


Review

IoT and Machine Learning Techniques for Precision Beekeeping: A Review

Agatha Turyagyenda ^{1,*} , Andrew Katumba ¹, Roseline Akol ¹, Mary Nsabagwa ²
and Mbazingwa Elirehema Mkiramweni ³

¹ Department of Electrical and Computer Engineering, Makerere University, Kampala P.O. Box 7062, Uganda; andrew.katumba@mak.ac.ug (A.K.); roseline.nyongarwiziakol@mak.ac.ug (R.A.)

² Department of Networks, Makerere University, Kampala P.O. Box 7062, Uganda; mnsabagwa@cit.ac.ug

³ Department of Electronics and Telecommunication Engineering, Dar es Salaam Institute of Technology, Dar es Salaam P.O. Box 2958, Tanzania; mbazingwa.mkiramweni@dit.ac.tz

* Correspondence: agatha.turyagyenda@gmail.com

Abstract: Integrating Internet of Things (IoT) devices and machine learning (ML) techniques holds immense potential for transforming beekeeping practices. This review paper offers a critical analysis of state-of-the-art IoT-enabled precision beekeeping systems. It examines the diverse sensor technologies deployed for honeybee data acquisition, delving into their strengths and limitations, particularly regarding accuracy, reliability, energy sustainability, transmission range, feasibility, and scalability. Furthermore, this paper dissects prevalent ML models used for bee behaviour analysis, disease detection, and colony monitoring tasks. This paper evaluates their methodologies, performance metrics, and the challenges involved in selecting appropriate machine learning algorithms. It also examines the influence of sensing devices, computational complexity, dataset limitations, validation procedures, evaluation metrics, and the effects of pre-processing techniques on these models' outcomes. Building upon this analysis, this paper identifies key research gaps and proposes promising avenues for future investigation. The focus is on the synergistic use of IoT and ML to address colony health management challenges and the overall sustainability of the beekeeping industry.

Keywords: beekeeping; honeybees; internet of things; machine learning; precision apiculture



Academic Editor: Walter Richardson

Received: 20 November 2024

Revised: 16 December 2024

Accepted: 26 December 2024

Published: 4 February 2025

Citation: Turyagyenda, A.; Katumba, A.; Akol, R.; Nsabagwa, M.; Mkiramweni, M.E. IoT and Machine Learning Techniques for Precision Beekeeping: A Review. *AI* **2025**, *6*, 26. <https://doi.org/10.3390/ai6020026>

Copyright: © 2025 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

The alarming decline in honeybee populations and diversity directly threatens global food security and the stability of ecosystems. These essential pollinators, responsible for 75% of crops consumed by humans, are dwindling due to various factors [1]. Knowledge of the diversity of the honeybees is vital for conservation efforts to determine how threats like climate change would affect them [2]. One of the tools that can address the causes of this decline is precision beekeeping/apiculture (PB/PA), defined as “an apiary management strategy based on the monitoring of individual bee colonies to minimise resource consumption and maximise the productivity of bees” [3]. This study focuses on the specific technologies of the Internet of Things (IoT) and machine learning underlying these innovations. Consequently, it does not explore commercially available solutions utilising these technologies, which are often proprietary. These techniques collect data from select colony parameters and synthesise it to provide insights into honeybee behaviours, indicating various occurrences within the colony. Several studies have established important colony parameters that provide useful insights into the honeybees' behaviours and health. These include the following: Weight,

which provides important information on the weight of the honey reserves; swarm departures [4]; beehive population growth and food consumption [5]; hive abandonment [6]; and the effect of pesticides on bee colonies, nectar, and pollen variation [7]. Temperature and humidity provide important information on metabolic processes [4], health and brood, mortality, and honey production [8]. Gases, especially carbon dioxide (CO₂) and oxygen, give an indication of the metabolic processes, activity level, and health status of the colony and mortality [4]. Vibration and acoustic signals provide information on the ability to differentiate between infected and healthy bees, the presence of pests, swarming [9], and the queen's behaviour during swarming [4]. Images/video provides information on food demand, food availability, colony age structure [4], and the impact of pests and pesticides [10]. A vital prerequisite in collecting data from honeybees is to carry it out non-invasively so as not to disrupt their natural rhythm and to obtain reliable data.

With their miniature nature and ability to collect large quantities of data with minimal disturbance to the surroundings, coupled with the ability of machine learning techniques to process this data and provide insights, IoT devices have been made suitable for monitoring honeybees.

Figure 1 shows the general architecture of an IoT-based honeybee monitoring system. An IoT device typically consists of (1) a microprocessor/microcontroller, which is the main processing part of the system. It collects the data from peripheral devices connected to it at set intervals. When the data have been collected, they can be sent directly to a remote server for further processing, or the processing is performed on the microcontroller/microprocessor, and information is sent to a remote server. (2) Sensors are components that acquire the physical elements from the variables being acquired. (3) Communication module(s) connect to the internet, and (4) energy sources power the system.

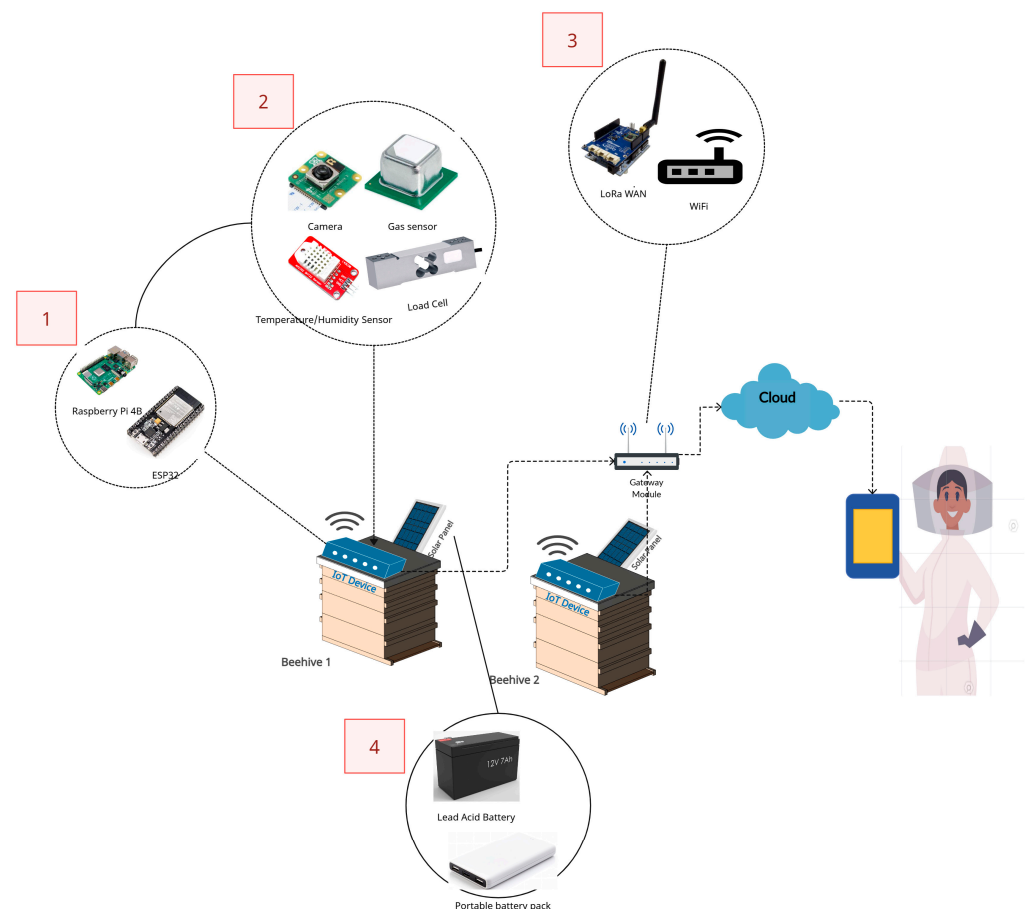


Figure 1. Typical IoT-enabled system for honeybee monitoring.

Machine learning (ML) is the study of algorithms and statistical models that enable computers to perform tasks without explicit instructions based on pattern recognition and inference [11]. It utilises statistical analysis, clustering algorithms, data transformations, and deep learning techniques involving Artificial Neural Networks.

A dataset is required to train the model, using either supervised learning or unsupervised learning, as shown in Figure 2. In supervised learning, the model is trained on labelled data, where all the output is known prior to training. In unsupervised learning, the model is trained on unlabelled data with unknown output [12]. These can be categorised into classical/traditional and deep learning techniques:

- i. The classical techniques are based on statistical methods and concepts to achieve their goals, for example, Support Vector Machine (SVM) [13], K-Nearest Neighbor (K-NN) [14], Random Forest (RF) [15], and Linear Discriminant Analysis (LDA) [14].
- ii. Deep learning methods are based on Artificial Neural Networks (ANNs) that are composed of multiple layers of interconnected nodes (artificial neurons or units). Examples of this type include Recurrent Neural Networks (RNN) and Convolutional Neural Networks, composed of multiple layers of interconnected nodes [16].

