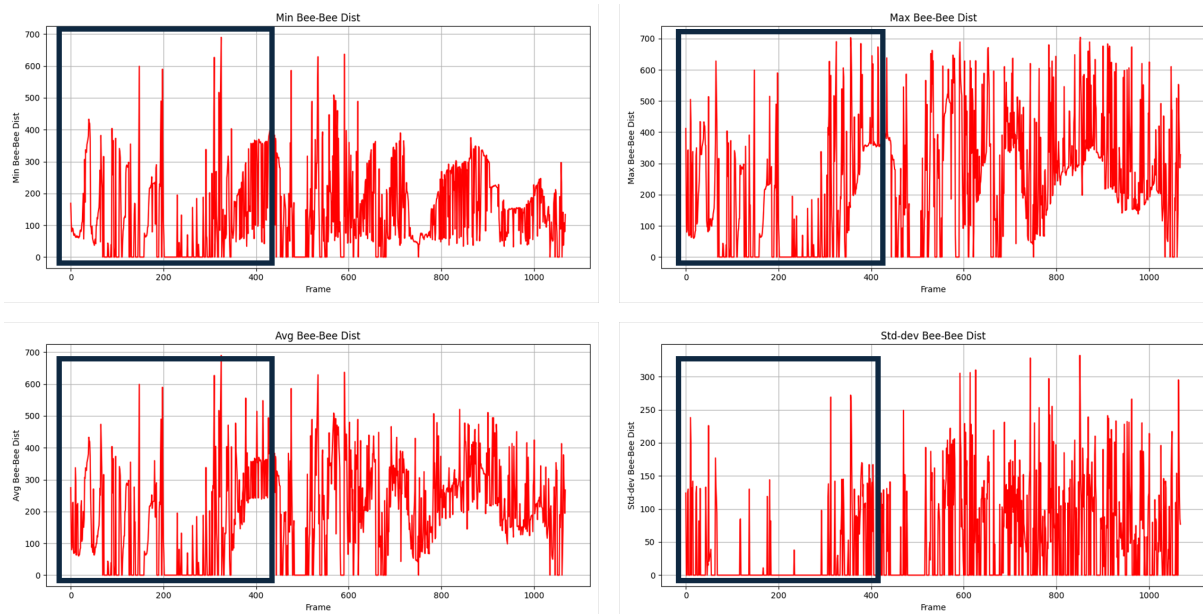


Figure 4.27: Intra-bee distance metric for timeframe 1 of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.

It can be confirmed that the constant intra-bee distance statistics are consistent with bees moving fast throughout the arena, and rapidly going towards and away from each other during this process of this exploration. This further supports the hypothesis. To strengthen the claim, the heatmap of bee positions can be examined.

The heatmap (Figure 4.28) shows three starting positions, this indicates that at the start of the test two bees were already out of frame. Furthermore no clear trajectories start from these starting points, meaning the bees stood static and then started to explore the arena quickly. In the center of the heatmap a hotspot is visible, where a bee remained static for a while. These are also visible at the left edge of the arena. These findings are in line with those of the previously analyzed metrics, and the hypothesis. The bee-to-center and mite-to-center distance metrics will now be analyzed to confirm the edge bias, and the hotspot at the center of the arena.

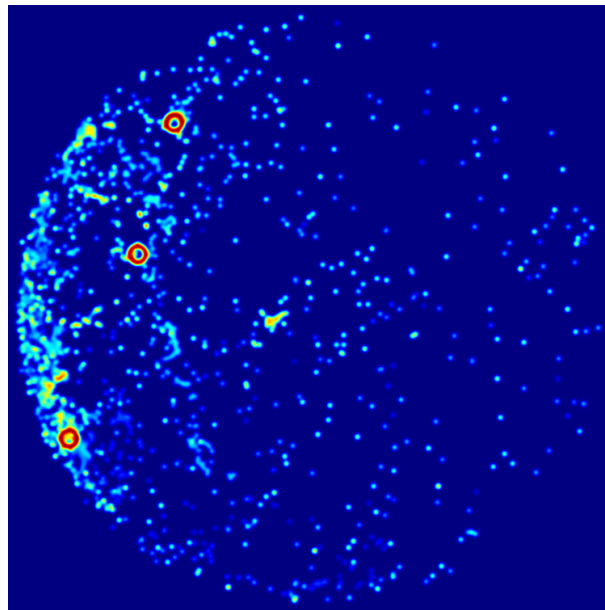


Figure 4.28: Heatmap of bee positions of the arenatest performed on 25/04/2025

The bee-to-center distance statistics (Figure 4.29) confirm the findings from the heatmap and the other metrics.

Note: since for this analysis the bee-to-center distance statistics were used, the bee-to-edge distance statistics do not have to be analyzed since these give the same info in a circular arena setup and this would only be confusing. To be complete, the bee-to-edge distance statistics will be shown once (Figure 4.30).

- The mean bee-to-center distance starts high, and dip with significant variability. This is consistent with the suspected edge bias during this timeframe. The dip explains the hotspot in the center of the arena, where a bee moved quickly into the center and remained there for some time before moving away again. The quick peaks and dropouts support the findings of the previous metrics.
- The standard deviation exhibits erratic behavior with many quick peaks to different values, with a notable peak occurring around the time a bee is suspected to be in the center. Afterward, the value drops to zero and rises again toward the end of the timeframe after approximately 100 frames. The high standard deviation peak during the period when the bee is in the center shows a large spread of the bees. The erratic behavior and dropouts around frame 200 align with the patterns observed in the intra-bee distance statistics.
- The minimum and maximum distances follow a similar trend, starting high, then dropping to about 50 px (± 5 px) before rising again and stabilizing. This supports the findings from the mean distances. At times, the minimum and maximum distances are the same, indicating that only one bee is detected momentarily during this timeframe.

4 Arenatest

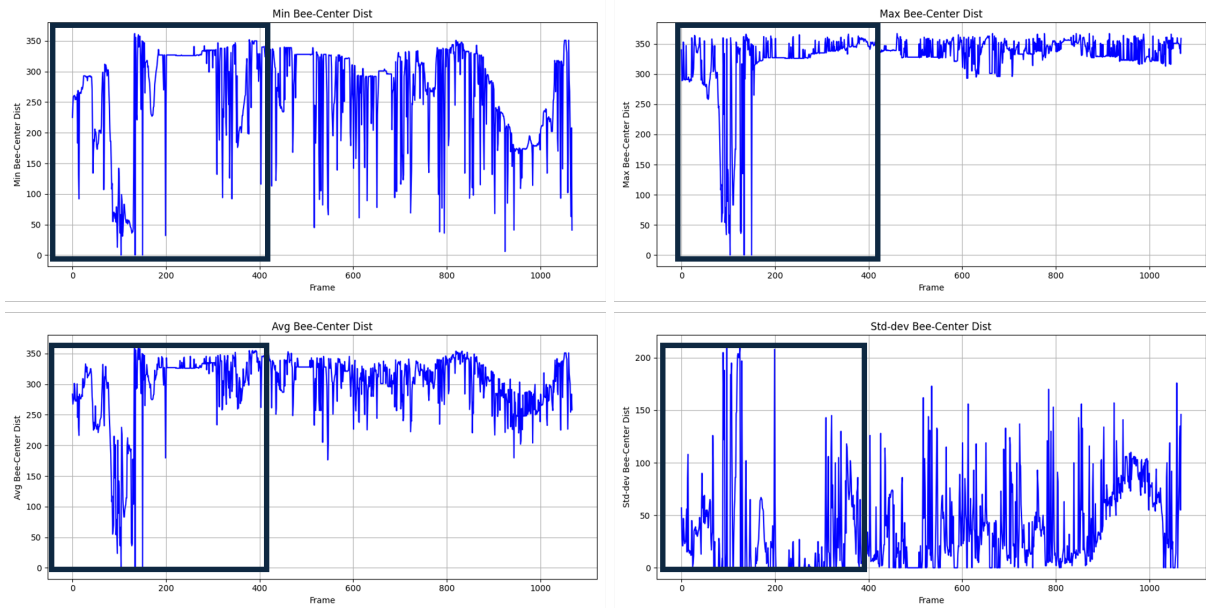


Figure 4.29: Bee-to-center distance metric for timeframe 1 of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.

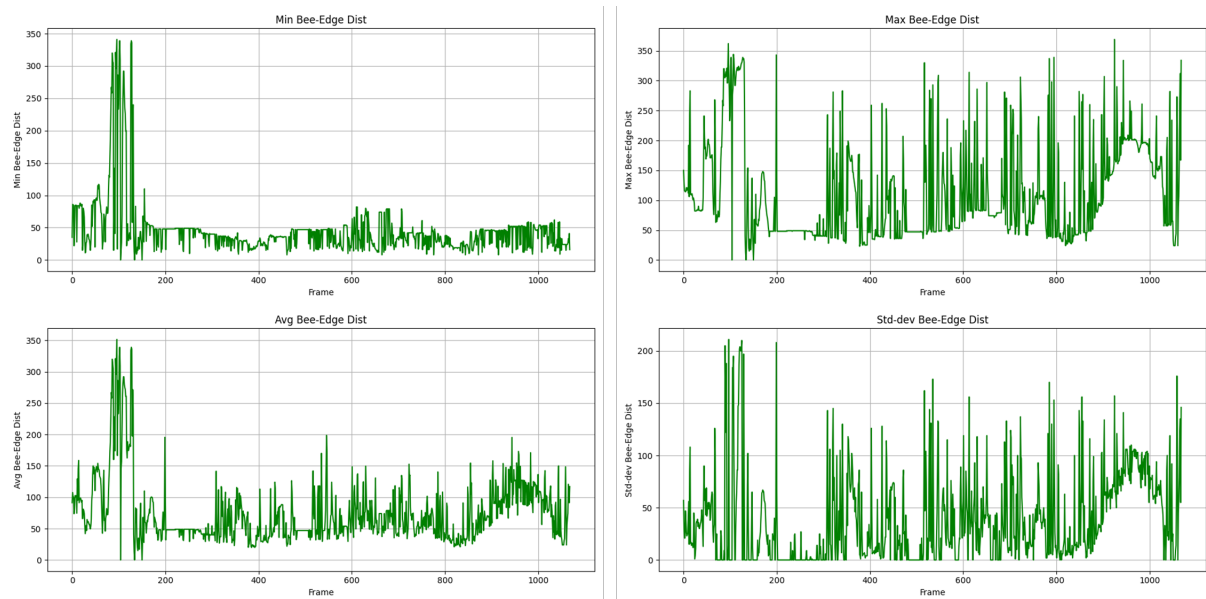


Figure 4.30: Bee-to-edge distance metric of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.

The mite-to-center distance statistics (Figure 4.31) support the findings from the heatmap and the other metrics. As expected, the mean, minimum and maximum values all have the same shape. Due to this arenatest only containing one mite. The standard deviation should be perfectly zero throughout the whole experiment in the case

4 Arenatest

of perfect detection accuracy. However when a false positive mite detection occurs multiple mites are supposedly detected, which gives diverging values. The standard deviation of the mite-to-center distance thus gives an indication of when false positive mite detections occur, which is valuable information.

Metrics show that the mite was first located at the edge of the arena, during the time the interaction with a bee occurred. This suggests that the interaction took place with the bee that was close to the edge at the start of the timeframe, which, as observed in the bee-to-center distance statistics, then quickly went towards the center of the arena. After the bee relocated to the center, the mite-to-center statistics show a fast jump. This could be explained as a bee blowing the mite closer to the center, by narrowly flying by it. This was observed to happen sometimes during this, and other performed arenatests.

After this the mite laid still for about 200 frames. Then the graph shows a slight drop, and a steady rise. This indicates that the mite was moved to a nearby spot by a bee flying by, and then walked towards the edge of the frame. After this a big peak happens, which could mean the mite was again moved, this time closer towards the arena walls.

The findings of this metric support the interaction happening between the bee and mite, and show that bees are possibly passing by the mite close enough to move it several times.

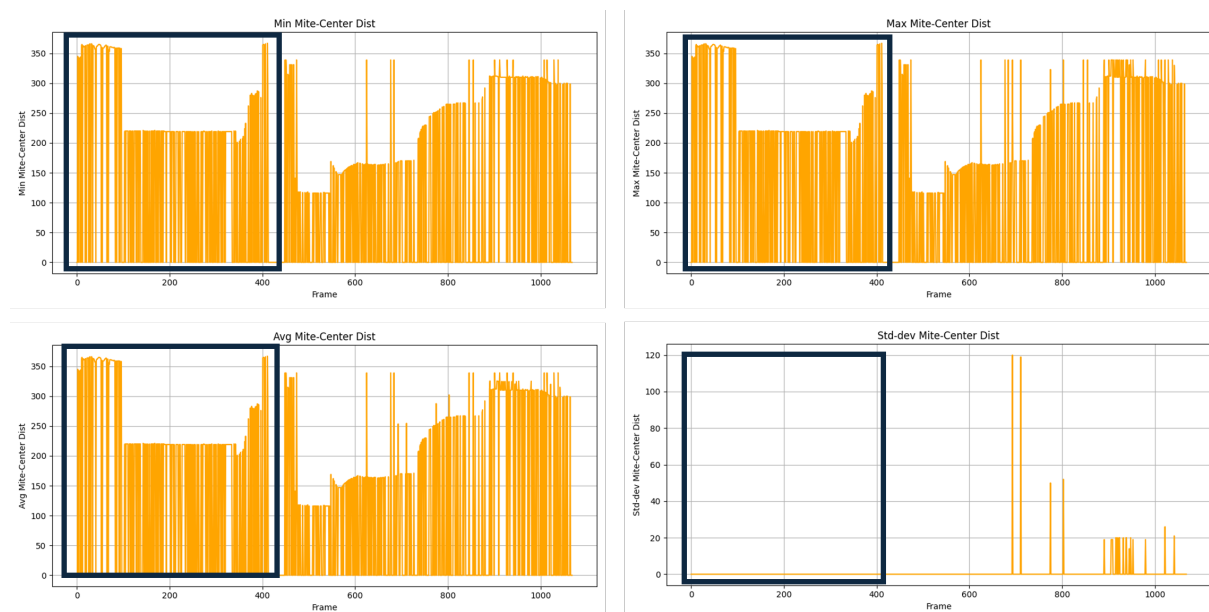


Figure 4.31: Mite-to-center distance metric for timeframe 1 of the arenatest performed on 25/04/2025. From top to bottom, left to right: (a) Minimum. (b) Maximum. (c) Mean. (d) Standard deviation.

Finally the graphs of the bee and mite interactions and of the amount of bees detected metrics will be analyzed.

The graph showing the amount of bees detected (Figure 4.32) matches up with the suggested values. It starts at

4 Arenatest

three, as suggested by the heatmap, jumps a lot between two and three, as suggested by the standard deviation of the intra-bee distance metric. Then it jumps between one and two for a time period, with a couple dropouts to zero and stays stable at one for a while around frame 200. This is in line with the findings from the intra-bee and bee-to-center metrics, and the standard deviation of the bee-to-mite distance metric.

The graph showing the amount of frames with potential bee and mite interactions (Figure 4.32) strengthen the hypothesis suggested by the bee-to-mite distance statistics of there being interaction at the start of the time-frame, and then no more as bees moved away from the mite.

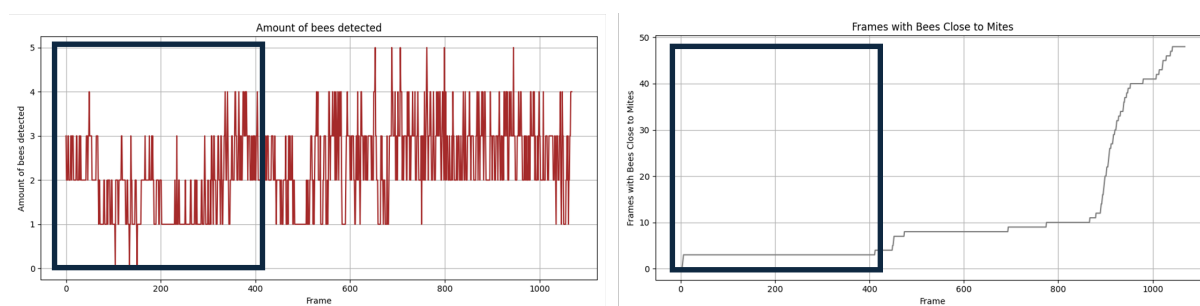


Figure 4.32: Amount of bees detected & amount of frame where bees are close to mites for timeframe 1 of the arenatest performed on 25/04/2025

The hypothesis of exploratory behavior, in which moved fast and went in and out frame often, during this time-frame was strengthened. A bias for the left side of the arena is established, as well as possible mite avoiding behavior in the bee moving away after its interaction with the mite ended. Although the fact that bees are passing by the mite close enough to move it several times could suggest non-Varroa resistant behavior. These will be kept in mind during the analysis of the second time frame.

Timeframe 2: frames 400 - 800

The data from this timeframe highlights a continuation of exploratory behavior among the bees, marked by frequent movement, changes in detected bee count and dynamic spatial distributions. A brief period of bee-mite interaction occurred early on, followed by a general avoidance pattern, although this avoidance does not appear intentional or consistent enough to indicate Varroa resistance. The bee-to-mite distance statistics, supported by the mite-to-center data, reveal that early interaction stemmed from the mite being hidden beneath a stationary bee suggesting no active recognition of the mite. Throughout the timeframe, intra-bee and bee-to-center distance metrics strongly support extensive exploration, a strong bias toward the arena's left and edge regions and intermittent clustering behaviors. The heatmap and detection graphs corroborate these findings, while the mite's own movement patterns, including a shift toward the arena edge, might suggest host-seeking activity. Overall, this timeframe offers no conclusive evidence of mite-targeted behaviors, but rather reinforces a pattern of general exploration with incidental proximity events.

4 Arenatest

Timeframe 3: frames 800 - 1000

This timeframe presents further evidence of continued exploratory behavior and increased bee-mite interactions, particularly around frame 900. Bee-to-mite distance statistics indicate two notable periods of proximity, which may suggest interaction but do not definitively confirm Varroa-resistant behavior. Instead, these interactions are more plausibly incidental, consistent with bees exploring the arena and occasionally approaching the mites. Intra-bee distance statistics and heatmap data indicate signs of social behavior among bees toward the end of the timeframe, with some clustering observable. The concurrent movement patterns of both bees and mites, particularly the alignment of their heatmap hotspots, strongly support the hypothesis that the mite is exhibiting host-seeking behavior. However, the bees' lack of consistent avoidance or reactive behavior diminishes the case for active Varroa recognition. Variability in detected bee numbers and erratic motion patterns further underscore the dynamic, non-directed nature of the observed behaviors. Overall, the findings suggest increased interaction and support host-seeking behavior in the mite, but no conclusions can be drawn about possible Varroa resistance in the bees.

Conclusions

Looking at frames 0 - 400, all bee-to-mite distance statistics rise from their initial low values. Followed by sharp peaks and a standard-deviation dropout to zero around frame 100 - 200, indicating bees rapidly exploring the arena. During this process they intermittently leave the frame. Intra-bee distances support this exploratory behavior. A bias from the bees for the left side of the arena for was established using the heatmap.

In the second timeframe (frames 400 - 800), a brief period of zero values in the bee-mite distances indicates the mite briefly went undetected (with the footage, it was found to be hiding under a stationary bee). After this bee-to-mite and intra-bee distances again display quick peaks, drops and plateaus. Around frame 750 a dip in intra-bee distances and a matching drop in the detected bee count reveal a momentary social cluster away from the mite. A bias from the bees for the arena edges for was established using the heatmap.

In frames 800 - 1000, mean and minimum bee-to-mite distances dip sharply around frame 900 while the maximum peaks signaling a close bee and mite encounter followed by rapid separation. The intra-bee distances briefly decline, suggesting bees clustered again before resuming exploratory dispersion. Comparing the heatmaps of bee and mite positions revealed that mites moved toward regions where bees the bees were, establishing a hypothesis of potential host-seeking behavior. This hypothesis was supported by the data.

These findings matched closely to what can be seen in the footage of the arenatest. The bees showed no signs of noticing the Varroa mite, or the threat it presented. The behaviors that were labeled as potentially being Varroa resistant behavior, seem to have been coincidences. This was also the conclusion from the analysis of the metrics extracted from the experiment.

Conclusions about the arenatest analysis

4 Arenatest

Both the control test with dead mites and the real arenatest with live mites can be split into three phases (although the first phase is missing in the real experiment). Comparison of their distance metrics highlights the differences in behavior between live and dead mites.

In the dead-mite control, bee-mite distances, intra-bee distances and bee-center distances remain essentially flat at first (the awakening phase) and later only show mild exploration. There were practically no real bee and mite interactions with the dead mites. Heatmaps show bees lingering where they were released, with occasional random clustering but no directed approach or avoidance of the mites.

In the live-mite trial multiple bee and mite interactions occur. Additionally, the intra-bee distances indicate possible evidence of social clustering and dispersal, as they fan out to explore. Spatially, bees show a strong left-side and edge bias. The mite itself exhibits host-seeking by moving towards the bees, as suggested by the overlap of both their heatmaps.

When comparing the second half of the dead-mite control arenatest (to account for the missing awakening phase) to the real arenatest, similar patterns show up. Exploratory behavior is strongly present in both experiments. In the case of the control no bias is shown towards a specific region of the arena, this differs from the real arenatest. There, a clear bias is visible for the left side of the arena, and more specifically the edges. This is however no indication of Varroa resistant behavior, since the mite also exclusively was located in this region. In the case of Varroa resistant bees avoiding behavior would have been a much expected reaction. Another difference between the two tests is the amount of interactions between bees and mites. The dead-mite control showed no meaningful interactions, while the real test showed multiple cases. Although this is also not a sign of Varroa resistant behavior, since it was established in the analysis of the real arenatest that the mite showed signs of host-seeking behavior. The interactions can subsequently be addressed to the mite looking for a host to attach to. This is strongly supported by the footage of this test, where bees show no recognition of the mite even when they are right next to it. Finally the tests both showed social interactions between bees.

In conclusion, the real test shows differences when compared to the control, but also a lot of similarities. Moreover, the differences can all potentially be addressed to the live mite behavior or are simply no indication of Varroa resistance. The similarities, on the other hand, are very clear. In both test bees show continued exploratory behavior and social interactions between bees. Avoidance behavior could be present, but can not be confirmed through this analysis alone. It can thus be concluded that, using this quantification, the hypothesis about Varroa resistance can be neither confirmed nor denied. Further steps must be taken to definitively prove or disprove Varroa resistance in the bees. By doing an arenatest with bees that are proven to be Varroa resistant and comparing these results more conclusions could be drawn about the true nature of the phenomenon that were observed in this analysis.

Finally, the conclusions drawn about the behaviors and movement patterns from the analysis of the metrics extracts from these tests matched closely to what can be seen in the footage. This proves that the arenatest can

be quantified and analyzed by these distance, detection and interaction based metrics.

4.3.4 Data storage and -access

The following data is extracted using the arenatest detection and classification algorithm:

- Folder containing the captured frames (in PNG format), without detection and classification.
- Folder containing the captured frames (in PNG format), with detection and classification.
- Heatmap of bee locations (in PNG format).
- CSV file containing all the extracted metrics.

This data is automatically saved to the home screen of the Raspberry Pi after the code is done executing. The folders containing the frames were first turned into videos (in WebM format), using Python code run in the Google Colab environment. After this, all the data is collected into a folder that is named by the format 'DDMMYYYY' and placed into the 'arenatestdata' folder on the Desktop of the Raspberry Pi. It can then be retrieved by the server to be displayed on the website, where the graphs are generated automatically using the JavaScript frontend code. The raw data will stay accessible on the Raspberry Pi, as well as in Appendix A ⁵.

4.4 Limitations, challenges and future perspectives

The biggest limitation of the arenatest setup and monitoring system was the camera quality. The Raspberry Pi camera module V2.1 can capture images at high resolution, but has the limitation of only capturing objects sharp from 0.5 m of distance. Using this full distance was crucial to be able to see the arena (and primarily the mites) as clear as possible, but also resulted in less pixels per bee or mite. This of course made the mite detection a challenge, since they were essentially only visible as a small collection of grayish pixels on the images. Good pre-processing became a very important and necessary step in order to limit the amount of false positives in mite detection. The full resolution range was additionally not available, because with higher resolutions the framerate dropped noticeably fast. Capturing frames fast enough is of course also important in being able to accurately quantify the arenatest. A phone camera was used to capture the arenatest of the 25th of April, as to compare if this gave better image quality. This was the case, but only very slightly so because of the need to zoom in considerably from the height of the camera stand. If the camera could have been placed lower the difference would have been greater.

The largest improvement on the monitoring method would be the use of machine learning, and specifically the use of convolutional neural networks (CNN) for bee and mite segmentation. This would greatly improve the

⁵GitHub repository for arenatest, pollen monitoring and website data

4 Arenatest

classification accuracy, and possibly near almost eliminate the occurrence of false positives in both bee and mite classification. The choice not to use CNN's was a deliberate one, because they also bring a few downsides. The labeling and training process takes a long time, and a lot of data would have been needed to be able to create a good model. A CNN would also need considerably more computing power and resources of the Raspberry Pi to run, and would have possibly limited the framerate of the camera even more. Another improvement would be the implementation of bee and mite tracking algorithms, and being able to give ID's to the subjects to be able to follow them on an individual level. This approach allows metrics to be extracted for each individual bee or mite, enabling a more detailed analysis of their respective behaviors. This was attempted but was quickly discontinued. Issues with false positives in detection, bees overlapping and being detected as one (not being distinguishable by the algorithm anymore) and bees going out of frame made this impossible on the current setup. In combination with the better detection accuracy from a CNN, this approach could be very interesting and gives great future perspectives in the use of computer vision for monitoring arenatests. A final improvement to the setup would be the use of an additional camera that monitors the underside of the arena. Since mites mostly attach to the underside of bees, sitting underneath their wings, this could provide the ability to detect mites on bees during the test.

An interesting future experiment that could be performed is doing an arenatest in the dark, or using red light since the bees don't recognize this. This would allow the use of young worker bees, that are only used to being in the dark inside the hive and could give different results. This does bring challenges in detection and big adjustments will need to be made to the setup, especially the camera used. Infra-red cameras could be used to capture images of the bees, but the mites would not be visible on these images since they are cold blooded (E. Danneels, personal communication, December 17, 2024).

Finally, using the algorithms developed in this chapter, large scale arenatests could be performed to be able to compare different honeybee hives. Bees from different regions could be compared and give insights into the differences in their behavior in the presence of the *Varroa destructor*.

4.5 Conclusion

In this chapter, a computer vision-based approach was developed to automate and quantify the arenatest, aiming to study honeybee behavior in the presence of the *Varroa destructor*. The process started with a simulation built in Python, which helped in the creation of the first version of the detection and analysis algorithms. From there the code was gradually tested and improved, starting with paper cutout of bees, then using dead bees and mites and finally moving to live subjects. This step-by-step process made it possible to fine-tune the algorithm in a controlled way and adapt it to real-world conditions such as noise.

Throughout the development, the focus was on using traditional computer vision techniques instead of machine learning. This decision was made to keep the computational load light enough to run on a Raspberry Pi. The visual differences between bees and mites made this possible. With the final algorithm, several key behavioral metrics

4 Arenatest

were calculated. These included distances between bees and mites, distances among bees and their positions within the arena. These metrics helped to objectively describe how bees moved, interacted with each other, and responded to mites.

When applied to both a control test (with dead mites) and a real test (with a live mite), the metrics showed clear differences in behavior. The bees did interact more with the live mite, but mostly in a passive way. No definitive conclusions could be drawn regarding potential Varroa-resistant behaviors, such as grooming or avoidance. Instead, the differences could be explained by the mite actively moving toward the bees. In both tests, the bees showed similar social and exploratory behaviors.

Although the system worked well, it had some limitations. The camera quality made mite detection difficult, and issues like overlapping bees and occlusions made individual tracking impossible. Still, the method proved to be effective in quantifying arenatest behavior and matched well with what was visible in the test footage. Looking forward, better tracking or extra cameras could improve accuracy and allow for even more detailed analysis. With these improvements, the system could be used to run large-scale tests comparing bee colonies (including ones from different regions), helping to study Varroa resistance in a more systematic way.

5

Pollen monitoring

In this chapter, a computer vision based approach for pollen counting is developed. The goal was to detect and annotate pollen loads on bees entering a hive, and estimate the pollen volume (in mm^3). This gives insights into the hive health, flower preference in bees and allows a beehives pollen collection abilities to be quantified and compared. First, the experimental setup is discussed, including the rationale behind the equipment choices, the constraints involved, and the assembly process. The heart of this setup is a custom built observation tunnel. An important step in the creation of the pollen segmentation algorithm was data collection. The code and reasoning behind the pollen segmentation and counting algorithm are explained in detail. The approach is based on background subtraction and thresholding to detect bees, and color based masking to segment pollen. The pollen are then color corrected, as to detect their real color accurately.

The goal of this chapter is to create a proof of concept, that shows computer vision is able to detect and count pollen with a good setup. This concept can be expanded upon, to be able to process the data live, and allow pollen entering the beehive to be monitored in real-time.

All data, code and results for this chapter can be found in Appendix B¹.

5.1 System setup and data collection

The main part of the setup for this experiment is the custom build bee beehive observation tunnel (see Figure 5.1). There were a few constraints and necessities that needed to be considered for this structure. First, it needed to provide a surface for the bees to land on, since this tunnel replaces the flying board of the beehive. This surface needed to be big enough to allow all bee traffic to flow in and out of the beehive easily. The shape of this surface also plays an important role in allowing bees to find the hive entrance. This is necessary, because when an adaptation is made to the beehive, the bees need some time to get used to this change. By choosing the shape in such a way that guides bees to the entrance, this process is accelerated. The shape used in the design is a long

¹GitHub repository for arenatest, pollen monitoring and website data

5 Pollen monitoring

rectangle that is placed perpendicular to the beehive entrance. This gives the bees a 'landing strip' that points directly at the entrance, without too much room for them to stray from the desired path.

Next, this structure needs to provide an entrance to the hive through which the bees can enter easily, and they can be captured by the camera. This was achieved by a small tunnel that mounts into the beehive entrance, which is opened to the exact width of the tunnel. This forces all bees to pass through the tunnel in entering the beehive, and get captured by the camera. The tunnel needed to be small enough so that it could be captured fully by the camera, but large enough so no congestion is formed when bees try to enter or leave the hive. The height was also important, since bees crawl on the ceiling and walls of this tunnel when trying to find their hive entrance. To make sure all bees appear as the same size in the images, and minimal occlusions occur. To be able to distinguish the bees on the images more easily, the ceiling of the tunnel was painted white. To allow the bees entering and leaving the hive to be captured by a camera the tunnel floor was made out of plexiglass.



Figure 5.1: Observation tunnel set up in front of beehive

The camera captures the bees from underneath, lying in a box (of which the top is the flying board on which the bees can land). The first setup for this experiment used a Raspberry Pi 5 and a Raspberry Pi camera module V2.1 to capture the images. A hole was drilled in the side of this box to be able to provide the Raspberry Pi with power. For this camera to capture the sharpest images, it needs to be at a distance of approximately 0.5 m from the bee it is imaging which of course is not possible in this setup. A compromise must also be made here: if the camera is placed further from the bees, the images, while sharper, contain less pixels per bee. If the camera is placed closer, the images are less sharp but there will be more pixels per captured bee. The ideal distance for this setup was determined to be about ten to fifteen centimeters through testing. For the box, a wine crate was used, which naturally had a height of ten centimeters in this orientation. However, after collecting data, it was determined

5 Pollen monitoring

that the images captured through this setup were unfortunately not good enough to be able to segment out the pollen. Because of this, a different approach was adopted, and a smartphone camera was used. This camera was capable of capturing images of sufficient quality to enable effective pollen segmentation.

The smartphone was placed inside the box, with its back camera pointing up. The flash was turned on, to provide enough light for the images. For optimal image clarity, the camera was manually focused at a distance of ten centimeters. Finally, to line up the tunnel with the beehive entrance, the box was placed on legs which were made to length. This allowed the observation tunnel to be installed in front of the hive in a quick and easy fashion, which was necessary as to not disturb the bees too much. The bees would react by exiting the hive and observing and exploring the changes made. This would disrupt their normal behavior and a time period will be needed to restore this. By minimizing the magnitude of this disruption, the time period is shortened and their normal behavior can be observed. Because bees have a tendency to try to squeeze themselves through every possible orifice, small pieces of foam were put between the small unavoidable gaps between the beehive opening slide and the observation tunnel.

Finally, because this is a proof of concept only a region of the total footage was analyzed. As to not have too many occlusions by bees on the walls of the tunnel, two lines were drawn on the plexiglass and the frames were cropped to be between these lines.

This setup was used to collect data for a beehive located in the bee hall of Honeybee Valley. Data was collected on the hives indicated on the images in Figure 5.2.

