

Article

Edge-Based Detection of Varroosis in Beehives with IoT Devices with Embedded and TPU-Accelerated Machine Learning

Dariusz Mrozek ^{1,*} , Rafał Górny ¹, Anna Wachowicz ¹  and Bożena Małysiak-Mrozek ² 

¹ Department of Applied Informatics, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland; avgorny@gmail.com (R.G.); Anna.Wachowicz@polsl.pl (A.W.)

² Department of Distributed Systems and Informatic Devices, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland; bozena.malysiak@polsl.pl

* Correspondence: dariusz.mrozek@polsl.pl; Tel.: +48-32-237-1339

Abstract: One of the causes of mortality in bees is varroosis, a bee disease caused by the *Varroa destructor* mite. *Varroa destructor* mites may occur suddenly in beehives, spread across them, and impair bee colonies, which finally die. Edge IoT (Internet of Things) devices capable of processing video streams in real-time, such as the one we propose, may allow for the monitoring of beehives for the presence of *Varroa destructor*. Additionally, centralization of monitoring in the Cloud data center enables the prevention of the spread of this disease and reduces bee mortality through monitoring entire apiaries. Although there are various IoT or non-IoT systems for bee-related issues, such comprehensive and technically advanced solutions for beekeeping and *Varroa detection* barely exist or perform mite detection after sending the data to the data center. The latter, in turn, increases communication and storage needs, which we try to limit in our approach. In the paper, we show an innovative Edge-based IoT solution for *Varroa destructor* detection. The solution relies on Tensor Processing Unit (TPU) acceleration for machine learning-based models pre-trained in the hybrid Cloud environment for bee identification and *Varroa destructor* infection detection. Our experiments were performed in order to investigate the effectiveness and the time performance of both steps, and the study of the impact of the image resolution on the quality of detection and classification processes prove that we can effectively detect the presence of varroosis in beehives in real-time with the use of Edge artificial intelligence invoked for the analysis of video streams.

Keywords: Internet of Things (IoT); *Varroa destructor*; precision beekeeping; machine learning; cloud; image analysis; Edge computing



Citation: Mrozek, D.; Górny, R.; Wachowicz, A.; Małysiak-Mrozek, B. Edge-Based Detection of Varroosis in Beehives with IoT Devices with Embedded and TPU-Accelerated Machine Learning. *Appl. Sci.* **2021**, *11*, 11078. <https://doi.org/10.3390/app112211078>

Academic Editor: Eui-Nam Huh and Agostino Forestiero

Received: 20 September 2021

Accepted: 16 November 2021

Published: 22 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, we have been observing rapid growth of the advancement in technology, including Internet of Things (IoT) solutions. These solutions are developed all over the world for almost every area of our lives. Daily use devices, such as smartphones, smartwatches, TVs, household appliances, or cars are obvious examples. For medicine, IoT solutions are used for remote monitoring of a patient's health state and performing complicated medical surgeries [1]. In agriculture [2], these systems are helpful for controlling environmental factors or monitoring animals [3].

On the other hand, people are becoming aware of environmental and climate changes. It is widely known that bees are crucial for the Earth. They are responsible for pollinating trees and plants, including food crops for humanity. Insects pollinate nearly 90% of global food production [4], while two-thirds of them are estimated to be honey bees. Unfortunately, for almost twenty years, the bee population all over the world has been decreasing. This phenomena was identified as Colony Collapse Disorder (CCD) [5]. The reasons have not yet been fully classified; however, among possible factors, scientists often mention wrong

bee nutrition [5,6], weather conditions (including climate changes) [5,7], pesticides that are used in agriculture more and more often [4,8,9], and various organisms (mites, bacteria, viruses etc.) that attack whole apiaries [4,8,10–12]. One of the mites that endangers bees' lives is *Varroa destructor* (*V. destructor*) [7,12,13]. It reaches up to 1.2 mm in length and is red or brown colored [14]. This is why the human eye can easily notice it without any magnification. *Varroa destructor* attacks both the bees and their larva, can move between workers and eggs inside a beehive, and feeds on hemolymph [13,15]. Bees infected with this mite live for a shorter time, are affected with wing and body deformations, have navigation problems, and work less efficiently than their healthy counterparts [14]. What is more, *Varroa destructor* may transfer other viruses, i.e., Deformed Wing Virus (DMV), Kashmir Bee Virus (KBV), or Acute Bee Paralysis Virus (ABPV) [13,14,16]. Currently, the most popular way of detecting the existence of *Varroa destructor* is regular and time-consuming bee observation in beehives performed manually by beekeepers. Responding to such needs, IoT solutions finally started to be also implemented for this field. However, detecting varroosis with IoT requires the development of devices that can constantly monitor beehives and analyze the video data stream in real-time. This necessitates the use of hardware-accelerated IoT devices since sending the video stream to the monitoring data center in many situations is impossible (due to limited access to the Internet).

This paper presents a large-scale IoT Edge- and Cloud-based solution for detecting the *Varroa destructor* mite among bees that enter and leave the beehive. We propose a system built on an IoT monitoring device and single-board computer that can record video streams, process video frames (images), use a Tensor Processing Unit (TPU) acceleration to perform on-edge analysis with pre-trained Machine Learning models, and send the data to the Cloud for the purpose of storage, further geo-wide analysis, and notifying beekeepers. Our solution, presented in detail further in this paper, in contrast with others, is a complex system designed for beekeepers who struggle with *Varroa destructor* mite. The use of TPU and machine learning (ML) models enables the recognition of possible threats quickly and efficiently and helps save bees from illness and death. Moreover, by performing the video stream processing (including the image analysis) at the Edge, not in the data center, our approach reduces the need for frequent communication with the data center and the storage space required to keep large streams of video captured by the camera-enabled IoT device. Our solution, unlike the other presented in the next section, entirely relies on edge and cloud processing, applies ML with two dedicated Convolutional Neural Network (CNN) models for bee identification and *Varroa destructor* detection, and can notify beekeepers directly in case of detected danger. In this manner, it complements the solutions published so far.

In the following section, we show particular component elements of our approach. Section 2 presents current state-of-the-art research regarding various solutions for bee monitoring developed during the last decades. Section 3 discloses details about the experimental environment that we developed to monitor bees and detect varroosis, which we experimentally tested. In Section 4, we present the results of the performed experiments. Finally, we discuss the solution and obtained results and compare the results with the results reported in the related works (Section 5).

2. Related Works

Bee monitoring has been a field for scientific research for more than a hundred years. The first proposal of a system that enabled measuring temperature and weight of beehive was described in 1914 by B.N. Gates [17]. The data were gathered hour by hour for one year. Nearly twenty years later, W. Dunham [18] used an electrical thermocouple method to monitor the temperature inside the beehive. Along with the development of technology, more and more solutions were proposed. Finally, there came a time for IoT systems.

The most common methods of bee monitoring used in Precision Beekeeping [19–21] include measuring temperature and other environmental factors inside and outside the beehive, including its weight. Solutions based on monitoring temperature enable the

detection of various situations inside the beehive—increased food consumption, beginning of brood rearing, pre-swarming state, or even death of the bee family while controlling the weight of beehive is helpful to indicate the time for honey harvesting, lack of food during winter or swarming state without enormous financial expenses. Systems that analyze temperature and humidity are described in [22,23]—both enable real-time monitoring; the one developed by Braga et al. [24] includes also measuring the weight of a beehive, whereas the system developed by Kontogiannis [25] monitors conditions inside the beehive and can regulate them when necessary. The approach described by Debauche et al. [26] allows web monitoring not only for temperature and humidity but also for additional factors such as air pollution or wind speed. A similar solution is proposed in [27] and gathers data from various sensors for air gases. The solution proposed by Meitalovs et al. [28] enables video recording along with temperature and humidity monitoring, while the system described by Balta et al. [29] also allows audio recording. Beehive weight-based IoT systems are presented in [30,31]. A solution presented in [32], apart from controlling temperature, humidity, air pressure, and beehive weight, also uses a gesture sensor to analyze bee flight behavior. In recent years, some solutions were developed that use chemical analysis of the air in the beehive that enable detection of the *V. destructor* mite inside the beehive [33,34]. They assume that *Varroa destructor* influences gas composition. Although the above solutions do not belong to IoT systems, they are worth mentioning.

Another category of IoT systems developed for beekeeping includes those that allow for audio analysis. The analysis is useful for detecting a swarming state inside a bee colony [35–38]. Qandour et al. [39] classify the behavior of queens, scouts, forgers, and the whole bee family based on bee noises. An approach worth mentioning is presented by Van Goethem et al. [40]. Although this system analyses bee sounds to determine bee pollination efficiency, it is dedicated to fruit growers.

As the development in the electronics field proceeds, camera devices are becoming more accessible and more and more often used in IoT systems. Beekeeping systems are not an exception for this. Video monitoring solutions that monitor the number of bees entering and leaving a beehive are presented in [41,42]. Campbell et al. [41] described encountered problems, such as recognizing bee bodies and their shadows, while Chen et al. [42] proposed a solution with tags attached to the bee legs, which eliminated the difficulty. Babic et al. [43] developed a system that detects bearing honey bees, along with counting the number of bees. The real-time monitoring system proposed by Bjerger et al. [44] records bees going through a passageway and enables counting them as well as identifying mites on their bodies. Apart from the camera, the solution uses spectral waves that illuminate the bee body and ML methods to classify bees affected by the *Varroa destructor* mite. The accuracy of classification reached up to 99%, depending on classifier parameters. The solution developed by Elizondo et al. [11] locates *V. destructor* mite inside a honeybee cell. However, it does not acquire video nor image data by itself. The last two works show that there remains a space for developing Precision Beekeeping solutions that enable identifying possible threats such as *V. destructor* mites. Some existing solutions for bee monitoring will be extended for the capability to detect *V. destructor* mites. Such an approach is presented in [45] with a Raspberry Pi platform-based, energy-efficient, and scalable solution that enables tracking bees locally, sending cropped images to the Cloud. It consists of modules for pollen detection, bee position, and bee type recognition, and a classification module. The system described in [46] is also based on the Raspberry Pi platform and will be extended with a module for *V. destructor* detection using visual monitoring at the entrance and gas sensors devices. Although the system is not an IoT solution yet, the authors aim to extend it to that type of device. Table 1 presents a short comparison between the most important of the above works.