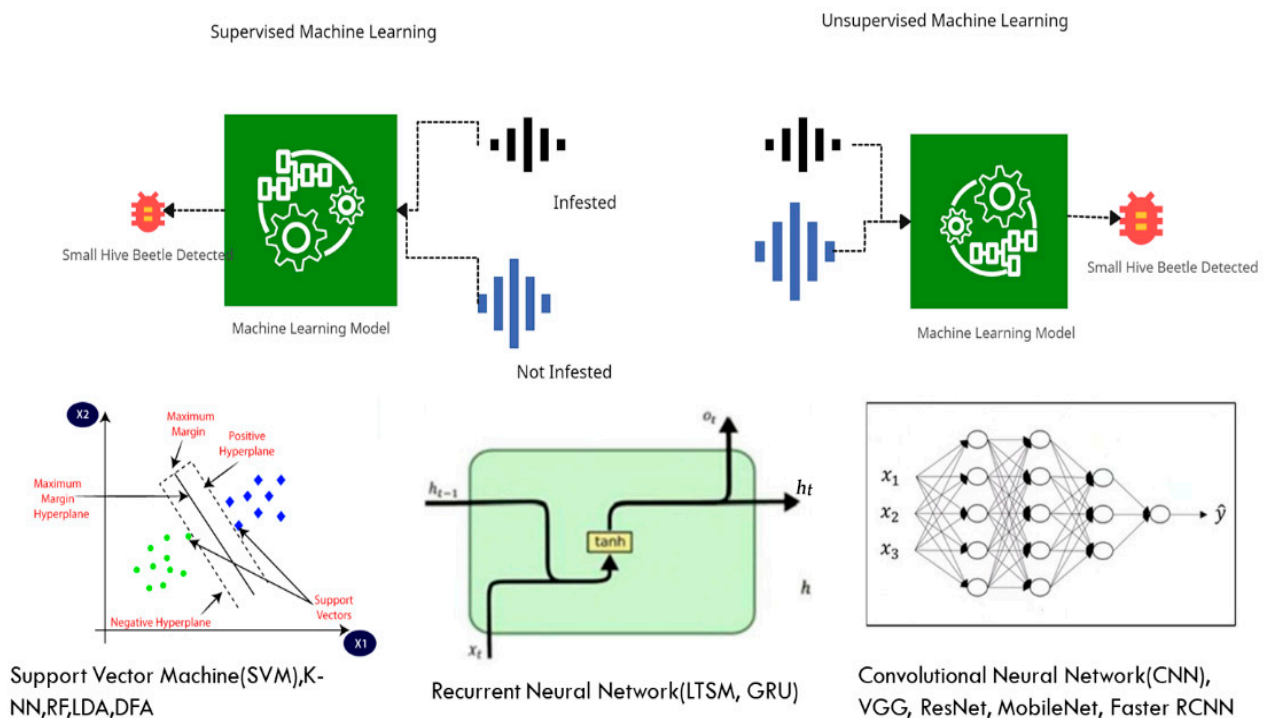


Figure 2. Machine learning techniques.

Figure 3 depicts the machine learning workflow for beekeeping applications, consisting of the following five key phases: The first phase is the data acquisition, where diverse datasets are collected from bee colonies. This includes gathering indicators such as weight measurements, temperature readings, gas concentrations, images, sound, and video recordings from hives. The second phase is data pre-processing, which involves cleaning and preparing the collected data to ensure their suitability for model training. This includes removing noise from audio recordings, annotating images and videos with relevant labels, and formatting all types of data consistently. This step is important for improving the quality of the datasets and enabling accurate model training. The third phase is model training, where the pre-processed data are used to train machine learning models that have been selected prior depending on the desired results. Data are input into the models, and parameters are iteratively adjusted to optimise performance. This enables

the models to identify patterns and make accurate predictions related to bee behaviour and hive conditions, such as detecting signs of disease or predicting a swarming event. The fourth phase is model testing, where the models undergo testing to evaluate their performance. This involves using test datasets to assess the models' accuracy and their ability to generalise to new, unseen data. The final phase is deployment and evaluation, which involves deploying the trained models in actual beekeeping environments. This includes integrating the models into hive monitoring systems and validating their predictions through laboratory analyses and field observations.

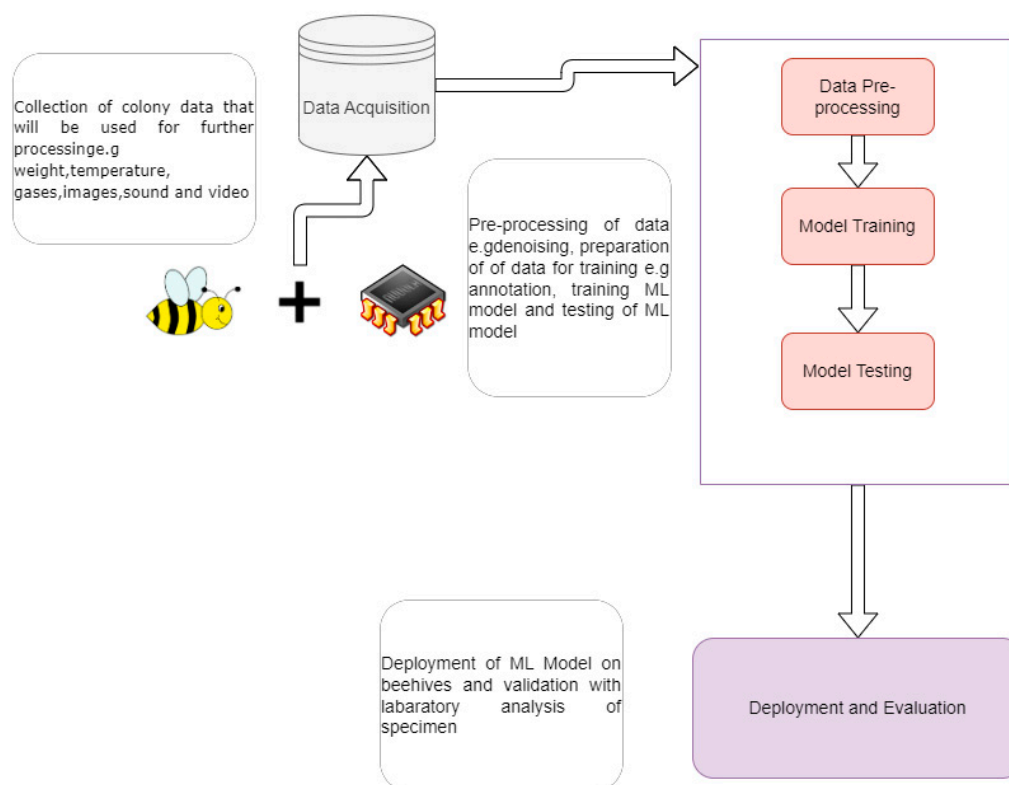


Figure 3. Typical machine learning workflow.

There are several review papers concerning IoT and machine learning techniques, namely the following: machine learning applications to adulterated honey and honeybee health status [17]; survey on automated or smart systems' design, development, deployment, feasibility, and associated costs for precision apiculture [3]; overview of the state-of-the-art computer vision and machine learning in bee monitoring [18]; comparison of machine learning classification algorithms for their suitability in low powered solutions [19] A review on recent developments of precision beekeeping [20] is presented but does not examine in detail the IoT and machine learning technologies driving these developments unlike this study.

This review paper distinguishes itself from previous works by comprehensively analysing the specific technological components within IoT and techniques in machine learning systems that enable precision beekeeping. It critically examines the limitations of existing technologies and proposes novel research avenues focused on developing intelligent edge devices tailored for precision beekeeping applications. (An intelligent edge device is a system designed to perform data collection, pre-processing, and decision-making directly at the edge of a network near the source of data generation. These devices utilise embedded computing resources, such as microprocessors or AI accelerators, to process data locally and execute tasks without extensive reliance on cloud computing [21,22]).

It aims to address the following questions:

- (1) What parameters are being acquired by IoT-based systems for precision beekeeping, and what insight is derived from them?
- (2) What are the strengths and limitations of the devices and equipment used for data acquisition?
- (3) Which pre-processing techniques are being applied to the collected data, and how have these affected the outcome?
- (4) Which machine learning techniques have been developed, what insights have been afforded by their application to precision apiculture, and what are their strengths and limitations?

We conducted studies from 2016 onwards to account for advancements in IoT and machine learning technologies and to focus on state-of-the-art devices. Additionally, the studies must have employed an IoT-based system or machine learning algorithms for processing honeybee monitoring data.