5 Pollen monitoring

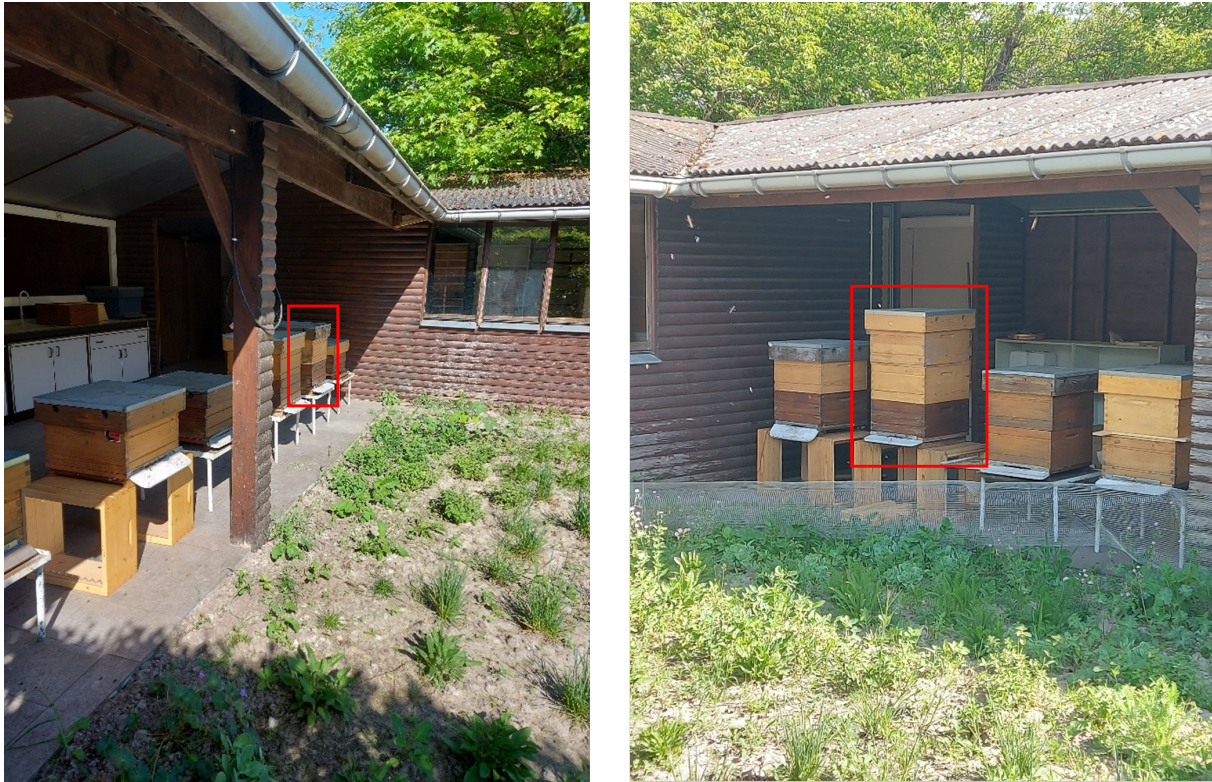


Figure 5.2: Beehives in the bee hall on which data was collected

Data collection was an important first step to be able to create an accurate pollen detection algorithm, and for the final setup, where data would be collected continuously.

For the first setup, a Python program was made that would start to capture images from the moment the Raspberry Pi was provided with power. This allowed the headless (using the Raspberry Pi without a screen) collection of data, which is necessary in the bee hall. Power was provided through an expansion cord. Ideally, the bees would first be given time to get used to the change to their hive, but providing power to the Raspberry Pi after the placement of the observation tunnel without disturbing the bees again was impossible. This is why the data collection is started immediately, but is done long enough as to be able to capture the bees' normal behavior. Data was collected for about an hour each time to ensure enough data storage was available. In the live processing setup, this would of course not be an issue. In the final setup, this was done in a similar way, by putting the smartphone in the box while recording and then sliding it in front of the beehive. To extract the data, the observation tunnel needed to be removed so the Raspberry Pi or smartphone could be accessed. In a live setup there would be no need for this, since the data would be processed in real-time without the need to extract it from the recording device.

Figure 5.3 shows a technical drawing of the observation tunnel.

5 Pollen monitoring

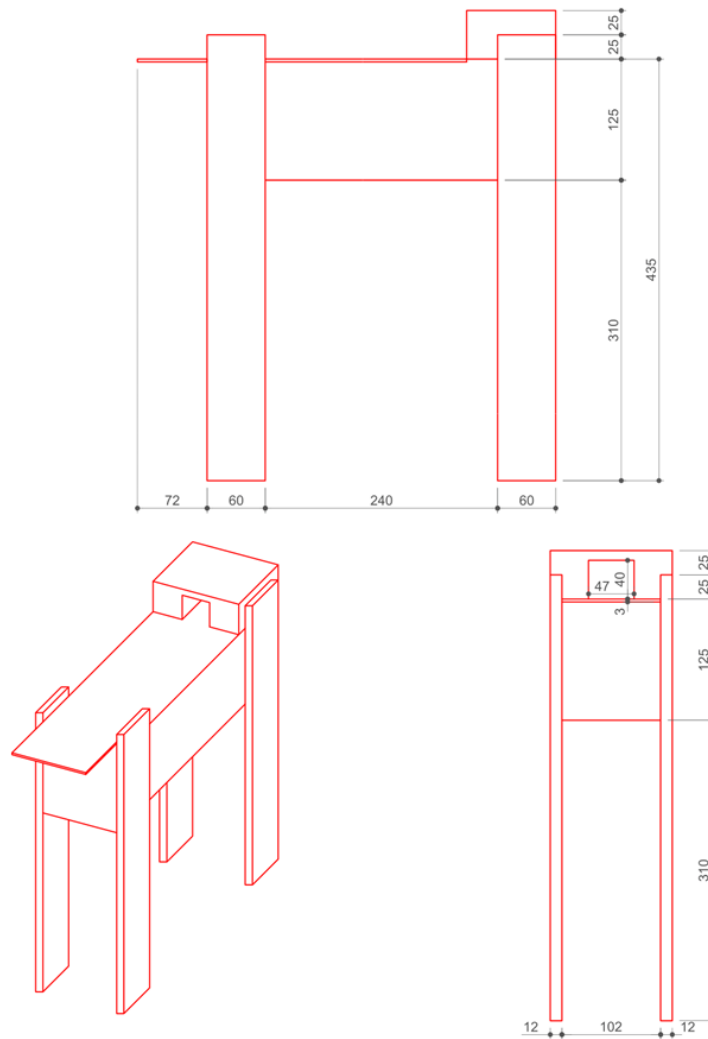


Figure 5.3: Observation tunnel technical drawing

5.2 Detection and counting algorithm

The pollen detection and counting algorithm was developed in the Google Colab cloud environment. The captured raw footage of bees entering the hive was uploaded here so it can be processed.

Each frame from this footage is read and processed one at a time. First, the frame is resized to be 540 x 960, and cropped. Resizing reduces memory usage and speeds up processing, which would be necessary for real-time applications. This also makes sure that the input for the processing pipeline is standardized. Then, this resized frame is converted to gray, which is a necessary step for some computer vision techniques. Next, background subtraction is done.

Background subtraction is a technique used to separate (moving) foreground objects from a static background.

5 Pollen monitoring

To do this, the background must first be defined, to which each frame can then be compared. Next, changes are detected: if a pixel has changed significantly, it's considered foreground. Finally, a mask is generated where foreground pixels are white, and background pixels are black. Two approaches were used to implement this technique. The first one used OpenCV's built-in background subtraction algorithm. This takes two arguments: history and varThreshold. History is the number of frames used to build the background model. A larger value means slower adaptation to changes. VarThreshold is a threshold on the distance between two pixels that are in the same location in the background model and frame. This determines how different a pixel must be from the background model to be considered foreground, where lower values result in a higher sensitivity. The problem with this approach is that bees would sometimes stay still in the tunnel for a period of time, which caused them to be recognized as background when they obviously were not. This leads to the adoption of another approach: frame differencing. In frame differencing, the background remains static and is not updated dynamically. The absolute difference between the frame and the background image is calculated, and similarly pixels with significant change are considered foreground pixels. This approach is simpler, but more accurate since the background is static throughout all frames and bees that stay still, are now able to be correctly recognized as foreground. With a threshold, a foreground mask can be created, which allows the processing of the recognizing of bees in the frame. This process is explained through the pseudo-code shown in Figure 5.4.

```
# Step 1: Load the background reference frame
background = read image from frame_paths[0]

# Step 2: Crop the region of interest from the resized background
background_cropped = crop region from background

# Step 3: Process each frame for background subtraction
for each frame_path in frame_paths:
    frame = read image from frame_path

    # Step 4: Crop the current frame
    frame_cropped = crop frame using (crop_y1:crop_y2, crop_x1:crop_x2)

    # Step 5: Compute absolute difference between current frame and background
    diff = absolute difference between frame_cropped and background_cropped

    # Step 6: Threshold the difference to get foreground mask
    fg_mask = apply binary threshold to diff
```

Figure 5.4: Pseudo-code that illustrates the background subtraction process

This is done through contour detection. The contours are first checked for their inner area. Those with an inner area that is too small to represent a bee contour, are disregarded as noise. The contours that pass this check are then used to define a region of interest (ROI) and crop out the bee from the frame. Because bees carry pollen on their back legs, only the lower 60% of the ROI is kept. On this image, the pollen segmentation is done. This segmentation is color-based. This approach assumes that researchers or beekeepers intending to implement the system have prior knowledge of the pollen colors specific to their region. In the footage captured, two pollen

5 Pollen monitoring

colors were visible: yellow and dark-red. The goal of the pollen segmentation is to create a mask that ideally contains all pixels that belong to pollen in the ROI. For this, a mask will be made for each available pollen color, that segments all pollen from this color. Finally, these masks are combined using a bitwise 'or' operation, and a combined pollen mask is created.

For the yellow pollen, segmentation is done with the use of the LAB color space. LAB shows color as three values: L for perceptual lightness, and A and B for the four unique colors of human vision: red, green, blue and yellow. The LAB mask is created by using the B-channel, because this represents the blue–yellow color opponent axis. Since no blue appears in the footage, this can be used to segment the yellows. A mask is created by thresholding the B-channel. This mask only keeps the yellows in the ROI. The process of yellow pollen segmentation is shown in Figure 5.5. For dark-red pollen, LAB is used again. This time, the A-channel is thresholded to create a mask. This channel represents the green to red axis. Since no green appears in the footage, this can be used to segment the reds.

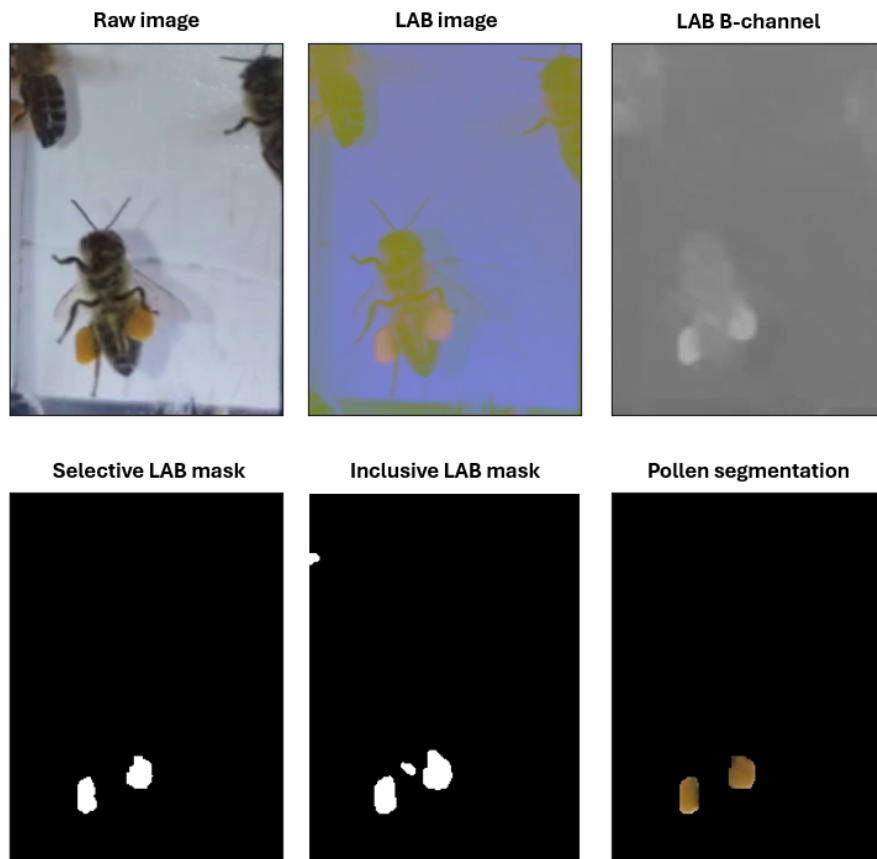


Figure 5.5: Yellow pollen segmentation process

To now segment pollen from the ROI, these masks are contour detected and contour analysis is performed. For the yellow pollen, first, the LAB mask is contour detected. The LAB mask will capture all pollen pixels in well-defined

5 Pollen monitoring

cases, such as in the raw image shown in Figure 5.5, but in more challenging cases, it may only detect the central pixels. It is essentially dialed in to avoid detecting non-pollen pixels as such, and thus only keep pixels it is sure of. To be able to segment all pollen pixels, even in hard cases, a more processed LAB mask is used. The goal is to create a mask that contains all pollen pixels, by also allowing some non-pollen pixels to be included. The first (selective) LAB mask can then be used to select pollen candidates. The alternatively processed (inclusive) LAB mask can then segment out the full pollen blob, knowing where the actual pollen is located and what area it can ignore. This alternative LAB mask also uses the B-channel, but processes it differently. It is cleaned up using histogram equalization to improve contrast, blurring and morphological opening to reduce noise. The contours resulting from the application of this method are put through inner area and perimeter thresholding to make sure no noise is included in the final segmentation. The contours that pass these thresholds are added to a mask that contains all the segmented pollen. For dark-red pollen, the dark-red mask is contour detected as well. These contours are put through the same inner area and perimeter thresholds as the yellow pollen contours, and are added to the complete pollen mask after passing. The red pollen segmentation process is shown in Figure 5.6.



Figure 5.6: Dark-red pollen segmentation process

For each of these pollen contours, the center point is calculated. Because the pollen contours missed a few pixels of the pollen, often on the edges since these were a bit darker, the pollen is approximated as a perfect circle in

5 Pollen monitoring

the image. This was an accurate approximation through visual confirmation, and helped more accurately count pollen that were semi-occluded. For this approximation the distance from the centroid to all contour points was calculated, and the largest distance was used as the radius. This allowed the pollen volume to be calculated as the volume of a sphere with this radius. This volume was converted from pixels cubed to mm³ with the use of a scaling factor which was determined with one of the captured frames, since all real world distances of the tunnel were known. The length (L) of the plexiglass was measured in millimeters on the tunnel itself and in pixels on the captured footage. The scaling factor that allows pollen volumes in pixels cubed to be converted to mm³ can thus be calculated by Equation 5.1.

$$\text{Scaling factor} = \left(\frac{L_{\text{real}}}{L_{\text{pixels}}} \right)^3 \quad (5.1)$$

The pollen volume can then be computed using Equation 5.2.

$$\text{Pollen-volume} = \frac{4}{3}\pi r^3 \quad (5.2)$$

The resulting pollen volume estimation thus contains some error due to the assumption that the pollen blob is a perfect sphere. To give an idea of the measured pollen volume of one pollen sphere, one of the pollen blobs shown in Figure 5.7 was determined to be 0.8 mm³ using Equation 5.2.

```
# Step 1: Find contours in the binary pollen mask
pollen_contours = find_external_contours in pollen_mask

# Step 2: Process each detected pollen contour
for each contour pollen_cnt in pollen_contours:

    # Step 3: Compute spatial moments of the contour
    M = calculate_image_moments of pollen_cnt

    if M["m00"] is not zero (i.e., valid contour):

        # Step 4: Calculate center of the contour using moments
        cx = M["m10"] / M["m00"]
        cy = M["m01"] / M["m00"]

        # Step 5: Estimate radius as max distance from center to any contour point
        for each point in pollen_cnt:
            compute distance from (cx, cy)
        radius = maximum of all distances

        # Step 6: Filter out contours that are too large to be pollen
        if radius < 10.0:

            # Step 7: Estimate pollen volume (assuming spherical shape)
            pollen_volume = (4/3) * pi * (radius ^ 3)
            pollen_volume_mm3 = pollen_volume * scaling_pollen
            add pollen_volume_mm3 to total_pollen_volume
```

Figure 5.7: Pseudo-code that illustrates the pollen volume calculation process

The final step was color correction of the pollen to accurately detect its color. This is necessary because of the shadows in the tunnel and use of camera flash, which distorted the colors in the captured images. This was done

5 Pollen monitoring

through color chart mapping. This technique uses a color chart of 24 standardized colors. An image is captured of this chart in an environment with good lighting, and in the observation tunnel. These two images are turned into two separate arrays, with the RGB values of each respective colors in the chart. The RGB color value of the pixel in the center of the pollen is taken, converted to LAB and compared to the colors in the array of colors captured in the observation tunnel. The distance to each of these colors is calculated and the closest match is determined. LAB is used for color distance calculation because it is designed to be perceptually uniform, which means that the Euclidean distance between two colors in the LAB color space correlates well with how different they appear to a human. The color is then corrected by changing it to the corresponding color in the array with of colors captured in the observation tunnel. This concept is shown in Figure 5.8.

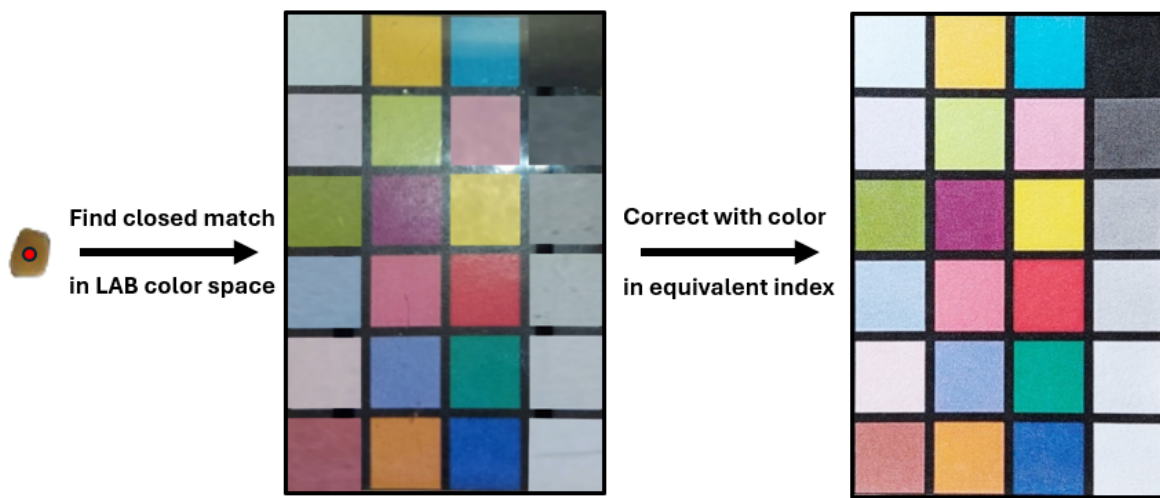


Figure 5.8: Color correction process (Xrite, n.d.)

The data extracted in this process is a video containing the detected pollen, drawn as a circle in the corrected color on the frames. Underneath the footage, a white banner is added which shows the cumulative counted pollen volume at each point in the video, and the original and corrected pollen color. To illustrate, a frame from this video is shown in Figure 5.9. This video is available in Appendix B ².

²GitHub repository for arenatest, pollen monitoring and website data

5 Pollen monitoring



Figure 5.9: Frame from the extracted pollen monitoring results video

5.3 Limitations, challenges and future perspectives

A first obvious limitation of this system, is that the pollen segmentation is limited to the database of colors. If a color was not included, then it will not be detected and counted. A segmentation not based on pollen color would generalize this system and allow it to be implemented in any environment. This was attempted but, did not succeed. However, with sharper footage, and if that still proves insufficient, AI models like CNNs could be used for segmentation to enable implementation.

Another limitation is the image quality. With better image quality, a better and more accurate segmentation can be done, where the spherical approximation could not be necessary and a more accurate pollen volume count could be achieved.

Next, in this setup, the pollen are counted multiple times, since the same pollen are visible over multiple frames. The use of a tracker would be necessary to get an accurate pollen count. This was again attempted but did not succeed, and is a prospective for the future.

The color correction process can also be further enhanced. Occasionally, errors occur, and some of these discrep-

5 Pollen monitoring

ancies are visually significant. By only taking the center pixel's RGB value, an approximation is made, although the results were better when the mean of the detected pollen was taken. This was possibly due to the pixels at the edges being darker, making the distance to other (incorrect) colors smaller and thus resulting in a fast changing closest match and correction. In its current implementation, it assumes linear lighting conditions and uniform illumination, which could be improved through the use of techniques such as Polynomial Color Correction and Thin Plate Spline (TPS) Color Mapping.

Furthermore, this proof of concept is limited by the static data extraction, data needs to be gathered on site. Then the observation tunnel needs to be removed to be able to access the data and upload it to the cloud. From there, it is uploaded in the Google Colab environment and processed. In the future, this can be upgraded to a real-time application. Live images could be processed by a Raspberry Pi and data could be extracted and visualized immediately. This could be done through the website (see 9.1.3: Pollen monitoring page), and a dummy implementation was already created through Node-RED. The pollen-data is saved to CSV-files with amounts of pollen detected for each color, per timestamp. The website uses these CSV files to provide a data history, and also shows the volume of pollen detected per color on the current day.

Finally, by deploying this system on multiple beehives and comparing their data, an analysis could be done showing the foraging efficiency of the hives.

5.4 Conclusion

This chapter showed how, using a custom-built observation tunnel and a color-based computer vision algorithm can be used to detect and quantify pollen collected by honeybees. The system was designed with attention to both the biological needs of the bees (including flight guidance and natural behavior preservation) and the technical constraints of image capturing and analysis. Although the initial Raspberry Pi setup had limitations in image quality, the use of a smartphone camera provided frames of sufficient clarity for accurate pollen segmentation and counting.

The developed algorithm successfully identified bees, isolated pollen on their legs and segmented it based on predefined color databases using the LAB color space. Furthermore, by approximating pollen as spheres and applying volume estimation and color correction, the pollen intake of a beehive could be quantified. Despite some limitations, such as dependence on known pollen colors, overcounting and offline processing, the groundwork has been laid for a real-time, scalable system that could monitor foraging behavior across multiple hives.

The goal of creating a proof of concept was achieved. Looking forward, improvements in tracking, segmentation accuracy and live data streaming could greatly improve the system's reliability and usability. When fully developed, this method has the potential to offer beekeepers and researchers a non-invasive tool for evaluating hive health and foraging efficiency.

6

Sound stimulus

6.1 Purpose

This section will discuss the use of a sound stimulus on a beehive. The aim of this experiment is to expose a beehive to a short, pulsed sound stimulus at a fixed frequency. This stimulus should trigger a reaction from the bees that will be recorded with a microphone. The recordings will then be analyzed to see how much effect the stimulus has on the bee colony. The reaction of the bees could contain information about the health of the colony. If this is the case, this sound stimulus could form an effective way to check the well-being of the bee colony without opening the beehive. This is a massive advantage because opening a beehive can seriously disrupt the bees. In winter, or when it is cold outside, opening the beehive is even dangerous and could cause the bees to get disoriented and die in the cold. This test is an alternative to the knocker test. The knocker test is explained in section 3.3. For the knocker test, a knocker is placed outside the beehive and an accelerometer is placed inside the beehive. For this test a microphone outside the beehive is used. If the reaction from the bees appears to be significant, then this test can be further examined with multiple beehives of different health states. Perhaps the reaction from the bee colony to the stimulus will then be dependent on their health. This could make the sound stimulus a good test for beekeepers to help them determine the health of their beehives.

All data, code and results for this chapter can be found in Appendix F¹ sound stimulus.

6.2 Test setup

The test are performed with a Raspberry Pi 3 B+ as the experiment control system. This Pi uses a USB microphone to make the recordings. So, for this test, the accelerometer is swapped for a microphone. Using a microphone instead of an accelerometer is a more practical solution. An accelerometer must be placed inside one of the honeycombs inside the beehive, and a wire must come out somewhere. This is a problem that disappears with the use of a microphone. The microphone is simply placed underneath the beehive and does not have to be

¹GitHub repository for sound stimulus test and tests with infrared imaging

6 Sound stimulus

placed inside the beehive. This makes the setup easier, although a microphone is much less precise than an accelerometer. An accelerometer is capable of capturing the signals much more precisely. The stimulus is sent by two speakers that are connected to an amplifier card. This card is connected to the Raspberry Pi via USB. These speakers are also placed underneath the beehive. Figure 6.1 shows the components used for the test. It also shows how the recordings are made underneath the beehive.



(a) components inside the setup



(b) Setup underneath a beehive

Figure 6.1: Setup used to sent a stimulus and record the reaction from the bee colony

Figure 6.2 shows the microphone and the speakers that are used for the tests.

6 Sound stimulus



(a) USB microphone



(b) speakers

Figure 6.2: The microphone and speakers used for the tests.

The stimulus used is 0,3 seconds long. This stimulus is a sound at one fixed frequency. The test is performed multiple times with multiple frequencies. This is necessary to see if the bees react stronger to a certain frequency. The frequency will vary between 500 Hz and 4000 Hz in steps of 500 Hz. The frequency is kept below 5000 Hz. This is because a sound at 5000 Hz is proven to scare hornets away. To minimize the risk of disturbing or hurting the bees, the maximum frequency that is used for the stimulus is 4000 Hz. The recording will start five seconds before the stimulus and continue until ten seconds after the stimulus.

6.3 Statistic analysis results

This section goes deeper into the test results and uses statistics to try and explain the recorded signals. The section is split into two parts. First, a comparison between recordings with bees and recordings without bees is made. This is to check if classification is possible. The second part covers the analysis of the reaction from the bees to the stimulus. Statistics will be applied to try and confirm that there is a reaction from the bees to the stimulus.

6.3.1 Comparison between bee and no bee recordings

This section will go through the analysis of the recordings. This section will try to go deeper into the effect of the sound stimulus on the bee colony. The first is to see if the presence of bees can be detected. For this, two sets of recordings were made. The first set contains 120 recordings. These are recordings made underneath a hive without any bees. The set contains 40 recordings at 35% of the maximum volume and 80 recordings at 70%. Each set also contains recordings with a stimulus at multiple frequencies. The frequencies go from 500 to 4000 Hz in steps of 500 Hz, which is eight different frequencies. The set contains ten recordings of each frequency at

6 Sound stimulus

70% of the volume and five recordings of each frequency at 35% of the volume. The second set is made by exactly the same recordings but then underneath a beehive that does contain a bee colony. Each recording contains five seconds before the stimulus, then the stimulus of 0,3 seconds and ten seconds after the stimulus. Now, a statistical analysis will be executed on the two different data sets.

The first test is a three-way ANOVA. A three-way ANOVA is employed to examine how the mean of a continuous dependent variable varies across the levels of three categorical independent variables. This method is appropriate when the goal is to understand the combined effects of three factors on a single outcome measure. This ANOVA has the presence of bees as the first variable, frequency as the second variable and volume as the third variable. This test studies the effect of these three variables on five dependent variables. These five variables are all features that can be extracted from the recordings. The first is the rms value or the root mean square of the signal. The rms value contains information about the average energy of the signal, and is calculated using Equation 6.1. The ANOVA does not use the rms of the whole signal but uses the rms ratio. This is the ratio of the rms value before the stimulus and the rms value after the stimulus. For each value, a window of four seconds is used.

$$rms = \sqrt{1/N \sum_{i=1}^N x_i^2} \quad (6.1)$$

The second variable is the spectral centroid. A spectral centroid measures the center of mass from the sound spectrum. A high centroid means that the higher frequencies contain more energy than the lower ones. This term basically shows the average frequency of the signal. The next variable is the bandwidth of the signal. The bandwidth shows the spread of the spectrum around the centroid. So, a higher bandwidth means that the signal has a broad frequency content. The fourth variable is the spectral rolloff. This shows the frequency below which 85% of the total spectral energy is contained. So, a low rolloff means that the energy remains mainly in the lower frequencies. The last variable is the zero crossing rate. This value shows how many times the signal crosses the zero line. These last four variables only look at the signal after the stimulus.

The first column of Table 6.1 shows the five different variables on which the ANOVA is performed. The second column shows the p-value for the effect whether bees are present or not, the third column the p-value for the effect of the stimulus frequency and the fourth the p-value for the effect of the volume. Table 6.2 contains the p-values of the interactions between the three variables: frequency, bee presence and volume. This shows if the variables depend on each other.

6 Sound stimulus

Table 6.1: Three-way ANOVA bee presence

Feature	Bee Pres. (p)	Freq. (p)	Vol. (p)
rms ratio	0,00	0,00	0,00
centroid	0,36	0,13	0,04
bandwidth	0,01	0,29	0,32
rolloff	0,98	0,08	0,10
zcr	0,51	0,00	0,01

Table 6.2: Three-way ANOVA bee presence, interactions

Feature	Bee*Freq (p)	Bee*Vol (p)	Freq*Vol (p)	Bee*Freq*Vol (p)
rms ratio	0,27	0,00	0,26	0,88
centroid	0,74	0,17	0,51	0,15
bandwidth	0,28	0,91	0,66	0,73
rolloff	0,26	0,35	0,68	0,28
zcr	0,56	0,00	0,04	0,00

Table 6.1 and Table 6.2 shows the p-values for the three-way ANOVA. The p-value determines whether the feature has a significant effect or not. If the p-value is smaller than 0,05 then this means that the result is significant. Otherwise, the result has no significance for the test. Only a few of the values show significance in the features. The rms ratio seems to be significantly different for all three variables. This means that the rms ratio is not the same for the recordings with bees and the recording without bees and for the different volumes, which is expected. However, it also differs for the recordings at different frequencies. The spectral centroids only seem to differ significantly for the two different volumes. The bandwidth shows only a significant difference for the recordings with bees and those without bees. Rolloff does not show any significant results. The zero crossing rate shows significant results for different frequencies and different volumes. The interactions between the three variables don't show much significant results. There are only four results that are less than 0,05. Three for the zcr and one for the rms ratio.

Figures 6.3 to Figure 6.10 show the box plots for the five characteristics, the ones by frequency and presence of bees and once by volume and presence of bees. Figure 6.3 and Figure 6.4 show the box plots for the rms ratio.

6 Sound stimulus

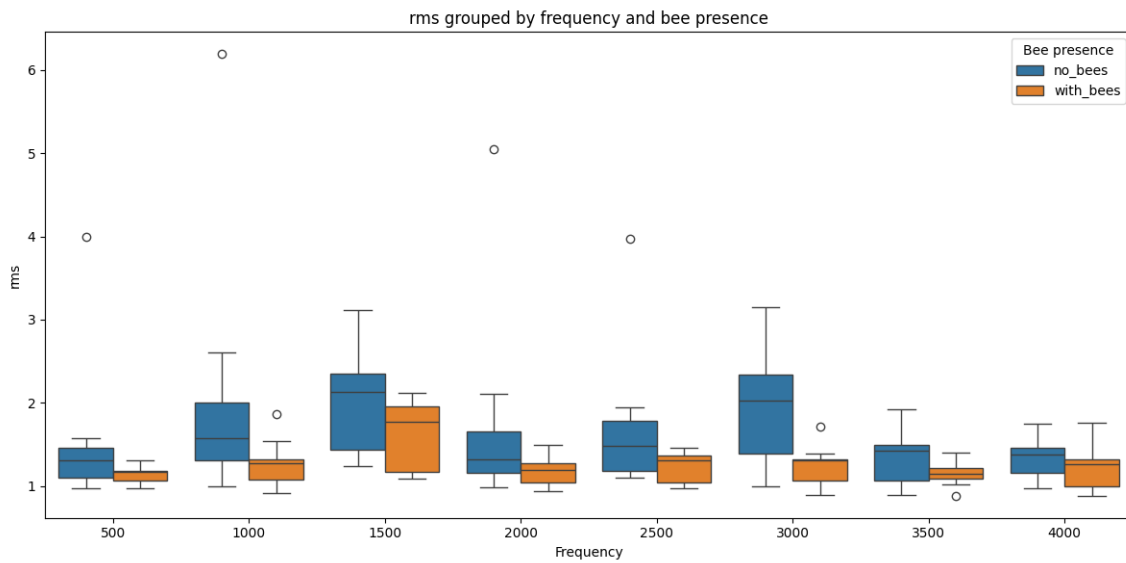


Figure 6.3: Box plots of the RMS value of the signal by frequency and bee presence.

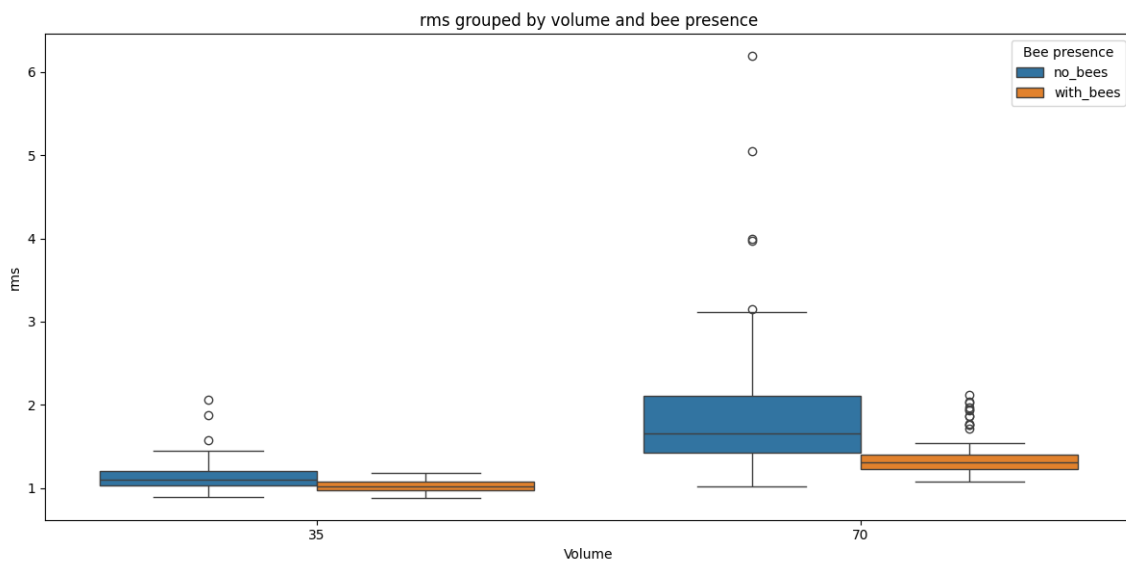


Figure 6.4: Box plots of the RMS value of the signal by volume and bee presence.