Table 1. Comparison of bee monitoring systems

Paper Authors	Data Type	Data Source	Purpose	Algorithm/Device
Campbell et al. [41]	Video	Own images	Bee counting	No information
Chen et al. [42]	Video	Own images	Bee counting	SVM/CCD camera, infrared LED light source, PC
Babic et al. [43]	Video	Own recordings	Bee counting, detect bees with pollen	MOG, NMC/Raspberry Pi, camera
Bjerge et al. [44]	Video, Spectral waves	Own recordings	Bee counting, identifying the varroa mite	LDA/Passageway, camera, mirror, LED arrays, laptop
Elizondo et al. [11]	Video	DB of Research Center of Tropical Apiculture Studies	Varroa detection inside honeybee cell	Background Subtraction, Binarization Process/No information
Marstaller [45]	Video	iNaturalist, own data	bee tracking, pollen detection, bee position, bee types recognition	cloud-based CNN/Raspberry Pi
Konig [46]	Sound, environmental conditions, video	Own data	Bee state, flight monitoring, Varroa detection	kNN/environmental conditions sensors, Raspberry Pi

The solutions mentioned above do not fully respond to beekeepers' needs. Although state-of-the-art papers about bee monitoring exist, the vast majority of them refer to controlling the conditions in- and outside the beehive. Moreover, the technological stack applied in most of the above does not stand with the cutting edge devices and algorithms. There are no comprehensive solutions that include image analysis based on edge-processing that could be easily scaled for multiple beehives and would not affect bees at the same time. Meanwhile, it is worth noting that scalability and fault tolerance are important issues while developing complex, network-based systems, such as the one presented in this paper. The decentralized solutions that use Peer-to-Peer connections (P2P) instead of the traditional Master-Slave approach feature good time performance with basically no loss of accuracy of data transmission [47]. Moreover, the P2P types of frameworks are less susceptible to network breakdowns due to the dysfunction of one of the peers, which makes the solution more scalable. The lack of such properties among the existing solutions led us to develop the presented approach that responds to the mentioned needs and fills these technical gaps. Our approach satisfies the scalability of performed analyses through TPU acceleration and the scalability of the monitoring process through IoT-dedicated cloud components. Fault tolerance is achieved by a redundancy of the IoT components, which is native to the cloud. Our proposal, presented in this paper, extends the current spectrum of existing works by:

- Providing a large-scale, Cloud-based infrastructure for monitoring bees and detecting varroosis that can be easily extended for multiple beehives;
- Enabling real-time analysis and detection of *Varroa destructor* with embedded artificial intelligence (AI) on the TPU-accelerated Edge IoT devices;
- Ensuring that gathering and processing data will not affect the regular activities of bees.

3. Experimental Environment

For the purpose of real-time and large-scale bee monitoring, we developed an experimental system composed of the monitoring IoT device and data center located in the Cloud

environment. Construction of the monitoring IoT device is presented in Figure 1. Software organization is shown in Figure 2. The architecture of the whole system is shown in Figure 3. Particular modules of the system are described in the following sections.

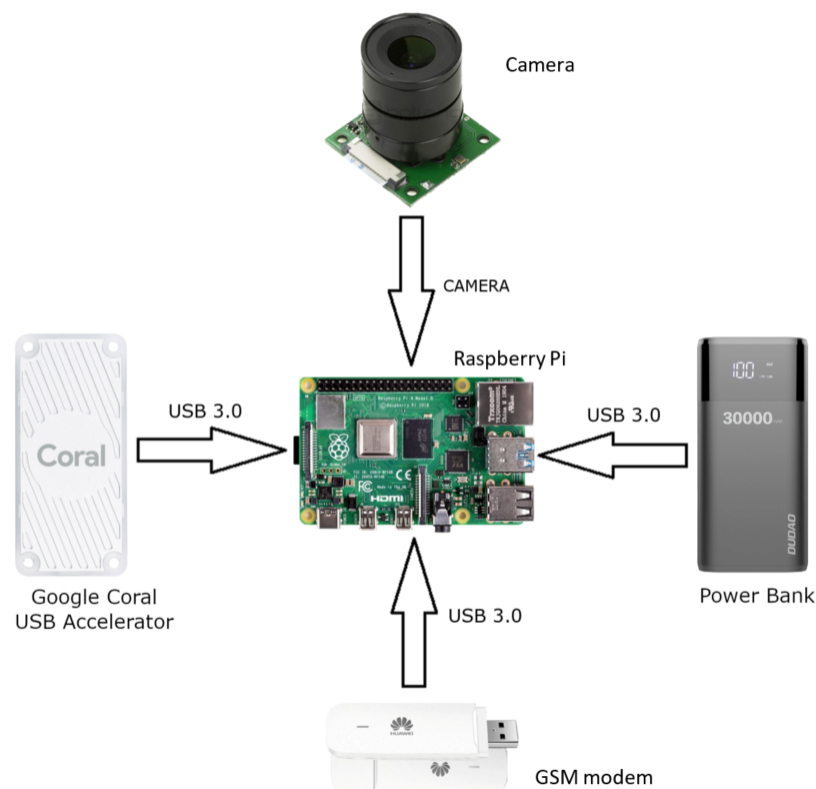


Figure 1. Overview of the IoT device used for monitoring a beehive.

3.1. Monitoring IoT Device with TPU Accelerator

We built the monitoring device based on the RaspberryPi 4 model B mini-computer with Broadcom BCM2711 quad-core 64-bit ARM-8 Cortex-A72 1,5 GHz processor and 4 GB of RAM (Raspberry Pi Foundation, Cambridge, UK) (Figure 1). The device is powered by an external source of electricity in the form of a power bank. Monitoring is performed with the use of an ArduCam OV5647 5Mpx camera with an LS-2718 CS lens sourced from Botland, Bralin, Poland. The camera supports the HD 1080 px mode, providing 30 frames per second. The camera is connected to a dedicated connector that the RaspberryPi is equipped with, ensuring fast data transfer. An additional GSM modem is used for transmitting data to the data center. The modem is connected to the central unit via a USB 3.0 connector, and it is equipped with its own SIM card. In such a way, it is independent of the availability of the broadband network in the vicinity of the beehive.

While having a network connection, it is possible to connect to the Cloud data center through the AWS IoT Cloud gateway service. For each frame of the recorded video stream, the IoT device performs bee identification (segmentation of bees) and *Varroa destructor* detection (the classification process). If the bee is classified as infected, the IoT device sends an MQTT message to the Cloud data center for further processing and notifying a beekeeper. The software organization is presented in Figure 2.

Detection of bees and of varroosis is carried out through the use of the Google Coral USB Accelerator. Thanks to the built-in Edge TPU co-processor, the Coral accelerator enables quick inference with Machine Learning (ML) models. The Edge TPU co-processor allows performing up to 4 trillion operations per second using only 2 watts of power [48]. This enables the implementation of modern mobile vision models with a speed of almost 400 frames per second in an energy-efficient manner. Moreover, the accelerator is compat-

ible with the Tensor Flow Lite library. A general model of the monitoring IoT device is shown in Figure 1.

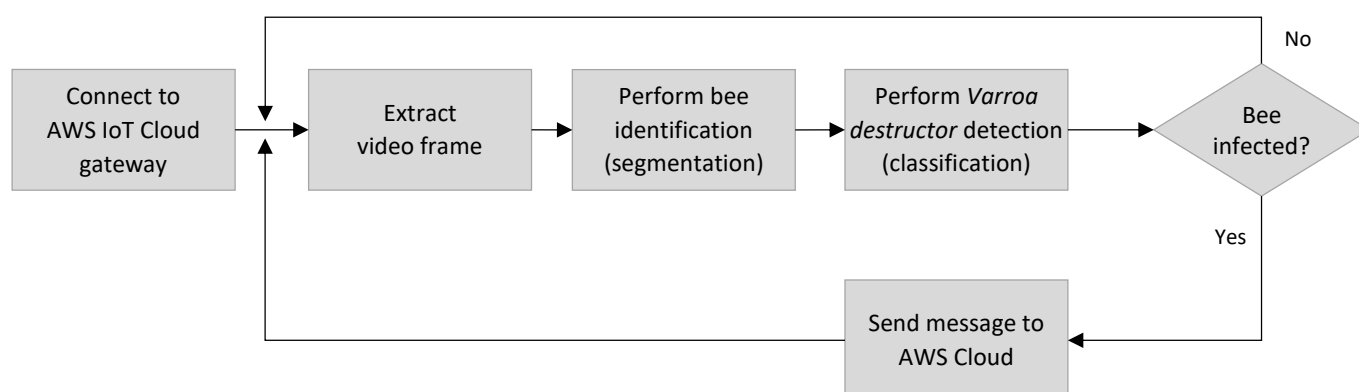


Figure 2. A software organisation for the *Varroa destructor* detection performed on the Edge IoT device.

The IoT device monitors the beehive and records a video data stream. Selected video frames of the video stream (images) are then analyzed in order to identify bees. Identified bees are marked with a surrounding rectangle (a frame indicating the region of interests, ROI) and extracted from the image. Then, extracted pictures of bees (ROIs) are analyzed again to detect *V. destructor* on bees (classification process). In such a way, each frame from the video stream delivers many smaller pictures of bees classified as infected or not. Both phases of the analysis are TPU-accelerated. In case of infection, the pictures of the infected bees are transferred to the data center for further analysis, storing, upgrading ML models, and sending a notification to the relevant beekeeper.