This paper is organised as follows: Section 2 reviews IoT-based technologies used in precision beekeeping, focusing on their components, capabilities, limitations, comparative analysis, and feasibility for beekeeping. Section 3 discusses machine learning techniques for honeybee monitoring, including data pre-processing, model development, evaluation, performance analysis, and feasibility in honeybee keeping. Section 4 identifies key knowledge gaps and proposes future research directions to enhance precision beekeeping technologies. Finally, Section 5 concludes the review by summarizing the findings and emphasizing the role of IoT and ML in addressing beekeeping challenges.

2. IoT-Based Technologies for Precision Beekeeping

Table 1 summarises these state-of-the-art IoT-based systems, highlighting their intended purpose and the most vital components of the system, which are the processor type, communication module type, sensors, and purpose of the system, and also highlights the limitations of the systems.

The systems developed to monitor apiaries generally consist of three types of data acquisition systems: (1) Specialised instrumentation that detects one parameter; they are accurate and precise but often need highly skilled knowledge for their operation and are processed using proprietary software. For example, ref. [23] deployed piezoelectric accelerometers placed in the honeybee hives to predict if the bee swarming was imminent, placing them in the centre of the hive frames. The signals acquired were digitized by a conditioner and sent to a software application that presented power spectra averaged over three minutes. (2) Non-IoT-based novel acquisition systems that are bulky and would be cumbersome for the beekeeper to use; for example, [24] developed a computer vision system to count the number of honeybees and Varroa mites and then determine the level of infestation of Varroa mites in a beehive. A Video Monitoring Unit with multispectral LED lights next to a camera and perpendicular to a mirror illuminated a glass passage that the bees used to enter the beehive. The unit was connected to a computer that collected the data. Ref. [25] developed a similar video monitoring system, and the data were acquired by a Raspberry Pi and stored on an external storage card. Ref. [26] developed an electronic nose to detect varroosis composed of an array of gas sensor elements that detect volatile organic compounds. The gas was collected for one second and stored on an external storage card. Finally, (3) IoT-based devices that process the data collected and processed locally or transmitted and processed remotely is the focus of discussion for this review.

Table 1. IoT-based systems for precision beekeeping.

Bee Monitoring Event (Author)	Processor/Microcontroller	Sensors	Communication Module	Limitations
Bee entry and exit vs. environmental parameters [27]	Arduino Mega 256 and ESP32	AM23032 (DHT22) for temperature and humidity; BME280 for temperature, humidity, and air pressure; MH-RD for raindrops; MQ135, MICS6814, and MICS5524 for air quality, smoke, carbon monoxide, noise, sound shocks; SW-420 for vibrations; VEMML6750 for UV index; S11145 IR for InfraRed; BH1750 for daylight intensity and photo-resistors.	Cellular Modem	Numerous sensors are included in the design without justification, and this affects the power consumption
Frequency of entrance and exit of individual bees [28]	NVIDIA Jetson TX2	Web Camera (Logitech C920), temperature sensor (SHT20), humidity sensor (SHT20), rainfall sensor (WH40), and light intensity sensor (BH1750).	4G LTE Router	Bulky and invasive solution that is running on mains
Detection of Swarming [29]	Raspberry Pi-3/Raspberry Pi zero/NVIDIA Jetson Nano	5 MP Camera	WiFi/LoRAWAN	Images/video notifications are not useful to beekeepers for swarm detection.
Detection of Varroa mites inside the hive [30]	Raspberry Pi Zero W version 2, microcontroller	5 MP Camera	WiFi/LTE 3G, Bluetooth	No night vision capability, therefore impractical. No considerations for energy efficiency
Identification of honeybees and Varroa mites [31]	Raspberry Pi 4B, Google Edge TPU co-processor	5 MP Camera	Cellular Modem	Off-the-shelf proprietary components without holistic integration were used and redesigned to obtain integrated solutions.
Monitoring several colony activities [8]	Raspberry Pi 3B	Load cells for weight acquisition, AM23032 (DHT22) for temperature and humidity; T6615 for carbon dioxide, microphone	Zigbee, Local computer	Use of mains supply that is inappropriate.
Differentiating pollen from non-pollen-bearing bees [32]	Jetson TX2	camera; SHT20 for temperature, humidity; WH40 for rain level; GY-30 BH1750 for light intensity	WiFi	The system is bulky and invasive.

2.1. Trends in IoT System Components

2.1.1. Trends in Processor Selection

Table 2 provides an overview of the processing speeds and storage capacities of the various processors and microcontrollers evaluated in this study. Among the processors, different versions of the Raspberry Pi (Zero, 2W, 3B, 4B) were the most commonly used [8,29–31]. The Raspberry Pi series is favoured for its balance between cost, performance, and ease of use. In contrast, high-performance modules such as the NVIDIA Jetson TX2 [28] and Jetson Nano [29] and the Google Edge Tensor Processing Unit (TPU) co-processor (an application-specific integrated circuit designed specifically for machine

learning acceleration) were selected for tasks requiring more intensive computational power. Due to its dedicated Graphical Processing Unit (GPU) functionality, the Jetson Nano outperformed the Raspberry Pi 3B in high-precision tasks. This capability enables the Jetson Nano to handle parallel processing efficiently, which is crucial for applications involving machine learning and computer vision [29]. However, this improved performance comes at the cost of increased power consumption, an important consideration in apiaries, especially in remote areas. The Arduino Mega 2560 and ESP32 were the primary choices for microcontrollers [27].

Table 2. Development platforms for IoT systems.

Platform	Processor Chipset	RAM
Raspberry Pi 4B	Quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.8 GHz	8 GB
Raspberry Pi 3B	Quad-core Cortex-A53 64-bit SoC @ 1.2 GHz	1 GB
Raspberry Pi 2W	Quad-core 64-bit Cortex-A53 SoC @ 1 GHz	512 MB
Raspberry Pi Zero	Single-core ARM1176JZF, @ 1 GHz	512 MB
NVIDIA Jetson TX2	Dual-Core 64-Bit Quad-Core ARM Cortex®-A57 MPCore with GPU	8 GB
NVIDIA Jetson Nano	Quad-core ARM Cortex-A57 @ 1.5 GHz	4 GB
Arduino Mega 256	Clock speed @ 16 MHz	256 KB
ESP32	Dual-core, Clock speed @ 160 MHz	520 KB

Microcontrollers typically consume significantly less power than processors, making them ideal for applications that do not require the acquisition or processing of videos or images. The application's requirements largely determine the choice between using a processor or a microcontroller. Processors, while more power-intensive, are more suitable for applications that involve complex tasks, such as video or image processing, where their superior computational capabilities are required.

The selection of processing units for IoT-based systems is a critical decision that balances the trade-offs between computational power and energy efficiency. The Raspberry Pi series and Jetson modules offer advantages catering to different application needs. At the same time, microcontrollers like the Arduino Mega 2560 and ESP32 provide energy-efficient solutions for less demanding tasks.

2.1.2. Trends in Sensor Selection

The most commonly used sensing devices were temperature, humidity, acoustics, vibration, camera, gas, and weight sensors. Temperature and humidity sensors like the AM2302(DHT22) and BME280 [27] are crucial for monitoring environmental conditions inside and outside the beehive, aiding in colony health and brood development. However, their accuracy can be affected by sensor placement within the hive and the materials of the hive structure [33]. For acoustic sensing, MEMS microphones are commonly used to detect bee activity and monitor hive health, providing insights into behaviours like swarming and stress. However, they are sensitive to environmental noise, affecting data accuracy. Robust pre-processing is needed to filter out unwanted noise and enhance reliability [19]. Vibration sensors and piezoelectric or MEMS-based accelerometers [34] are used to monitor hive vibrations caused by bee movement or stress. These sensors can detect subtle vibrations related to colony disturbances but may also pick up non-bee-related vibrations, requiring filtering to retrieve relevant signals. Cameras are widely used to detect pollen-bearing bees or Varroa mites, mostly of 5MP resolution images. The image resolution significantly affects both the computation resources required to process them and the accuracy of the processing techniques [29]. Their high computational requirements,

dependence on lighting, and significant energy consumption limit scalability, especially in large or remote deployments [30]. Gas sensors, such as those detecting CO₂ [35] and VOCs [36], are useful for monitoring hive metabolic processes and identifying potential issues like Varroosis. However, these sensors often require frequent recalibration [26]. Weight sensors, such as load cells, detect weight fluctuations, which can provide insights into colony activity, nectar intake, and honey production. These sensors enable beekeepers to track food reserves and detect swarming events. While weight sensors are relatively low power, their performance can be affected by environmental factors such as uneven hive placement, wind, and temperature changes, requiring proper calibration and setup [37].

2.1.3. Trends in Communication Modules

Many of the reviewed studies used cellular modems as communication modules. Cellular modules offer wider coverage and greater reliability but at the cost of increased energy consumption and higher operational costs. Other communication modules include Wi-Fi-enabled devices, which are easy to integrate and provide reliable communication in areas with existing network infrastructure. However, it is often impractical for rural or remote beekeeping sites due to its limited range. The authors of [29] investigated the use of a Long-Range Wide Area Network (LoRaWAN) [38]; it allows for efficient data transmission over long distances with minimal energy usage, which is crucial for battery-operated devices in beehives located in isolated areas. However, they are constrained in the amounts of data they can reliably transmit. Zigbee [39] was used in some studies for the local network of the sensors.