The next variable is the spectral centroid of the signal. This is shown in Figure 6.5 and Figure 6.6. These box plots don't show much difference in centroids for the different frequencies and volumes.

6 Sound stimulus

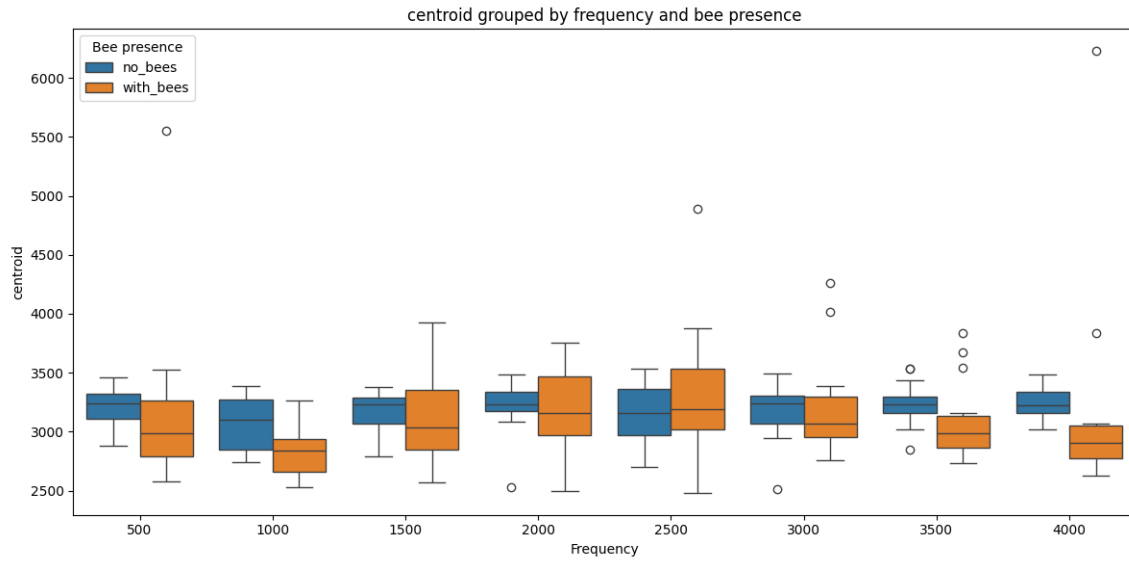


Figure 6.5: Box plots of the spectral centroid of the signal by frequency and bee presence.

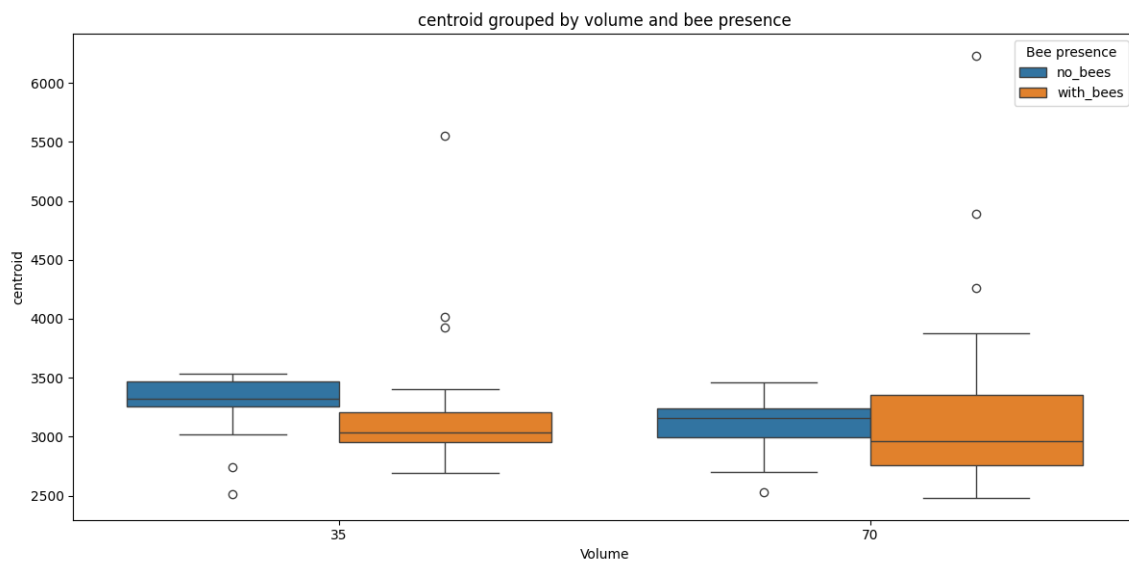


Figure 6.6: Box plots of the spectral centroid of the signal by volume and bee presence.

Figure 6.7 and Figure 6.8 shows the box plots of the bandwidth by frequency and bee presence. Here, only bee presence returns a significant p-value. This means that the presence of bees alters the bandwidth of the recorded signal. This is another feature that can be used to distinguish bee recordings from no bee recordings.

6 Sound stimulus

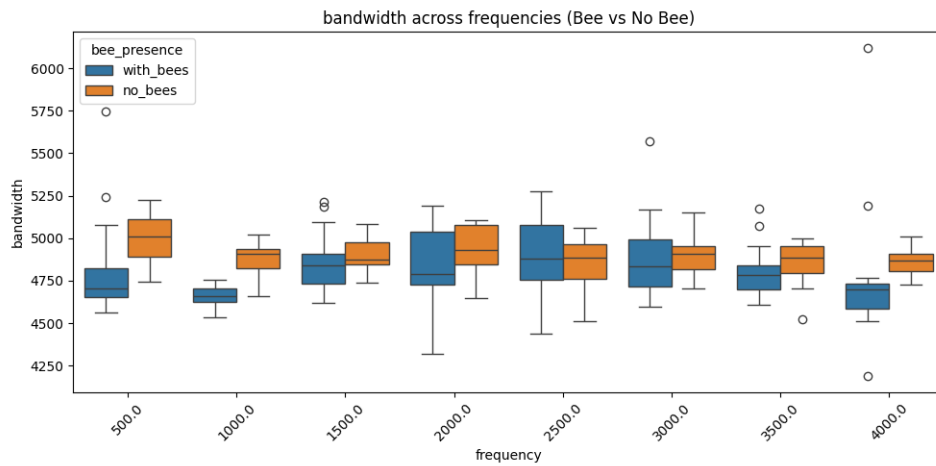


Figure 6.7: Box plots of the bandwidth of the signal by frequency and bee presence.

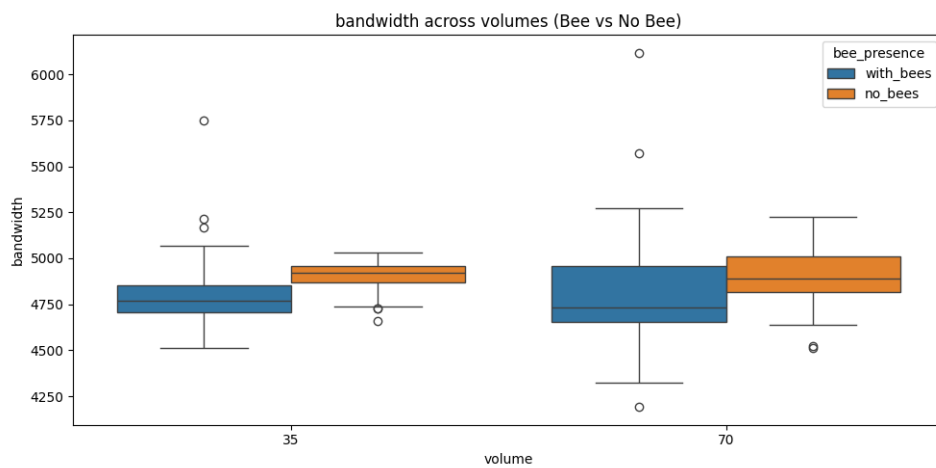


Figure 6.8: Box plots of the bandwidth of the signal by volume and bee presence.

The fourth feature is the rolloff. This feature has three p-values that are all larger than 0,05 so none of them are significant. In Figure 6.9 and Figure 6.10 the box plots are shown to confirm this.

6 Sound stimulus

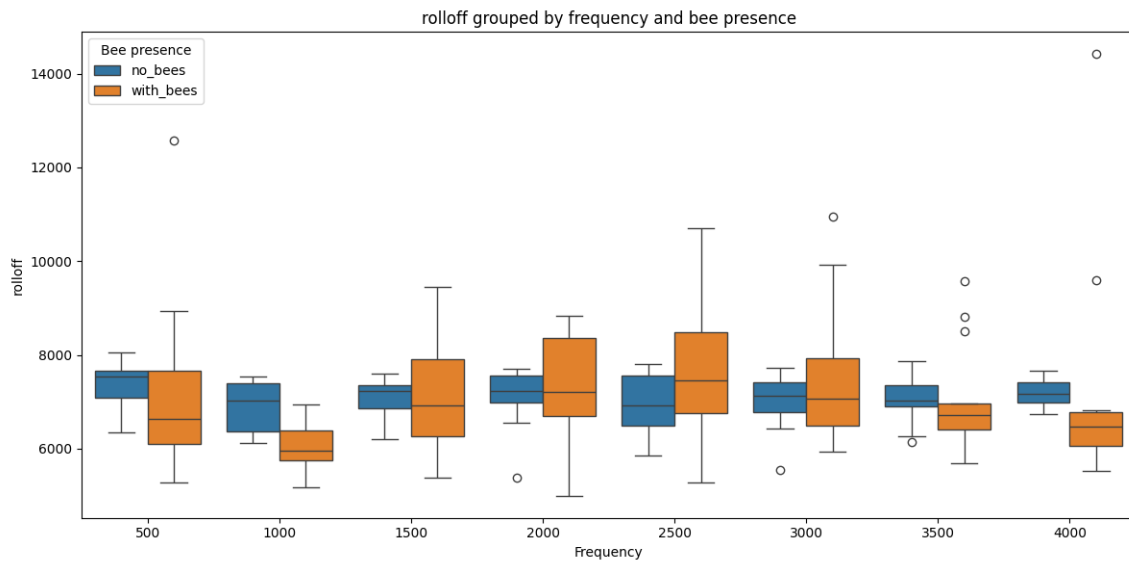


Figure 6.9: Box plots of the spectral rolloff of the signal by frequency and bee presence.

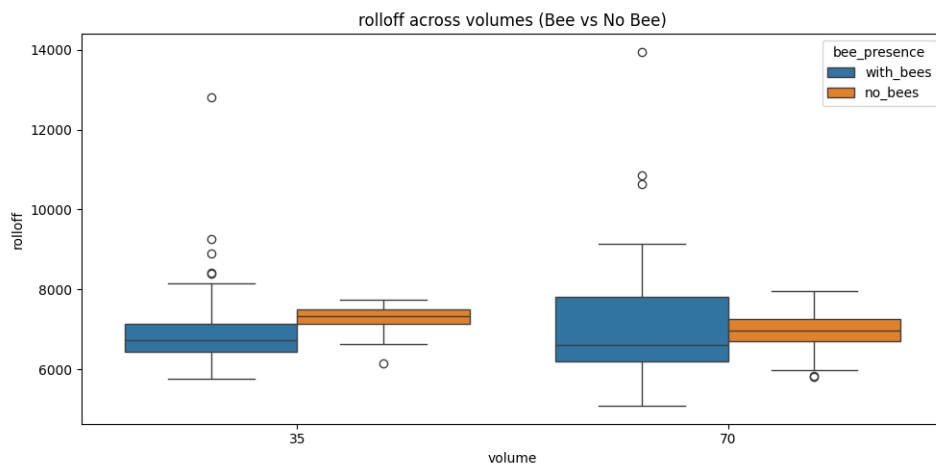


Figure 6.10: Box plots of the spectral rolloff of the signal by volume and bee presence.

The last feature is the zero crossing rate. This is displayed in Figure 6.11 and Figure 6.12.

6 Sound stimulus

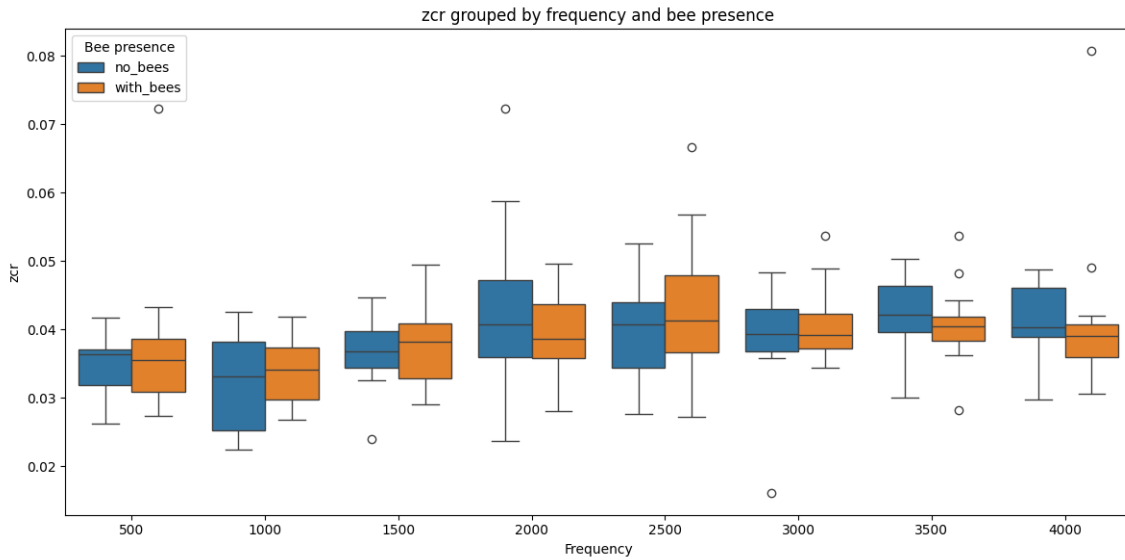


Figure 6.11: Box plots of the zcr value of the signal by frequency and bee presence.

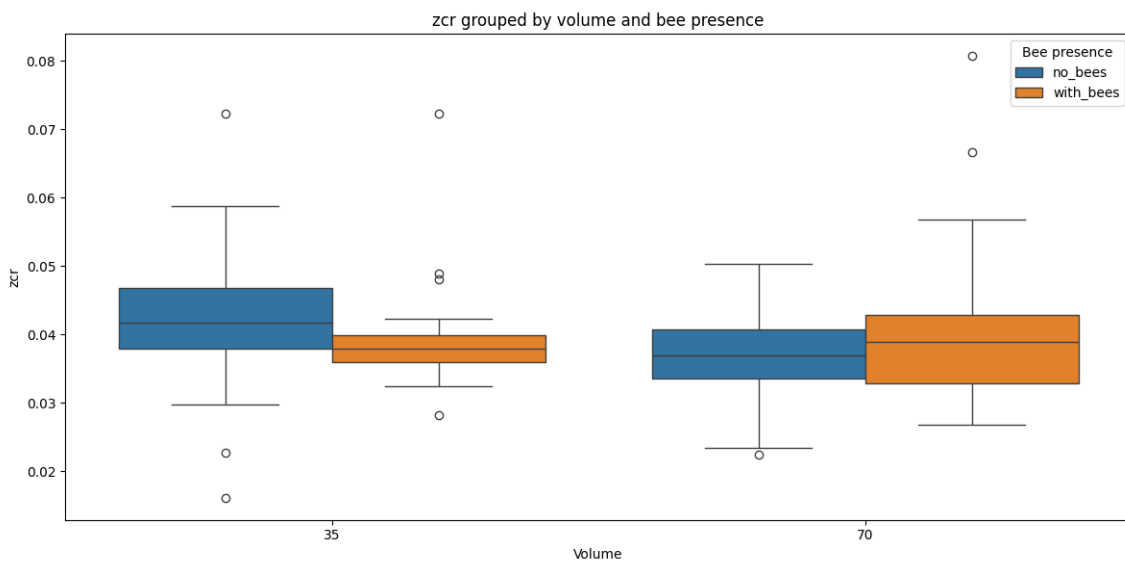


Figure 6.12: Box plots of the zcr value of the signal by volume and bee presence.

The box plots for the different box plots show no huge differences between the recordings with and without bees. There is a lot of overlap between both cases. The box plots showing the rms ratio seem to display the biggest difference with and without bees. This means that the rms ratio is a very useful factor in classifying recordings with or without bees. The box plots show the same results as Table 6.1.

To get a closer look at which pairs of recordings differ, a Tukey Honestly Significant Difference (HSD) test was

6 Sound stimulus

executed. This test performs a pairwise comparison between the different recordings after the ANOVA to show which pairs differ significantly. This test did not return significant results.

6.3.2 Bee reaction to the stimulus

The three-way ANOVA test confirms that there is a difference between the recordings with and without bees. Now, the thesis will look deeper into the recordings and try to confirm if there is a reaction from the bees specifically to the stimulus. Before using any statistics, this section will begin with a visual analysis without using any tests. The next step is a principal component analysis (PCA) on the recordings. These PCA results will be used to perform a deeper statistical analysis on the data set.

The first step in looking for a reaction of the bees is to make a few different plots. The plots are shown here for one example, a recording at 72% volume and a stimulus of 1500 Hz. Figure 6.13 shows the waveform of this recording.

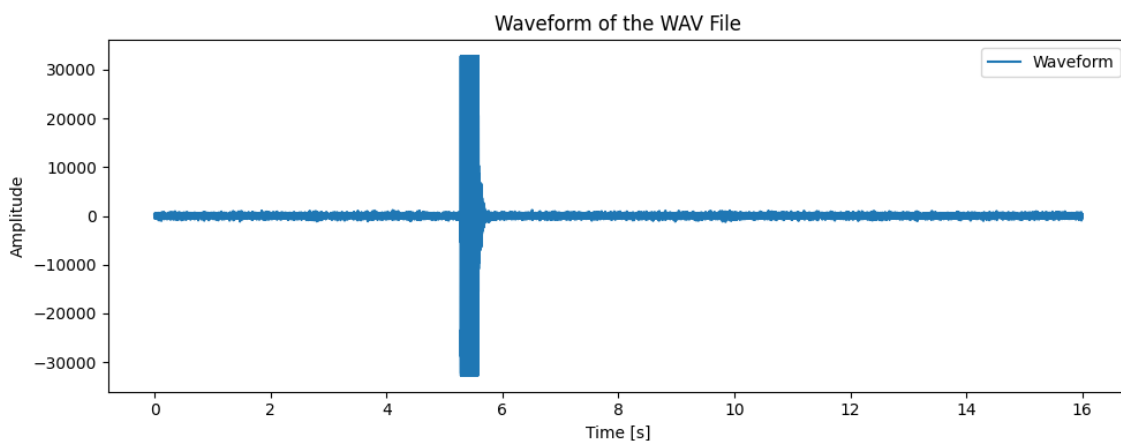


Figure 6.13: RMS value from a recording at 70% volume and 1500 Hz

The first plot that is made shows the RMS value of the signal. Figure 6.14 shows an example for one of the recordings at 70% volume and a stimulus at 1500 Hz. The Figure shows an increase in the RMS value just after the stimulus. For the rest of the stimulus, there do not seem to be any changes in the RMS value.

6 Sound stimulus

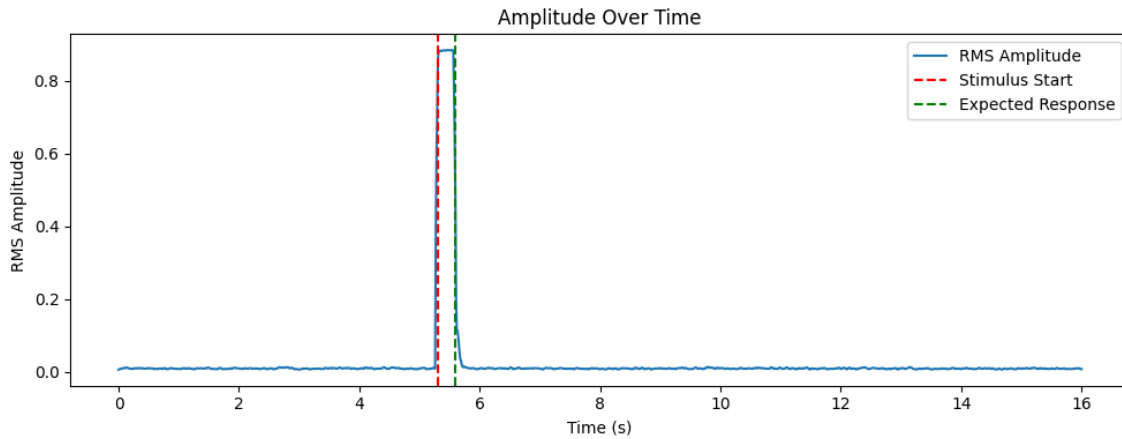


Figure 6.14: RMS value from a recording at 70% volume and 1500 Hz

The next plot shows the spectral centroid of the signal. Figure 6.15 shows the result. The plot shows that the spectral centroid stays around a value of 3000 Hz. The value does drop when the stimulus starts. This is logical because the stimulus has a frequency of 1500 Hz. This makes the mean frequency of the signal drop. More than that, the figure does not show.

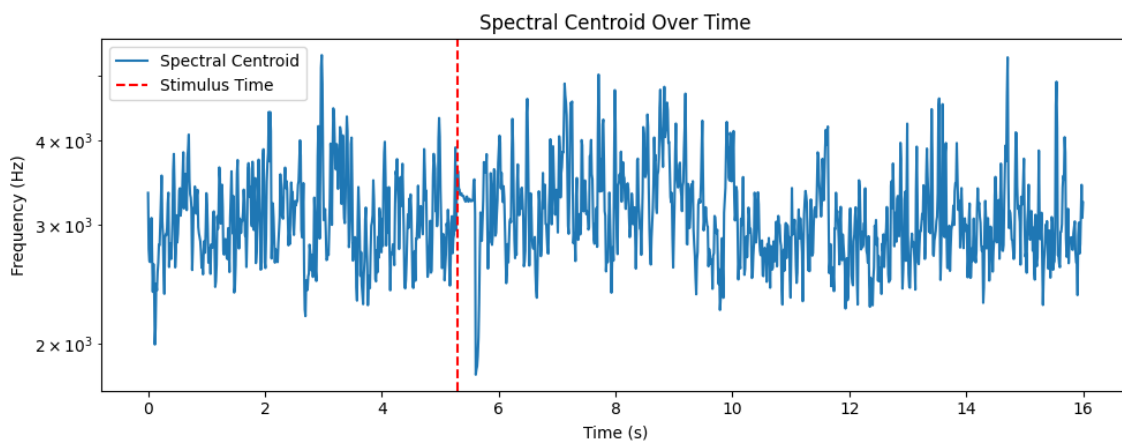


Figure 6.15: RMS value from a recording at 70% volume and 1500 Hz

Figure 6.16 shows the different frequencies that appear in the signal on a logarithmic scale. The graph is split into two. The first one shows the frequency distribution in the signal before the stimulus and the second one shows the frequency distribution after the stimulus. This clearly shows the frequency of the stimulus. There are also some low frequencies that show up, and are visible after the stimulus. These do have a significantly lower magnitude than the stimulus. There are no other frequencies that show up. The other recordings show a similar output. If there are other frequencies that show up in these graphs, these are harmonic frequencies of

6 Sound stimulus

the stimulus frequency.

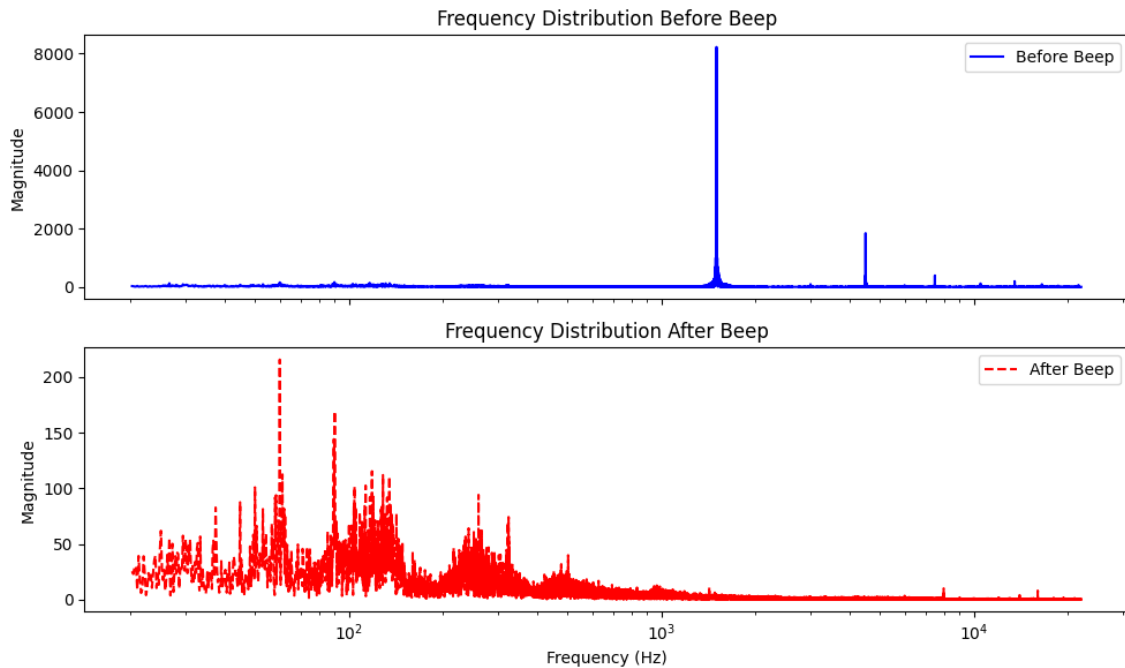


Figure 6.16: RMS value from a recording at 70% volume and 1500 Hz

The next figure is the spectrogram of the signal. Figure 6.17 shows the results. The spectrogram shows all the frequencies that appear in the signal. All values below 1024 Hz show up in the signal but do not seem to show any changes in the part before the stimulus compared to the part after the stimulus. The stimulus frequency of 1500 Hz is clearly visible. The harmonic frequencies are also very visible. There do not appear very visible changes before and after the stimulus.

6 Sound stimulus

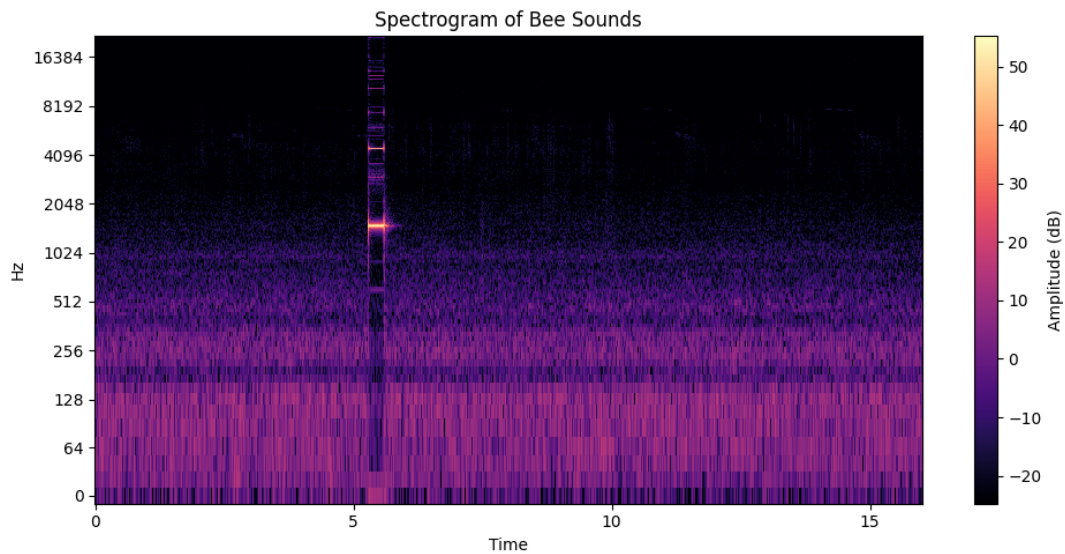


Figure 6.17: RMS value from a recording at 70% volume and 1500 Hz

The last plot is a zoom image of the full waveform. Figure 6.18 shows the part of the waveform just after the stimulus. The stimulus ends at 5.6 seconds. After that, there is still an increase in the signal. There even is a dip between this increase and the stimulus. This is a result that was also accomplished in the knocker test that is explained in section 3.3. This could of course also be the speakers that are still vibrating after the stimulus.

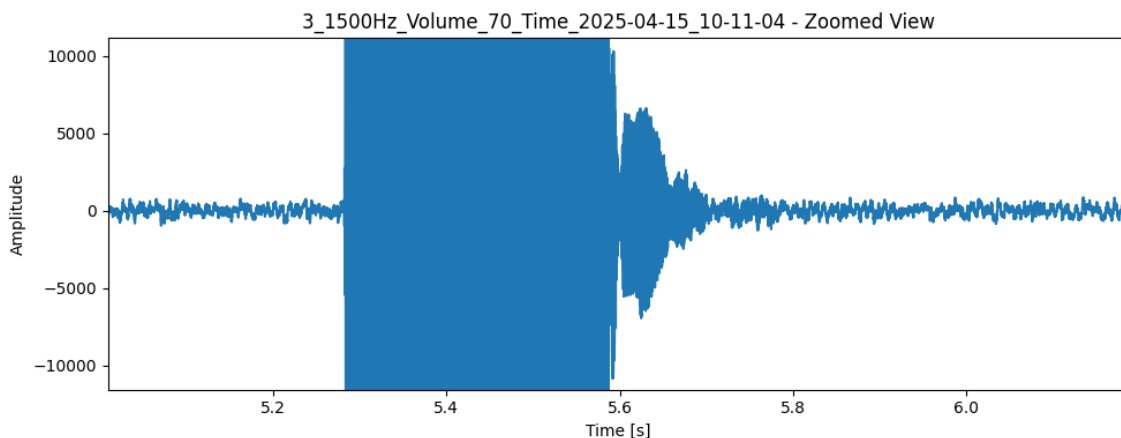


Figure 6.18: RMS value from a recording at 70% volume and 1500 Hz

These results show some signs of a reaction from the bees. Most of the plots show no visible changes in the signal. Only Figure 6.14 and Figure 6.18, the figures of the rms value and the zoomed image, show some signs of a reaction. This could both be a vibration of the speakers that is detected. However, it could also contain a reaction from bees. During testing with the sound stimulus, a recording was made with a knock on the beehive. Figure 6.19 shows the

6 Sound stimulus

spectrogram of this recording. The knock occurs just before the five-second mark. This spectrogram shows that different frequencies appear after this stimulus. The entire spectrum lights up for almost five seconds. This is a sign that never happens in any of the recordings with the sound stimulus. This does not mean that bees do not react to the sound stimulus. This means that the bees react less strongly to this stimulus or react differently than to the knock.

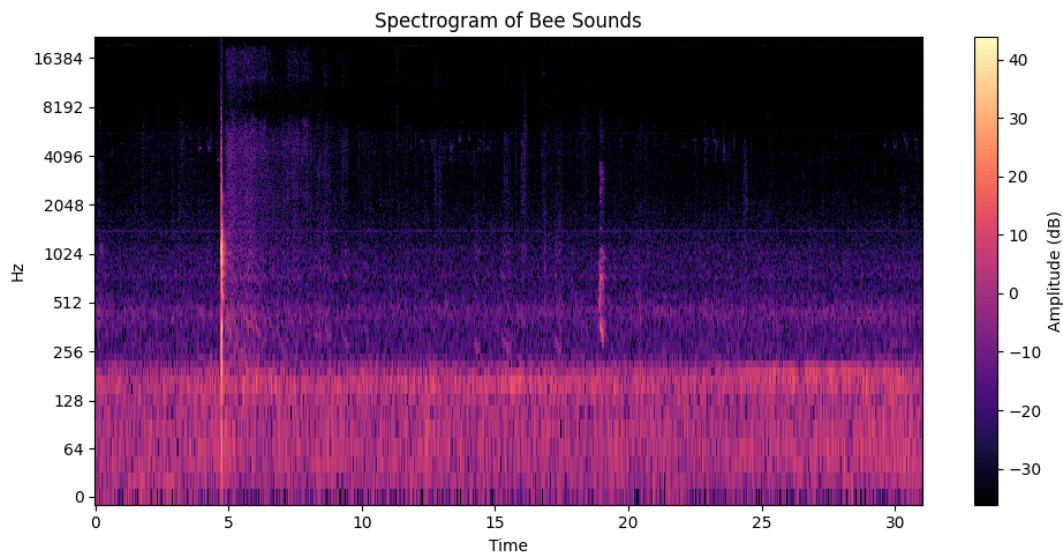


Figure 6.19: knock on the beehive

Visual analysis does not prove that there is a reaction from bees to the stimulus. So, a deeper analysis using statistics and PCA is required.