3.2. Cloud Environment for Bee Monitoring

The main data center for collecting data, geo-wide monitoring of many apiaries, and notifying beekeepers is located in the Amazon Web Services (AWS) cloud. Communication between the monitoring IoT device and the data center occurs mainly when the varroosis is detected and performed using the MQTT protocol. The data center acquires MQTT messages from IoT devices, saves the information in the database, and finally informs the end user (a beekeeper) with an email message that a bee infected with varroa virus was detected.

To perform these tasks, the cloud-based data center consists of several modules. First, the MQTT messages from monitoring IoT devices are received by the AWS IoT Core. The AWS IoT Core plays the role of the Cloud gateway. It enables the management of the information sent in the MQTT messages. The received data is stored in the AWS DynamoDB database. This is a key-value store that provides flexibility to changes in the data structure. It stores the data such as the timestamp of the message, the health state of the bee (healthy or sick), and the photo of a bee in the *Base64* binary format. Finally, if an infected bee is detected, the serverless AWS Lambda service launches a code responsible for sending an email message to the user with the AWS Simple Notification Service (AWS SNS). The whole process is presented in Figure 3.

Importantly, the detection of varroosis is performed at the Edge, at the TPU-accelerated IoT device. The device-to-cloud data transmission is initiated only in case of detecting *Varroa destructor* mites on bees. In such a way, the data transmission is limited to necessary cases—when there is a necessity to notify the beekeeper. The image data are transmitted together with the alert message to be stored in the global repository to constitute a reservoir of images of infected bees. These images supply the training phase and updates in the machine learning models that detect *Varroa destructor* infections. We train the ML models using the Google AI of the Google Cloud Platform, which is compatible with the Google

Coral Accelerator we use for the detection. The accelerator integrates well with its native Cloud environment.

The trained ML model for varroosis detection is then deployed to the Google Coral Accelerator. This is performed periodically when the detection model is updated to include new cases.

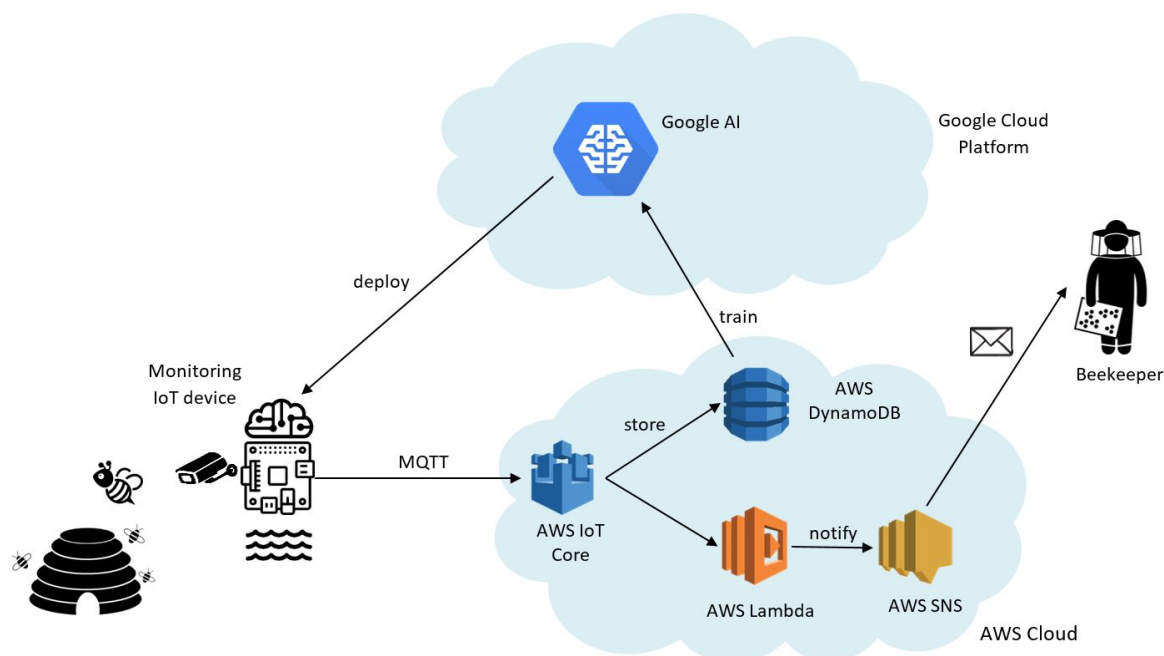


Figure 3. The architecture of the Edge- and Cloud-based experimental environment for monitoring bees and detection of varroosis.

3.3. ML Models for Bee Identification and Varroosis Detection

Both ML models for bee identification and *Varroa destructor* detection rely on convolutional neural networks (CNN). As CNNs are used mainly for image classification, the idea is to apply filtering on images, reducing the number of parameters involved in the classification process without losing essential features. The ML models creation relied on a reinforcement learning [49]. The general idea of this process is presented in Figure 4. A *parent* neural net controller generates architectural hyper-parameters of neural networks and proposes a *child* model architecture. Then, a CNN model is trained—at first, features are filtered and extracted into feature maps (convolution step) and combined within smaller structures (pooling step). These steps may be repeated a few times. Then, a fully connected layer is added, which enables the learning of non-linear combinations of the most important features. Finally, a softmax algorithm is executed to differentiate between high and low-level image details and perform classification. As feedback, the *child* returns to the *parent* quality parameters and, based on that controller, decides whether to generate an improved model or not. Google AutoML Vision maximizes the accuracy of the model architecture by implementing the policy gradient method to automatically search for the optimal CNN model. Real ML models for bee identification and *Varroa destructor* detection are presented in Appendix A, Figure A1 and Appendix B, Figure A2. The models developed in such a way expect images of 512×512 px size with 3 channels for the bee identification and 224×224 px size with 3 channels for *V. destructor* detection. Then, 32 (bee identification) and 48 (varroa detection) filters designed as $3 \times 3 \times 3$ structures are applied for the first convolutional layer. The bee identification model is wide and results in ten convolutional layers of 48 and 24 filters before reshaping and concatenating the output. Then, a quantization process is performed to reduce model size with only a little loss of accuracy. This optimization operation is important for the models that will be used on

devices such as ours. The *Varroa detection* model is smaller, and the model output consist of a 2-dimensional structure with the number of neurons equal to 1280.

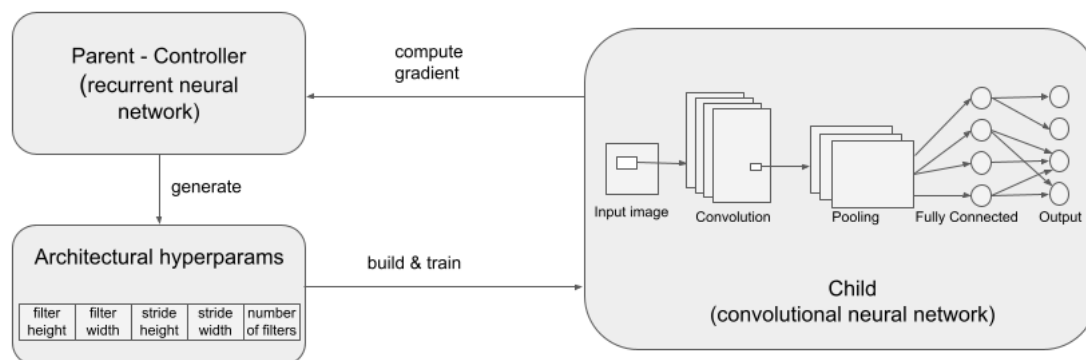


Figure 4. General image analysis employing CNNs with reinforced learning for bee identification and varroosis detection.

4. Experimental Evaluation

The capabilities of the developed IoT device and the entire environment for constant monitoring of beehives with the real-time detection of varroosis through the analysis of video data stream were evaluated in a series of experiments. Within these experiments we investigated:

- The effectiveness of bee identification (segmentation of ROIs, Section 4.1) and *Varroa destructor* detection (the classification of bee images, Section 4.2);
- Time performance of both operations (Sections 4.3 and 4.4);
- The impact of camera resolution on both detection processes (Section 4.5);
- Device resources consumption (Section 4.6).

4.1. Effectiveness of Bee Identification

The bee identification process tested within these experiments was performed with the AutoML Vision Edge service in the Google Cloud Platform. Due to its compatibility with Edge TPU, AutoML is recommended for preparing detection and classification models. The AutoML Vision Edge service builds models through transfer learning and searching neural architecture to find the best network architecture. Moreover, it optimizes hyperparameters of that architecture to minimize the loss function of the trained models.

As a training set for the bee segmentation model, we used 300 images (video frames) extracted from the video data stream, where bees were marked with rectangles (frames). The images were taken by the camera mounted on the monitoring IoT device described in Section 3.1. Since images were acquired during the whole day, the training set included pictures taken in various light conditions, with different numbers and positions of bees. The example of an image included in the training set is shown in Figure 5.

While evaluating the effectiveness of the detection processes, we used the following rates and effectiveness indicators:

- A confusion (error) matrix—presents the performance of the classification algorithm in a visual manner, describes the number of actual and predicted cases classified properly as true positives (TP) and true negatives (TN) or in a wrong manner as false positives (FP) and false negatives (FN).
- Precision (positive predicted value—PPV)—the number of properly classified positive results (e.g., correctly identified bees) to a sum of all positives, including those incorrectly classified as positives, which is defined as:

$$PPV = \frac{TP}{TP + FP} \quad (1)$$

- Sensitivity (recall, true positive rate—TPR)—the number of properly classified positives (e.g., correctly identified bees) divided by all samples that should have been classified as positives, defined as:

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN} \quad (2)$$

- F1-score, which stands for the harmonic mean of precision and sensitivity:

$$F_1 = 2 * \frac{PPV * TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN} \quad (3)$$

The effectiveness of bee identification was evaluated with a test set containing 20 images with various numbers of bees, their positions, and in different light conditions. These images were processed by the bee identification model. After this step, we obtained a copy of the image where detected bees were marked with ROIs (bee segmentation). To ensure the readability of presented data, the images were named with numbers, beginning from 1 up to 20. Then, the results were analyzed manually to determine whether all the bees were detected properly. For every single image, we prepared the confusion matrix that allowed us to calculate the values of precision (1), sensitivity (2), and F1 score (3) rates, which are presented in Table 2.