2.1.4. Trends in Energy Source Selection

The most common source of power deployed in the studies was a photovoltaic and rechargeable battery [27,30] because of its sustainability property; however, optimisation is required to maintain continuous operation during low sunlight periods. The mains supply [28] is reliable but impractical for rural beekeeping due to limited grid access and scalability challenges. A portable power pack [31] was found to be suitable for short-term use but requires frequent recharging, making it less viable for beekeepers' long-term management.

The next section examines the bee activities monitored in the studies, including the sensors and devices used to capture the relevant parameters, and elucidates the associated IoT systems' strengths and limitations.

2.2. *Frequency of Entry and Exit of Honeybees*

The studies majorly use photo-resistors and cameras to determine honeybees' entry and exit frequency. Ref. [27] developed a system for counting bees that operates on the principle of two photo-reflective resistors per gate. The system counts a bee when both resistors activate simultaneously, signifying a single pass. The direction of movement into or out of the hive is influenced by the sequence in which the activation occurs, and other parameters to acquire the atmospheric elements were also acquired. The system includes a user interface with predictive and analytical features. Despite the study's claim of low power consumption and its use of solar and rechargeable batteries, it did not present any power consumption investigations. Numerous sensors were deployed in this device without justifying their selection concerning monitoring bee activity. This is especially important for this solution because it has a direct implication on power consumption; Ref. [28] presents a study that uses a large black observation box that contains a web camera, an LED panel, and a transparent passageway to restrict the bees to the camera's region of visibility and prevent disruption by other bees go through. This construction

seems to disrupt the bees' natural rhythms, it would face scalability challenges, and its power supply is from the mains, which is impractical for long-term monitoring.

2.3. Prediction of an Eminent Swarm or Detection of an Ongoing Swarm

Several sensing devices have been used, for example, microphones [40,41], accelerometers [23], and cameras. The authors of [29] developed a system that includes a camera module connected to a microprocessor and sent to a remote database. However, using the camera for swarming detection is unsuitable as it would only be detected as it is happening, which is not very useful to the farmers, who prefer to know prior so they can intervene.

2.4. Detection of Pests, Predominately the Varroa Mite

Sensing devices deployed are cameras and gas sensors; Ref. [30] developed an imaging system for the early detection of Varroa consisting of two 5MP cameras placed at different angles inside the bee frames. Although no power consumption studies have been reported, a solar panel and rechargeable battery power the system. The sole reliance on the camera means the system becomes ineffective in case of obstruction or change in illumination. Ref. [31] developed an image-based detection system to detect Varroa mites using a video captured by the camera, although the study does not mention its position inside or outside the beehive. The pre-processing was performed on the edge with the aid of a Tensor Processing Unit. However, this system is not integrated; it is a combination of off-the-shelf plug-and-play devices. Although the study mentions a power bank as its energy source, it does not present a power consumption analysis.

2.5. Detection of Pollen and Non-Carrying Bees

The detection of pollen predominately used cameras to capture videos at the beehive entrance. In their study to differentiate pollen and non-pollen-carrying bees, Ref. [32] developed an embedded imaging system that consisted of an off-the-shelf camera at a restricted entryway of the bee beehive that captured a video stream that was then processed with a Jetson TX2 processor. The system was then enclosed in a black observation box and fitted with red LED lights for illumination. An additional environment sensing module that consisted of temperature and humidity sensors placed inside and outside the beehive, a light sensor, and a rain level sensor connected to a Raspberry Pi 3 processor, which transmitted the data via WiFi to a remote database where it was further processed and displayed on a website. Additional wind information was obtained from external sources. The setup is huge and disrupts the bees' natural way of life.

2.6. General Colony Activity Monitoring

In [42], a system was developed that monitors weight using load cells placed in a customised frame below the beehive, temperature, humidity, carbon dioxide, and bee sounds using Micro-Electro-Mechanical System (MEMS) microphones. The mains supply was the power source, which is unsuitable for remote monitoring or scalability.

2.7. Discussion

Table 3 presents a summary of the reviewed performance of the systems for IoT-based precision beekeeping using the following metrics: accuracy, reliability, energy sustainability, transmission range, feasibility, and scalability.

Table 3. System performance evaluation.

Bee Monitoring Event	Energy Sustainability	Feasibility	Scalability
Bee entry and exit and other vs. environmental parameters [27]	Not available.	Yes. Sensors placed so as not to disrupt bee rhythms and protected from bee wax and propolis.	None.
Frequency of entrance and exit of individual bees [28]	Mains. Not available.	Bulky and invasive.	Yes. Star topology.
Detection of swarming [29]	Solar PV but without studies conducted.	Not feasible. Camera module placed inside the hive with no night vision capability. Not suitable.	Not shown.
Detection of Varroa mites inside the hive [30]	Solar PV but no power sustainability studies conducted.	Not feasible. Camera placed inside the hive with no night vision capabilities.	Not shown.
Identification of honeybees and Varroa mites [31]	Power bank. No power consumption analysis.	Not a fully integrated system. Plug-and-play components.	Not shown.
Monitoring several colony activities [8]	Mains connection. No studies conducted.	Yes. Sensors are non-invasive.	Yes. Six systems connected in a star topology.
Differentiating pollen from non-pollen-bearing bees [32]	Mains. No power consumption studies conducted.	Bulky and disruptive.	Yes. Star topology.

2.7.1. Accuracy

Accuracy is important because incorrect data could lead to a misinterpretation of the beehive's status or activities. Considering all the studies reviewed, none of the studies show that any accuracy tests or calibration were conducted on selected sensors, a crucial step in ensuring confidence in the obtained results. It is important to note that none of the studies reviewed carry out accuracy tests, for example, by benchmarking with existing standardised systems/sensors. Moreover, the systems reviewed here utilise various sensors, each with varying degrees of accuracy depending on their design and application. For instance, [27,28] relied on photo-resistors and cameras to monitor the frequency of bee entry and exit. However, environmental factors such as lighting conditions, sensor obstruction, or changes in bee behaviour due to restrictive designs (e.g., black observation boxes) may compromise the accuracy of these systems. The image-based Varroa detection systems [30,31] are heavily reliant on cameras, which are prone to inaccuracies if the visual field is obstructed or illumination varies. Although the limitations of individual sensors may still affect them, systems that integrate multiple types of sensors, like [8] with load cells and MEMS microphones, are likely to be more accurate as they can cross-verify data from different sources.

2.7.2. Reliability

Reliability in sensor systems is determined by their ability to consistently provide data over time without significant downtime or data loss. Notably, no reliability analysis was also provided by the studies reviewed, despite some of them operating in the field for extended periods [8]. Systems powered by mains electricity, such as those developed by [8,28], might offer reliable operation in terms of data transmission and sensor uptime but are impractical for remote or off-grid applications. Solar-powered systems [27,30] present a more sustainable option for remote beekeeping but can be less reliable in areas with limited sunlight or during periods of poor weather. Additionally, reliance on wireless transmission introduces another point of potential failure, particularly in areas with poor network coverage or interference.

2.7.3. Energy Sustainability

Energy sustainability is crucial, especially for systems deployed in remote locations where frequent battery changes or maintenance are impractical. While [27,30] report low power consumption and solar-powered designs, the absence of detailed power consumption analysis in their studies raises questions about the long-term sustainability of these solutions, but also the practicality of solar-based systems is a concern as many apiaries are often found in areas with vegetation cover. On the other hand, systems like those developed by [28] that depend on mains power are inherently unsustainable for remote or large-scale deployment. Off-the-shelf components, as seen in the Varroa detection system by [31], raise concerns about energy efficiency, as these components may not be optimised for low-power operation. The studies often used more sensors than the actual application required without justification [28]; this is pertinent because of the low power consumption that practical systems for apiculture should have. Additionally, continuous data monitoring and transmission to cloud platforms or local hubs impose high energy demands, reducing device lifespan and increasing maintenance requirements. While real-time edge processing minimises cloud dependency, it heightens local power consumption due to computational overhead.

In Internet of Things (IoT) systems, power consumption is predominantly determined by three main components: sensors, communication modules, and processing units. (1) Sensors used for monitoring colony parameters, such as temperature and humidity, consume energy during data acquisition, with power usage influenced by sampling rate and accuracy. High-resolution sensors require more energy due to increased data processing demands, whereas lower sampling rates can significantly reduce energy consumption. (2) Communication modules, which transmit data to servers, cloud platforms, or nearby gateways, account for a significant portion of energy usage. Factors such as the frequency of data transmission and transmission range affect power demands, with long-range protocols like GSM, LTE, or satellite communication consuming more energy than short-range alternatives like Zigbee, LoRa, or Bluetooth Low Energy (BLE). (3) Processing units, such as microcontrollers or on-device processors, also contribute to power consumption. Systems that process data locally, for instance, through lightweight machine learning models or rule-based algorithms, consume more energy compared to devices that transmit raw data for remote analysis.