Now, a principal component analysis will be performed on the recordings. PCA is a strategy to reduce the dimensions of a large dataset to a set of principal components that hold most of the original information. This is done by extracting a set of features from the data. These principal components will be used in statistical tests to help determine the effect of the stimulus on bees.

The first principal component analysis is run using the following features: RMS, zcr, spectral centroid, spectral bandwidth and spectral rolloff. These are the same five features that are used in the three-way ANOVA.

6 Sound stimulus

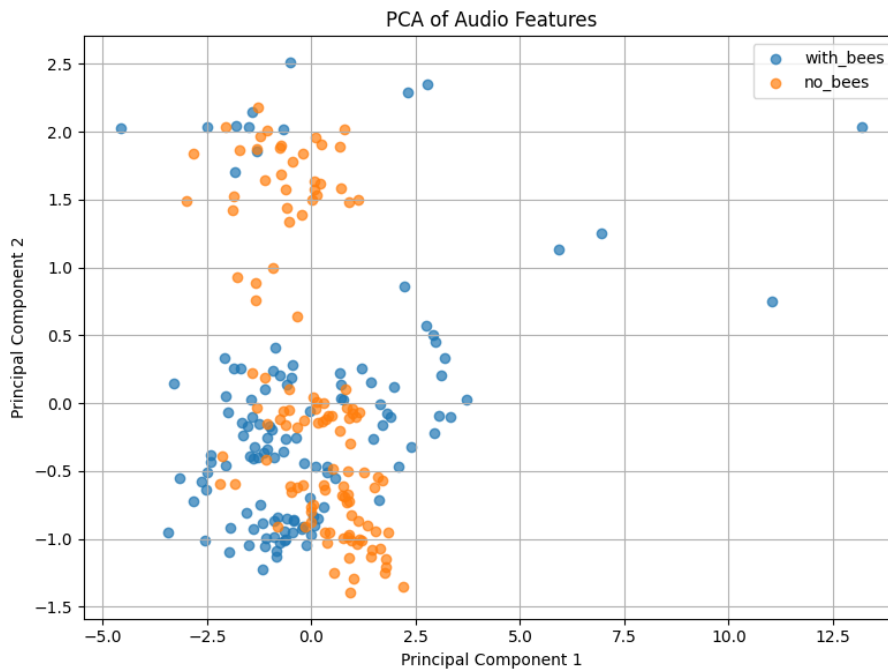


Figure 6.20: Scatter plot that shows the distribution of recordings by PC1 and PC2 components for PCA with 5 features

Figure 6.20 shows the distribution of the recordings in a two-dimensional space by PC1 and PC2. PC1 captures the direction of maximum variance in the data set, and PC2 captures the second highest variance, orthogonal to PC1. The plot shows that the data points for the recordings with and without bees overlap quite a bit. However, it is possible to detect some clustering even with the naked eye. To better separate both recordings, more features are added to the PCA. The first feature that is added is the spectral flatness. This feature quantifies how much a signal resembles a certain tone. The next features that are added are the mfcc features, Mel-frequency cepstral coefficient. The mfcc features from one to thirteen are used. These thirteen coefficients capture the short-term power spectrum of the sound. The lower coefficients, such as mfcc1, capture the overall shape of the signal, while the higher coefficients capture fine spectral details. These coefficients are often used for classification tasks related to speech recognition or music identification. Other features that will now be used are chroma features. Chroma features capture pitch-class information from a signal. These features can be used for harmonic similarity estimation and chord recognition. The chroma features of one to twelve are now used in the PCA.

6 Sound stimulus

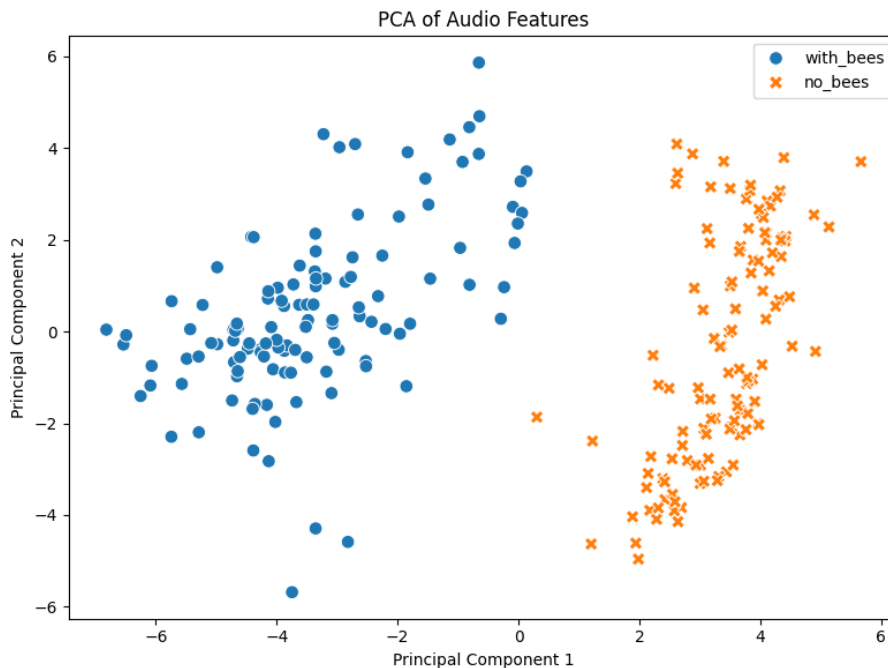


Figure 6.21: Scatter plot that shows the distribution of recordings by PC1 and PC2 components for PCA with 31 features

Figure 6.21 shows the new PCA scatter plot with 31 features instead of five. This makes the separation between the data points for the recordings with bees and the recordings without bees very clear. From now on, the analysis will work with the large set of features instead of the small one of six features. For the last PCA, PC1 explains 42,8% of the variance, and PC2 explains 16% of the variance. Together, this is good for 59% of the variance, which makes PC1 and PC2 enough to go on. For now it doesn't seem to be necessary to look at more principal components as these could work as noise on the data and make it harder to interpret.

To confirm the findings of the scatter plot, a t-test is performed on the resulting PC1 values. This t-test returns a t-value of -41,66 and a p-value of 4,53e-95. This is a very strong result. The t-value is far from the zero hypothesis, and the p-value is zero, which means that the difference is very significant. This means that PC1 contains enough variance to separate the bee recordings from the no-bee recordings. Figure 6.22 shows the resulting box plot from this t-test.

6 Sound stimulus

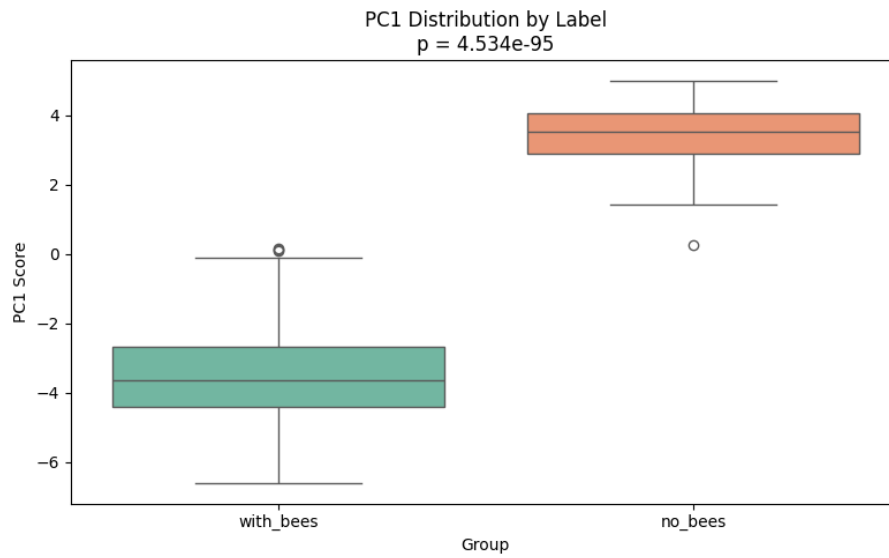


Figure 6.22: Box plot for t-test on PC1, bee presence

It's clear that with these features it is possible to discriminate the recordings with bees from the recordings without bees. To gain more insight in which features exactly capture the most variance, Figure 6.23 and Figure 6.24 show the weight of the top ten features in the linear combinations that form PC1 and PC2. Many of the higher coefficients of the mfcc features recur in the loadings for PC1. This shows that the difference between the recordings is found in finer spectral details and not in the overall spectral shape. Also, a few chroma features recur, which shows that the recordings with bees contain some specific pitch-class content.

6 Sound stimulus

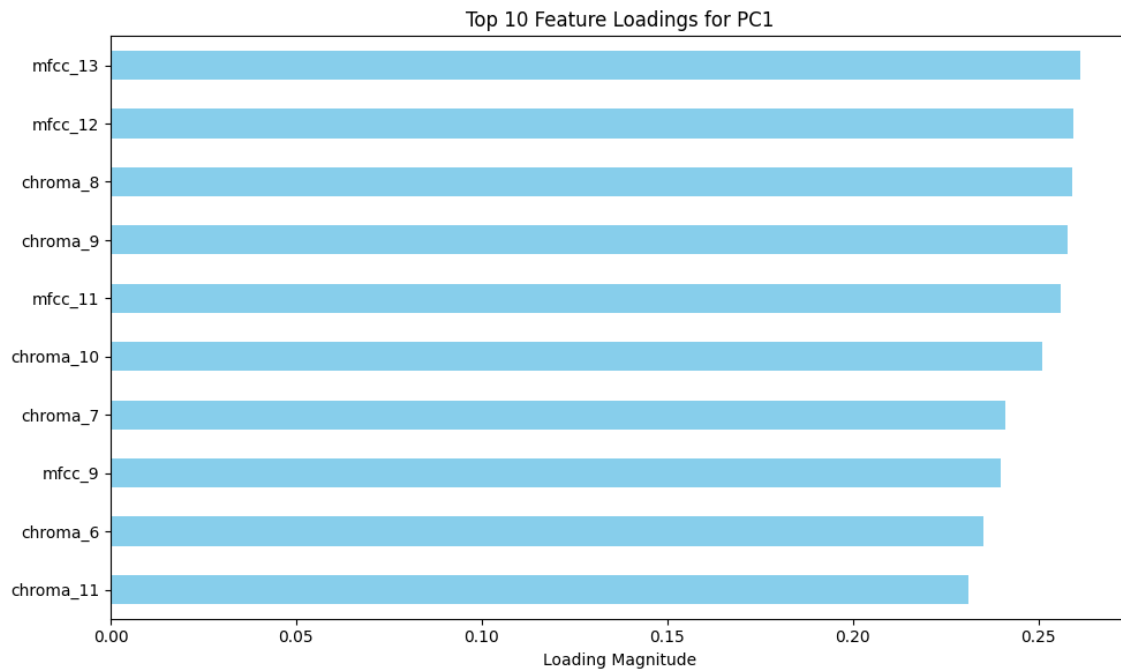


Figure 6.23: Contribution of the different features to principal component 1

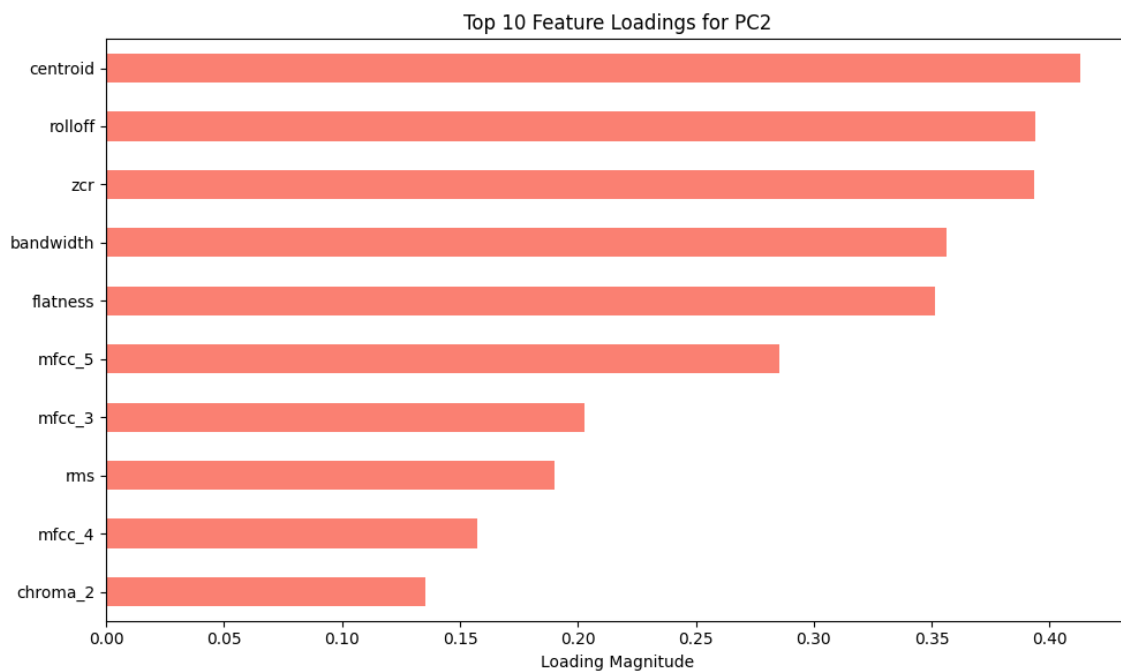


Figure 6.24: Contribution of the different features to principal component 2

Now, this analysis will be extended to look at the effects of the stimulus on bees. Now, the analysis will focus

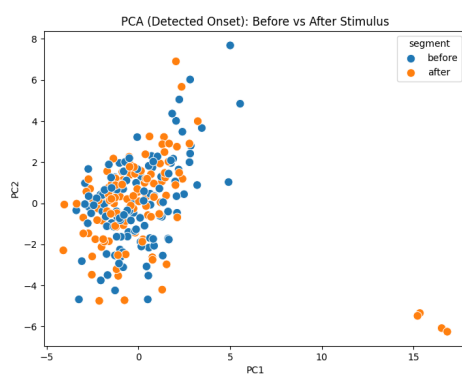
6 Sound stimulus

on comparing the part of the recording just before the stimulus and the part just after the stimulus. Principal component analysis is executed on these two parts of the recordings. This is done for different time windows ranging from 0,01 seconds to five seconds. With these PCA results, a paired t-test is performed on the PC1 values to check if the variance between the data sets is significant. Table 6.3 shows the results of the t-tests on the PC1 values for different time frames.

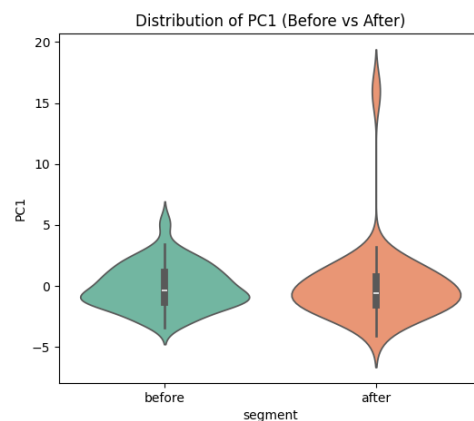
Table 6.3: Paired t-test on PC1

Time window [s]	<i>t</i> -value	<i>p</i> -value
0,01	-0,41	6,85e-1
0,10	-3,30	1,28e-3
0,30	3,99	1,13e-4
0,50	-1,98	5,06e-2
1,00	-2,21	2,94e-2
2,00	-2,49	1,41e-2
3,00	-2,65	9,07e-3
5,00	-4,07	8,41e-5

Table 6.3 shows that for all time windows except 0,01 seconds, the *p*-values are less than 0,05. This means that for those time frames the difference in the signal before and after the signal is significant. The difference seems to be the most significant with a time window of 0,3 or 5 seconds, and these windows also return the highest *t* values. Now, the scatter plots and violin plots of these tests are displayed to confirm the values of Table 6.3.



(a) Scatter plot PCA



(b) Violin plot PC1

Figure 6.25: Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 0.01s before and after the stimulus

6 Sound stimulus

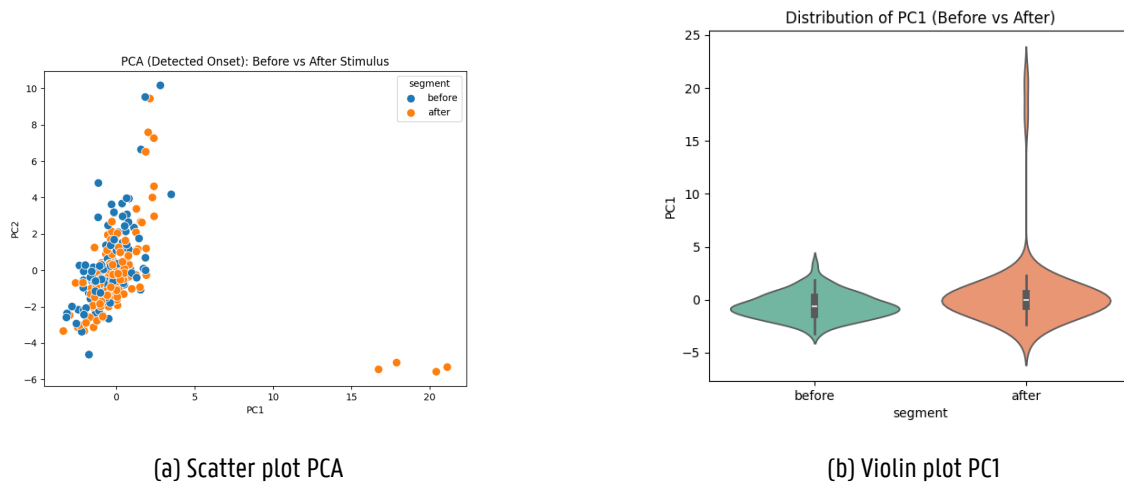


Figure 6.26: Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 0.1s before and after the stimulus

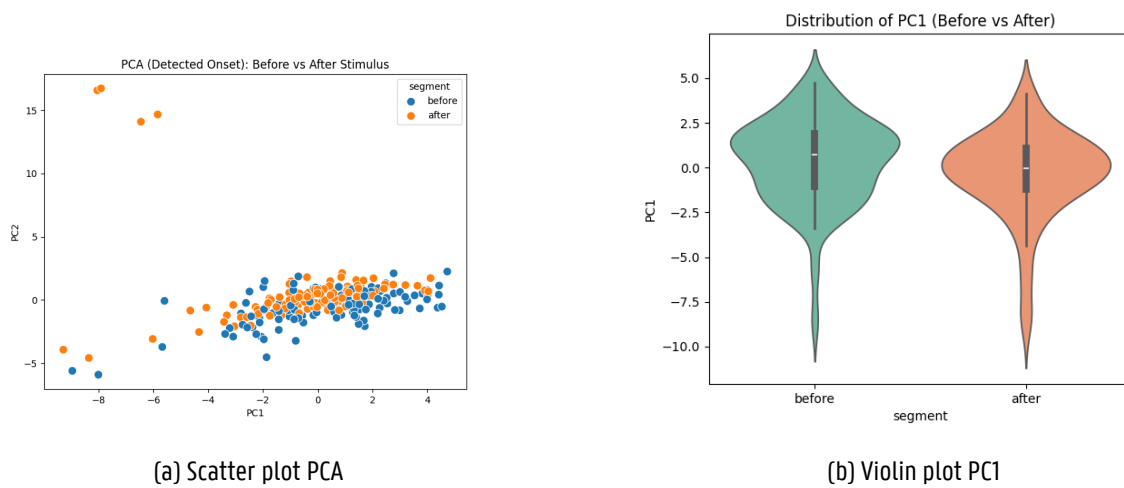


Figure 6.27: Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 0.3s before and after the stimulus

6 Sound stimulus

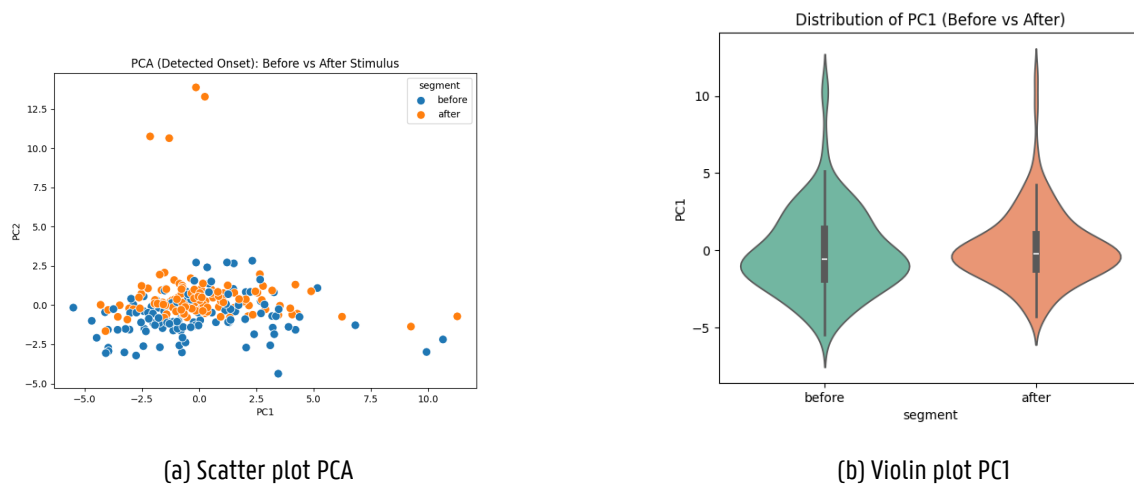


Figure 6.28: Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 0.5s before and after the stimulus

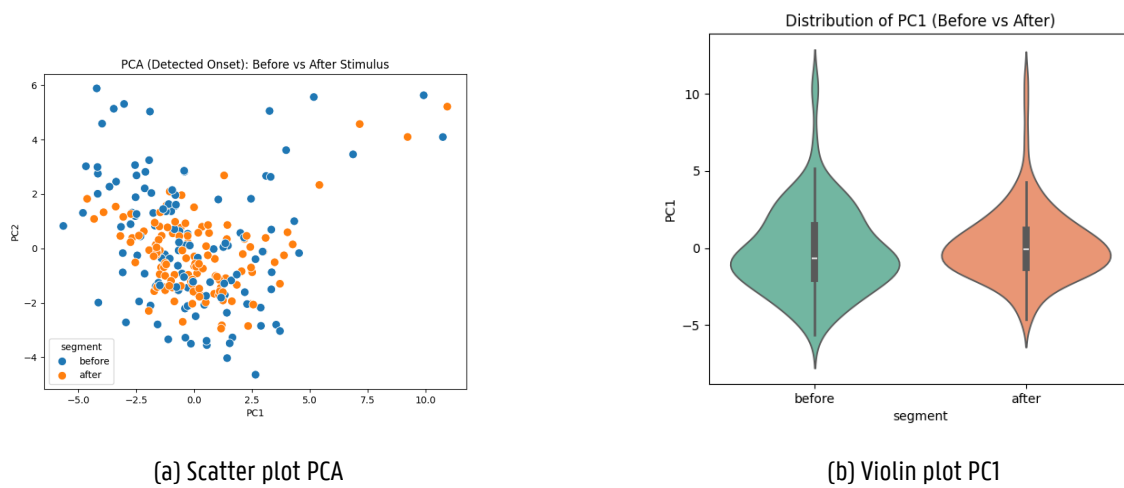


Figure 6.29: Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 1s before and after the stimulus

6 Sound stimulus

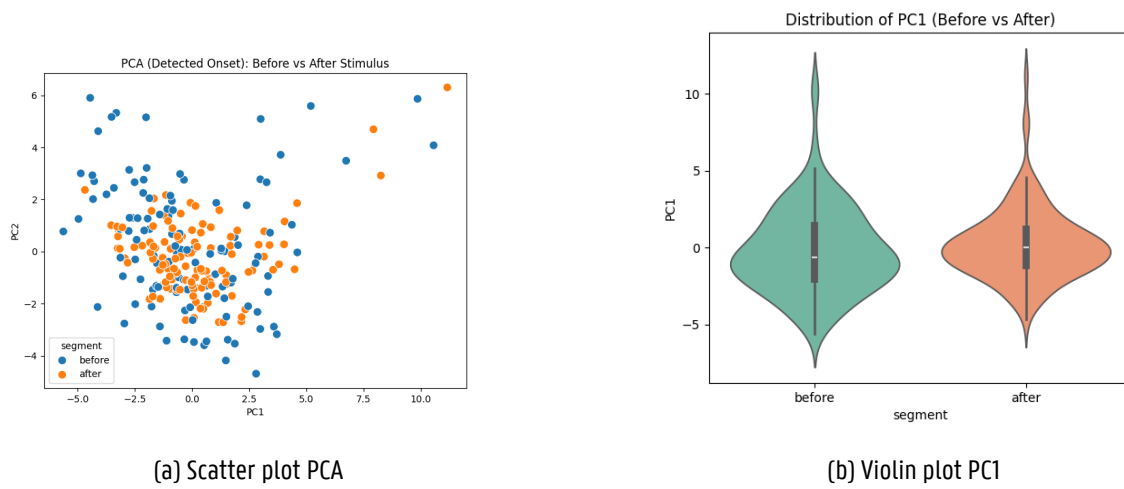


Figure 6.30: Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 2 s before and after the stimulus

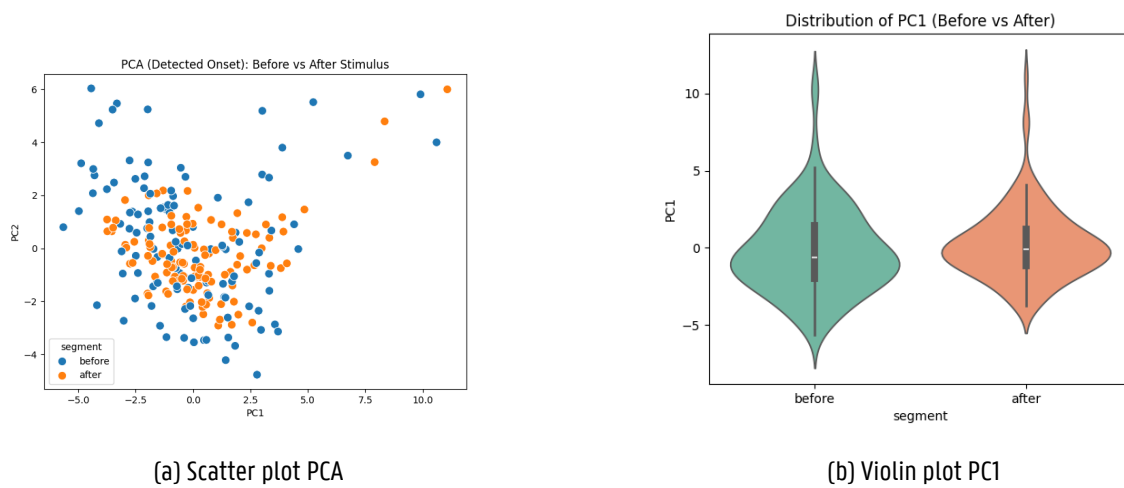


Figure 6.31: Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 3 s before and after the stimulus

6 Sound stimulus

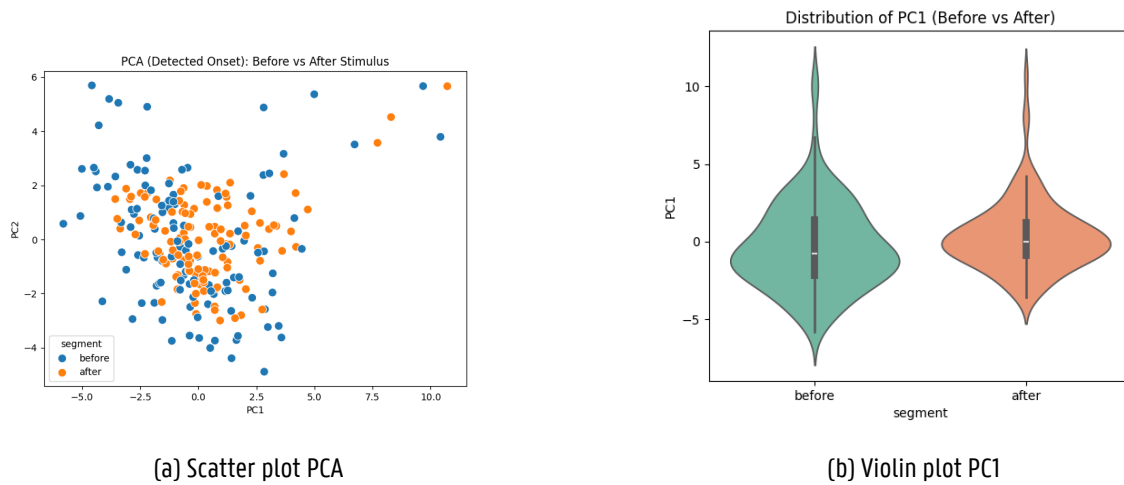


Figure 6.32: Scatter plot and violin plot displaying the results of the principal component analysis and paired t-test on a time frame of 5 s before and after the stimulus

Figure 6.25 to Figure 6.32 show the scatter and violin plots in their respective time frames. In all of the pictures, the data points from before the stimulus overlap with the data points after the stimulus. With the naked eye, it's hard to see clear clustering in two groups. But for the plots at a time frame of 0,1 to 0,5 seconds, there does seem to be more clustering than on the other images. This corresponds partially to the values found in Table 6.3. There, the p-values for the time windows from 0,1 to 0,3 seconds have some of the lowest p-values which shows the difference of the signal before and after the signal is strong in these windows. A window of 0,01 seconds is probably too short to detect any change in the signal. The time windows longer than one second still return p-values that indicate significant differences, but on the plots it is harder to see this difference. The p-value does seem to get smaller for the bigger time frames as well, such as time frames of three and five seconds. The reason for this could be that as the windows grow larger, the chance that some background noise is captured grows. Background noise in the part before or after the stimulus would trigger a false significant result. The reason is that the signal change would not come from bees. The results appear to show that the sound stimulus triggers a response from the bees. This response does not seem to be very strong and maybe not perfectly timed for each recording. But there is a slight change in the signal that makes the PC1 component vary significantly after the stimulus.

The next step is to make a comparison for the different frequencies. This will help to clarify whether bees have a stronger reaction to a certain frequency or not. This comparison is done by running a paired t-test for PC1 again, but this time for each frequency. Table 6.4 shows the results of the paired t-test. These results are also visualized in Figure 6.33. Figure 6.34 and Figure 6.35 are the result of the same t-test but only for the recordings at 35% volume and, respectively, 70% volume.

Table 6.4 shows that all p-values are less than 0,001 which means the results are significant.