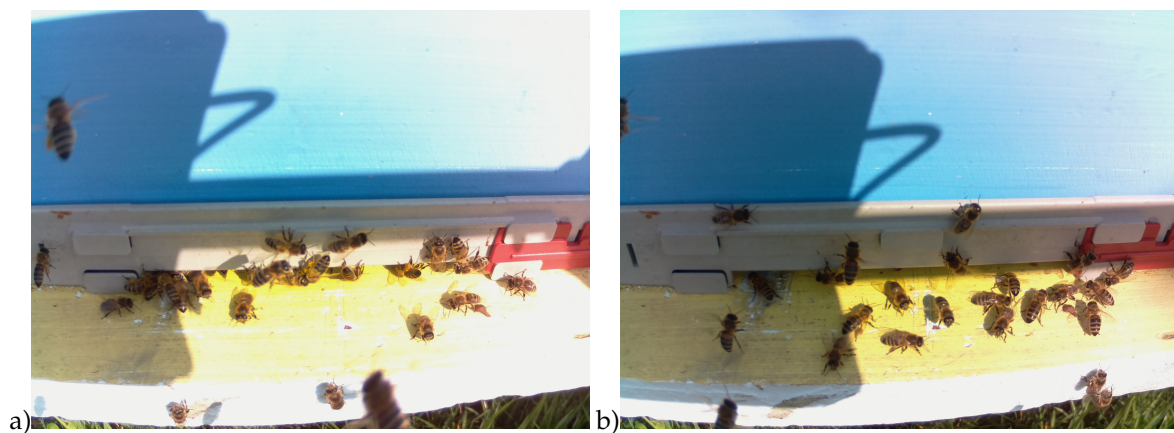
Analyzing the obtained results, we can observe that only two bees were identified for one image (no. 19), even though they were not present in the image (False Positive rate). The bee identification model performed a bit worse while finding bees assembled in groups, staying close to each other. The sensitivity rate (TPR) and F1 score reflect this situation—both of them have a wide range of values for different images. Figure 6 presents images no. 7 and no. 10, for which we obtained the worst detection results. However, considering that bees are leaving and entering a beehive a couple of times a day, and the IoT device will take their pictures several times, the outcome of this experiment is auspicious.



Figure 5. A sample picture extracted from the video data stream with bees marked with rectangles (frames) used for training the model of bee detection.

Table 2. Effectiveness of bee identification for various images (video frames).

Image No.	TP	FP	FN	TN	PPV	TPR	F1
1	9	0	2	0	1	0.81	0.9
2	8	0	2	0	1	0.80	0.89
3	4	0	0	0	1	1	1
4	11	0	1	0	1	0.92	0.96
5	11	0	4	0	1	0.73	0.85
6	13	0	2	0	1	0.87	0.93
7	12	0	10	0	1	0.54	0.71
8	14	0	6	0	1	0.7	0.82
9	12	0	5	0	1	0.71	0.83
10	12	0	11	0	1	0.52	0.69
11	11	0	2	0	1	0.85	0.88
12	9	0	3	0	1	0.75	0.86
13	6	0	1	0	1	0.86	0.92
14	5	0	1	0	1	0.83	0.91
15	6	0	1	0	1	0.86	0.92
16	7	0	0	0	1	1	1
17	2	0	1	0	1	0.67	0.8
18	10	0	0	0	1	1	1
19	14	2	0	0	0.86	1	0.8
20	8	0	2	0	1	0.8	0.89

**Figure 6.** Images no. 7 (a) and no. 10 (b).

4.2. Effectiveness of Detecting Varroosis

The classification model for detecting *Varroa destructor* was also prepared with the AutoML Vision Edge Service. This time, the training set consisted of 10,743 images that included 5748 pictures of healthy bees and 4995 pictures of bees infected by *Varroa destructor* (unhealthy ones). Since there was no sufficient amount of pictures with infected bees, we used image augmentation techniques to prepare the training set by randomly rotating, shifting, and locating the *V. destructor* on chosen parts of the bee bodies. To assess the effectiveness, the experiment itself took under analysis 50 pictures of bees (the test set) that were not a part of the training set used during the learning phase. For this series of experiments, we obtained the following results—the precision (PPV) of the model reached nearly 0.7, sensitivity (TPR)—0.94, while the F1 score was at the level of 0.8.

Additionally, for presenting the effectiveness of the varroosis detection model, we also calculated the specificity (True Negative rate, TNR):

$$TNR = \frac{TN}{N} = \frac{TN}{TN + FP}. \quad (4)$$

The specificity of the varroosis detection model was 0.92. Taking into account all the above rates, the outcome of this experiment is satisfactory.

4.3. Time Performance of Bee Identification

Time performance measurements for both image analysis steps (bee identification and the *V. destructor* mite detection) were performed according to a similar schema presented in Figure 7. The step *Perform operations* stands for image segmentation (bee identification) process and image classification for the *Varroa detection* process. Time measurements covered the TPU inference for a particular step.

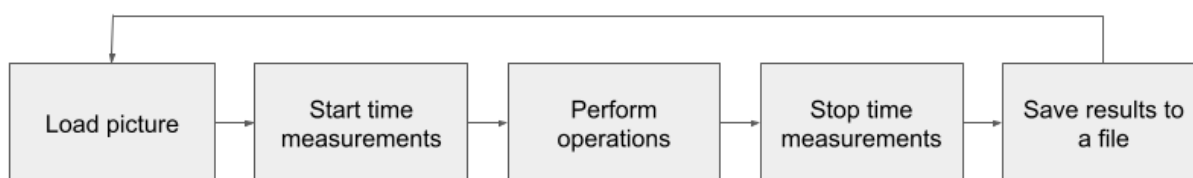


Figure 7. Time performance computing steps for bee identification and *Varroa destructor* detection.

The time performance of the bee identification process was analyzed for each of the 20 images (video frames) that were input as data for the bee identification classification model. In addition, for the experiments, we chose images in the highest possible resolution of 2592×1944 supported by the device's camera, which gives 5 Mpx. The outcome is presented in Figure 8. We can notice that the time needed to process an image was similar for most of the video frames. It took a maximum of 795 ms to analyze image no. 1 and a minimum of 769 ms to analyze image no. 17. The average time of processing a single image extracted from the video stream was 778.25 ms, whereas the standard deviation was 5.37 ms. It is also worth mentioning that there was no clear relationship between the number of identified bees and processing time for the video frame. Taking all of the above into account, the obtained results are satisfactory.

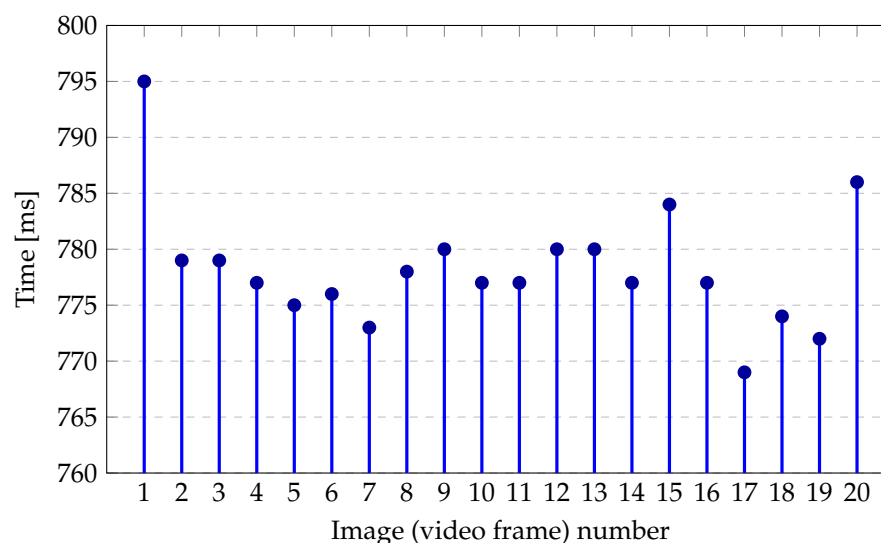


Figure 8. Time performance for bee identification.

4.4. Time Performance of Varroosis Detection

To examine the time performance of varroosis detection, we used 20 pictures of bees selected from the test set used for the experiment described in Section 4.2. Pictures had various sizes, resolutions and represented both healthy and infected bees to check if any of the above factors impacted the time performance. The results are shown in Figure 9. For presentation purposes, pictures were numbered as follows: numbers from 1 to 10 for pictures with healthy bees, 11 to 20 for pictures presenting the *V. destructor*-infected bees.