2.7.4. Transmission Range

The ability to transmit data over long distances is essential for beekeepers who manage hives in remote areas. Systems that rely on WiFi [32] may face limitations in transmission range, particularly in rural or forested areas where WiFi signals are weak or nonexistent. Conversely, systems using cellular networks or long-range radio transmission could offer greater coverage but at the expense of higher power consumption and potentially higher operational costs. As a result, the choice of transmission technology has significant implications for the system's practicality and cost-effectiveness.

2.7.5. Feasibility

Feasibility refers to the practicality of implementing these systems in real-world beekeeping operations. Systems that require complex setups or interfere with the natural behaviour of bees, such as those with restricted entryways or black observation boxes [28,32], maybe less feasible for widespread adoption. Similarly, systems that depend on a power source, such as mains electricity, may not be viable in remote or off-grid environments. In contrast, systems that integrate seamlessly into the hive without significant disruption

(e.g., load cells placed under hives, as in [8]) are more likely to be adopted by beekeepers, provided they are cost-effective and easy to maintain.

2.7.6. Scalability

Scalability is the capability to extend the monitoring system to cover multiple hives or larger apiaries. Systems that rely on expensive or bulky components, like Jetson TX2 processors [32], may face scalability challenges due to their high cost and large size. Additionally, systems with significant power resources or complex installations are less likely to be scalable. Conversely, systems that use cost-effective, low-power sensors and can be easily deployable across multiple hives (e.g., those using MEMS sensors or low-cost cameras) are more likely to scale effectively. However, scalability must also consider the ease of data aggregation and analysis, as managing data from numerous hives can become challenging without robust data management systems.

2.8. Comparative Evaluation of Machine Learning Inference Machines on Edge-Class Devices

The integration of edge-class devices into IoT-based systems facilitates real-time processing in resource-constrained environments, reducing reliance on cloud infrastructure. This section evaluates the performance of edge-class devices and ML models in precision beekeeping.

2.8.1. Lightweight ML Models on Raspberry Pi Devices

Raspberry Pi devices offer a balance between affordability and computational capability, making them suitable for running lightweight ML models, such as Support Vector Machines (SVM) or simplified Convolutional Neural Networks (CNNs) like SSD-MobileNet v1 [29]. While their quad-core processors enable moderate inference capabilities, their reliance on CPUs (Central Processing Units) rather than GPUs can hinder efficiency for deep learning tasks. Furthermore, their higher power consumption than microcontrollers limits their feasibility in off-grid apiaries [31].

2.8.2. Deep Learning Acceleration with NVIDIA Jetson Nano

The NVIDIA Jetson series is equipped with integrated GPUs to handle high computational loads efficiently. These devices are particularly effective for image-based applications, such as Varroa mite detection, where deep learning models like Faster R-CNNs are required [31]. However, their enhanced performance comes at increased energy consumption and higher operational costs, which may restrict their scalability for large apiaries and feasibility in off-grid apiaries [29].

2.8.3. TPU-Accelerated Inference with Google Coral

The Google Edge TPU is an application-specific integrated circuit (ASIC) designed to accelerate ML inference. It facilitated fast and efficient inference of ML models for both bee identification and varroosis detection [31]. With its ability to process up to 4 trillion operations per second at just 2 watts, the Coral TPU proved ideal for real-time edge processing. This approach reduced data transmission to the cloud, limiting bandwidth usage and ensuring real-time alerts for beekeepers. However, the proprietary nature of the Edge TPU introduces potential challenges in system integration.

2.8.4. Microcontrollers for Energy-Constrained Tasks

Microcontrollers provide an energy-efficient alternative for tasks requiring low computational power. Devices such as the ESP32 and Arduino Mega 2560 are well-suited for deploying simple ML models (e.g., K-Nearest Neighbors, SVM) or performing non-ML data aggregation tasks. However, their limited memory and processing capabilities make

them unsuitable for tasks involving high-resolution image analysis or complex pattern recognition. Notably, none of the reviewed studies had utilised them for ML-related tasks, but the development of versions of ML techniques like Tiny ML [43] offers opportunities for further exploration with these devices.

Table 4 summarises the comparative evaluation of these devices based on key performance metrics, including latency, power consumption, accuracy, cost, and scalability. Raspberry Pi devices demonstrate moderate latency and power consumption but are inefficient with deep learning workloads. In contrast, NVIDIA Jetson Nano provides superior accuracy and lower latency for computationally intensive tasks, although it has higher energy requirements. Microcontrollers like ESP32 are cost-effective and highly scalable but are restricted to lightweight inference tasks. The Google Edge TPU offers an optimal balance between accuracy and energy efficiency, particularly for deep learning-based applications, yet its scalability depends on cost and compatibility considerations.

Table 4. Comparative evaluation of Edge class devices.

Device	Latency	Power Consumption	Accuracy	Cost	Scalability
Raspberry Pi 4B	Moderate	Moderate	High	Low	Moderate
NVIDIA Jetson Nano	Low	High	Very High	High	Low
ESP32	Low	Very Low	Moderate	Very Low	High
Google Edge TPU	Very Low	Low	High	Moderate	High

3. Machine Learning-Based Techniques for Precision Beekeeping

Table 5 summarises the honeybee activity detected or predicted, the associated machine learning models, pre-processing techniques, and the studies' limitations in precision beekeeping. The majority of the studies utilise deep learning-based models, especially the Convolutional Neural Networks and their variations. The models achieved good accuracy between 73% [44] and 94% [41], clearly showing their potential to provide good insights into bee activities and behaviour. Pre-processing/feature extraction techniques are predominately for the acoustic, vibrational, and image data, and studies have shown that they significantly affect the model's performance [45,46].

Convolutional Neural Networks (CNNs) were effective for image-based tasks, achieving accuracies as high as 94% [31] in detecting Varroa mites or 93% [44] in identifying pollen-bearing bees due to their ability to recognize complex patterns in visual data. However, their high computational requirements make them better suited for cloud-based processing rather than deployment on low-power edge devices. Support Vector Machines (SVMs) are ideal for low-power applications, such as gas detection [36] or acoustic signal classification [19], as they are computationally efficient and perform well with smaller datasets. This makes them a practical choice for real-time processing on resource-constrained edge devices. Recurrent Neural Networks (RNNs) are well-suited for time-series data analysis, such as predicting bee activity from environmental parameters like temperature and humidity [27]. While RNNs effectively capture temporal dependencies, their computational intensity often limits their feasibility for deployment on low-power systems.

Evaluating model performance is essential to ensure their effectiveness. A common issue observed is the inconsistent presentation of performance metrics across studies. Providing a complete set of performance metrics allows for a more accurate comparison of models. While most research emphasises basic performance indicators like accuracy, precision, recall, and F1-score, a thorough evaluation should include additional metrics to understand the model's capabilities. The comprehensive evaluation presentation should

include the accuracy, recall, precision, confusion matrix, F1-score, and the Area Under the ROC [19].

The Confusion Matrix is a table used to analyse the performance of a classification model. It summarises the predictions' outcomes by comparing the actual and predicted classifications. The matrix consists of four components: true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). Each element provides insight into the model's accuracy and the types of errors it makes.

Table 5. Machine learning-based models for precision beekeeping.

Bee Event Identified/Classified	Dataset	Pre-Processing Technique/Feature Extraction	Machine Learning Model	Evaluation	Limitations
Bee entrance and exit activity [27]	20 days—time series	None	LSTM, ARIMA, Facebook, Prophet	RMSE—426.9 (entry) RMSE—378.464 (exit)	Limited dataset and lack of diverse environmental conditions undermine the purpose of the study.
Beehive entrance activity [18]	1000 images	Renet-50 CNN	SVM	Accuracy—85%	Inadequate discussion of the accuracy metrics of the model. No field evaluations.
Prediction of an imminent swarming event [23]	30 mil freq. points	PCA	DFA	Accuracy—80%	Other potential effects on vibration were not investigated. Numerous false alarms suggest the low sensitivity of models.
Swarming detection [29]	5000 images	bilateral filtering, cubic interpolation	SSD, Faster-CNN	mAP(SSD)—0.4223 mAP (Faster-RCNN)—0.7308	Validation based on a laboratory simulation does not reflect real field events.
Eminent swarming detection [40]	2400 audio recordings (122 min)	MFCC, LPC	Hidden Markov Model (HMM), Gaussian Markov Model (GMM)	<u>HMM</u> Accuracy—82.7% F1—0.90, Recall—0.92, Precision—0.89 <u>GMM</u> Accuracy—79.78% F1—0.89, Recall—0.88, Precision—0.89	The model is evaluated in a controlled environment. The diversity of the dataset is not fully explored.
Varroa mite detection [26]	30 days	None	SVM, K-NN	<u>SVM</u> TPR—0.93, TNR—0.95 <u>KNN</u> TPR—0.86, TNR—0.95	The level of infestation is unspecified, and the model is unreliable due to the variance of accuracy affected by different groups of bees.
Varroa mite detection [31]	11,000 images	None	CNN with reinforcement learning	Precision—0.7 TPR—0.94, F1 score—0.8	Limited dataset diversity from individual hives.
Varroa mite monitoring [47]	Not available	Histogram, Hough transform	Convolutional Neural Network	Accuracy—93%	Synthetic images were utilised and not tested in real-world environments.
Classification of pollen and non-pollen-bearing bees [32]	3500 images	None	Tiny- Yolo v3	F1-score—0.94 Precision—0.91 Recall—0.99	The control used for this study is not an appropriate choice because it differs from the one being monitored for different reasons, e.g., having a healthier population
Detection of pollen sacs in bees [44]	125 videos	Background subtraction, Hough transform	Faster-RCNN with VGG-16-CNN	Accuracy—93% Recall—56%	System performance evaluated in a controlled environment
Detection of pollen-bearing bees [48]	1000 images	Colour based segmentation, SIFT algorithm,	SVM	(Area under the Curve) AUC—0.91	Limited dataset and diverse bee environments not investigated