6 Sound stimulus

Table 6.4: Paired t-test results for each frequency

Frequency [Hz]	<i>t</i> -value	<i>p</i> -Value	<i>n</i>
500	7,02	< 0,001	15
1000	7,15	< 0,001	15
1500	11,50	< 0,001	15
2000	7,59	< 0,001	14
2500	8,41	< 0,001	15
3000	6,63	< 0,001	15
3500	8,19	< 0,001	15
4000	10,96	< 0,001	15

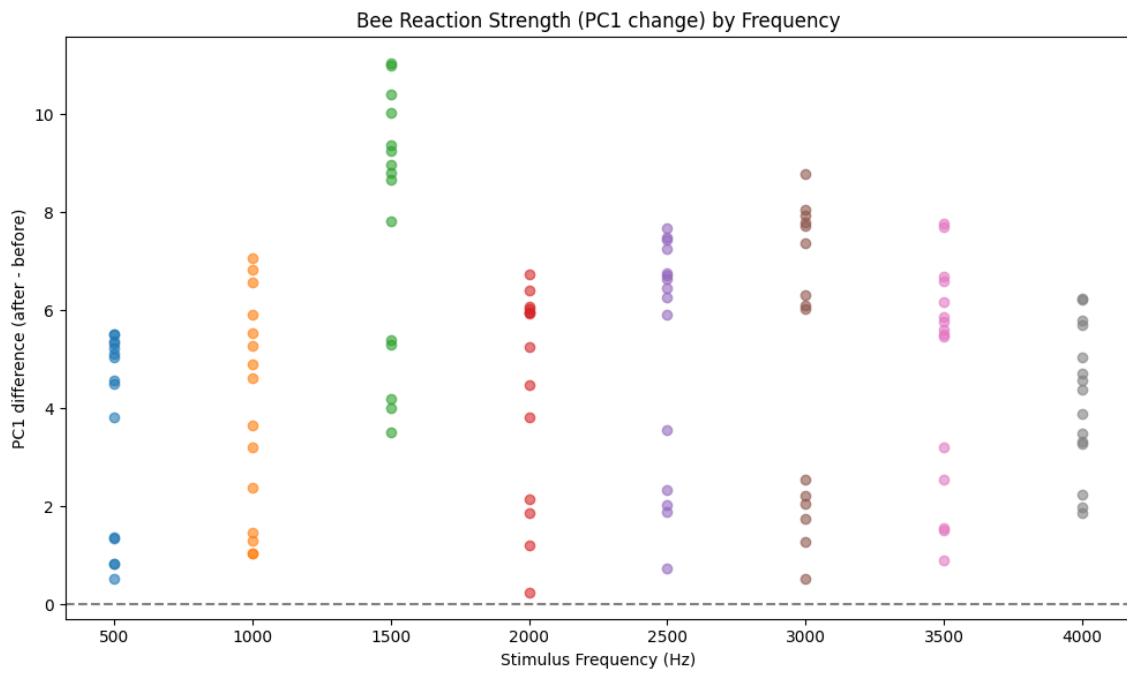


Figure 6.33: PC1 shift for each frequency

6 Sound stimulus

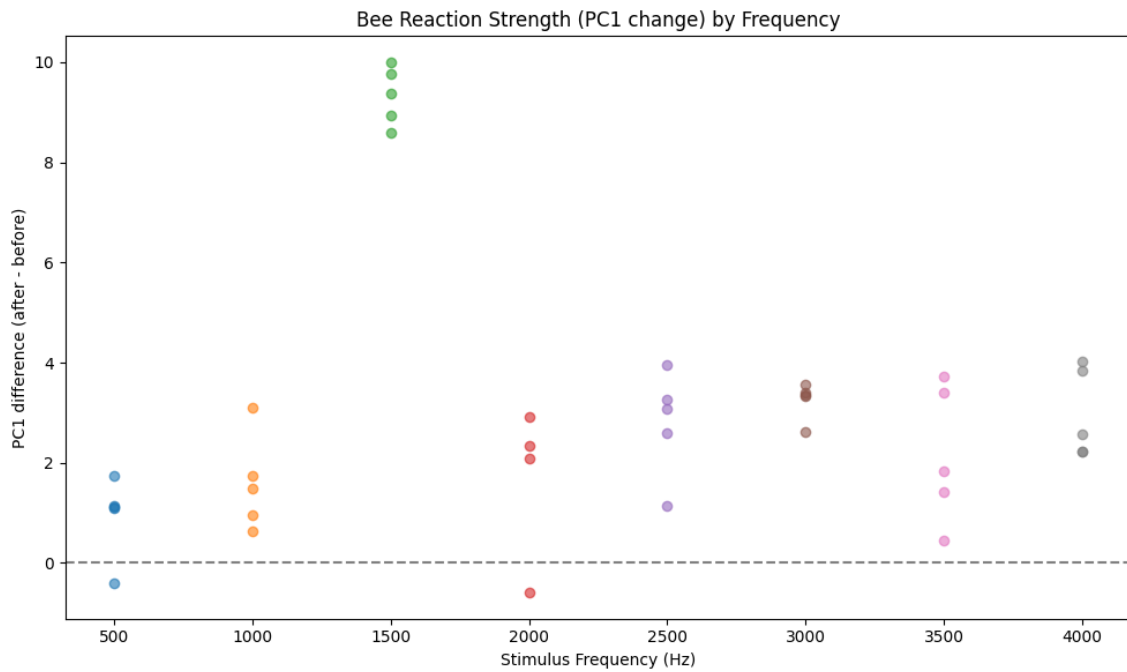


Figure 6.34: PC1 shift for each frequency at 35% volume

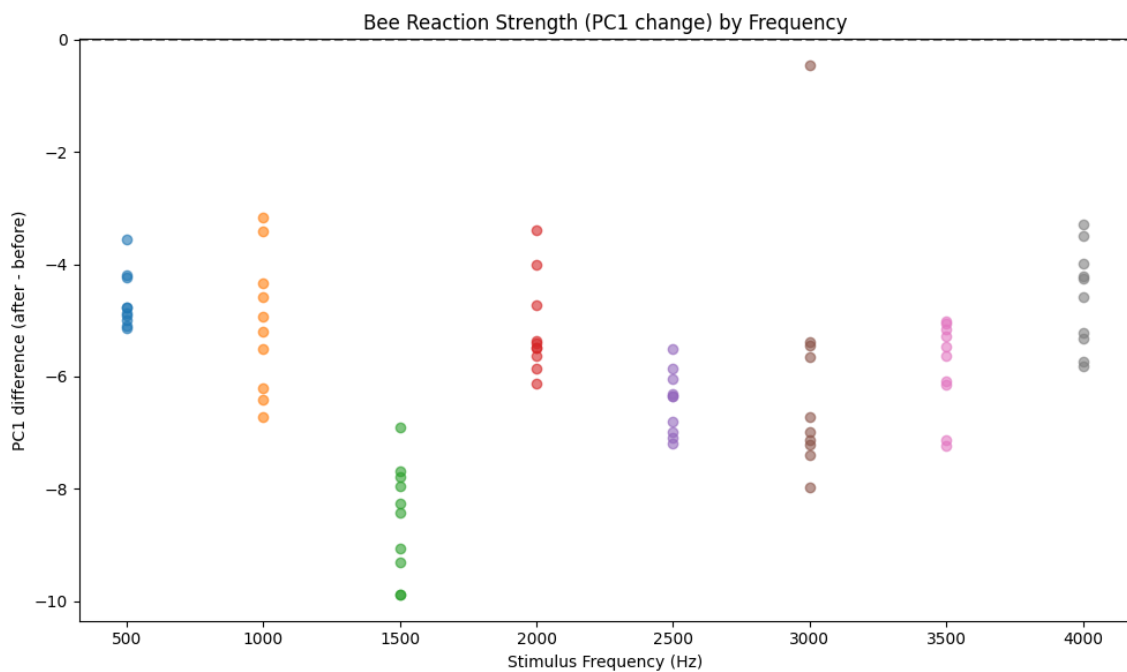


Figure 6.35: PC1 shift for each frequency at 70% volume

Figure 6.33 shows that the bees consistently react to the stimulus at each frequency. The further the points are

6 Sound stimulus

from the zero line the more significant the reaction is. The significance of the change after the stimulus appears to be consistently greater at 70% of the volume. Between the different frequencies, there are slight differences in significance, but mostly there is a lot of overlap between the data points. The recordings at 1500 Hz seem to show the biggest difference in signal before and after the stimulus. To check if there is a significant difference in delta PC1 or delta PC2 between the different frequencies, a one-way ANOVA is run for both principle components. Delta PC1 is the difference in PC1 before and after the stimulus, the same counts for delta PC2.

Table 6.5: One-way ANOVA on delta PC1

Source	SS	df	F	p-value
Frequency	27,57	7	1,34	0,24
Residual	328,36	112	–	–

Table 6.6: One-way ANOVA on delta PC2

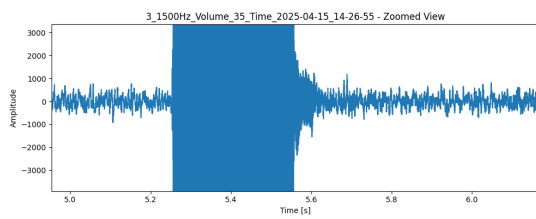
Source	SS	df	F	p-value
Frequency	94.05	7	3.12	.005
Residual	482.48	112	–	–

Table 6.5 and table 6.6 show the results of the one-way ANOVA tests on delta PC1 and delta PC2. These tables show a few different columns and rows. The first column gives the source of the variability. The variability either comes from the difference between the groups, these are the different frequencies here, or the variance is unexplained and comes from within-group variance, this makes it residual. The second column shows the sum of squares which says how much the variance is explained by each source. The next column shows the degrees of freedom. The first row says seven in both Table 6.5 and Table 6.6, this is because there are eight different frequencies and eight minus one equals seven. The second column shows 112, this is the total sample size of 120 minus the amount of groups which is eight. The next column shows the F-statistic, this is the ratio of between-group variance and within-group variance. The last column shows the p-value which explains the probability of observing an F this large if all groups had the same mean. Again, if the p-value is smaller than 0,05 this means the difference is significant. The results in Table 6.5 and Table 6.6 shows that there is only a significant difference in delta PC2 and not in delta PC1. So this proposes that there is a different reaction for the different frequencies, this difference appears in PC2.

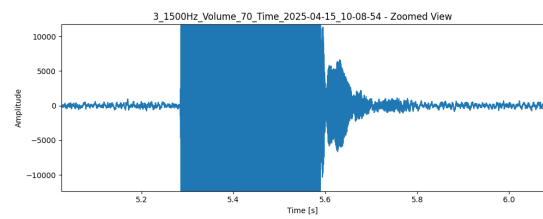
These results appear to indicate that there is a reaction from bees to the stimulus. All of the above tests return p-values that show the significance of the difference in the signal before and after the stimulus and even between the different frequencies. However, these results might not be completely trustworthy. When a closer look is

6 Sound stimulus

taken at the plots of the recordings, the signal has a larger rms value just after the stimulus. This is seen for all recordings at 70% volume and for some at 35% volume but is less strong. The problem with this is that this peak is also visible in recordings without bees. This indicates that the increase comes from the speakers and not from the bees. The increased rms just after the stimulus is probably a resonance phenomenon which is higher at increased volume. Figure 6.36 and Figure 6.37 show the waveforms from recordings with bees and Figure 6.38 and Figure 6.39 show the waveforms from recordings without bees. These plots show that the resonance phenomenon returns for the recordings without bees as well. But at first sight it does seem less strong. This visual assessment says nothing, but it gives hope that there might still be a response from the bees after the stimulus at this peak as well.

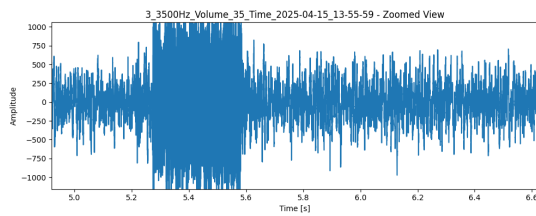


(a) 1500 Hz, 35% volume

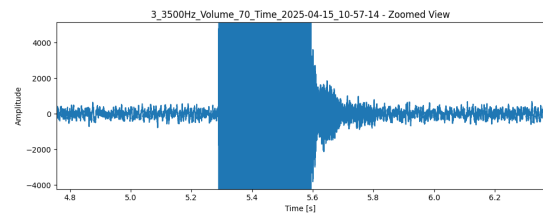


(b) 1500 Hz, 70% volume

Figure 6.36: Plots of two recordings at 1500 Hz for both volume levels with bees

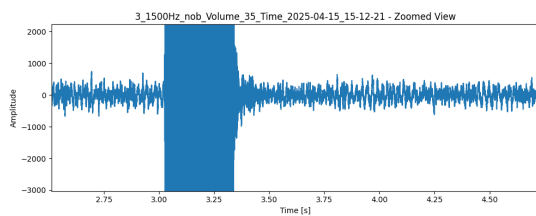


(a) 3500 Hz, 35% volume

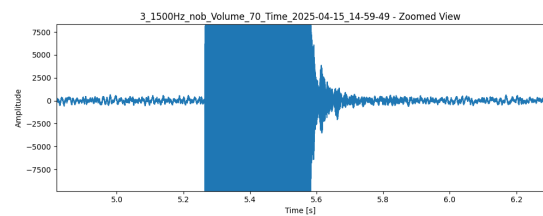


(b) 3500 Hz, 70% volume

Figure 6.37: Plots of two recordings at 3500 Hz for both volume levels with bees



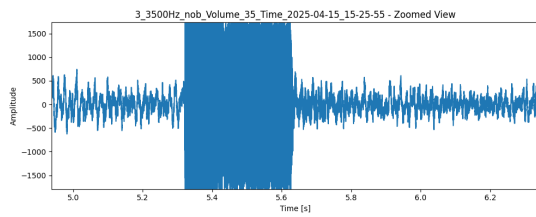
(a) 1500 Hz, 35% volume, without bees



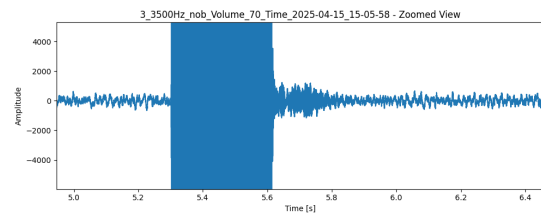
(b) 1500 Hz, 70% volume, without bees

Figure 6.38: Plots of two recordings at 1500 Hz for both volume levels without bees

6 Sound stimulus



(a) 3500 Hz, 35% volume, without bees



(b) 3500 Hz, 70% volume, without bees

Figure 6.39: Plots of two recordings at 3500 Hz for both volume levels without bees

In the next step, the tests will be done again, but now the resonance phenomena are left out of the analysis. The final vibrations of the box never seem to last longer than 0,2 seconds after the stimulus. To remove this signal pollution, the analysis will only look at the signal 0,2 seconds after the stimulus. Table 6.3 shows that there is not much difference in significance for the different time windows. For the next steps, a time window of three seconds will be used. The number of principal components will also increase from two to five components.

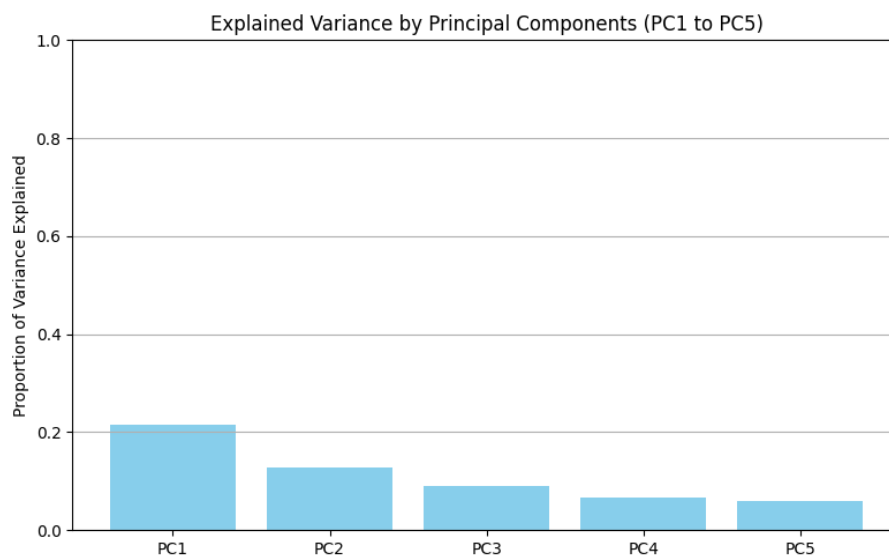


Figure 6.40: The explained variance for the first five principal components

Figure 6.40 shows the variance that each of the five principal components explains. None of them explain very much, but PC1 does explain significantly more than the others.

6 Sound stimulus

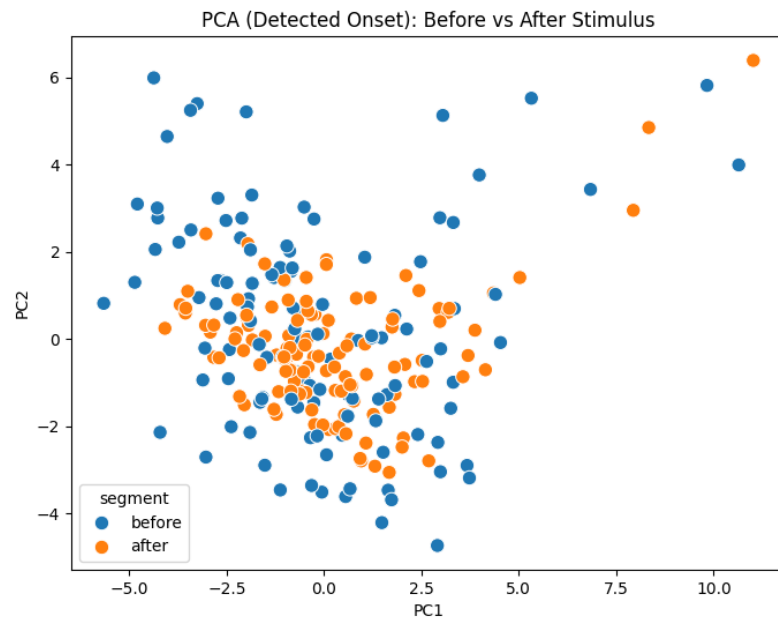


Figure 6.41: Scatter plot PCA 0,2 seconds after the stimulus

Figure 6.41 shows the scatter plot of the data points for PC and PC2. Again, visually it is not possible to discriminate two separate clusters.

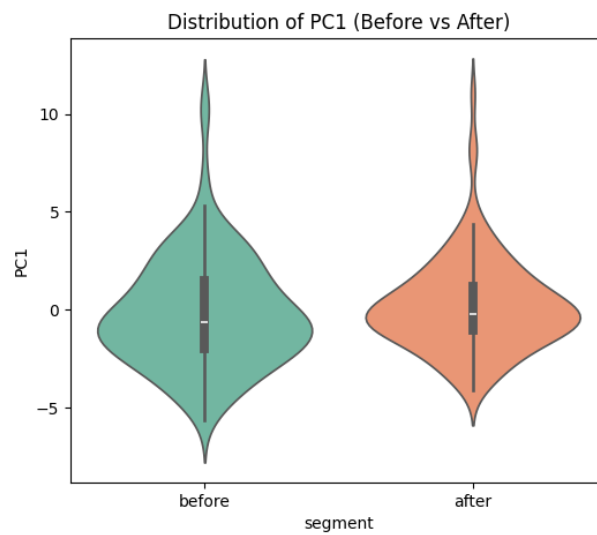


Figure 6.42: Violin plot for PC1 0,2 seconds after the stimulus

6 Sound stimulus

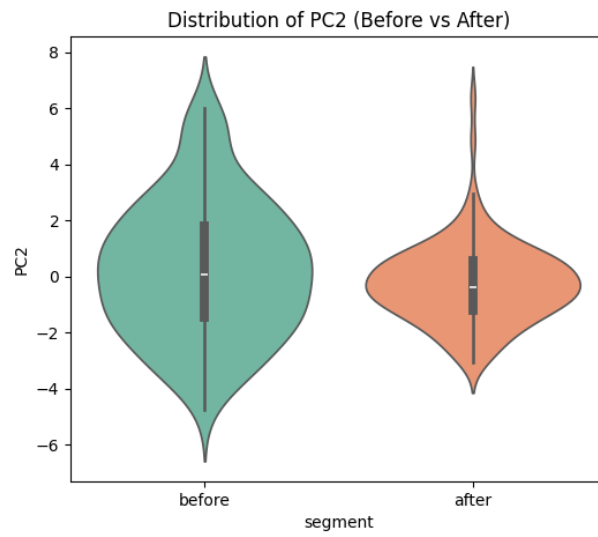


Figure 6.43: Violin plot for PC2 0,2 seconds after the stimulus

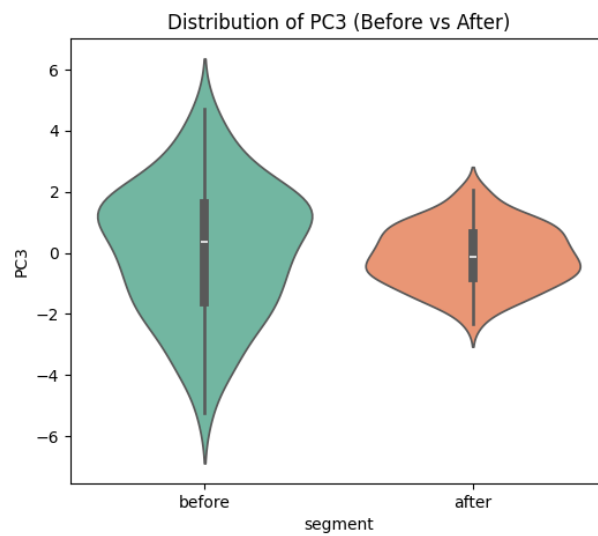


Figure 6.44: Violin plot for PC3 0,2 seconds after the stimulus

6 Sound stimulus

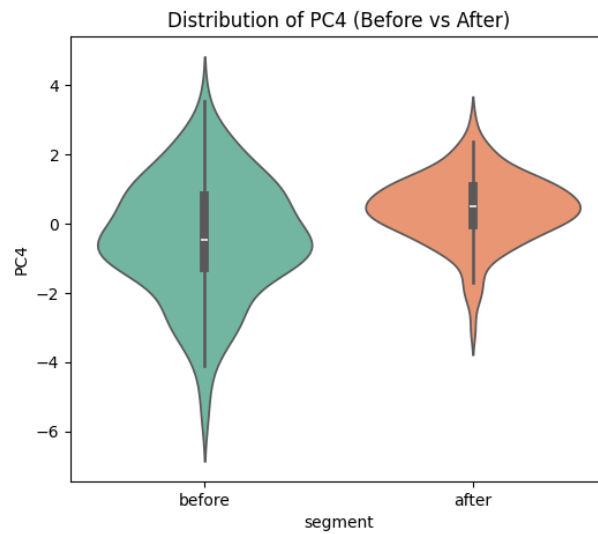


Figure 6.45: Violin plot for PC4 0,2 seconds after the stimulus

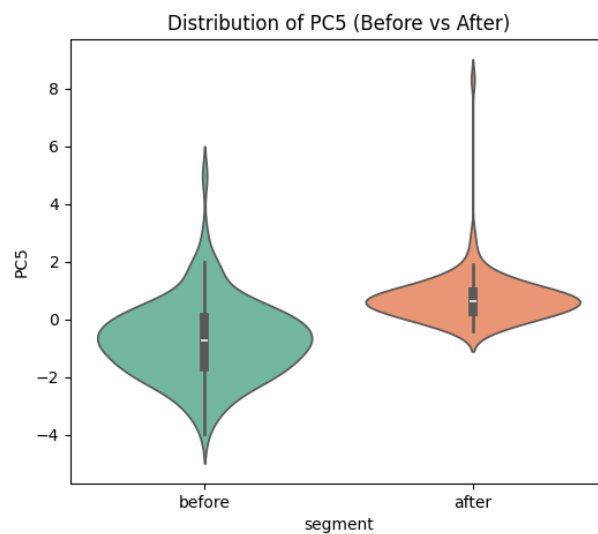


Figure 6.46: Violin plot for PC5 0,2 seconds after the stimulus

Figure 6.42 to Figure 6.46 show the violin plots for the different principal components. These show the distribution of the PC values before and after the stimulus. The values of all PC components seem to be closer together after the stimulus than before the stimulus. This indicates that the bees react to the stimulus. So, the three seconds after the stimulus do vary from the three seconds before the stimulus. A t-test on the PC1 values returns a t-value of -2,39 and a p-value of 0,019 which shows that the difference before and after the stimulus is significant. But the medians of the violin plots are not far apart, which shows that the difference is not that big.

6 Sound stimulus

Now, the test to see if the frequencies deliver different results is also repeated. Figure 6.47 shows the results. All data points are around the zero line. This indicates that none of the frequencies show a significant difference in the PC1 value before and after the stimulus. A t-test that is performed for all these frequencies delivers p-values that are all larger than 0,05 except for 4000 Hz. So, none of the results show a significant difference except for 4000 Hz. But the p-value for 4000 Hz is only just smaller, so it is still not a big difference. There also does not seem to be a difference between the frequencies. There are some small variations, but none of them produce significant results.

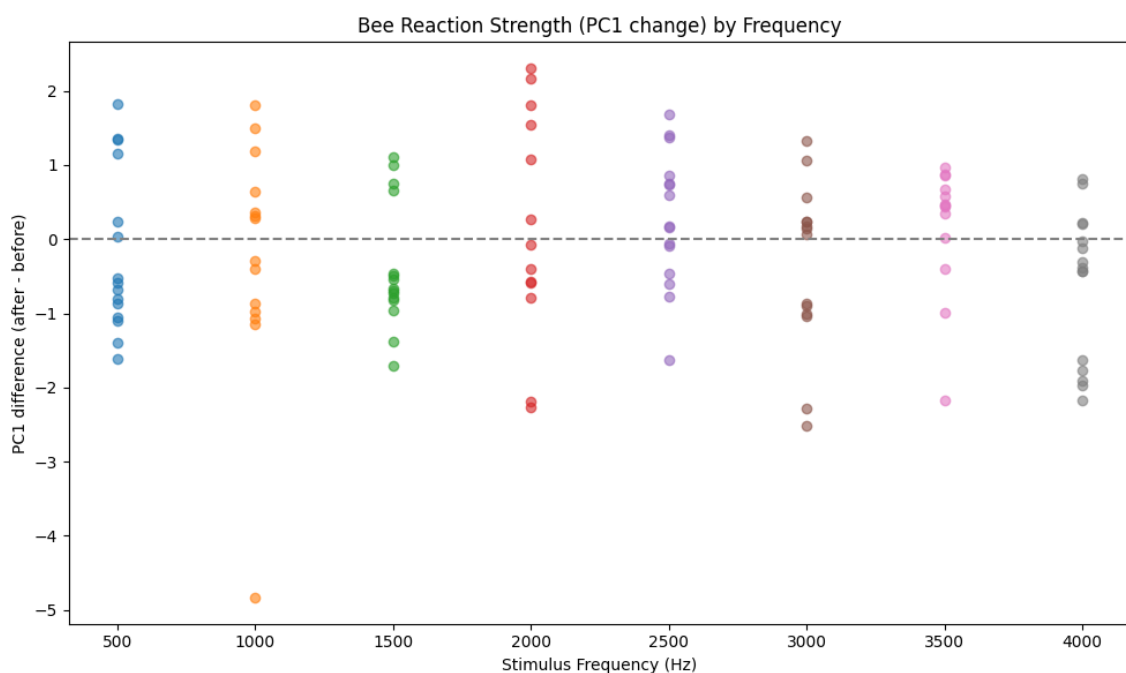


Figure 6.47: PC1 shift for each frequency 0,2 seconds after the stimulus

The next step is to perform a MANOVA (multivariate analysis of variance) test. This test is used with five dependent variables, the five principal components, and one independent variable, the segment before and after the stimulus. The MANOVA tests if the combination of these five PC's differ significantly between before and after the stimulus. MANOVA returns several multivariate statistics. Wilks' Lambda shows the proportion of the total variance that is not explained by group differences. Pillai's Trace shows the sum of explained variances. Hotelling-Lawley shows the same as Pillai but is more significant to group differences. And Roy's Greatest Root is based on the strongest linear combination. It focuses on the largest difference across any single dimension. Table 6.7 shows the results of the MANOVA.

The results of the MANOVA show that there is a strong statistically significant difference between the segments before and after. Wilks' Lambda shows that about 40% of the variance in the PCA features is explained by the difference between the segment before the stimulus and the segment after the stimulus. The other statistics

6 Sound stimulus

Table 6.7: MANOVA Test Results for Segment

Test	Value	Num DF	F	p
Wilks' Lambda	0.60	5	31.77	< .001
Pillai's Trace	0.40	5	31.77	< .001
Hotelling-Lawley Trace	0.68	5	31.77	< .001
Roy's Greatest Root	0.68	5	31.77	< .001

confirm this. The p-values are all smaller than 0,05 which confirms the results are significant. This MANOVA shows that the stimulus triggers a reaction of the bees. This means that MANOVA can be used to determine the strength of the reaction from a bee colony to the sound stimulus. This makes it a tool for comparing different colonies of bees with each other by looking at the different reactions to the stimulus.

6.4 Practical implementation

This section will provide more information about the techniques and codes that are used to perform the sound stimulus test and analyze its results.

To play the stimulus and record the sound, a Python script is used on the Raspberry Pi. The Pi is connected to both speakers via USB and a USB microphone. Both devices work simultaneously when the script is running. First, the recording starts. After five seconds, the stimulus plays. The stimulus is a tone of one single frequency and lasts 0,3 seconds. The microphone continues to record until ten seconds after the end of the stimulus. To make the speaker and microphone work simultaneously, the speaker must be connected to the Pi first. This speaker must be set to the default output device. The microphone can then be connected. The code also needs the card and device number of the microphone so it can select it when recording starts. The recordings are made using arecord and the stimulus is played using aplay. This is how it works practically. The code is now used to play the stimulus once a day. The stimulus will play at a randomized time between nine a.m. and six p.m., the recording and a plot of the signal are saved in a folder on the Raspberry Pi.

For the analysis of the recordings, there are multiple Python scripts. The first script makes several graphs from some of the features of the signal. This script plots the RMS value of the signal, the spectral centroids, the frequency distribution before and after the stimulus, the waveform and a spectrogram of the signal. These are a few plots that can be used to examine the recordings separately.

The other scripts contain code to run the statistic test from section 6.3. The first script makes a csv file with the features of all the recordings. This csv file is then used in another script that runs a three-way ANOVA. That script returns the results from the ANOVA and makes box plots for the different features.

6 Sound stimulus

The next scripts are for the analysis using principal component analysis. The first step is to use the script that extracts the necessary features from the recordings and puts them in a csv file. This csv file is then used to make other csv files with the values for the different principal components. Using both csv files, a t-test can be performed to verify the significance of the results. There is also a script that plots the loadings of the PCA analysis. This shows the weights of the different features in the components. The next script is the one that compares the biggest one. This code extracts the features from the recordings and runs a principal component test on these features. It plots a violin plot for the results of the principal components before and after the stimulus. It also runs a t-test and shows the variance explained by each of the principal components. In the end, it also runs a MANOVA for all the principal components. In this script, a time window before the stimulus is compared with a time window after the stimulus. The time window can be adapted. Finally, there are two more scripts. These scripts are used for the comparison between the different frequencies. The first script runs a paired t-test for the PC1 values of the recordings. The next script runs an ANOVA on the results for the different frequencies to check if there is a significant difference between the frequencies.

All these scripts return data that can be used to make a decision as to whether or not the bees give a significant reaction. They can also be used to classify recordings with and without bees. These statistical tests can be very useful to generate data, but a conclusion can only be drawn with all the results combined.

6.5 Conclusion

The use of a sound stimulus appears to be useful if the goal is to detect the presence of bees. By using three-way ANOVA it is possible to detect a difference between recordings with bees and recordings without bees. Using a principal component analysis works even better in distinguishing the recordings with bees from the ones without bees. The PCA works better with the large set of features. The small one only contains the RMS ratio, the spectral centroids, the bandwidth, the spectral rolloff and the zero crossing rate. This is already enough to detect the presence of bees, but with a large set the results are much stronger. The large set contains mfcc features and chromas. These results show that the sound stimulus can be used to detect the difference in recordings with and without bees. This makes it possible to use the results in a classifier that can detect the presence of bees. If a starting set is used to train a classifier, this can be used to check a beehive for the presence of bees. This can be useful in winter to check if there are actually still live bees inside a beehive.

The next step was to see if there is a reaction of the bees to the sound stimulus. The comparison between recordings with bees and without bees was a first step towards this goal. Visual analysis showed some signs of a reaction from the bees but nothing definitive. Then a principal component analysis is performed. This analysis calculates the first two principal components. Then a time window before the stimulus is compared with the same time window after the stimulus. The length of this time window is changed for different repeats of the test. The length of the time window does not make a significant difference in the results, so the following tests continue

6 Sound stimulus

with a time window of three seconds. Because there is a signal increase that shows up right after the stimulus in both the recordings with and the recordings without bees, the time window only starts 0,2 seconds after the stimulus. Then a t-test shows that there is a significant difference in the PC1 value before the stimulus and the PC1 value after the stimulus. This means that the principal components analysis does seem to show that there is a reaction from the bees to the sound stimulus. The next step was to check the difference in the PC1 value before and after the stimulus for each of the frequencies separately. These results show that there is no significant difference because all the values are in the vicinity of the zero line. There are a few differences between the different frequencies, but there is no significant difference. Then, the principal component analysis is extended to the first five components. With these five components, a MANOVA is performed. The MANOVA examines whether there is a significant difference for all five of these components before the stimulus and after the stimulus. The MANOVA returns all significant p-values. This means that there is a significant difference for the principal components between the part before the stimulus and the part after the stimulus.

The results of the tests seem to indicate that there is a reaction from the bees. The reaction does not appear to be very strong or consistent. Principal component analysis shows that there is not that much difference before and after the stimulus. However, MANOVA proves that there is a difference in the signal before and after the stimulus. This indicates that there is a reaction from the bees. This MANOVA test can be used to compare the reaction of multiple hives with each other. If a sufficient number of recordings are made for a beehive, the MANOVA test can be performed. Looking at the results of the MANOVA, one can put a number on the strength of the response. The different values show how significant the difference is. The smaller the significance of the difference in the principal components, the weaker the reaction of the bees. This shows that the test can be used to compare different colonies with each other. Maybe weaker colony react differently than stronger ones. This makes it a possible tool for beekeepers. They can use this during winter to check how strong their colonies are.

However, this test must be examined further to deliver a clear answer to its usefulness. Currently, it has not been proven to reproduce a strong reaction from bees. The test with the sound stimulus should be performed in different beehives of different strengths during different periods of the year. This is necessary to qualify the strength of the reaction. If the test is carried out on more beehives, the reaction could be quantified on a scale between zero and one. This could give a more conclusive answer to the question of whether bees react significantly to the stimulus. It will also show whether there is a practical use bound to this reaction. Currently, the test does not provide any successful results. But care must still be taken with this test. During the test with the sound stimulus on a weaker beehive, the colony died completely. It is not sure whether the stimulus has anything to do with this or not. But there is always the possibility that the stimulus is harmful to bees. This aspect and the fact that the reaction of the bees appears to be minimal shows that the knocker test might be the better alternative between the two tests. The knocker test showed a stronger response than the sound stimulus test. This could also be due to the microphone. This microphone might not pick up enough details. If this is the case, an accelerometer might show more response. The use of the accelerometer could provide more precise measurements. The disadvantage of this accelerometer is that there must be cables inside the beehive.