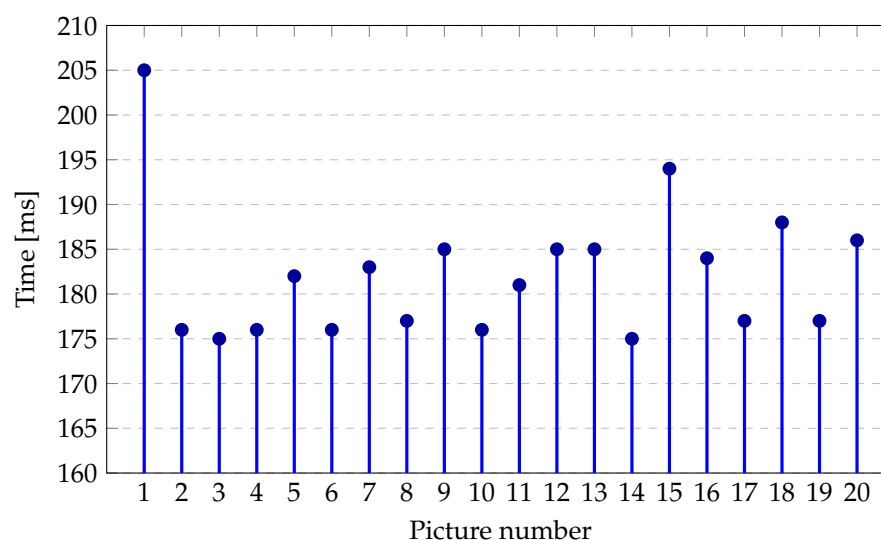


Figure 9. Time performance for the *V. destructor* detection.

For the obtained results, we calculated the average detection time, which took 182.15 ms with a standard deviation of 7.32 ms for all of the pictures, 181.1 ms with a standard deviation of 8.65 ms for pictures with healthy bees, and 183.2 ms with a standard deviation of 5.51 ms for pictures of infected bees. The average time was similar for both groups, and differences between standard deviations are acceptable. Moreover, the picture with the longest processing time (no. 1) had a resolution of 190×146 pixels, whereas the picture with the shortest processing time (no. 14) had a resolution of 1446×124 pixels. The largest picture (no. 18) with a resolution of 632×496 pixels did not generate the longest processing time. This information led us to conclude that none of the following factors—resolution, size, and the content of pictures—directly influence the performance of the *V. destructor* detection process.

4.5. Influence of Camera Resolution on the Identification and Detection Processes

The last series of experiments was focused on verifying the impact of camera resolution on the whole detection process. Here, we wanted to check:

- The influence of the quality of images obtained from video frames on bee identification process;
- The influence of the quality of video frames (camera resolution) on the varroosis detection;
- The influence of the camera resolution on the time performance of bee identification and the *V. destructor* detection.

For all of the previously described experiments (Sections 4.1:4.4), the analyzed (input) images had a resolution of 5 Mpx. In this part, we processed each of the video frames from the 20-element test set to lower the resolution to 1.2 Mpx, 0.3 Mpx, and 0.1 Mpx. For these four resolutions, we performed the bee identification again, checking its effectiveness. The effectiveness was measured in terms of precision, sensitivity, and the F1 score.

The precision achieved for the bee identification process was almost the same for all used camera resolutions. In most cases, it was close to 1.0. It differed only for image no.

19—we achieved the lowest value of 0.76 for the resolution 0.3 Mpx. The average precision, calculated for all images in particular resolution, is presented in Table 3.

Table 3. Average precision for bee identification depending on camera resolution.

Camera Resolution [Mpx]	Avg. Precision
0.1	1
0.3	0.98
1.2	0.99
5.0	0.99

We noticed much more diversity for sensitivity, which is presented in Figure 10. For camera resolution 0.1 Mpx, the sensitivity was the lowest for most of the analyzed images. Nevertheless, for the other three resolutions, obtained results are close to each other, and differences between them are rather small. The average sensitivity of 0.77 was noted for the two highest resolutions (5 Mpx and 1.2 Mpx), 0.78 for 0.3 Mpx, and 0.68 for the lowest resolution (0.1 Mpx).

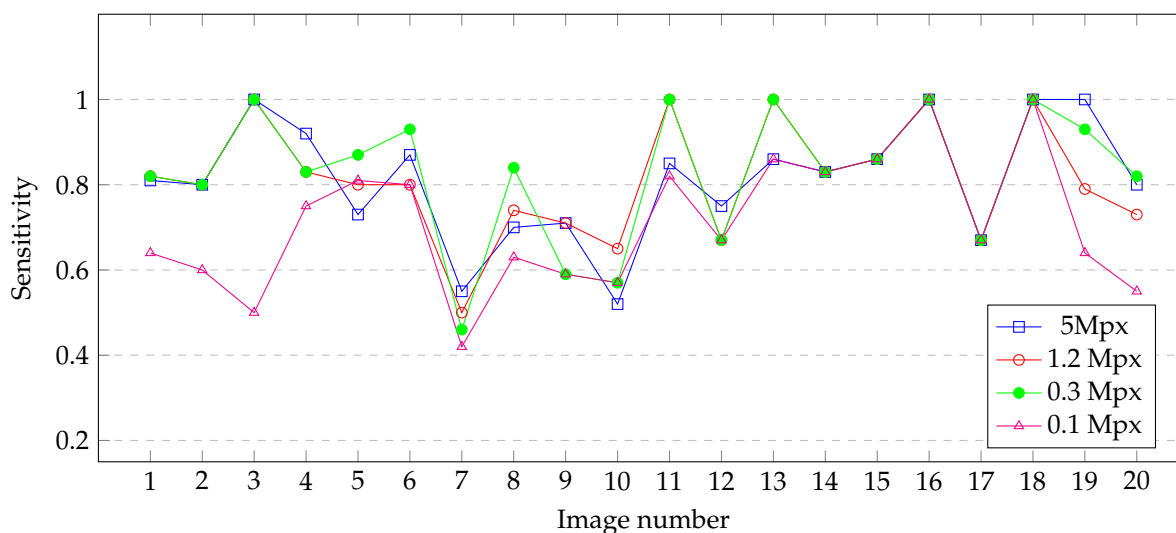


Figure 10. Sensitivity achieved for the bee identification process for different camera resolutions.

Figure 11 shows the F1 score measure obtained for particular images in tested resolutions. For some of the processed and analyzed images, the camera resolution does not influence the bee detection process. There are some images for which the camera resolution is important, which is also confirmed by the average value of the F1 score. As it was also visible for the sensitivity, the worst results (with the average F1 score reaching 0.81) were noticed for camera resolution 0.1 Mpx, while for other resolutions, the score reached 0.87.

The second part of this experiment covered the analysis of the impact of camera resolution on the *V. destructor* detection. Again, we used the initial test set with images taken with the 5 Mpx camera, which we processed to lower the resolution. The identified bees (there are usually many bees on each image) were manually labeled as healthy and infected. Then, we performed the detection of the *Varroa destructor* and calculated the effectiveness based on the confusion matrices obtained for different resolutions (5, 1.2, 0.3, 0.1 Mpx). Results are shown in Table 4.

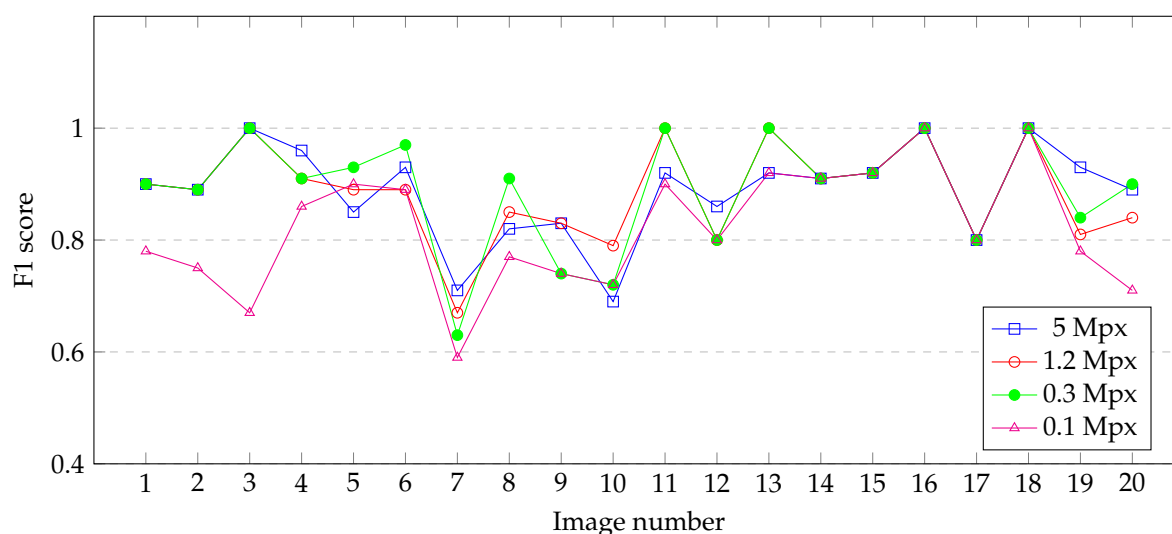


Figure 11. F1 score obtained for the bee identification process for different camera resolutions.

Table 4. Effectiveness of *V.destructor* detection depending on camera resolution (TPR—sensitivity, TNR—specificity, PPV—precision, ACC—accuracy, F1 score).

Camera Resolution (Mpx)	TPR	TNR	PPV	ACC	F1
0.1	1.000	0.000	0.185	0.185	0.313
0.3	1.000	0.000	0.167	0.167	0.286
1.2	1.000	0.043	0.185	0.214	0.313
5.0	1.000	1.000	1.000	1.000	1.000

Obtained results show that the detection capabilities depend highly on the camera resolution. For lower image resolutions, the classifier categorized most of the bees as infected (which could be observed in confusion matrices). Only for the highest resolution of 5 Mpx, the *V. destructor* detection was fully successful and reflected the actual state. This proves that, unlike bee identification, the detection of varroosis depends on the quality of the video stream. We expected that better quality images would result in better detection capabilities. However, we wanted to confirm how large the influence is to confirm the camera choice.

In the last part of the experiments, we investigated the influence of the image quality on the time performance of bee identification and *V. destructor* detection processes. Experiments were performed for the same four resolutions (5 Mpx, 1.2 Mpx, 0.3 Mpx, and 0.1 Mpx) and the same initial test set of 20 video frames. The analysis time (covering bee segmentation and varroosis detection) was measured for each image separately and averaged, first for all pictures of bees segmented from the image (ROIs) and then for all pictures of bees obtained for a particular resolution. Results are presented in Figure 12.