Table 5. Cont.

Bee Event Identified/Classified	Dataset	Pre-Processing Technique/Feature Extraction	Machine Learning Model	Evaluation	Limitations
Detection of the queen's presence [19]	360 min of audio	SVD, MFCC	SVM, KNN NN	<u>SVM</u> Accuracy—0.98 F1—0.97, Recall—0.97 Precision—0.98 AUC—0.95 <u>KNN</u> Accuracy—0.98 F1—0.98, Recall—0.98 Precision—0.98 AUC—0.98 <u>NN</u> Accuracy—0.97 F1—0.97, Recall—0.97 Precision—0.97 AUC—0.97	Computation cost evaluation did not factor in pre-processing effects.
Detection of queen presence [49]	720 min of audio	MFCC and mel spectra	SVM, CNN	AUC—0.8	Limited model accuracy metrics were discussed.

Accuracy is a metric used to measure the overall correctness of a classification model. It is the ratio of correctly predicted instances to the total cases.

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})$$

While accuracy is useful, it may be misleading in datasets where the number of instances differs greatly between classes, a common occurrence in beekeeping.

Precision, also known as positive predictive value, measures the accuracy of positive predictions. It is the ratio of true positive predictions to the total positive predictions (true positives and false positives).

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

Precision is important in scenarios where the cost of false positives is high, such as pests or diseases.

Recall, also known as sensitivity or true positive rate, measures the ability of a model to identify all relevant instances. It is defined as the ratio of true positive predictions to the total actual positives (true positives and false negatives)

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

Recall is crucial in situations where missing positive instances is costly.

F1-Score is the harmonic mean of precision and recall, providing a single metric that balances both concerns. It is especially useful when the class distribution is imbalanced.

$$\text{F1-score} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

It provides a more comprehensive measure of a model's performance than either precision or recall alone.

The Area Under the Receiver Operating Characteristic Curve (AUC-ROC) is a performance metric for binary classification models. The ROC curve plots the true positive rate (recall) against the false positive rate (1-Specificity) at various threshold settings. The AUC

indicates the likelihood that the model will assign a higher rank to a randomly selected positive instance compared to a randomly selected negative instance. An AUC of 0.5 indicates a model with no discriminatory ability, while an AUC of 1.0 represents a perfect model.

The next section examines the studies that have utilised different machine learning models, presenting their outcomes and stating the limitations observed.

3.1. *Bee Entrance and Exit Activity*

In [27], using time series data on entrance and exit, a model was developed that predicted the entrance and exit of bees' activities based on environmental conditions. Three models were explored: Long Short-Term Memory (LSTM) [50], Facebook Prophet [51], and Autoregressive Integrated Moving Average (ARIMA) [52]. Although the details of its performance metrics were not shown, the LSTM model is reported to have missed 8.9 bees on exit and 7.8 on entry per hour. The parameters that had the most significant impact on movements used to develop the prediction model included temperature and relative humidity inside and outside the hive, the occurrence of rain, air quality, the range and intensity of daylight, UV radiation, and the transition between night and day. The analysed data were collected over 20 days during a period that was not a flowering season so that bees would naturally not be as active, and insufficient time meant that the developed model would most probably not be able to accurately predict bee behaviour relative to varying metrological conditions. Using images from the beehive entrance, Ref. [18] developed a model that used a Resnet-50 Convolutional Neural Network without the fully connected layers for extraction and a Support Vector Machine (SVM) for classification. They achieved an accuracy of 85%. The dataset with 1000 images gave poor results when trained on Resnet-50, VGG-16 [53], or DenseNet-201 [54]. The study reported the limitation of pre-trained models led to overgeneralisation, especially when coupled with limited data, which affected the performance of the algorithm. The algorithm/model's performance provided limited information, and there was no ground truth comparison to validate the accuracy of their model.

3.2. *Prediction of an Imminent Swarming Event or Detection of an Ongoing Swarming Event*

In [23], using colony vibration data, a machine learning-based model was developed based on Principal Component Analysis (PCA) [55] for dimension reduction and Discriminant Functional Analysis (DFA) [56] for classification, which predicts with an accuracy of up to 90% whether an imminent swarm is about to occur or not. However, they did not consider several other factors that could affect the beehive's vibrational spectra of the beehive, such as brood levels, honeycomb levels, and infestation of pests. They also experienced false positive alarms for all the hives that were studied, which could indicate that their models were not sensitive or robust enough, and the hives that had experienced power shortages were wrongly predicted, suggesting that they fell short in the practicalities of field use; Ref. [29] detected an ongoing swarm event using images collected from the beehive. The algorithm for detection consisted of Single-Shot MultiBox Detector (SSD) [57] and Faster-RCNN principal Resnet [58] algorithms coupled with pre-trained models of MobileNetV1 [59] and Inception V2 [60] models. The study achieved the highest accuracy of 70% with Faster R-CNN and Inception V2 models. The pre-processing was performed using bilateral filtering and cubic interpolation or super-resolution using ESDR for their preparation as inputs. The training set used 6627 images, and 100 were used for validation. The validation was performed in a laboratory beehive using the number of bees detected for a population increase and clustering upon the introduction of a new queen to the colony. A dataset of 5000 images was used for training, and the validation was performed using a laboratory simulation that could affect the model's performance and generalisability;

Refs. [40,41], using 1800 datasets and the test set of 643 datasets, sought to detect a swarming event. Various feature extraction methods, such as Mel-Frequency Cepstral Coefficients (MFCC) [61] and Linear Predictive Coding (LPC) [62], were investigated. They evaluated the proposed system across three levels of acoustic model complexity, low, medium, and high, defined by the number of Gaussian mixtures per state. The low-complexity models achieved moderate classification accuracy, with the GMM model outperforming the others at 60.50%. Precision ranged between 0.72 and 0.79, indicating a low false positive rate, which is important for minimising incorrect swarm activity detections. However, recall values (0.70–0.78) were comparatively lower, reflecting limited swarm detection capability. The F1-scores (0.74–0.75) indicate a balanced but moderate performance, suitable for scenarios with constrained computational resources. The medium complexity models showed an improvement in classification accuracy, with the 15-state HMM model achieving the highest value at 75.43%. Precision and recall metrics also achieved higher precision, ranging from 0.85 to 0.88 and recall from 0.81 to 0.89. This indicates better swarm detection capabilities while maintaining low false positives. The F1-score of 0.86 for the 15-state HMM model reflects the optimal trade-off between precision and recall, making this configuration well-suited for IoT-based implementations with moderate computational capabilities. The high complexity models delivered the best classification performance, with the 15-state HMM model achieving the highest accuracy of 82.27%. Precision values (0.89) and recall (0.91–0.92) underline the superior detection ability of these models, particularly for identifying swarm events. The highest F1-score of 0.90 was observed with the 15-state HMM model, demonstrating its effectiveness for high-accuracy classification tasks, especially in server-based or cloud-supported applications where computational resources are not a limiting factor. This study showed that while higher complexity models yield superior performance, medium complexity models offer a favourable balance between accuracy and computational efficiency, making them a practical choice for IoT systems [11]. The study notes the misclassification due to the effect of the bees having direct contact with the recording devices. The author did not thoroughly investigate the diversity of the dataset, which only included data from a single beehive and its impact on the accuracy of the model. There were no real-world field studies to validate the results of any of their models.