7

Infra-red imaging

This chapter will handle all the tests about the use of an infra-red camera.

All data, code and results for this chapter can be found in Appendix G¹ infrared images.

7.1 Purpose

The following section discusses the experiments using an infra-red camera. For this experiment, an infra-red camera is set up underneath the beehive. There, it captures images of the bee colony. This data contains information about the temperature throughout the beehive. These infra-red images can show quickly where the majority of the bees are located in the hive. When more images are taken throughout the day, even more information can be extracted. The images then also contain information about the movement of the bees throughout the day. Even the temperature outside the hive can be captured with the infra-red camera. infra-red images can show the behavior of bees together with the temperature change throughout the day. This information may be useful for researchers and beekeepers. It can help them get a better understanding of the behavior of honeybees. It can also be used to capture the reaction of bees to certain circumstances. So, the thermal camera can definitely be a useful tool for inspecting the behavior of a bee colony. Thermal images can also be used to view the movement of bees through the different honeycombs inside the beehive. Another application is to use the thermal camera on different beehives made of different materials. Each material has a certain capacity to isolate the beehive and keep the temperature constant. The thermal data can be used to see if bees move more or less with the use of a different material for the beehive.

7.2 Test setup

For the tests, a MLX90640 thermal camera is used. Figure 7.1 shows one of these thermal cameras.

¹GitHub repository for sound stimulus test and tests with infrared imaging

7 Infra-red imaging



Figure 7.1: MLX90640 thermal camera

This camera is connected to the Raspberry Pi using the IO pins. The camera uses the I2C protocol to send its data to the Pi. For the tests a Raspberry Pi 3B+ is used. Using Python and the corresponding libraries to the thermal camera, the images were made and stored on the Pi. For the tests, this camera is placed underneath the beehive. This is done at a distance so that the camera also captures the surroundings of the beehive so that the external temperature can also be monitored. When the camera is set up it can be used to make constant images of the beehive at a frame rate of 0,5 fps or it can be used to make images at fixed times. The camera is used to take pictures every minute, so it captures a lot of bee movement.

7.3 Results

Several different features are extracted from the thermal images. First, the images are analyzed as a whole. Later, the images will be compartmentalized in different zones. These zones will include the space between the honeycombs inside the beehive. But first the image will be examined in its entirety.

The first thing that is made with the results is a GIF that displays all the images. The camera takes pictures every minute. By placing these images after one another in a GIF format, the overall movement in temperature inside the beehive is displayed. The edges also show the change in the outside temperature throughout the day. This can be used to link the movement of the bees to the time of day. Figure 7.2 shows one of the thermal images that is used to make a GIF. This image shows a clear difference in temperature inside the beehive and outside the beehive. The camera captures images at a resolution of 32x24 pixels, but the captured data are interpolated, so the image is smoothed over 160x120 pixels.

7 Infra-red imaging

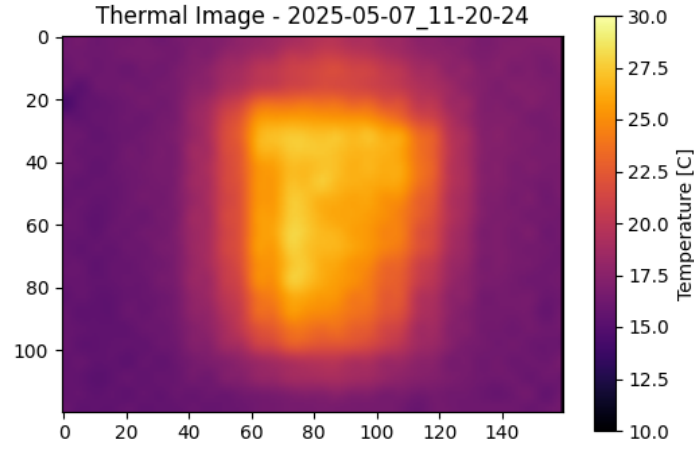


Figure 7.2: Thermal image used in GIF

The first analysis performed on the thermal data is to track the position of the bees through the beehive. This is done by tracking the hottest point in the data. But it is not simply the hottest point that is taken for this analysis, as this would make the results sensitive to outliers. So, instead, the analysis uses a mean position with the temperatures as weights as shown in Equation 7.1.

$$(x, y)_G = \sum_{i=1}^N \frac{T(x, y) * (x, y)}{\sum_{i=1}^N T(x, y)} \quad (7.1)$$

This is the formula that is used to calculate the coordinates of the hottest point in the thermal image. N is the number of pixels and T is the temperature value at a certain point. In this way, the temperature of all pixels is taken into account. Figure 7.3 shows all the calculated points for every minute for the entire day. The points are all very close to each other. In the x direction they lay between pixel 15 and 15,6 and in the y direction they lay between pixel 11,15 and 11,6. The x-direction has a maximum of 32 pixels, and the y-direction has a maximum of 24 pixels. This shows that the hottest point was consistently in the middle of the beehive on this day.

7 Infra-red imaging

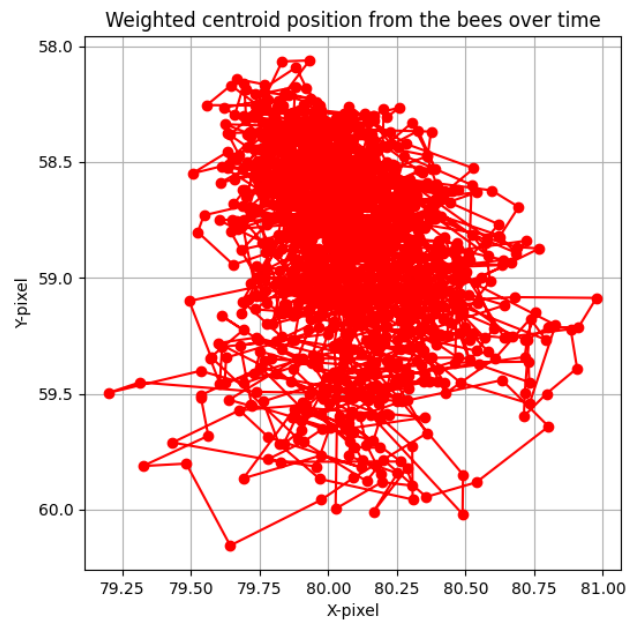


Figure 7.3: Hottest location in the hive throughout a day

Figure 7.4 shows the plot of the hottest points, but in one of the thermal images. This is to give a better understanding of the scale.

7 Infra-red imaging

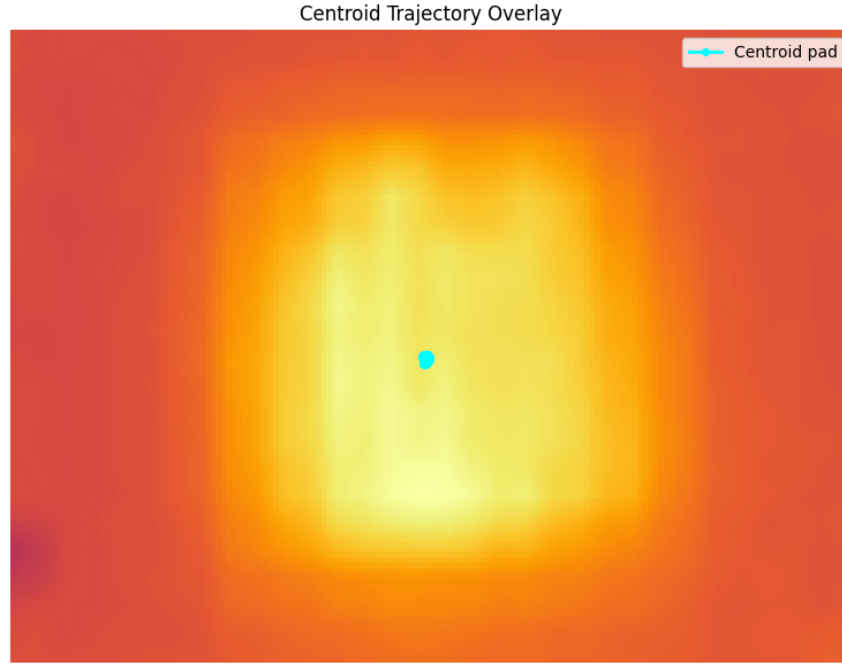


Figure 7.4: Hottest location in the hive throughout a day on a thermal image

This analysis can show where bees cluster during winter periods. At such times, it is not possible to open the beehive. This can be useful for beekeepers so that they know where the bees are. It also gives a sign of how large the colony is. This can help decide whether the colony needs extra food during winter and where this food can be placed.

The next step is to calculate the standard deviation of the temperature values. This is done using Equation 7.3, supported by Equation 7.2.

$$var = \frac{\sum_{i=1}^N T(x, y) * [(x, y) - (x, y)_G]^2}{\sum_{i=1}^N T(x, y)} \quad (7.2)$$

$$\sigma = \sqrt{var} \quad (7.3)$$

The standard deviation is a spatial measure defined in a number of pixels. It shows how widely the temperature is distributed around the thermal centroid by returning a value with the number of pixels as unit. Figure 7.5 shows the standard deviation for an entire day. For example, a value of 55 means that the average heat is spread out over 55 pixels around the calculated weighted thermal centroid. A high value means that heat is spread more and that the bees are less clustered. A lower value means that the bees are more concentrated together. This graph should have a shape similar to the outside temperature. Because at night the bees are less active and are tightly

7 Infra-red imaging

clustered. This would result in a low standard deviation. At night, the temperature will also be lower. So, both graphs should show a minimum at night.

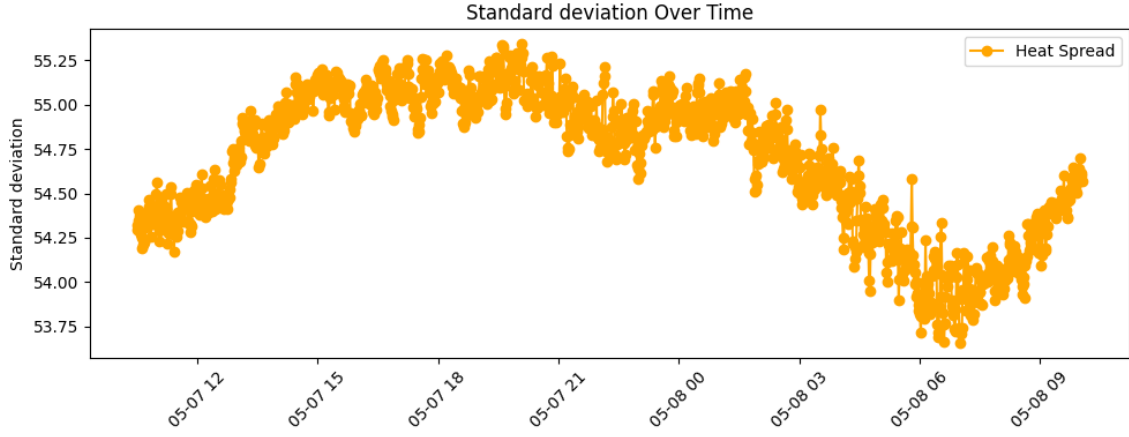


Figure 7.5: Standard deviation of the position

To show the variation of the thermal centroid from its mean position, Figure 7.6 is made. This is done by calculating the Euclidean distance from the mean position from all the thermal centroid positions. The distance is shown in a number of pixels and is computed through Equation 7.4.

$$distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (7.4)$$

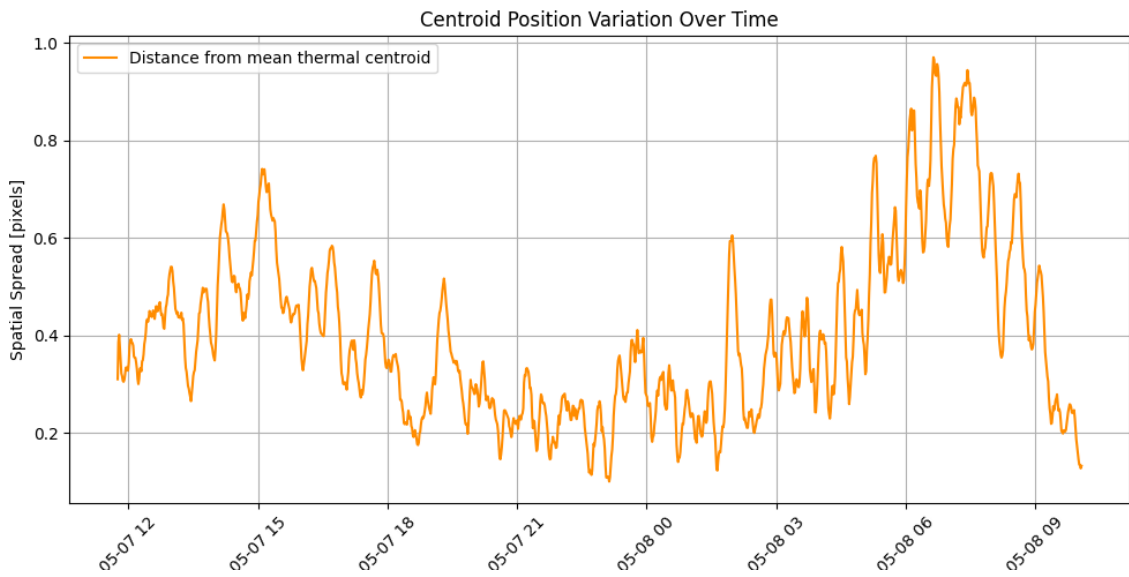


Figure 7.6: Distance from mean thermal centroid

7 Infra-red imaging

The next step is to calculate the temperature of the surrounding air using the edges of the thermal images. The outside temperature is calculated by looking at the band of three pixels at the top and bottom of the image. Figure 7.7 shows the results. This image has a very similar shape as Figure 7.5 as expected. Both show a minimum at night and a maximum in the afternoon. This graph shows the temperature outside the beehive. This helps to interpret the movement of the bees. When its colder outside, the bees will fly less and will be clustered inside.

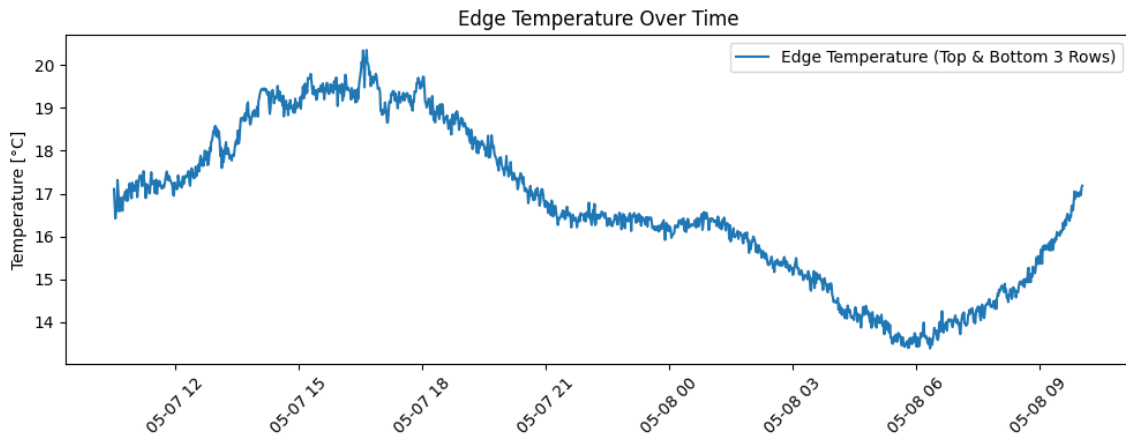


Figure 7.7: Temperature at the edges of the thermal

The next analysis looks at how many clusters there are inside the beehive and how large these clusters are. The results are shown in Figure 7.8 and Figure 7.9. This shows all clusters that have a temperature higher than 14 °C. This is done by running through all the pixel values in an array. All pixels that have a temperature value greater than 14, the chosen threshold, are set to True in a binary mask. Then the binary image is scanned and all the groups of adjacent True pixels get a label. In this way, it is possible to count all the different labels and get the number of clusters. To get the size of the clusters, the number of pixels with the same label is counted.

There seem to be more clusters at night. This means that during the day the bees are spread out more evenly through the beehive.

7 Infra-red imaging

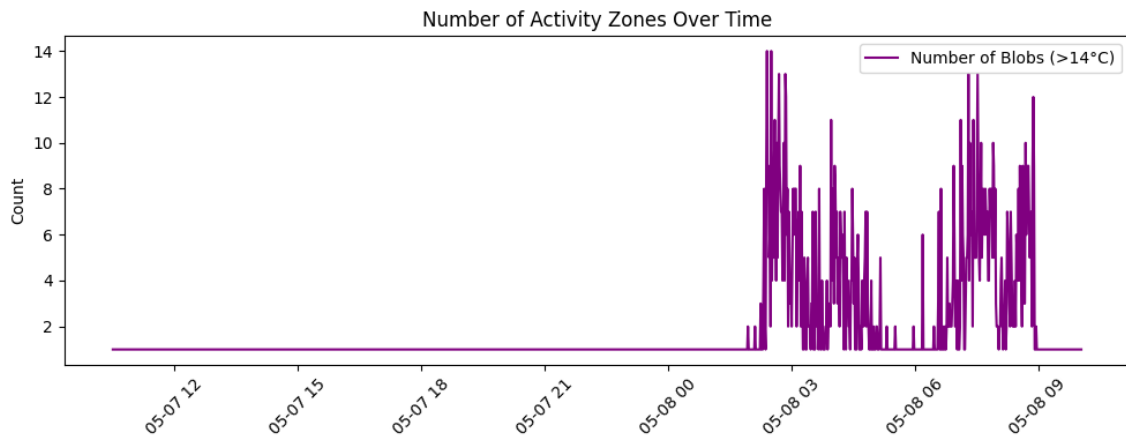


Figure 7.8: Number of clusters in the beehive

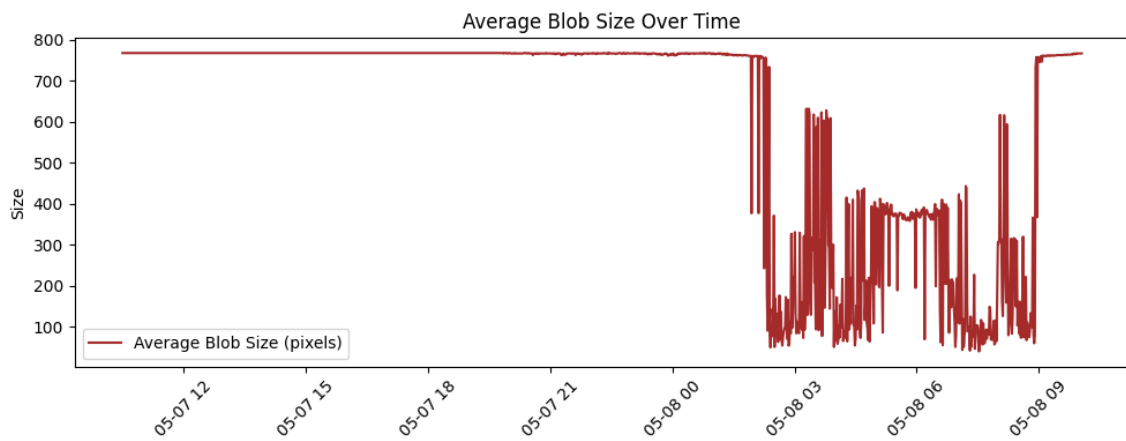


Figure 7.9: Size of the clusters

Now, the analysis will start looking at the individual zones in the beehive. These zones are the spaces between the honeycombs. One beehive contains eleven honeycombs. The thermal images will be divided into fourteen zones. These zones are the twelve spaces between the honeycombs and the zones left and right of the beehive. These last two zones do not contain valuable information about the bees. They serve as a reference for the other zones. Figure 7.10 shows one of the thermal images that is divided into the different zones. The blue lines show the honeycombs and the white lines show the edges of the beehive.

7 Infra-red imaging

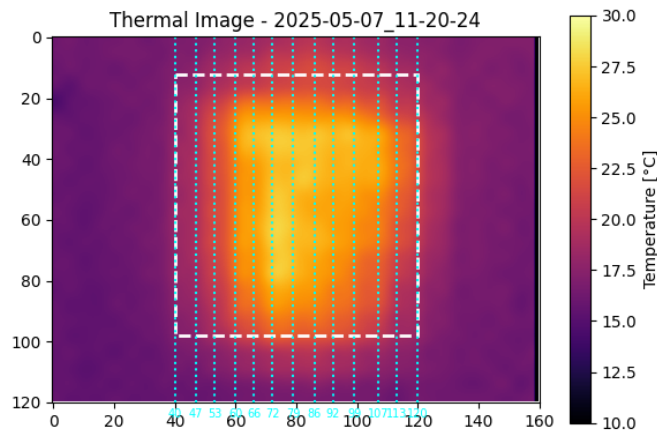


Figure 7.10: Thermal image divided in zones

Now an analysis is performed to look at several features within these zones. First, the mean temperature in each zone is shown. Figure 7.11 shows the results of this test. This image shows the data for one day. These curves show a similar trend to Figure 7.7. They look the same, but for different temperature values. Zone one and zone fourteen are the two zones outside the beehive. These are the two lowest curves, showing that its hotter inside the beehive. The other zone curves are closer together, but show differences as well. The zones in the middle of the beehive give the hottest values. However, these also seem to show more variations in temperature. This is probably because there is more movement from the bees there. This causes the bigger variations in temperature.

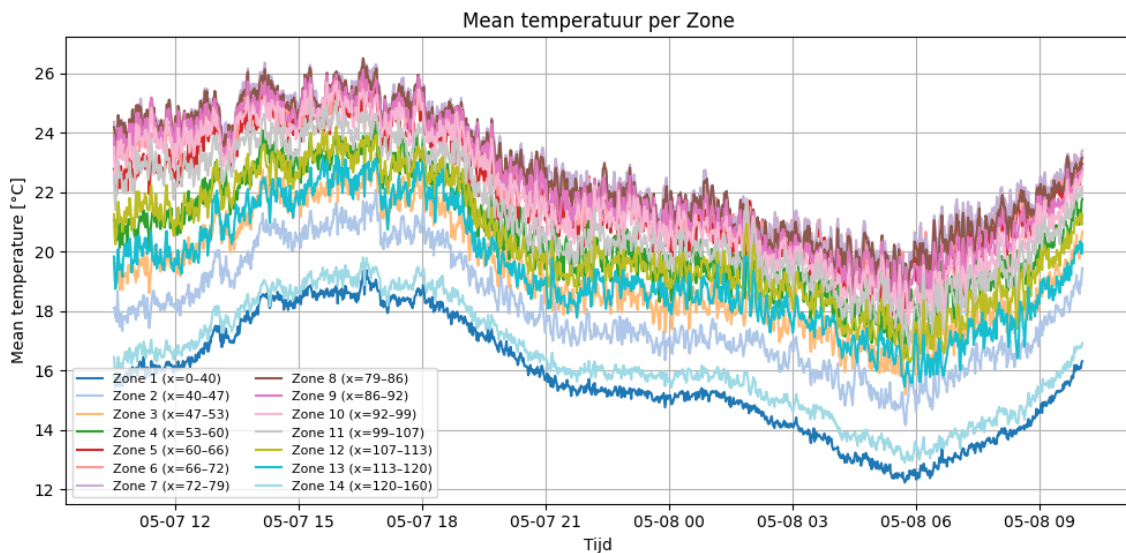


Figure 7.11: Mean temperature for each zone

Then, the standard deviation for the temperature in each zone is calculated. The same technique as for the whole

7 Infra-red imaging

thermal image is also used here. Figure 7.12 shows the different standard deviations for the fourteen different zones. This confirms the findings of Figure 7.11. The zones in the middle of the beehive vary more in temperature than the ones on the borders.

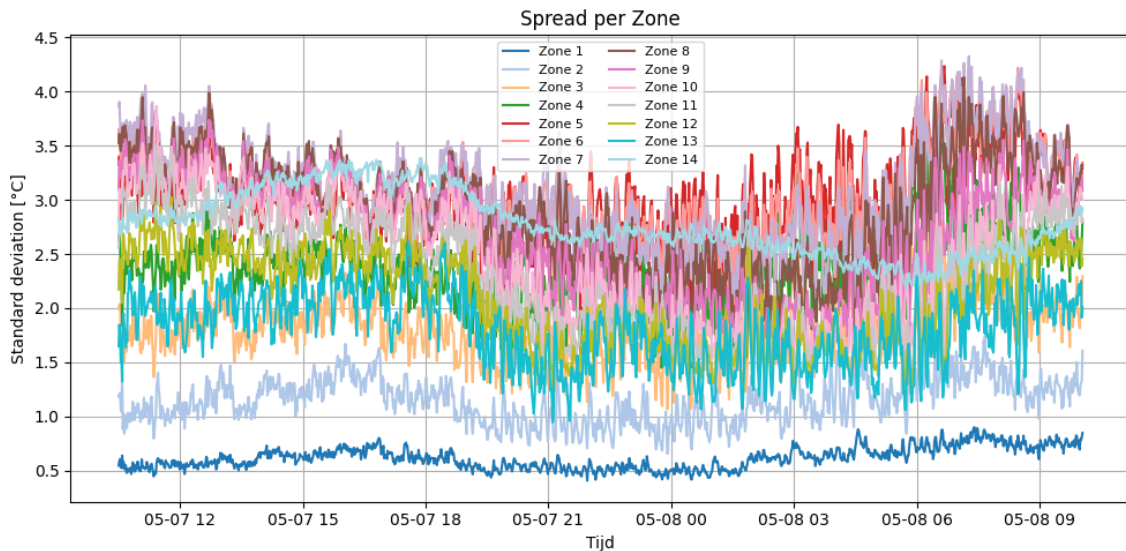


Figure 7.12: Standard deviation for each zone

The next step is to examine the movement of the bees through the different zones. The same method that is used to track the location of the bees in the complete thermal image is now performed on all the different zones. This gives a graph for the movement in the y-direction and a graph for the movement in the x-direction. Figure 7.13 and Figure 7.14 show these graphs.

7 Infra-red imaging

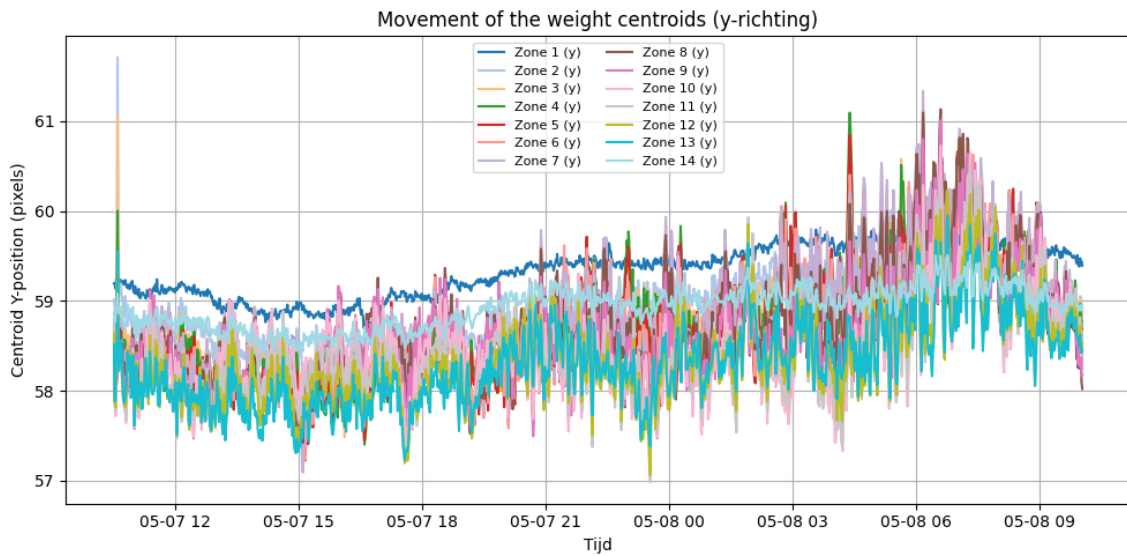


Figure 7.13: Movement in y-direction for each zone

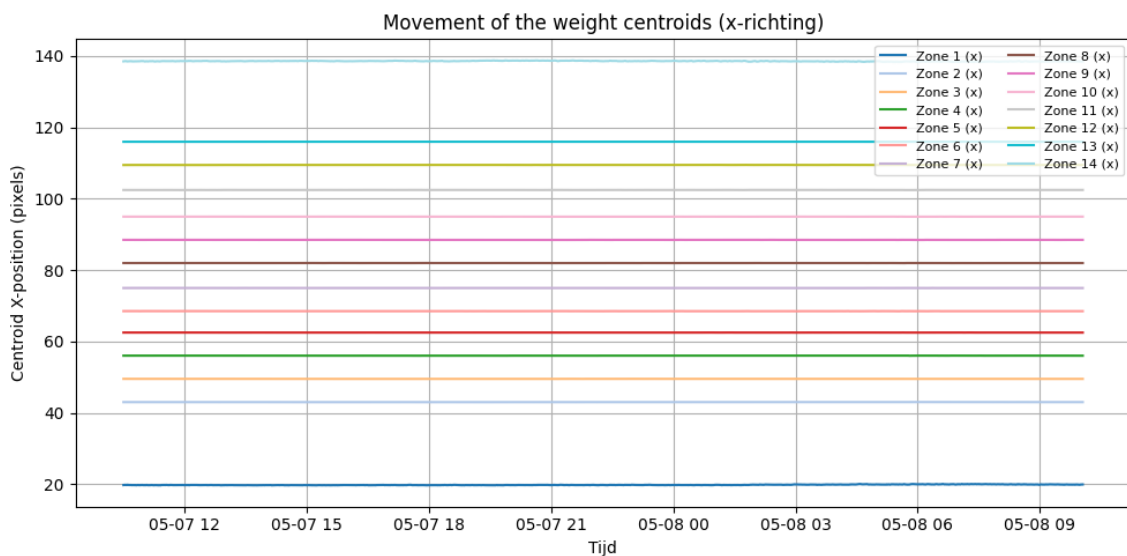


Figure 7.14: Movement in x-direction for each zone

Figure 7.14 which shows the movement in the x-direction, just shows straight lines. This makes sense, the space between the honeycombs is very narrow, so the bees can only move up and down. Figure 7.13 shows more results. There is a clear difference between zone one and zone fourteen compared to the other zones. The other zones show much more movement. The graphs show that the hottest part in each zone moves up and down around a certain point. The overall hottest zone does seem to move upward during the night. All of these results are the

7 Infra-red imaging

results of data for an entire day. This technique can also be used on data for shorter durations of time to see specific movement to an event. Or over a longer period of time to see the overall movement of the bee colony.

Figure 7.15, Figure 7.16 and Figure 7.17 show the movement in the y-direction, the mean temperature per zone and the standard deviation of the temperature for each zone for a duration of ten minutes. Figure 7.15 shows that the bees move up in several zones to move down again near the end. This is also possible to see in the thermal images. Some of the other zones show that the bees are moving down. so not all zones move together in the same direction. The temperature seems to follow the same trend for all zones. Only the mean value is different. The standard deviation is pretty constant for a window of ten minutes. but after seven minutes the standard deviation shows a minimum. This minimum shows up in several zones, but not in all of them.