We observed that the whole execution times for images taken with camera resolutions of 5 Mpx are 5–30% higher than for images in the resolution 0.1 Mpx. However, the number of correctly segmented bees on images in the resolution 5 Mpx is higher than for the resolution 0.1 Mpx. When looking at the average time per segmented bee (Figure 12), we can see that for the lowest resolution 0.1 Mpx, the average processing time of a picture was around 296 ms, which was the highest of all results. For the other three resolutions, average times were similar—about 277 ms. Taking into account the results from previous experiments, it seems that analyzing video streams with a resolution of 5 Mpx would be the best choice regarding both efficiency and quality of bee segmentation and varroosis detection.

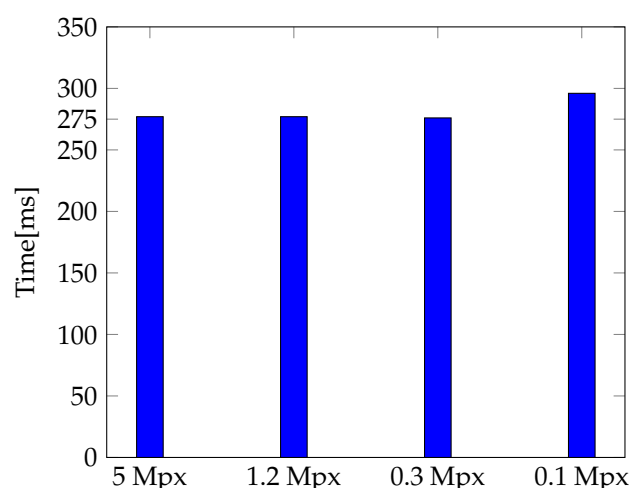


Figure 12. Average processing time of a single bee picture for different resolutions.

4.6. IoT Device Power Consumption

The IoT device we developed was powered by a power bank with a capacity of 30,000 mAh. To check the power consumption of the device, we placed it in the front of a beehive to capture the video stream constantly. Since the varroosis rarely occurs in well-performed beekeeping, measuring power consumption in such a state would not give a good view of the possible requirements for energy (the IoT device would not transmit anything to the Cloud). For this reason, we simulated the occurrence of *Varroa destructor* on some video frames by swapping every 10th frame in the video stream by a randomly chosen frame from the test set containing 20 frames with some infected bees. This approach led to sending an MQTT message with images of affected bees to the data center in the Cloud after analyzing every ten frames of the video stream.

Under the above simulation conditions, the IoT device could work for 10 hours and 35 minutes, which means the whole day of monitoring the beehive without a need of changing or recharging the power supply. Taking into account the fact that in natural conditions, the device would not send MQTT messages so often, the obtained results are satisfactory but lead to the conclusion that the best choice would be to power the device with an additional photo-voltaic panel.

5. Discussion

IoT devices that allow for the monitoring of beehives and detection of varroosis are becoming an important example of the use of advanced IT technologies and electronics in the natural sciences. Such devices can help prevent the spread of *Varroa destructor* between hives and whole apiaries worldwide, which is important not only for beekeeping itself but also indirectly influences the cultivation of plants and the food industry.

Our solution, likewise presented in [45,46], is built on a single-board computer (Raspberry Pi) platform and aims to analyze video streams with bees and detect varroosis. These are the only similarities between the three systems. However, our solution extends the capabilities of the mentioned solutions by including the modules for the *V. destructor* detection. The method of image processing in [45] includes sending cropped images of bees to a cloud for further analysis (i.e., genus identification and pollen detection), whereas our solution is based on Edge processing, and cloud operations are performed only when the *V. destructor* mites are detected. This Edge computing-based approach allows limiting the data transmitted to the data center and the storage space occupied. Indus Bees 4.0 [46] is rather a prototype of a system. The author does not provide any results for *V. destructor* detection yet. However, he mentions Intel Neural Compute Stick (NCS-2) as a potential accelerator. Currently, he provides only the first no-detailed results for the *V. destructor* counting system, which still uses camera images taken during routine hive checks. The experiment was taken on a prepared clean slider that included 84 mites and used a k-NN

classifier. The solution was more focused on the IoT device than on cooperation with a cloud data center.

On the other hand, compared with existing solutions for *V. destructor* detection, our solution complements them through providing IoT devices capable of data acquisition and early analysis, and through a cloud-based data system for geo-wide analyses and training of detection models. In contrast to the solution proposed by Elizondo et al. [11], who analyzed video samples taken from DB of Research Center of Tropical Apiculture Studies (National University of Costa Rica), we can collect the data with the designed IoT device and detect *V. destructor* mite on it. Unlike Bjerger et al. [44], who recorded images of bees and illuminated their body with spectral waves to locate the *V. destructor* mite with a special passageway (which can be noticed as invasive), our approach is non-invasive since bees are monitored from some distance without any modifications in the beehive construction. Our solution is also comprehensive—it provides image acquisition, on-edge machine learning-based detection, and cloud processing by notifying the user about detected *Varroa destructor*. By using the Cloud platform, it can also be easily used on a wider scale and globally available (although it requires a cloud subscription for keeping the data center alive on the Cloud). We managed to obtain satisfactory results for identifying bees in the video frames and classifying them as infected or healthy ones using only the camera mounted at the beehive entrance. That approach does not affect bees and their bodies being effective at the same time.

In terms of the bee and *Varroa destructor* detection quality, we also obtained promising results. Whereas in [33], Szczurek et al. used gas sensors and the SVM classifier and achieved the sensitivity (TPR) between 0.67 and 0.75, depending on the category of *V. destructor* infestation level (low, medium, high), and specificity (TNR) between 0.93 and 0.97, our solution for *V. destructor* detection reached 0.94 for TPR and 0.92 for TNR. These are significantly better results for TPR and slightly worse for TNR. What is also worth noting is that in the solution presented in [33], the authors recorded the data locally on an SD card, and they trained the SVM model in the MathWorks environment. Although the authors mention the possibility of remote data transmission with the GSM module, there is no information that the solution, unlike ours, uses any Cloud or Edge processing. Additionally, for kNN and LDA classifiers and the same method of data acquisition, Szczurek et al. [34] achieved a bit worse results of *V. destructor* detection than we did.

In terms of detection quality, Bjerger et al. [44] reached the F1 score at a maximum level of 0.91 when using a CNN classifier for *Varroa destructor* detection based on image analysis. Although this is a better result than ours (F1 = 0.8), their solution affected bees' natural environment—they used a passageway for bee illumination and recording. The main differences between various solutions for bee monitoring, including detection of the *Varroa destructor* mite, and our approach are summarized in Table 5. We compared the approaches according to the capability of detecting the *Varroa destructor*, the type of analyzed data, the algorithm used for the analysis, the device platform, and Cloud- and Edge-based processing capabilities.

While analyzing four different camera resolutions in our research, we concluded that the most accurate results were obtained with 5 Mpx as image resolution. The detection and classification results are the best, and the average time needed for processing a single image is similar to two lower resolutions. Measured processing times are also satisfactory and do not differ significantly for both healthy and infected bees.

Table 5. Comparison of bee monitoring solutions, including detection of the *Varroa destructor*.

Solution	Varroosis Detection	Data Type	Algorithm	Device	Cloud Processing	Edge Processing
Marstaller et al. [45]	Mentioned, not implemented	Video	CNN	Raspberry Pi	Yes	No
Konig [46]	Mentioned, not implemented	Video, temperature, humidity, weight	kNN	Raspberry Pi, sensors	No	No
Elizondo et al. [11]	In honeybee cells	Video	Background subtraction	No information	No	No
Bjerger et al. [44]	Yes	Video, Spectral waves	LDA	Passageway, camera, mirror, LED arrays, laptop	No	No
Szczurek et al. [33]	Yes	Gas measurements	SVM	Gas sensors	No	No
Szczurek et al. [34]	Yes	Gas measurements	kNN, LDA	Gas sensors	No	No
Our (proposed) solution	Yes	Video	Neural network (AutoML)	RaspberryPi + TPU	Yes	Yes

6. Conclusions

In this paper, we presented a comprehensive and efficient solution for detecting and classifying the *Varroa destructor* mite on bees' bodies. Our approach enables acquiring image data by IoT device, performing machine learning operations related to bee and varroosis detection at the device, transferring the suspicious cases to the data center located in the Cloud, and informing a beekeeper when the infection is detected. The results of our experiments confirm that with the presented approach relying on Edge-Cloud computing hybrid architecture, we can perform the detection of the disease effectively and efficiently. For most of the analyzed video frames, the precision rate (1) reached the value of 1 for bee identification and about 0.7 for the *Varroa destructor* detection process. For the sensitivity rate (2), most values exceeded 0.7 for bee recognition and 0.9 for the *Varroa destructor* detection. The F1 score value was similar for both processes and reached around 0.8. Although the effectiveness of the detection processes is still not perfect, we have to remember that the point is to detect *Varroa destructor* in the beehive as quickly as possible before the disease starts to spread. This is more important than detecting every case of the disease since, after the first notification, the beekeeper can start the healing procedure. In this regard, our solution works sufficiently well.

In the future, we can extend our solution in several directions. First of all, we think about improving device powering by adding solar panels. That would make our solution more eco-friendly and limit the frequency of power bank recharging. Another idea can refer to extending a system with bee activity monitoring. Bee behavior can reveal some abnormalities, including various mite infections. For that purpose, using context predictions would be quite an innovative concept regarding bee monitoring. Several context-aware applications, such as those presented in [50] and context-history based systems [51–53] have proved promising results in other fields (project management—[52], competences management—[50], thus predicting bee condition based on history of their behaviour may bring interesting conclusions. Finally, our further works consider trying other placements of the device to find the best location for obtaining good results, provide easy access for a beekeeper and be non-invasive for bees at the same time. Each approach from the above will bring some new findings and enrich smart beekeeping.