3.3. Detection of Pests, Predominately Varroa Mites

In [24], a Convolutional Neural Network was developed to detect Varroa mites, and an image processing algorithm capable of estimating the number of bees infested was developed, which had an accuracy of 90%. Based on a video sequence with 1775 bees and 98 visual mites, Linear Discriminant Analysis was used to train and test a classifier from images with both bees and mites. The algorithm extracted the bee shape using background subtraction and segmentation techniques, a morphological open and close operation, an Implicit Shape Model, and a Scale-Invariant Feature Transform (Lowe) (SIFT). The use of SIFT in the pre-processing could come with a high computational cost and might not be optimal for very large datasets. This study also used a limited dataset that does not take into consideration different environmental conditions, e.g., lighting, that might affect the appearance of bees and mites. Ref. [30] described the detection of Varroa mites utilising several algorithms and pre-trained machine learning models from images that were collected by their embedded device and images from publicly available datasets. The techniques involved in the pre-processing of the images were bilateral filtering and cubic interpolation. The detection of the bees is performed by using the Single-Shot MultiBox Detector (SSD) and Faster R-CNN with pre-trained models (MobileNetV2, MobileNetV3, and RESNet-50 FPN), followed by the use of colour masking and Hough transform to detect the Varroa mites. The training phase was carried out using 100 bees and 100 images

of Varroa mites. The testing dataset consisted of 200 images, where half of them contained Varroa mites and the other half did not. The authors demonstrated that using R-CNN coupled with MobileNetV2, they could predict Varroa mites with an accuracy of 77% and a precision of 86%. They also presented a comparison of online detection, which was 3–4 times faster than offline detection on an embedded node. However, their model was not evaluated on some real-time incidents of Varroa mites' invasion of a beehive. Ref. [26] reported the development of a system that utilised SVM (Support Vector Machine) and K-Nearest Neighbors (K-NN) to detect the presence of Varroosis (disease due to Varroa mites) using volatile gaseous elements with an accuracy of 93%. The study determined that the number and type of gas sensors, which should be at least four, affected the performance of the algorithm for Varroa detection. The developed device is huge and would face scaling-up challenges that are suitable for field use. A feature was defined as the exposure of an individual gas sensor to beehive air for two minutes. The feature vector was a combination of the multiple gas sensors composed of the electronic nose, and a classifier was built for each feature. The K-NN classifier was chosen because of its simplicity and ease of deployment on the measuring system, while SVM was chosen because it was presumed to achieve higher accuracy; it is also well suited for two classification problems like this one. SVM showed better results than k-NN in both TPR and TNR with balanced datasets, making it the more reliable model for accurate classification. However, it had lower TNR with imbalanced data, suggesting it is more affected by unbalanced class distributions, which could be the typical scenario in the apiaries. Although this study was able to detect an infected colony, the level of infestation could not be specified. The influence of different colony groups on performance could potentially impact future predictions and meteorological conditions, as the study demonstrated their impact on the collected measurements. Ref. [31] detected Varroa mites from a video stream and developed a CNN-based ML model with an accuracy of 80%. The precision of the model was approximately 0.70, indicating a moderate rate of false positives in the detection of infected bees. The sensitivity (TPR) was 0.94, demonstrating the model's ability to accurately identify nearly all infected bees within the dataset. The specificity (TNR) was 0.92, showing the system's capability to correctly classify non-infected bees, reducing false alarms. The F1-score was 0.80, showing a balance between detection accuracy and false-positive. The researchers used 300 extracted images from the video stream as the training dataset for bee detection. The researchers used a further 10,743 images as a training set for the detection of the Varroa mite, which included 5748 pictures of healthy bees and 4995 pictures of bees with mites. The set containing mites was insufficient, so they modified the pictures to place mites in different places of its body. Their study showed that the size, resolution, and content of video frames did not affect the identification of the bees; however, the resolution impacted the detection of the mites, with lower resolution providing a worse performance. The study reported poor prediction for bees that are close to each other. This study does not mention how long the field evaluation lasted; it was performed on only one beehive. Although their study identified and detected Varroa mites, it did not show if it could rule out other types of mites that might not be harmful to the honeybees. Ref. [47] reported the development of algorithms for the detection of Varroa mites based on a CNN with an accuracy of 93%. Several techniques were investigated for the initial pre-processing of the images, including Histogram, Hough Transformation, and region labelling/colour identification. The study opted for region labelling because of its superior results for identifying mites on the bees. This system was, however, not tested on real field conditions.

3.4. Differentiation of Pollen-Bearing and Non-Bearing Bees

In their study to differentiate pollen and non-pollen-carrying bees, the authors of [32] developed a deep learning model based on the Tiny You Only Look Once (YOLOv3) [63] model to identify them from a video with an accuracy of 90%. The model achieved a precision of 0.91, indicating the model's ability to ensure that the majority of honey bees identified as pollen-bearing were accurately classified, thus reducing erroneous detection. A recall (TPR) of 0.99 exhibited the model's ability to recognise true positives, ensuring high detection performance. The F1-score was 0.94, showing the model's balanced performance in both minimising false positive rates and maximising true positive detections. The images to train the model were collected from different hives at different times and then divided into a training set of 3000 and a test set of 500 images. An integrated algorithm comprising a Kalman and Hungarian filter was used to track and count the bees. During a five-month field experiment, two other hives under control collected pollen using traps. The equation used to estimate the pollen carried by the beehives is an estimated value from a beehive different from the one where the images were obtained, potentially making the accuracy stated unreliable. Ref. [44] developed a model based on Faster RCNN with a VGG 16 Core Network to detect the presence or absence of pollen sacs on honeybees from a video captured at the entrance of the beehive. This model achieved a maximum sensitivity of 73%, with a measurement error of 7%, compared to another model that employed image processing techniques and statistical analysis. In videos with relatively low numbers of non-pollen bees, the deep learning model achieved high sensitivity (0.70). This suggests that the model is effective at correctly identifying pollen-carrying bees when they are more easily distinguishable from non-pollen-bearing bees. Where the number of non-pollen bees was significantly higher (1107 compared to 46 pollen bees), the model's sensitivity decreased to 0.46. This decrease reflects the challenge of distinguishing pollen-carrying bees in high-density scenarios, where the bees may overlap or be difficult to detect. The system's performance was evaluated in a laboratory-based beehive that might not reflect real-world conditions. Ref. [48] detected the presence of pollen sacs on bees; to this end, they deployed the use of segmentation and classification using SVM. The pre-processing techniques included segmentation by CIE Lab space and the K-means clustering algorithm, followed by morphological post-processing and dilation. Three types of images were investigated for the training set, which consisted of 500 images with pollen and 500 without pollen for the classification process: original RGB images, images using the b component, and decorrelation using Principal Component Analysis. The Vector of Locally Aggregated Descriptors (VLAD) [64] was used to compute the descriptors, followed by SVM to classify the descriptors. Three methods were evaluated for classification: The non-processed images achieved an AUC of 0.8700 and a confusion matrix indicating 86 true positives (TP) and 14 false positives (FP) with 12 false negatives (FN) and 88 true negatives (TN). The segmented images achieved a higher AUC of 0.9100, with 91 TP, 9 FP, 9 FN, and 91 TN, indicating that segmentation significantly improved classification performance. The decorrelated images achieved the highest AUC of 0.9150, with 90 TP, 10 FP, 7 FN, and 93 TN. This suggests that decorrelation further enhanced classification accuracy, especially in distinguishing between pollen-bearing and non-pollen-carrying bees. There was significant difficulty in identifying bees too close to each other and at the boundary of the recording. One of the challenges of using K-clustering is that you obtain different results based on the optimisation parameters used. The use of VLAD can be computationally expensive and requires careful parameter tuning for optimal performance.

3.5. Detection of Queen Presence in the Beehive

The authors of [19] reported the detection of the queen bee by investigating the performance of four machine learning models: SVM, K-Nearest Neighbors (KNN), Random Forest (RF), and CNN. The study reports that SVM had the best results, with an accuracy of 95%. A dataset was acquired from five beehives from their setup for 15 days. The pre-processing initially used Single Value Decomposition (SVD), and no significant patterns were noticed. They then extracted features using MFCCs, trained the model using 720 samples, and evaluated it using K-cross validation. The study reported that SVM and RF provided the highest recall, which is vital for identifying colonies with critical issues (e.g., queenless colonies). This ensures that colonies in distress are correctly identified for intervention, minimising the risk of missing out on failing colonies. Models with higher precision, such as SVM (0.92) and KNN (0.89), are vital for preventing false positives, especially in large colonies where there are many healthy bees. This reduces unnecessary interventions and allows for better resource allocation. SVM with an AUC of 0.94 offers better overall classification performance, making it more reliable in real-world monitoring systems where the threshold for classification might need to be adjusted dynamically. The computational efficiency of these models is crucial for real-time monitoring on devices like Raspberry Pi (RPi 3), especially in autonomous beekeeping systems. Although CNN showed high performance, SVM and KNN were more computationally efficient for use on an RPi 3, where processing power is limited. This ensures that real-time colony monitoring remains feasible. Although this study sought to determine the computation resources required for the learning and classification of the RPi3, it did not include the pre-processing, feature extraction, and data splitting time. A relatively small and similar, unbalanced dataset was used that could affect the representativeness and generalisation of the beehive. The developed models were not tested on real field conditions to validate them. Ref. [49] reported the development of SVM and CNN-based models with 80% and 90% accuracy, respectively, to detect the presence of the queen bee. The study pre-processed the data using MFCCs, Empirical Mode Decomposition (EMD), and the Hilbert Huang Transform (HHT). The AUC scores show the ability of both models to distinguish between the different colony states. However, the models showed challenges with generalization to unseen hives, which is critical in practical beekeeping applications where new hives might be encountered regularly. The SVM model was more prone to overfitting, while the CNN showed slightly better generalization ability, suggesting that CNNs may be more adaptable for deployment in real-world beekeeping scenarios. Both models struggled with unbalanced datasets and hive-independent splits, highlighting the need for better datasets in practical beekeeping. Ensuring that the training set includes a representative sample of hives from different environments and populations is crucial to improving model robustness and reliability. However, the study did not provide any explanations or interventions and apart from the AUC, no other metrics were discussed to gain a comprehensive understanding of the model's performance.