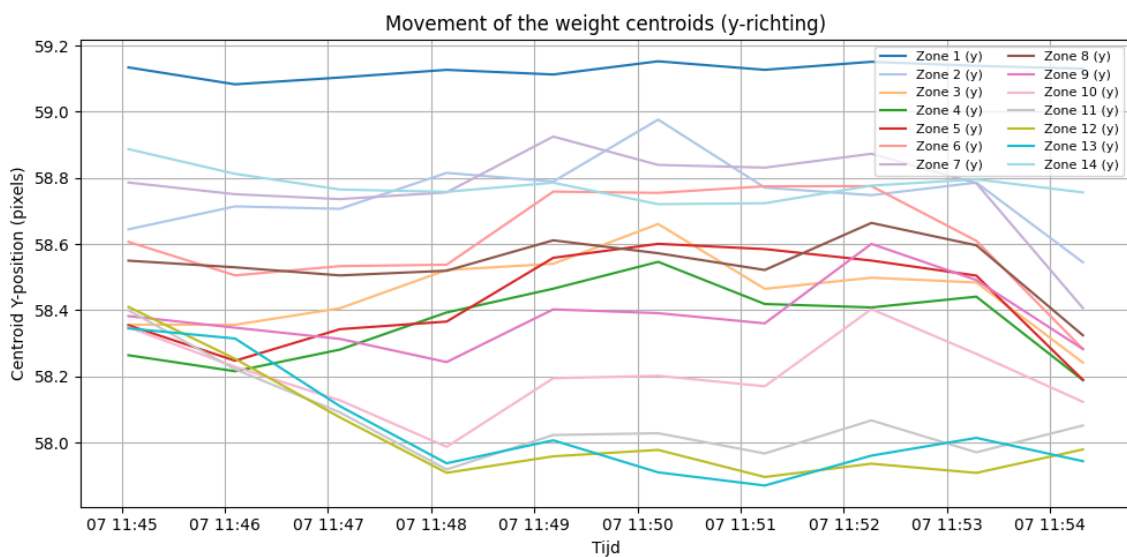


Figure 7.15: Movement in y-direction for each zone during ten minutes

7 Infra-red imaging

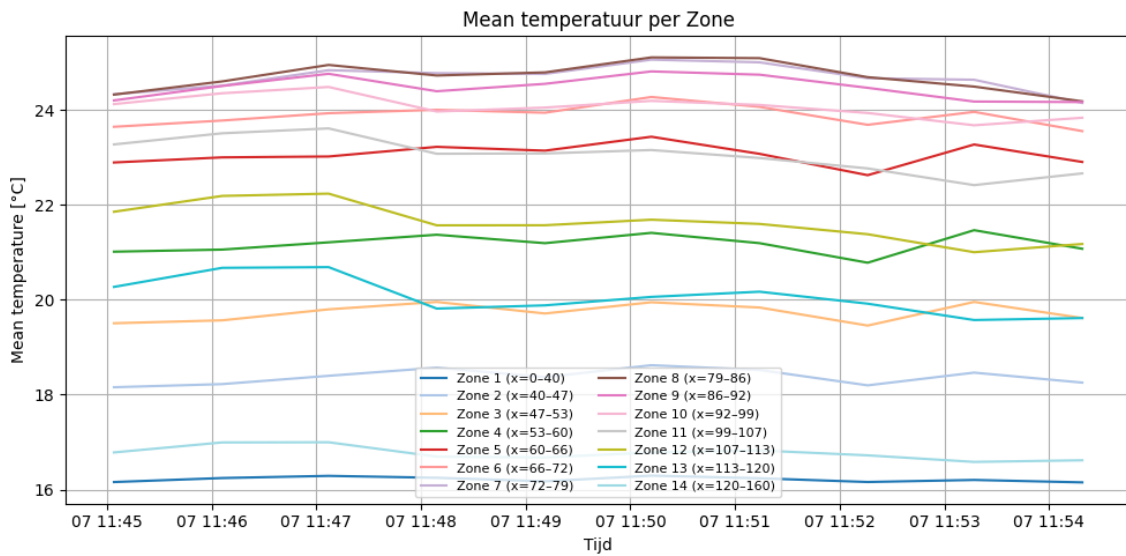


Figure 7.16: Mean temperature for each zone during 10 minutes

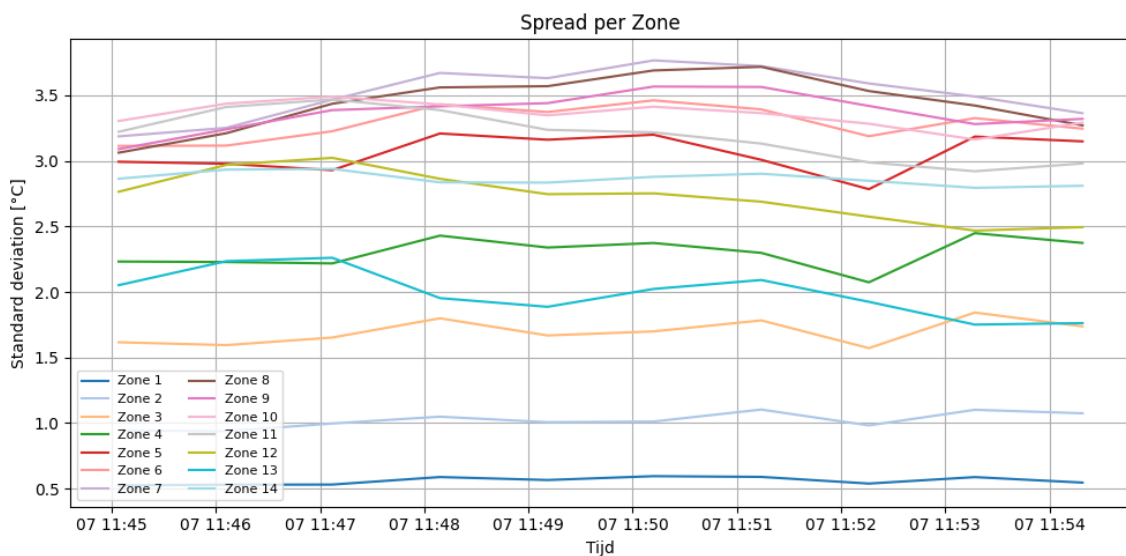


Figure 7.17: Standard deviation for each zone during 10 minutes

7.4 Practical implementation

This section will provide additional information on how the analysis is performed in practice.

To capture the images, an mlx90640 thermal camera is used. This sends its data to a Raspberry Pi via I2C communication. The camera is activated by a Python script on the Pi. This script initializes the camera using the

7 Infra-red imaging

libraries from Adafruit (Adafruit Industries, 2005). The camera then starts to capture images every minute. The time between capturing the images is adjustable. The camera originally captures images at a resolution of 24x32. To better visualize the image, the data is interpolated to a resolution of 120x160. After interpolation, the array with all the thermal data is saved. From this array the thermal image is made. Both arrays and images are stored on the Pi.

The analysis is done with another script. This script contains a number of functions that interpret the data from the array and make graphs and visualizations. All the graphs in section 7.3 are made with this Python script. The code needs two inputs, a start time and an end time. Then it makes graphs for all the infra-red images between these two dates. The first thing this script does is run through all the arrays with thermal data in a for loop. There, a function is used to calculate the weighted thermal centroids to determine the position of the bees. How this is done mathematically is shown in section 7.3. The thermal centroids are stored in an array. In the for loop, there are four other arrays that are filled with data. An array gets filled with the standard deviation of the thermal centroids which is also shown in section 7.3. Another array is filled with the time stamps of the thermal images. Two other arrays are filled with data about the number of clusters and the size of these clusters. The number of clusters is calculated by looking at every pixel and assigning a label to each pixel that has a higher value than a given value. Then the function counts all the pixels that are connected and form one cluster. This generates multiple clusters of a certain number of pixels. The functions that analyze the entire thermal image then use these arrays to plot the graphs.

For the graphs that describe the data in the different zones, a set of other arrays is used. These are filled inside a for loop inside the other for loop. This for loop runs through an array that defines the different zones. Like that, the thermal centroids are calculated for each zone every minute. The data collected in these for loops is then used by the functions to plot the graphs. These graphs are shown in section 7.3.

7.5 Conclusion

A thermal camera can be used to keep track of the movement and temperature in a beehive. This is useful information for beekeepers. It gives them an idea of the size and well being of their bee colony when they cannot open the beehive. The thermal camera can be used to show the mean location of the bees by calculating the hottest zone inside the beehive. The location can also be displayed by placing the thermal images after one another so that they display the changes inside the beehive. The standard deviation of the temperature can also be calculated. This shows how much the bees spread out through the beehive. It shows how far from the calculated thermal centroid they are positioned. This gives an idea of how much space is actually occupied in the beehive. Another feature that can be calculated is the temperature outside the beehive. This is useful to use as a comparison with the other graphs. This helps to link the behavior of the bees with the circumstances outside of the beehive. Another part that can be analyzed in the thermal data is the amount of heat clusters there are in

7 Infra-red imaging

the image. The amount and size of these clusters give information about the distribution of the bees inside the beehive.

The thermal images can be further analyzed. This is done by dividing the images into zones, zones that contain the space between two honeycombs. The same analysis as in the complete picture can be performed for the different zones. This gives more information about which honeycombs contain the most activity. The temperature and the standard deviation of the temperature in these zones can be calculated. This shows which zones contain more bees. The standard deviation of the temperature shows how much the temperature changes in each zone. It also gives an idea of how capable the bees are of keeping the temperature inside the hive constant. In these zones, it is also possible to look at the movement of the thermal centroids for each of these zones. This is another source of information about the movement of bees between honeycombs. The experiments can be further expanded to test on beehives made of different materials. This is to see if the kind of isolation has an effect on the movement of the bees.

In conclusion, infra-red imaging is a very good tool to get a lot of information from a closed beehive. It does not give much exact information, but it gives a good image of the status of the beehive. The images can be explored deeper to look at smaller parts of the beehive. This makes infra-red imaging a possible tool for beekeepers.

8

Prototype bee monitoring system

A prototype for a beehive monitoring system is made. This is still a very simple construct. The system is built to be placed underneath a beehive. The system is made up of two shelves. The bottom one holds all the electronics. This contains the Raspberry Pi, the transformer and the amplifier card for the speakers. The upper shelf contains the sensors. For the moment, this is only the infrared camera and the microphone. Later, this can be extended to accommodate more sensors. The upper level has to be quite low for the thermal camera to capture the entire beehive and the edges around it, but it needs to be close to the bottom of the beehive for the microphone. So, the shelf is placed in the correct position for the infrared camera. And for the microphone, there is an arm that can be moved underneath the beehive that can hold the microphone. It is also possible to move the shelf to a higher level. This system is a prototype that works for the sensors in the tests in this thesis. It is possible to use it with other sensors as well. Figure 8.1 shows the prototype. The beehive should be placed on top of this construction.

8 Prototype bee monitoring system



Figure 8.1: Setup for all the sensors underneath the beehive

9

Website

To visualize, and allow analysis of the data collected from the different tests performed in this thesis, a website was created. Using different pages for each test, the researchers of Honeybee Valley can easily access all data at any time. This website is hosted on the Raspberry Pi, and can be accessed through its IP address.

All data, code and results for this chapter can be found in Appendix C¹.

9.1 Website overview

The website consists of the following eight pages:

- Homepage
- Sensordata
- Pollen monitoring
- Arenatest
- Knocker test
- Infrared imaging
- Activity
- Logs

These each fulfill their own purpose, and provide a way to access and visualize the data gathered in each respective part of this thesis. The goal was to provide an easy interface, that allows all data (past and current) to be retrieved quickly at command. The sensordata page, pollen monitoring page and part of the homepage include functionalities that were not fully implemented due to time constraints and changes in objective focus. However,

¹GitHub repository for arenatest, pollen monitoring and website data

9 Website

these were retained, and to demonstrate the full capabilities of the website, an implementation using dummy data generated by a Node-RED flow was developed.

Each page has a similar layout: a top banner with the title “Beehive monitoring system”, the title of this thesis, University of Ghent’s logo and Honeybee Valley’s logo. Just beneath the top banner are the navigation tabs, that can be used to access each page. At the bottom of the page, a small section with info about this thesis, and contact information of all contributors was added.

The different pages will now be discussed, as to give an overview of the website’s features.

9.1.1 Homepage

The homepage has two purposes: show the current sensordata, and give information about the project and its goals.

The live sensordata is shown using six different cards, this amount can easily be scaled to any number of sensors used. All of these cards feature an icon to be able to quickly tell which sensor value is displayed. Depending on the current value, the border of the cards will change, as to give an indication of any alarming values that need to be looked at. The colors used are:

- Green: this value falls into the expected normal range.
- Orange: this value is still fine, but might need looking into.
- Red: this value is outside of the expected normal range, and needs urgent attention.

However, since no such sensors were implemented, this part is only kept as a possible future expansion. To show its capabilities, a Node-RED flow was used to send dummy data to the website. By showing the live sensordata on the homepage, a quick overview can be given and allow the status of the beehive to be assessed in the glimpse of an eye.

Finally, a short paragraph is added giving information about this project. The problem statement is briefly framed, showing the need for these tools, and the goals of this thesis are stated.

This page is shown in Figure 9.1.

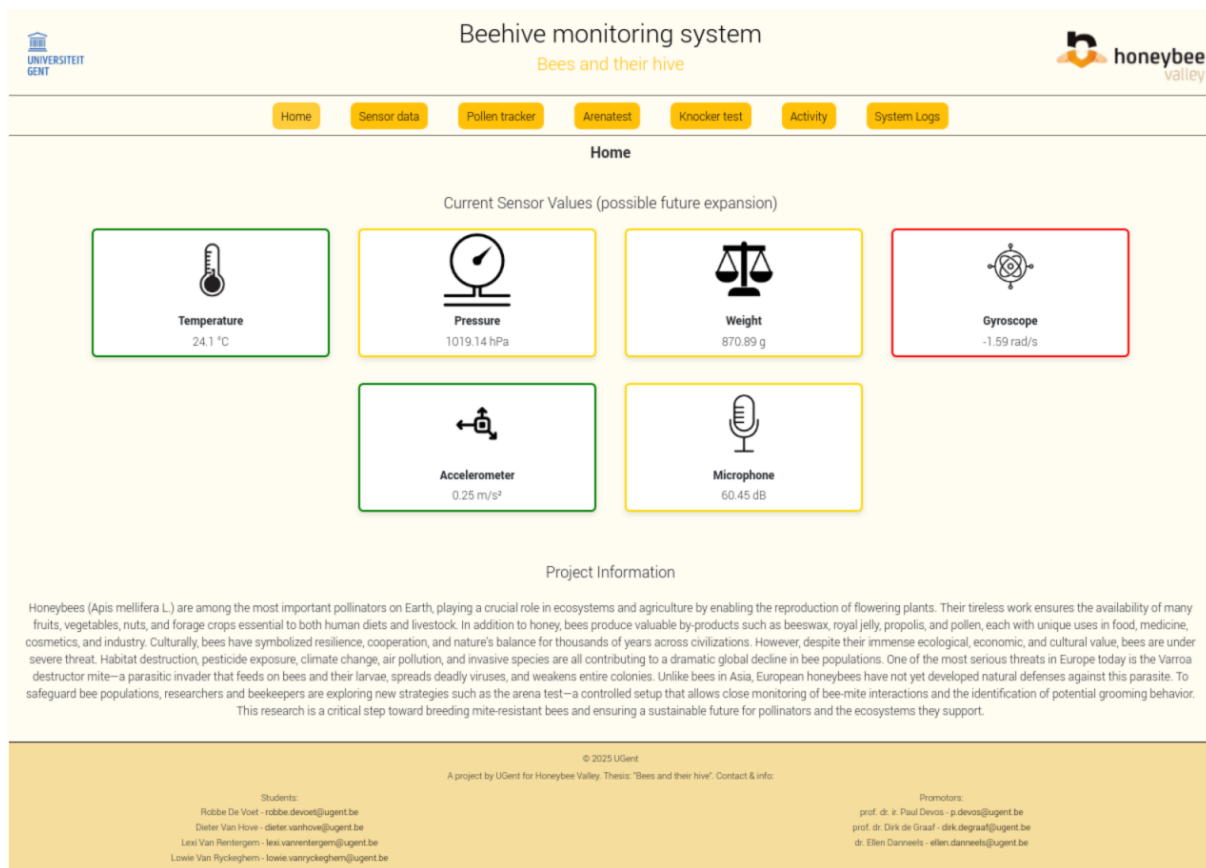


Figure 9.1: Website homepage

9.1.2 Sensordata page

The sensordata page gives an overview of all sensordata, past and present. On the left hand side, a table of the current sensor values is shown. The same icons as on the homepage are used. On the right hand side, the data history is accessible.

The main part of this page is the sensor history. This includes graphs that present the data for each sensor, for the selected date. A date can be selected through the date picker placed next to the title of this section. If the date of the current day is selected, the data of that day (so far) will be displayed. Any new data that is acquired while having this page open, will be added in real-time. This page can be used to see trends in the sensordata, and look back on previously recorded data. New, or different, sensors can easily be added to this page, to be able to showcase and analyze all the necessary sensordata.

As mentioned before, since no such sensors were implemented, this page is only kept as a possible future expansion. The Node-RED flow can also be used here to show the capabilities of this page.

This page is visible in Figure 9.2.

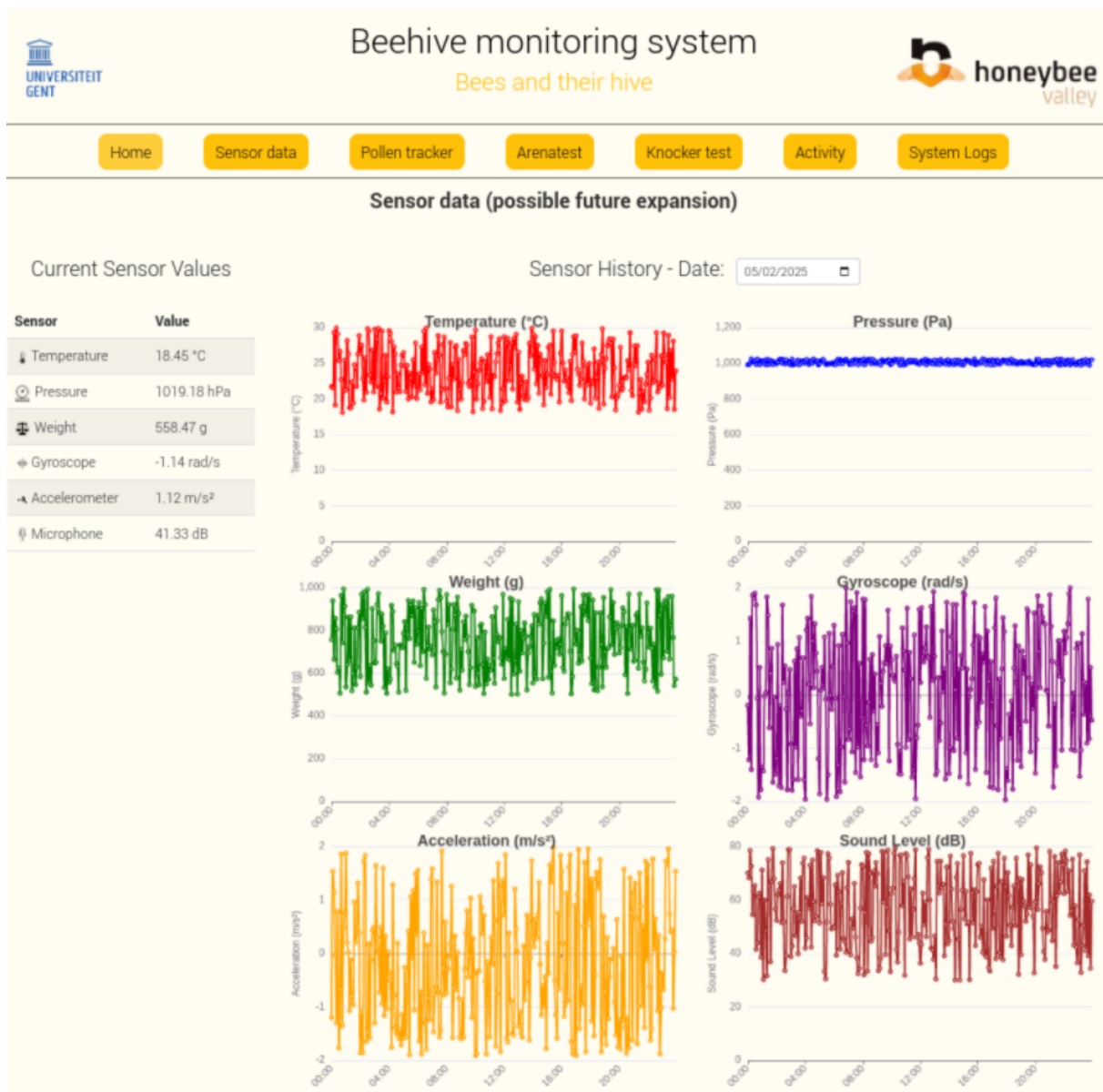


Figure 9.2: Website sensordata page

9.1.3 Pollen monitoring page

The pollen monitoring page shares a layout similar to the sensordata page. The implementation and functionalities are very similar, but instead of sensordata now the amount of pollen counted at the beehive entrance is shown. The amount of pollen, displayed in mm³, is shown in a cumulative way. The pollen color types can be expanded upon or changed, depending on the pollen colors that appear in the region in which the monitoring setup is deployed.

9 Website

Since pollen monitoring could not be fully implemented (see 5. Pollen monitoring) this page is only kept as a possible future expansion. A Node-RED flow was also used here to show the capabilities of this page.

This page is displayed in Figure 9.3.

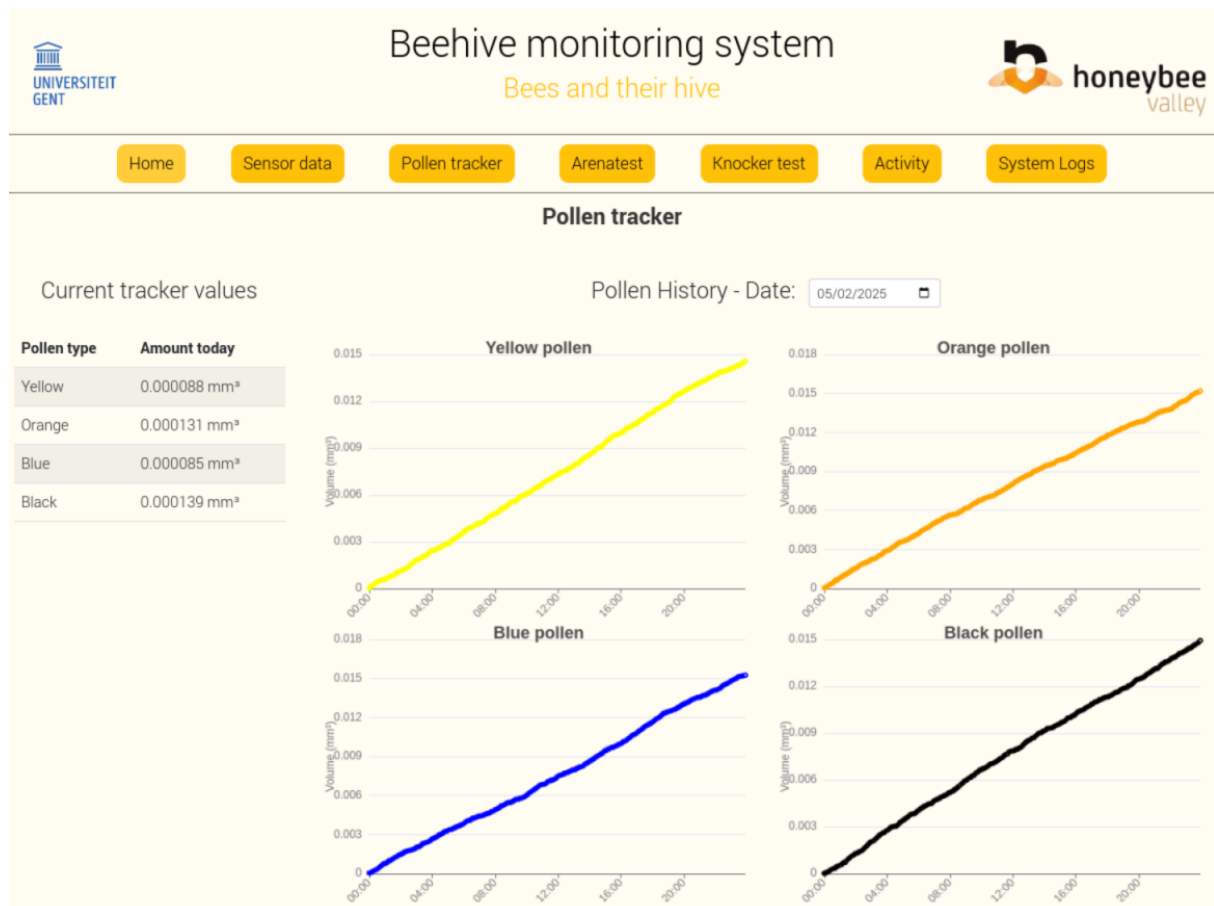


Figure 9.3: Website pollen monitoring page

9.1.4 Arenatest page

The arenatest page gives an overview of the data collected from the arenatest monitoring algorithm, and allows for quick evaluation and review of the tests performed.

All performed tests, from which data was found, can be selected through a drop down list at the top of the page. Upon selection of a certain test, its respective data is retrieved, and displayed on the page. The heatmap of bee positions is displayed on the left hand side, the video created from the frames with detection is displayed on the right hand side. Finally, the graphs showing the trends of the extracted metrics are shown underneath the heatmap and footage of the test. New arenatests can easily be added to the database, and are immediately available on the website once added. This page is shown in Figure 9.4.

9 Website



Figure 9.4: Website arenatest page

9.1.5 Knocker test page

The page about the knocker test shows some of the results from the test with a sound stimulus, section 6. The page will show the plot of the waveform from the recording. The page has a selection bar where the dates with available images appear. When a date is chosen, all the images made from recordings at that date will appear. The images that can be shown are: the full plot of the waveform, a zoomed image of a specific part, the rms value of the signal, the spectral centroid, the frequency distribution of the signal, and a spectrogram from the recording. These images are used for visual analysis in section 6.3.

The page also has the option of making a distinction between different beehives. The correct beehive has to be selected before the images appear. In this way only data from recordings from a specific beehive appears for the correct date. The page also has the option of adding and removing beehives. It is also possible to change the name of a beehive. These beehives are folders on the Raspberry Pi that contain the images. The images all have the correct timestamp in their filenames, so the website knows which images to show for a selected date. Figure 9.5 shows a print screen of the knocker page.

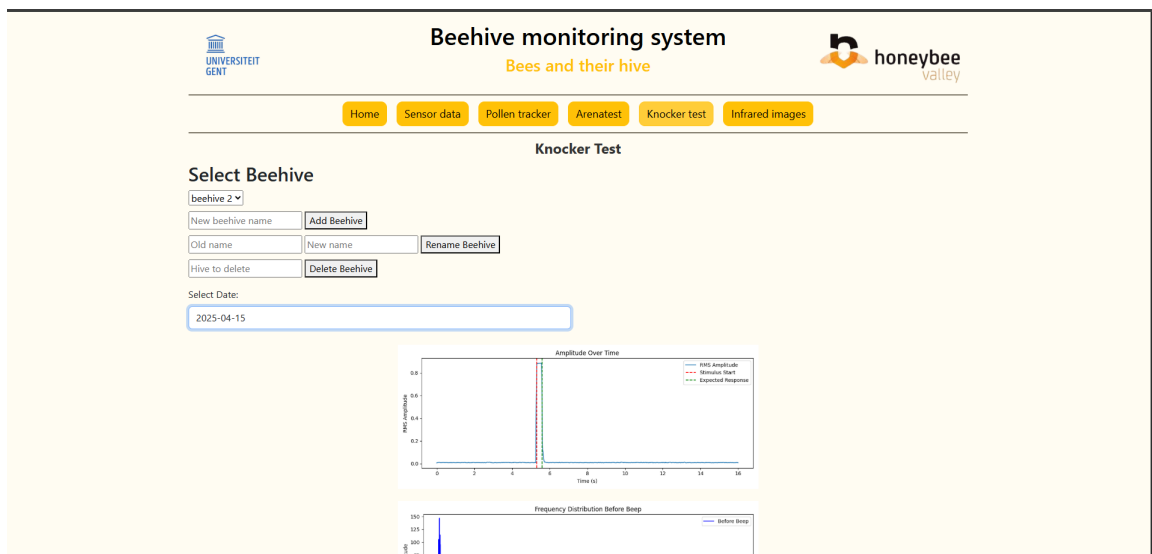


Figure 9.5: Print screen from the knocker page

9.1.6 Infrared imaging page

The page about infrared imaging is used to display data captured by the infrared camera, section 7. This page is designed to show the thermal images captured by the infrared camera in a certain time window. The page has two calendars for selecting a start date and time and an end date and time. Then there is a button that, when pressed, displays all the thermal images taken between these two selected moments.

The infrared image page also has the possibility of analyzing the thermal images. The functions that execute

9 Website

this analysis are explained in section 7.3 and section 7.4. For this to work, a Python code that contains all the functions to make the graphs and images is saved on the Raspberry Pi, this file is called analyzer.py. When the website gives the sign to start the analysis, the server backend activates this Python file to start running on the data between the two chosen dates. Analyzer.py does not run directly, it is run through a test wrapper named infrared_analysis.py. This is done to make it easier to test the different functions. It is also the first step to make an extending to the infrared page that makes it possible for the user to select which exact function it wants to run and return graphs. The frontend server, the JavaScript file, then displays the images from the analysis on the website. Figure 9.6 shows a print screen of the infrared page displaying several thermal images.

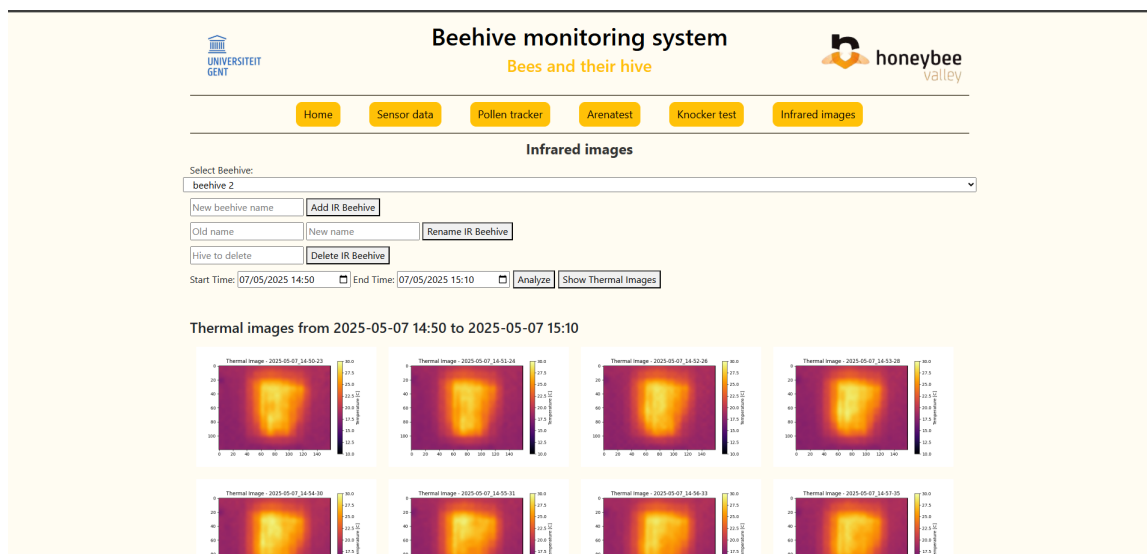


Figure 9.6: Print screen from the infrared page

9.1.7 Activity page

This page is a contribution by Lexi Van Rentergem and Robbe De Voet, and is a part of their thesis: "Smart Hive Monitoring: a Multichannel Microphone System with Machine Learning for Bee Activity Tracking". This page is visible in Figure 9.7.

9 Website

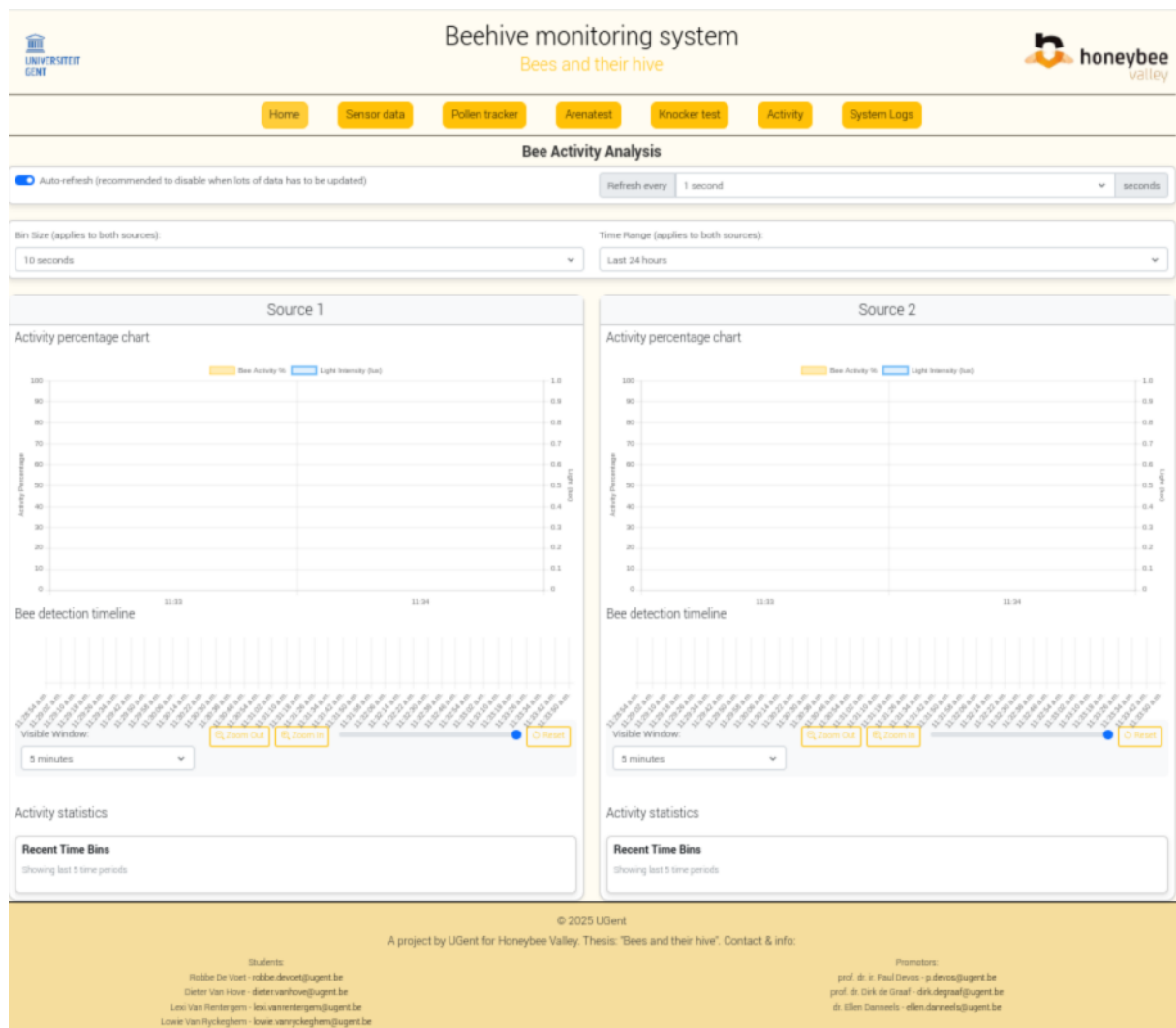


Figure 9.7: Website activity page

9.1.8 Logs page

This page is a contribution by Lexi Van Rentergem and Robbe De Voet, and is a part of their thesis: "Smart Hive Monitoring: a Multichannel Microphone System with Machine Learning for Bee Activity Tracking". This page is displayed in Figure 9.8.