Author Contributions: D.M. conceived and designed the experiments; R.G. performed the experiments; A.W., R.G. and D.M. analyzed the data; B.M.-M. designed and implemented the data analysis workflows; A.W. and D.M. wrote the paper; A.W., B.M.-M. and D.M. corrected the paper. All authors have read and agreed to the published version of the manuscript.

Funding: The work reported in the paper is partially supported by the European Social Funds; project no POWR.03.02.00-00-I007/17-00 “CyPhIS – the program of modern PhD studies in the field of Cyber-Physical Systems”, and partially, by the professorship grant (02/020/RGP19/0184) and the pro-quality grant for highly scored publications or issued patents of the Rector of the Silesian University of Technology, Gliwice, Poland (grant No 02/100/RGJ21/0009), and by Statutory Research funds of Institute of Informatics, Silesian University of Technology, Gliwice, Poland (grant No BK/204/RAU7/2021).

Institutional Review Board Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

ABPV	Acute Bee Paralysis Virus
API	Application Programming Interface
AWS	Amazon Web Services
CCD	Colony Collapse Disorder
CCN	Convolutional Neural Networks
DLL	Dynamic-Link Library
DWV	Deformed Wing Virus
F1	F1 score
FN	False Negative
FP	False Positive
IoT	Internet of Things
KBV	Kashmir Bee Virus
PPV	Positive Predicted Value or precision
ROI	Region of Interests
SDK	Software Development Kit
TN	True Negative
TNR	True Negative Rate
TP	True Positive
TPR	True Positive Value or sensitivity (recall)
TPU	Tensor Processing Unit
VMs	virtual machines

Appendix A. ML Model for Bee Identification

The structure of the convolutional neural network (CNN) used for bee identification is presented in Figure A1. The high-quality figure presenting the structure of the CNN is available as a supplementary file.

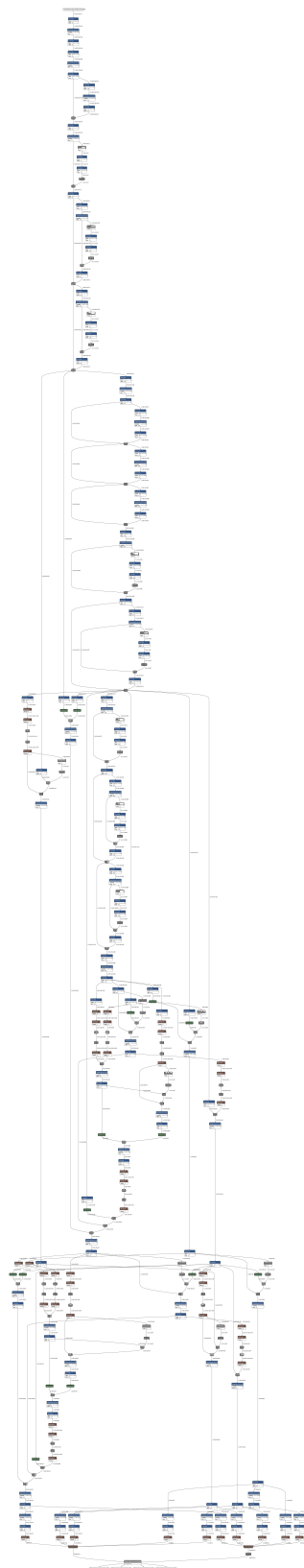


Figure A1. Structure of the convolutional neural network (CNN) used for bee identification.

Appendix B. ML Model for *Varroa destructor* Detection

The structure of the convolutional neural network (CNN) used for bee identification is presented in Figure A2. The high-quality figure presenting the structure of the CNN is available as a supplementary file.



Figure A2. Structure of the convolutional neural network (CNN) used for *Varroa destructor* detection.

References

1. Yin, Y.; Zeng, Y.; Chen, X.; Fan, Y. The internet of things in healthcare: An overview. *J. Ind. Inf. Integr.* **2016**, *1*, 3–13, doi:10.1016/j.jii.2016.03.004.
2. Talavera, J.M.; Tobón, L.E.; Gómez, J.A.; Culman, M.A.; Aranda, J.M.; Parra, D.T.; Quiroz, L.A.; Hoyos, A.; Garreta, L.E. Review of IoT applications in agro-industrial and environmental fields. *Comput. Electron. Agric.* **2017**, *142*, 283–297, doi:10.1016/j.compag.2017.09.015.
3. Unold, O.; Nikodem, M.; Piasecki, M.; Szyc, K.; Maciejewski, H.; Bawiec, M.; Dobrowolski, P.; Zdunek, M. IoT-Based Cow Health Monitoring System. In Proceedings of the Computational Science—ICCS 2020, Amsterdam, The Netherlands, 3–5 June 2020; Krzhizhanovskaya, V.V., Závodszy, G., Lees, M.H., Dongarra, J.J., Sloat, P.M.A., Brissos, S., Teixeira, J., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 344–356, doi:10.1007/978-3-030-50426-7_26.
4. van der Sluijs, J.P.; Simon-Delso, N.; Goulson, D.; Maxim, L.; Bonmatin, J.M.; Belzunces, L.P. Neonicotinoids, bee disorders and the sustainability of pollinator services. *Curr. Opin. Environ. Sustain.* **2013**, *5*, 293–305, doi:10.1016/j.cosust.2013.05.007.
5. VanEngelsdorp, D.; Underwood, R.; Caron, D.; Hayes, J. An Estimate of Managed Colony Losses in the Winter of 2006–2007: A Report Commissioned by the Apiary Inspectors of America. *Am. Bee J.* **2007**, *147*, 599–603.
6. Seitz, N.; Traynor, K.S.; Steinhauer, N.; Rennich, K.; Wilson, M.E.; Ellis, J.D.; Rose, R.; Tarpy, D.R.; Sagili, R.R.; Caron, D.M.; et al. A national survey of managed honey bee 2014–2015 annual colony losses in the USA. *J. Apic. Res.* **2015**, *54*, 292–304, doi:10.1080/00218839.2016.1153294.
7. Guzman, E.; Eccles, L.; Calvete, Y.; McGowan, J.; Kelly, P.; Correa-Benítez, A. Varroa destructor is the main culprit for the death and reduced populations of overwintered honey bee (*Apis mellifera*) colonies in Ontario, Canada. *Apidologie* **2010**, *41*, 443–450, doi:10.1051/apido/2009076.
8. Cornman, R.S.; Tarpy, D.R.; Chen, Y.; Jeffreys, L.; Lopez, D.; Pettis, J.S.; van Engelsdorp, D.; Evans, J.D. Pathogen Webs in Collapsing Honey Bee Colonies. *PLoS ONE* **2012**, *7*, e0043562, doi:10.1371/journal.pone.0043562.
9. vanEngelsdorp, D.; Traynor, K.S.; Andree, M.; Lichtenberg, E.M.; Chen, Y.; Saegerman, C.; Cox-Foster, D.L. Colony Collapse Disorder (CCD) and bee age impact honey bee pathophysiology. *PLoS ONE* **2017**, *12*, e0179535, doi:10.1371/journal.pone.0179535.
10. Barron, A.B. Death of the bee hive: Understanding the failure of an insect society. *Curr. Opin. Insect Sci.* **2015**, *10*, 45–50, doi:10.1016/j.cois.2015.04.004.
11. Elizondo, V.; Briceno, J.; Travieso, C.; Alonso, J. Video Monitoring of a Mite in Honeybee Cells. *Adv. Mater. Res.* **2013**, *664*, 1107–1113, doi:10.4028/www.scientific.net/AMR.664.1107.
12. Doublet, V.; Labarussias, M.; de Miranda, J.R.; Moritz, R.F.A.; Paxton, R.J. Bees under stress: Sublethal doses of a neonicotinoid pesticide and pathogens interact to elevate honey bee mortality across the life cycle. *Environ. Microbiol.* **2015**, *17*, 969–983, doi:10.1111/1462-2920.12426.
13. Francis, R.M.; Nielsen, S.L.; Kryger, P. Varroa-Virus Interaction in Collapsing Honey Bee Colonies. *PLoS ONE* **2013**, *8*, e0057540, doi:10.1371/journal.pone.0057540.
14. Rosenkranz, P.; Aumeier, P.; Ziegelmann, B. Biology and control of Varroa destructor. *J. Invertebr. Pathol.* **2010**, *103*, S96–S119, doi:10.1016/j.jip.2009.07.016.
15. Rinkevich, F.D.; Danko, R.G.; Healy, K.B. Influence of Varroa Mite (*Varroa destructor*) Management Practices on Insecticide Sensitivity in the Honey Bee (*Apis mellifera*). *Insects* **2017**, *8*, 9, doi:10.3390/insects8010009.
16. Dainat, B.; vanEngelsdorp, D.; Neumann, P. Colony collapse disorder in Europe. *Environ. Microbiol. Rep.* **2012**, *4*, 123–125, doi:10.1111/j.1758-2229.2011.00312.x.
17. Gates, B.N. *The Temperature of the Bee Colony*; United States Department of Agriculture: Washington, DC, USA, 1914; p. 622. Available online: <https://www.biodiversitylibrary.org/bibliography/64474> (accessed on: 19-09-2021).
18. Dunham, W. *Hive Temperatures for Each Hour of a Day*; The Department of Zoology and Entomology: Columbus, OH, USA, 1931; pp. 181–188.
19. Kviesis, A.; Zacepins, A. System Architectures for Real-time Bee Colony Temperature Monitoring. *Procedia Comput. Sci.* **2015**, *43*, 86–94, doi:10.1016/j.procs.2014.12.012.
20. Zogovic, N.; Mladenovic, M.; Rašić, S. From Primitive to Cyber-Physical Beekeeping. In Proceedings of the 7th International Conference on Information Society and Technology, Kopaonik, Serbia, 12–15 March 2017; pp. 38–43.
21. Zacepins, A.; Kviesis, A.; Pecka, A.; Osadcuks, V. Development of Internet of Things concept for Precision Beekeeping. In Proceedings of the 2017 18th International Carpathian Control Conference (ICCC), Sinaia, Romania, 28–31 May 2017; pp. 23–27, doi:10.1109/CarpathianCC.2017.7970365.
22. Gil-Lebrero, S.; Quiles Latorre, F.; Ortiz, M.; Sánchez Ruiz, V.; Gamiz, V.; Luna-Rodríguez, J.J. Honey Bee Colonies Remote Monitoring System. *Sensors* **2017**, *17*, 55, doi:10.3390/s17010055.
23. Rustia, D.J.; Ngo, N.; Lin, T.T. An IoT-based information system for honeybee in and out activity with beehive environmental condition monitoring. In Proceedings of the Conference on Bio-Mechatronics and Agricultural Machinery Engineering, Niigata, Japan, 23–25 May 2016; pp. 1–2.
24. Braga, A.R.; Hassler, E.E.; Gomes, D.G.; Freitas, B.M.; Cazier, J.A. IoT for Development: Building a Classification Algorithm to Help Beekeepers Detect Honeybee Health Problems Early. In Proceedings of the AMCIS 2019, Cancún, Mexico, 15–17 August 2019.