9 Website

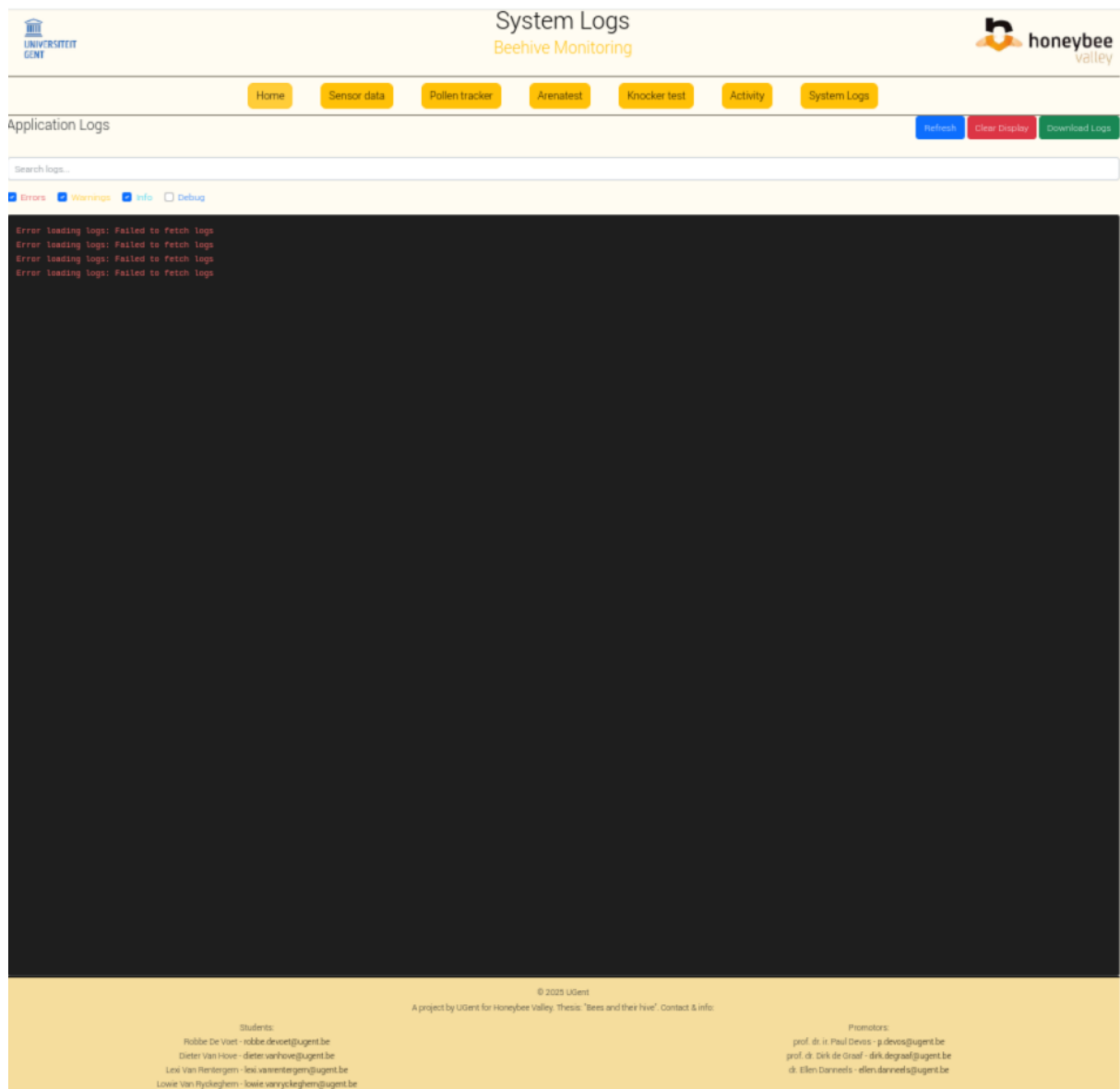


Figure 9.8: Website logs page

9.2 Technical implementation

9.2.1 Frontend

Each page is written in Hypertext Markup Language (HTML) in the Visual Studio environment. The Bootstrap library was used to help build the page layouts with its twelve-column grid, and enhance the look of the pages through the provided templates for e.g. tables and navigation bars. The pages are styled using Cascading Style Sheets (CSS). A global CSS file is used for every page to style shared elements like the title, navigation bar, logo

9 Website

and footer. For each of the pages where this was necessary, a page specific CSS file was added. For the creation of charts, the Apache ECharts framework was used.

To make the website dynamic, JavaScript is used. This is done page specific, according to the needs of the page. The Fetch API is used to get the necessary data from the server, which is the backend of the website and will be explained later.

Arenatest JavaScript overview

To be able to change the page, DOM, or Document Object Model, is used. The DOM represents the document as nodes and objects, and allows programming languages, like, JavaScript to interact with the page. When the HTML document is finished constructing, the DOM fires an event that allows the JavaScript file to be executed.

First, the drop down list is selected, and filled with a list of choices through a helper function. That function issues an HTTP GET request to the server endpoint called “get-tests”. When the server replies with JSON text, the browser automatically converts that text into a JavaScript array of objects. Each object in the array represents one performed arenatest of which data is saved on the Raspberry Pi. The function then iterates through this array. For each test object, it creates an 'option' element, sets the option's underlying value to the raw folder name and sets the label as “Arenatest DD/MM/YYYY”. Each option is then added to the drop down list. Once the list is built, the code selects the very first option by default and immediately loads that test's data.

The process of loading a given arenatest's data consists of three parallel requests. One fetches a PNG heatmap image, the second fetches a WebM video file and the third fetches a CSV file of the extracted metrics. Each of those fetches returns a promise. When the fetches return, the script takes the URL provided by the fetch response object, and shows it on the website page. When the CSV file arrives, it is first converted from a raw string into a table of JavaScript objects. The text is then parsed into the right format using a helper function. Then another function is used to create charts of the statistics.

The chart creation function first locates the right container <div> for each metric by its identifier (e.g. “chart1” or “chart2”). It then extracts the numeric values for that metric from the array of objects and, gives those to ECharts to render a line graph. Finally, the drop down list itself has an event listener for when the user selects a different option. If a different test is selected, the same three-part data-loading sequence repeats: fetch the heatmap, fetch the video, fetch and parse the CSV, then update the DOM and redraw all charts.

Sensordata and pollen counting JavaScript overview

Because the pollen counting and sensordata JavaScript files work in an identical way, the code overview is only discussed once. This will be done for the sensordata page.

Once the HTML document is loaded and the DOM is created, the DOMContentLoaded event is activated. There are three code blocks attached to this event that will react by executing. The first code block gets the date for today,

9 Website

concatenates it into a string in the ISO-style format “YYYY-MM-DD” and assigns this value to the `dateSelector`. This ensures that the date-picker on the page is defaulted to be today's date, and thus the most relevant data is shown by default.

The second code block is responsible for drawing six line charts, one for each of the sensor readings. For each chart, a helper function is called with the container ID, a label and a color. Inside that helper, the ECharts library is initialized on the target chart container. An ECharts options object is created with the chart title, axis configurations, grid margins, ... Then Echart's `setOption` method is called, which creates a blank chart that can now be filled with data.

The third code block is used to load the `sensordata`, depending on the selected date. That date is extracted as a string, which is then parsed back into the `Date` object. A comparison is made: if the chosen day and month match today's date day and month, the code shows today's live charts by calling the function `createChartsToday`. Otherwise, it issues a fetch request to a server endpoint that returns a CSV file of the `sensordata` of the selected date. The raw CSV text is retrieved, parsed and finally charts are created with this data.

To retrieve the live `sensordata`, a WebSocket connection is used. A function is used to handle and process this data. First, the WebSocket connection is opened. Whenever a new message arrives, it is parsed from JSON, into a JavaScript object containing the value readings for each of the sensors. These values are used to update the table that contains the latest sensor values. The code again compares the date that the user selected in the `dateSelector`, with today's date. If they match, the CSV for today's date is fetched and all charts are rendered again. This ensures that whenever new data comes in, the charts are automatically updated if the today's data is being looked at (since new data is also written to a CSV file in real-time).

9.2.2 Backend

The backend consists of a server, that is used to serve the data stored on the Raspberry Pi to the website, using HTTP GET requests. The server is implemented as a Node.js application, using the Express framework. It listens on port 4000, bound to a certain host IP, making it accessible via the address: `http://IP-ADRESS:4000`, where 'IP-ADRESS' is the IP of the network to which the Raspberry Pi is connected. A cross-origin resource sharing (CORS) policy is applied, so that any origin may request resources, this allows any device to use the website.

Incoming sensor- and pollen data is handled via two POST endpoints. Each endpoint builds a file path for the CSV file based on today's date, and initializes a CSV file with this file path, if this does not yet exist. The request body is deconstructed into the individual measurements (e.g. temperature, pressure, weight, gyroscope, accelerometer, microphone for `sensordata`; yellow, orange, blue, black for pollen), and a timestamp is generated in “yyyy-MM-dd HH:mm:ss” format. These are then added to an array that is used to create a new row in the CSV file. Two dynamic routes under `“/sensor/:filename”` and `“/pollen/:filename”` catches requests for this data, and serve the CSV files from the Raspberry Pi storage to the website.

9 Website

For retrieval of the arenatest data, three endpoints are used. A GET on “/get-tests” reads the arenatest data directory and returns a list containing the names of the subdirectories (the individual test folders). A GET on “/get-files” has a folder query parameter that will be the name of one of these subdirectories. It verifies that the folder exists, then scans its contents to locate one CSV, one PNG and one WebM file. These are then bundled together into a JSON object. Finally, a dynamic route catches requests for any filename in an arenatest folder, and serves those data from the Raspberry Pi storage to the website.

9.2.3 Data flow

The data folder structure on the Raspberry Pi consists of folders on the desktop, which allows quick data access. There are four main folders:

- Sensordata
- Pollen counting data
- Arenatest data
- Website data

The sensordata folder consist of all the CSV data files. These are named by the format “sensor_data_DDMMYYYY.csv”, making them easy to browse through if you are looking for data from a specific day. To demonstrate the capabilities of the sensordata history, files with dummy data were generated using Python code to fill this directory. This also means that data can easily be added, removed or changed if necessary. The pollen counting data folder is organized in a similar way, using the format “pollen_data_DDMMYYYY.csv” for the CSV files, and also containing dummy files to demonstrate the abilities of the pollen monitoring page.

The arenatest data folder contains subfolders for each conducted arenatest, that are named by the date they were performed. These contain their respective data: the CSV file with the extracted metrics data, the heatmap image and the WebM video with the test footage. New test data can be easily added by simply adding a new folder.

Finally, the website folder contains the HTML files for each page and three subfolders. The subfolders are the ‘css’ folder, containing the stylesheets, the ‘js’ folder containing the JavaScript code for each page and the ‘img’ folder containing the images used on the website, such as the logo’s, icons and the favicon.

These folders can be accessed by the server to serve the data to the website, if requested.

9.2.4 Node-RED testing

Node-RED was used to create an implementation with dummy data, that showcases the capabilities of the sensordata and pollen counting data processing. It is a flow-based development tool built on Node.js, that provides

9 Website

a browser-based editor for wiring together hardware, APIs and services. In this case, it is used to run JavaScript code which generates the dummy data, puts it in the correct format, and sends it to the website. For the display of the current sensor values, Websockets are used. A WebSocket is a persistent, full-duplex communication channel established over a single TCP connection between a client (in this case, the website) and a server (in this case the Node-RED server). This allows both sides to push messages to each other in real-time, without repeated HTTP requests. Whenever the Node-RED server sends a message with new sensordata, an event is triggered on the WebSocket. Using a function attached to that event, the data this message contains can be read and processed, and shown on the website.

To generate the CSV files containing real-time sensor and pollen data, HTTP POST requests are sent to the server endpoints. These requests transmit the same data that are sent via the WebSocket, and the server automatically appends the data to the appropriate CSV file.

An overview of the flow implemented is given in Figure 9.9.

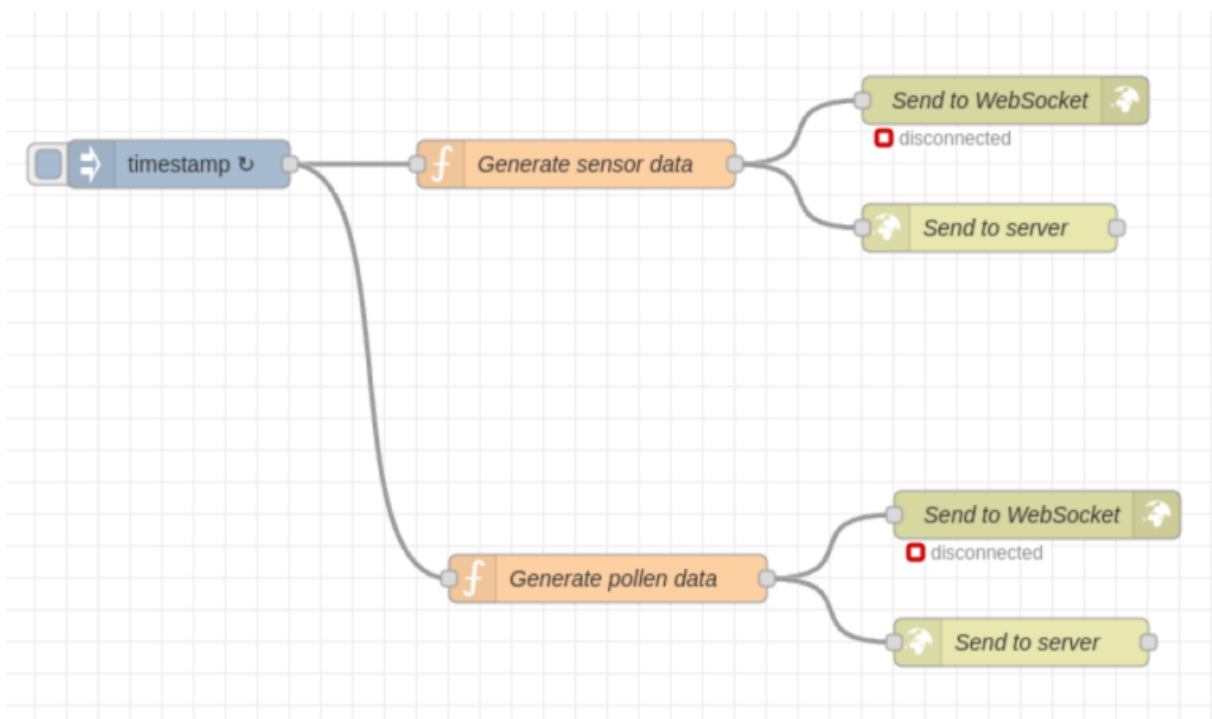


Figure 9.9: Node-RED pollen- and sensor dummy data generating flow

9.3 Deployment and accessibility

The entire web application is hosted on the Raspberry Pi 5. During system initialization on boot up, the Node.js runtime is started automatically through a systemd service. Node.js immediately loads the Express application and binds its HTTP to the Pi network interface at the IP address, to which it is connected on port 4000. This

allows the website to be accessed at "http://IP_ADRESS:4000", where 'IP_ADRESS' is the IP address of the network to which the Raspberry Pi is connected, if you are also connected to that network. Because CORS is used, all devices can access the website. If this were not implemented, the HTML and CSS would load in, but the server would deny requests to fetch data from the Raspberry Pi, and therefore the dynamic processes of this website would not be functional.

9.4 Limitations, challenges and future prospectives

The biggest limitation of the website is that it can only be accessed in the vicinity of the Raspberry Pi. Before this beehive monitoring system can be fully functional, it would be a necessary improvement to allow data to be accessed from anywhere, anytime. For this, the website would need to be accessible from anywhere, and on any network. An idea for a partial solution was the use of the University's "Eduroam" network, which would allow access to the website from all University buildings (including Honeybee Valley and the hall with beehives). However, this could not be done, since the network did not allow this. Certain permissions are needed to be able to host the website on the network, to which there was no access. There exist (paid) tools that allow website hosting on Raspberry Pi's, the implementation of which is a great future prospective.

Other improvements could also be made to the website's features, allowing better analysis and access to the data. More options for data visualization and allowing data to be downloaded to any device would be a first step. The security of the website would also need to be improved, if it was made available to anyone. The use of HTTPS instead of HTTP would be a useful first improvement for this, as well as adding user authentication.

The biggest future prospective is the implementation of an interface through which more than one beehive can be accessed and monitored. Using a drop down list, through which can be selected which hive data are shown, could be an option to allow this to be implemented in the current website layout. This would allow the researchers of Honeybee Valley (and possible beekeepers) to monitor the status of all their beehives in a quick, and accurate fashion. Building on this, an app could be developed to allow easier access to the system.

9.5 Conclusion

In this chapter a website hosted locally on a Raspberry Pi was developed, to visualize and manage data from the beehive monitoring experiments. The site provides researchers with quick access to test results, including areatest data, pollen counts and sensor readings. Although some features were only implemented using dummy data due to time constraints, the website has potential to be used for real-time, multi-hive monitoring.

It was made with HTML, CSS and JavaScript on the frontend, and Node.js with the Express framework on the backend. WebSockets were used in combination with Node-RED, to show the capabilities of the sensordata- and

9 Website

pollen monitoring pages, through dummy data generated and send in real-time. ECharts was used to create interactive graphs to track sensor values, pollen volume and arenatest metrics. Although currently limited to devices connected to the same local network as the Raspberry Pi, there are possibilities to scale this. Possible future enhancements could include enabling remote access through secure hosting, implementing HTTPS for data encryption and adding user authentication for security. A drop down-based hive selection interface would allow multi-hive monitoring from a single dashboard, and a mobile app could further improve accessibility.

10

Ethical and social reflection

The development of a multi-sensor and computer vision system for monitoring honeybee colony health, as presented in this thesis, intersects significantly with pressing ethical and societal concerns. At its core, this project addresses the urgent need to safeguard pollinators, which are essential for food security and ecosystem health, aligning directly with the United Nations Sustainable Development Goals (SDGs) — particularly SDG 2 (Zero Hunger), SDG 12 (Responsible Consumption and Production), SDG 13 (Climate Action) and SDG 15 (Life on Land).

Honeybees are vital to biodiversity and agricultural productivity. However, their populations are declining globally due to a combination of environmental stressors, including habitat loss, pesticide exposure, climate change and parasitic infestations such as *Varroa destructor*. This thesis contributes ethically to this challenge by proposing a non-invasive, data-driven monitoring solution that minimizes the disturbance to bee colonies while maximizing observational accuracy. Through the use of sound, thermal imaging and vision-based monitoring, beekeepers and researchers can detect early signs of colony stress and take timely action. Such technological interventions reduce the need for chemical treatments and frequent hive inspections, supporting more sustainable and ethical beekeeping practices.

From a societal standpoint, the system developed offers potential benefits beyond research. By democratizing access to advanced diagnostic tools through low-cost hardware and open-source software, this project supports inclusive innovation, resonating with SDG 9 (Industry, Innovation and Infrastructure). In particular, small-scale beekeepers may gain access to early warning systems that traditionally have only been available to well-funded institutions.

Ethically, it is also important to consider data privacy and responsible use. While this system does not collect personal data, future extensions involving network connectivity or cloud storage should take into account appropriate safeguards to ensure secure data handling. Furthermore, as with any technological intervention in biological systems, it is essential to monitor for unintended consequences, such as interference with natural behavior due to sensor placement or light exposure.

In conclusion, this work reflects a commitment to developing ethically responsible, environmentally sustainable

10 Ethical and social reflection

and socially impactful technology. By supporting ecological resilience and promoting responsible innovation, it makes a meaningful contribution toward achieving a number of global sustainability targets.

References

- Abdelrahman, E. (2025, March). *Beekeeping with computer vision: Smarter hive monitoring*. <https://www.ultralytics.com/blog/beekeeping-with-computer-vision-smarter-hive-monitoring>
- Adafruit Industries. (2005). *Adafruit mlx90640 ir thermal camera* [Accessed: 2025-02-20]. <https://learn.adafruit.com/adafruit-mlx90640-ir-thermal-camera/python-circuitpython>
- Amazon. (n.d.). *Hahhhha bee pollen traps collector easy install pollen traps pollen collector tray plastic rectangle outdoor beekeeping tool front mount pollen trap*. <https://www.amazon.co.uk/Hahhhha-Collector-Install-Rectangle-Beekeeping/dp/B0B2RGG19V>
- Aumeier, P. (2001). Bioassay for grooming effectiveness towards varroa destructor mites in africanized and carniolan honey bees. *Apidologie*, 32(1), 81–90. <https://www.apidologie.org/articles/apido/abs/2001/01/m1107/m1107.html>
- Bahreini, R., et al. (2021). New bioassay cage methodology for in vitro studies on varroa destructor and apis mellifera. *PLOS ONE*, 16(4), e0250594. <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0250594>
- Bayer AG. (2019, January). *A deadly honey bee parasite: The varroa mite*. https://www.bayer.com/sites/default/files/The_Varroa_Mite_2020.pdf
- Bencsik, M., McVeigh, A., Bouuaert, D. C., Capela, N., Penny, F., Newton, M. I., Sousa, J. P., & de Graaf, D. C. (2024). Quantitative assessments of honeybee colony's response to an artificial vibrational pulse resulting in non-invasive measurements of colony's overall mobility and restfulness. *Scientific Reports*, 1–11. <https://doi.org/10.1038/s41598-024-54107-8>
- Betterbee. (n.d.). *Using pollen traps to collect (and sell) bee pollen*. https://www.betterbee.com/instructions-and-resources/how-to-use-a-pollen-trap.asp?srsId=AfmB0op5pg3BtAmMQT2LU-eEL1MvxRIjiH5FoxL1Hg_09cUStVR6zBvI
- Brogioli, D., Crotti, M., Galli, A., Piccini, B., Tagliaferri, S., & Finazzi, S. (2011). Infrared imaging of honey bees for studying the hive thermal behavior [Accessed: 2025-05-13]. *Optics Express*, 19(1), 399–408. <https://doi.org/10.1364/OE.19.000399>
- Collison, C. (2018, May). *A closer look*. <https://beeculture.com/a-closer-look-19/>

10 References

- Company, F. B. (n.d.). *Bee pollen traps*. <https://www.foxhoundbeecompany.com/blogs/beekeeping-equipment/bee-pollen-traps?srsId=AfmB0orj1rxBJZ5xb5G9HCqIA6CTz18JkjOB1po9HdxNriW63CB0uwV>
- Connor, L. (2017). *A closer look: Sound generation and hearing* [Accessed: 2025-05-13]. Bee Culture. <https://beeculture.com/a-closer-look-sound-generation-and-hearing/>
- Currie, R. W., & Gatien, P. (2012). Timing acaricide treatments to prevent varroa destructor from causing economic damage to honey bee colonies. *The Canadian Entomologist*, 138, 238–252. <https://www.cambridge.org/core/journals/canadian-entomologist/article/abs/timing-acaricide-treatments-to-prevent-varroa-destructor-acari-varroidae-from-causing-economic-damage-to-honey-bee-colonies/887D7C98E3EDF54349D9529F72DC0406>
- Danneels, E. (n.d.). *Initiatie tot het imkeren*. <https://www.honeybeevalley.eu/doit/initiatie-tot-het-imkeren>
- Departement Omgeving. (2025). *Bijen zijn belangrijk want...* <https://www.vlaanderen.be/week-van-de-bij/waarom/bijen-zijn-belangrijk-want>
- DW. (2018, May). *Eu court upholds ban on pesticides threatening bees*. <https://www.dw.com/en/eu-court-upholds-ban-on-pesticides-threatening-bees/a-43821344>
- Fitzgibbon, A., Pilu, M., & Fisher, R. (1996). Direct least squares fitting of ellipses. *Proceedings - International Conference on Pattern Recognition*, 1, 253–257. <https://doi.org/10.1109/ICPR.1996.546029>
- Frings, H., & Little, F. (1957). Reactions of honey bees in the hive to simple sounds. *Science*, 125(3238), 122. <https://doi.org/10.1126/science.125.3238.122>
- Gladstone, S. (n.d.). *Module 15: Hive value added products*. <https://iyanolaapiculture.org/training/module15/>
- Goulson, D., Nicholls, E., Botias, C., & Rotheray, E. (2015). Bee declines driven by combined stress from parasites, pesticides, and lack of flowers. *Science*, 347(6229). https://doi.org/10.1126/SCIENCE.1255957/ASSET/25ACDBB7-03E0-4D87-BEC1-0935B850C29F/ASSETS/GRAPHIC/347_1255957_FA.JPEG
- Guelph, T. U. O. (n.d.). *Communication*. <https://hbrc.ca/honeybeecommunication/>
- Henry, M., Beguin, M., Requier, F., Rollin, O., Odoux, J., Aupinel, P., Aptel, J., Tchamitchian, S., & Decourtye, A. (2012). A common pesticide decreases foraging success and survival in honey bees. *Science*, 336(6079), 348–350. <https://doi.org/10.1126/SCIENCE.1215039>,

10 References

- Honing van de Imker. (n.d.). *Varroa destructor: Plage voor bij en imker*. <https://honing-van-de-imker.be/varroa-destructor-plaag-voor-bij-en-imker.asp>
- Hoeningkorf. (2020, December). *Be the hooneybee*. <https://www.opencaching.nl/viewcache.php?wp=0B1BE4>
- Kamakura, M. (2011). Royalactin induces queen differentiation in honeybees. *Nature*, 473(7348), 478–483. <https://doi.org/10.1038/NATURE10093>;SUBJMETA=136,142,1466,601,631,80,86;KWRD=CELL+SIGNALLING,DIFFERENTIATION,ENTOMOLOGY
- Kirchner, W. H. (1989). Hearing in honeybees: The mechanical response of the bee's antenna to near-field sound [Accessed: 2025-05-13]. *Science*, 244(4905), 686–688. <https://doi.org/10.1126/science.244.4905.686>
- Kralj, J., & Fuchs, S. (2006). Parasitic varroa destructor mites influence flight duration and homing ability of infested apis mellifera foragers. *Apidologie*, 37(5), 577–587. <https://www.apidologie.org/articles/apido/abs/2006/05/m6048/m6048.html>
- Le Conte, Y., & Navajas, M. (2008). Climate change: Impact on bee populations and health. *Apidologie*, 39(3), 183–194. <https://pubmed.ncbi.nlm.nih.gov/18819674/>
- LeDoux, M. N., et al. (2000). Development of a bioassay to test the orientation behavior of the honey bee ectoparasite, varroa jacobsoni. *Journal of Apicultural Research*, 39(1-2), 47–54. <https://www.tandfonline.com/doi/abs/10.1080/00218839.2000.11101020>
- Leponiemi, M., Freitak, D., Moreno-Torres, M., Pferschy-Wenzig, E., Becker-Scarpitta, A., Tiusanen, M., Vesterinen, E., & Wirta, H. (2023). Honeybees' foraging choices for nectar and pollen revealed by dna metabarcoding. *Scientific Reports*, 13(1), 1–15. <https://doi.org/10.1038/S41598-023-42102-4>;SUBJMETA=158,2452,2456,631;KWRD=AGROECOLOGY,ECOLOGY,MOLECULAR+ECOLOGY
- Martin, S. J. (1998). A population model of the ectoparasitic mite varroa jacobsoni in honey bee (apis mellifera) colonies. *Ecological Modeling*, 109, 267–281. [https://doi.org/10.1016/S0304-3800\(98\)00059-3](https://doi.org/10.1016/S0304-3800(98)00059-3)
- Martin, S. J., & Kemp, D. (1997). Average number of reproductive cycles performed by the parasitic mite varroa jacobsoni in apis mellifera colonies. *Journal of Apicultural Research*, 36, 113–123. <https://www.tandfonline.com/doi/abs/10.1080/00218839.1997.11100937>

10 References

- McKinnon, A., Collins, L., Wood, J., Murphy, N., Franks, A., & Steinbauer, M. (2023). Precision monitoring of honey bee (hymenoptera: Apidae) activity and pollen diversity during pollination to evaluate colony health. *Insects*, *14*(1), 95. <https://doi.org/10.3390/INSECTS14010095>
- Micinski, S., Wilson, E. R., & Levin, M. D. (1972). Honey bee behavior in a controlled sound field [Accessed: 2025-05-13]. *Annals of the Entomological Society of America*, *55*(1), 82–86. <https://doi.org/10.1093/aesa/55.1.82>
- Morawetz, L., et al. (2019). Health status of honey bee colonies (*apis mellifera*) and disease-related risk factors for colony losses in austria. *PLOS ONE*, *14*(7), e0219293. <https://doi.org/10.1371/journal.pone.0219293>
- Ngo, T., Rustia, D., & Lin, T. (2021). Automated monitoring and analyses of honey bee pollen foraging behavior using a deep learning-based imaging system. *Computers and Electronics in Agriculture*, *187*, 106239. <https://doi.org/10.1016/J.COMPAG.2021.106239>
- Oliver, R. (2006). *Ip3 fighting varroa: Strategy – understanding varroa population dynamics*. <https://scientificbeekeeping.com/ipm-3-strategy-understanding-varroa-population-dynamics/>
- Pasupuleti, V. R., Sammugam, L., Ramesh, N., & Gan, S. H. (2017). Honey, propolis, and royal jelly: A comprehensive review of their biological actions and health benefits. *Oxidative Medicine and Cellular Longevity*, *2017*, 1259510. <https://onlinelibrary.wiley.com/doi/10.1155/2017/1259510>
- Potera, C. (2008). Air pollution: Floral scents going off the air. *Environmental Health Perspectives*, *116*(8), A334. <https://doi.org/10.1289/ehp.116-a334>
- Prendergast, K., Garcia, J., Howard, S., Ren, Z.-X., McFarlane, S., & Dyer, A. (2021). Bee representations in human art and culture through the ages. *Art & Perception*, *10*(1), 1–62. <https://doi.org/10.1163/22134913-BJA10031>
- Ramsey, S. D., et al. (2019). Varroa destructor feeds primarily on honey bee fat body tissue and not hemolymph. *Proceedings of the National Academy of Sciences of the United States of America*, *116*(5), 1792–1801. <https://doi.org/10.1073/pnas.1818371116>
- Robinson, G. (1992). Regulation of division of labor in insect societies. *Annual Review of Entomology*, *37*(1), 637–665. <https://doi.org/10.1146/ANNUREV.EN.37.010192.003225/CITE/REFWORKS>
- Rodriguez, I., Chan, J., Rios, M. A., Branson, K., Agosto-Rivera, J., Giray, T., & Megret, R. (2022). Automated video monitoring of unmarked and marked honey bees at the hive entrance. *Frontiers in Computer Science*, *3*, 769338. <https://doi.org/10.3389/FCOMP.2021.769338/BIBTEX>

10 References

- Rusty. (n.d.). *Pollen traps require constant attention*. <https://www.honeybeesuite.com/pollen-traps-require-constant-attention/>
- Smith, L. (2022, October). *How to harvest bee pollen*. <https://backyardbeekeeping.iamcountryside.com/plants-pollination/how-to-harvest-bee-pollen/>
- Stefanec, M., Oberreiter, H., Becher, M. A., Haase, G., & Schmickl, T. (2021). Effects of sinusoidal vibrations on the motion response of honeybees. *Frontiers in Physics*, 9, 670555. <https://doi.org/10.3389/fphy.2021.670555>
- Texas Apiary Inspection Service. (n.d.). *Varroa mites*. <https://txbeeinspection.tamu.edu/varroa-mites/>
- ThermoMite. (n.d.). *Thermomite thermotherapy method for varroa control*. <https://www.thermomite.com/product>
- United Nations Environment Programme. (2022, May). *Why bees are essential to people and planet*. <https://www.unep.org/news-and-stories/story/why-bees-are-essential-people-and-planet>
- Veto-pharma. (2023, March). *Field bioassay to detect varroa mite resistance in the apiary*. <https://www.blog-veto-pharma.com/en/field-bioassay-to-detect-varroa-mite-resistance-in-the-apiary/>
- von Frisch, K. (1967). *The dance language and orientation of bees*. Harvard University Press.
- Wikipedia. (2025, March). *Hsl and hsv*. https://en.wikipedia.org/wiki/HSL_and_HSV
- WildBnB. (2022). *Belangrijkste bedreigingen*. <https://www.wildbnb.brussels/menaces-principales?lang=nl>
- Woodcock, B., Oliver, A., Newbold, L., Gweon, H. S., Read, D., Sayed, U., Savage, J., Bacon, J., Upcott, E., Howell, K., Turvey, K., Roy, D., Pereira, M. G., Sleep, D., Greenop, A., & Pywell, R. (2022). Citizen science monitoring reveals links between honeybee health, pesticide exposure and seasonal availability of floral resources. *Scientific Reports*, 12(1), 1–13. <https://doi.org/10.1038/s41598-022-18672-0>
- Xrite. (n.d.). *Colorchecker classic*. <https://www.xrite.com/categories/calibration-profiling/colorchecker-classic>