25. Kontogiannis, S. An Internet of Things-Based Low-Power Integrated Beekeeping Safety and Conditions Monitoring System. *Inventions* **2019**, *4*, 52, doi:10.3390/inventions4030052.
26. Debauche, O.; Moulat, M.E.; Mahmoudi, S.; Boukraa, S.; Manneback, P.; Lebeau, F. Web Monitoring of Bee Health for Researchers and Beekeepers Based on the Internet of Things. *Procedia Comput. Sci.* **2018**, *130*, 991–998, doi:10.1016/j.procs.2018.04.103.
27. Edwards-Murphy, F.; Magno, M.; Whelan, P.M.; O'Halloran, J.; Popovici, E.M. b+WSN: Smart beehive with preliminary decision tree analysis for agriculture and honey bee health monitoring. *Comput. Electron. Agric.* **2016**, *124*, 211–219, doi:https://doi.org/10.1016/j.compag.2016.04.008.
28. Meitalovs, J.; Histjajevs, A.; Stalidzans, E. Automatic Microclimate Controlled Beehive Observation System. In Proceedings of the 8th International Scientific Conference 'Engineering for Rural Development', Jelgava, Latvia, 28–29 May 2009, pp. 265–271.
29. Balta, A.; Dogan, S.; Ozmen koca, G.; Akbal, E. Software Modeling of Remote Controlled Beehive Design. In Proceedings of the International Conference on Advances and Innovations in Engineering (ICAIE), Elazig, Turkey, 10–12 May 2017; pp. 1–6.
30. Fitzgerald, D.W.; Murphy, F.E.; Wright, W.M.D.; Whelan, P.M.; Popovici, E.M. Design and development of a smart weighing scale for beehive monitoring. In Proceedings of the 2015 26th Irish Signals and Systems Conference (ISSC), Pittsburgh, PA, USA, 24–25 June 2015; pp. 1–6. doi:10.1109/ISSC.2015.7163763.
31. Zacepins, A.; Pecka, A.; Osadcuks, V.; Kviesis, A.; Engel, S. Solution for automated bee colony weight monitoring. *Agron. Res.* **2017**, *15*, 585–593.
32. Machamer, R.; Altenhofer, J.; Ueding, K.; Czenkusch, L.; Stolz, F.; Harth, M.; Mattern, M.; Latif, A.; Haab, S.; Herrmann, J.; et al. Visual Programming IoT Beehive Monitoring for Decision Aid by Machine Learning based Anomaly Detection. In Proceedings of the 2020 9th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, 8–11 June 2020; pp. 1–5, doi:10.1109/MECO49872.2020.9134323.
33. Szczurek, A.; Maciejewska, M.; Zajiczek, Z.; Wilk, J.; Wilde, J.; Siuda, M. The Effectiveness of Varroa destructor Infestation Classification Using an E-Nose Depending on the Time of Day. *Sensors* **2020**, *20*, 2532, doi:10.3390/s20092532.
34. Szczurek, A.; Maciejewska, M.; Wilk, J.; Wilde, J.; Siuda, M. Detection of Honeybee Disease: Varroasis using a Semiconductor Gas Sensor Array. In Proceedings of the 8th International Conference on Sensor Networks, Prague, Czech Republic, 26–27 January 2019; pp. 58–66, doi:10.5220/0007575600580066.
35. Meikle, W.; Holst, N. Application of continuous monitoring of honeybee colonies. *Apidologie* **2014**, *46*, 10–22, doi:10.1007/s13592-014-0298-x.
36. Mezquida, D.; Martínez, J. Short communication.: Platform for bee-hives monitoring based on sound analysis. A perpetual warehouse for swarming daily activity. *Span. J. Agric. Res.* **2009**, *7*, 824–828, doi:10.5424/sjar/2009074-1109.
37. Bencsik, M.; Bencsik, J.; Baxter, M.; Lucian, A.; Romieu, J.; Millet, M. Identification of the honey bee swarming process by analysing the time course of hive vibrations. *Comput. Electron. Agric.* **2011**, *76*, 44–50, doi:doi.org/10.1016/j.compag.2011.01.004.
38. Zgank, A. Bee Swarm Activity Acoustic Classification for an IoT-Based Farm Service. *Sensors* **2020**, *20*, 21, doi:10.3390/s20010021.
39. Qandour, A.; Ahmad, I.; Habibi, D.; Leppard, M. Remote Beehive Monitoring using Acoustic Signals. *Acoust. Aust./Aust. Acoust. Soc.* **2014**, *42*, 204–209.
40. Van Goethem, S.; Verwulgen, S.; Goethijn, F.; Steckel, J. An IoT solution for measuring bee pollination efficacy. In Proceedings of the 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 15–18 April 2019; pp. 837–841, doi:10.1109/WF-IoT.2019.8767298.
41. Campbell, J.; Mummert, L.; Sukthankar, R. Video Monitoring of Honey Bee Colonies at the Hive Entrance. In Proceedings of the Workshop Visual Observation and Analysis of Animal and Insect Behavior, Tampa, FL, USA, 8–11 December 2008; Volume 8, pp. 1–4.
42. Chen, C.; Yang, E.C.; Jiang, J.A.; Lin, T.T. An imaging system for monitoring the in-and-out activity of honey bees. *Comput. Electron. Agric.* **2012**, *89*, 100–109, doi:10.1016/j.compag.2012.08.006.
43. Babic, Z.; Pilipovic, R.; Risojevic, V.; Mirjanic, G. Pollen Bearing Honey Bee Detection in Hive Entrance Video Recorded by Remote Embedded System for Pollination Monitoring. *ISPRS Ann. Photogramm. Remote. Sens. Spat. Inf. Sci.* **2016**, *III7*, 51–57, doi:10.5194/isprs-annals-III-7-51-2016.
44. Bjerger, K.; Frigaard, C.E.; Mikkelsen, P.H.; Nielsen, T.H.; Misbi, M.; Kryger, P. A computer vision system to monitor the infestation level of Varroa destructor in a honeybee colony. *Comput. Electron. Agric.* **2019**, *164*, 104898, doi:https://doi.org/10.1016/j.compag.2019.104898.
45. Marstaller, J.; Tausch, F.; Stock, S. DeepBees—Building and Scaling Convolutional Neuronal Nets For Fast and Large-Scale Visual Monitoring of Bee Hives. In Proceedings of the International Conference on Computer Vision, Thessaloniki, Greece, 23–25 September 2019; pp. 1–8, doi:10.1109/ICCVW.2019.00036.
46. König, A. IndusBee 4.0—Integrated Intelligent Sensory Systems for Advanced Bee Hive Instrumentation and Hive Keepers' Assistance Systems. *Sens. Transducers J.* **2019**, *237*, 109–121.
47. Folino, G.; Forestiero, A.; Spezzano, G. A Jxta Based Asynchronous Peer-to-Peer Implementation of Genetic Programming. *J. Softw.* **2006**, *1*, 12–23, doi:10.4304/jsw.1.2.12-23.
48. Google Coral Documentation. Available online: <https://coral.ai/docs> (accessed on 30 September 2021).
49. Zoph, B.; Le, Q.V. Neural Architecture Search with Reinforcement Learning. In Proceedings of the ICLR 2017, Toulon, France, 24–26 April 2017.

-
50. Rosa, J.H.; Barbosa, J.L.V.; Kich, M.; Brito, L. A Multi-Temporal Context-aware System for Competences Management. *Int. J. Artif. Intell. Educ.* **2015**, *25*, 455–492, doi:10.1007/s40593-015-0047-y.
 51. Rosa, J.; Barbosa, J.; Ribeiro, G. ORACON: An adaptive model for context prediction. *Expert Syst. Appl.* **2015**, *45*, 1–45, doi:10.1016/j.eswa.2015.09.016.
 52. Filippetto, A.S.; Lima, R.; Barbosa, J.L.V. A risk prediction model for software project management based on similarity analysis of context histories. *Inf. Softw. Technol.* **2021**, *131*, 106497, doi:10.1016/j.infsof.2020.106497.
 53. Dupont, D.; Barbosa, J.; Alves, B. CHSPAM: A multi-domain model for sequential pattern discovery and monitoring in contexts histories. *Pattern Anal. Appl.* **2020**, *23*, 1–10, doi:10.1007/s10044-019-00829-9.