3.6. Discussion

The discussion focuses on evaluating the current state of research in precision beekeeping with an emphasis on aspects such as the choice of machine learning algorithms, the impact of sensing devices, computational complexity, dataset limitations, validation procedures, evaluation metrics, and the impact of pre-processing techniques on machine learning models' outcomes.

3.6.1. Current State of Machine Learning Algorithms

Figure 4 illustrates that the reviewed studies demonstrate an adoption of both classical machine learning models and deep learning models to determine honeybee activities. The most frequently used models were CNN and SVM. The choice between these algorithms often depends on trade-offs between computational complexity and accuracy. For instance, refs. [24,26] show that classical models like SVM can achieve high accuracy with lower computational demands, making them suitable for resource-constrained environments. However, the emergence of deep learning models such as those used by [30,31] offers improved accuracy and the ability to handle more complex patterns, although with higher computational costs and longer training times.

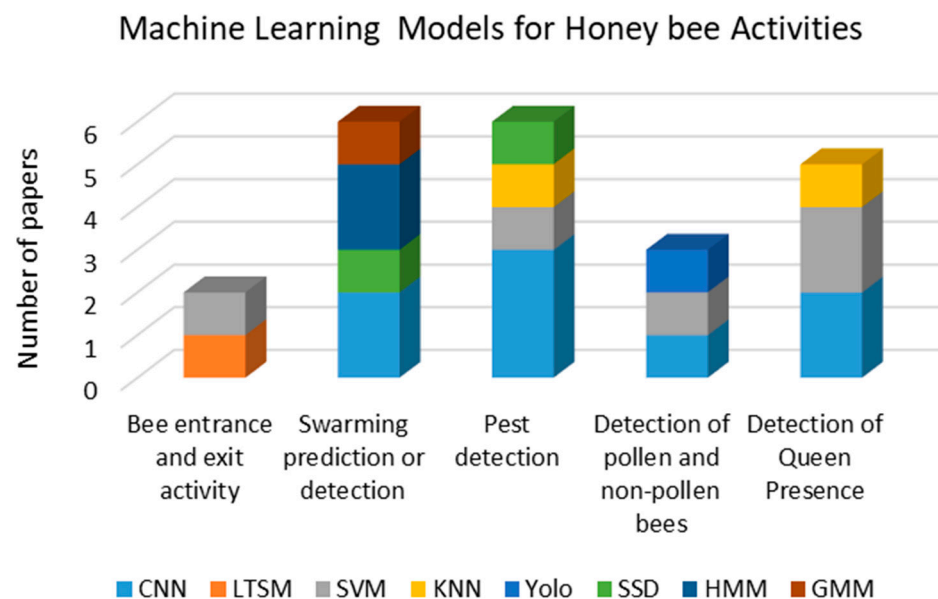


Figure 4. Machine learning models for honey bee activities.

3.6.2. Impact of Sensing Devices

The authors of [31] highlighted the impact of image resolution on the performance of machine learning models in detecting Varroa mites. Their findings indicate that lower image resolutions can significantly degrade the model's ability to detect mites, while high-resolution images improve detection but require more computational resources. This trade-off between image quality and processing speed is vital for real-time applications, where delays in processing could lead to missed detections or false positives. The study also pointed out challenges in accurately identifying bees in close proximity, a common issue in densely populated hives, further complicating the task of mite detection.

3.6.3. Impact of Pre-Processing Techniques

The pre-processing stage is critical in determining the effectiveness of machine learning models, particularly in handling the noisy and dynamic environment of beehives. Techniques such as those used by [24], including background subtraction, segmentation, and morphological operations, are essential for obtaining relevant features and improving model accuracy. However, these methods can introduce significant computational overhead, particularly when dealing with large datasets or high-resolution images. The choice of pre-processing methods, such as SIFT in [24] or bilateral filtering and cubic interpolation in [30], directly impacts the model's performance, especially in environments with variable lighting and background conditions.

3.6.4. Evaluation Metrics

The studies reviewed typically focus on basic evaluation metrics such as accuracy and precision, with limited attention given to more comprehensive measures that could provide deeper insights into model performance. For instance, metrics like Balanced Accuracy and Matthews Correlation Coefficient (MCC) are rarely reported despite their importance in assessing models trained on imbalanced datasets, a common scenario in beekeeping. Furthermore, while Area Under the Curve (AUC)–ROC and precision–recall curves are powerful tools for evaluating the trade-offs between different types of errors, they are often overlooked in favour of simpler metrics. The omission of these more detailed evaluations limits the ability to fully understand a model's strengths and weaknesses, particularly in diverse and unpredictable beekeeping environments. The reliance on a narrow set of evaluation metrics presents several challenges:

- Incomplete performance assessment: Basic metrics like accuracy provide only a partial view of a model's effectiveness. For instance, a high accuracy rate might obscure the model's poor performance on minority classes or its susceptibility to false positives. This can be critical in beekeeping, where misclassification of pests or other anomalies could lead to significant hive losses.
- Lack of generalizability: Models developed and tested under controlled conditions may not perform well in real-world environments, particularly those with varying meteorological conditions, different apiaries, or diverse bee populations. Without rigorous testing across multiple conditions and datasets, the generalisability of these models remains uncertain.

3.6.5. Computational Complexity and Resource Utilization

The authors of [24] utilized SIFT in their pre-processing pipeline, which, while effective, is computationally intensive. Such complexity might make it difficult for real-time processing and scalability, especially when dealing with large datasets or deploying on resource-constrained devices. Refs. [30,31] utilised deep learning models like Faster R-CNN and CNNs, respectively. While these models offer high accuracy, they require significant computational resources, which could be a barrier for in-field applications where power and processing capabilities are limited.

3.6.6. Dataset Limitations

Notable among the studies is the reliance on limited and sometimes unbalanced datasets. For instance, [24,30] trained their models on datasets that may not encompass the variability present in different environmental conditions, such as lighting variations or diverse hive structures. Ref. [31] encountered a lack of enough data for Varroa mite images, leading them to artificially augment the dataset by placing mites on bees in images. While data augmentation is a standard practice, it may not capture the complexity of real-world scenarios.

3.6.7. Validation Procedures

Several studies lacked extensive field validation. Refs. [30,47] did not test their models under real-world conditions, which raises concerns about the models' robustness and adaptability to varying environmental factors. Ref. [26] employed gas sensors and machine learning classifiers like SVM and K-NN to detect Varroosis. Although achieving high accuracy, the system's large size and the lack of field validation limit its practical applicability.

4. Knowledge Gaps and Recommendations for Further Research

IoT technologies coupled with machine learning techniques have shown great potential in improving the beekeeper's management of his apiaries through the detection of queen bee presence, pests, swarming detection and prediction, pollen-bearing bees detection, and ambient colony conditions monitoring, which are important indicators of the colony's health (Tables 1 and 5).

Integrating IoT technologies with machine learning in apiculture has demonstrated significant potential in enhancing beekeeping practices. However, several areas remain unexplored. Addressing these gaps will be key to realising the full potential of precision beekeeping systems. Below are suggested directions for future research.

4.1. Integration of IoT Systems and Machine Learning

While there is substantial research on the application of machine learning algorithms in precision beekeeping, there is a noticeable gap in exploring integrated IoT systems. The success of machine learning models in this domain depends not only on the algorithms themselves but also on how effectively they are integrated with IoT systems for data acquisition and processing. Current studies often overlook the complexity and computational demands of these models when deployed on IoT devices, especially in resource-limited environments like remote apiaries. Future research should focus on the following areas to achieve intelligent edge devices:

- Optimisation of machine learning models: research should focus on creating lightweight, energy-efficient versions of machine learning models that can perform effectively on low-power devices without neglecting accuracy.
- Development of real-time data processing capabilities: designing systems that can process and analyse data in real-time directly at the hive, reducing latency and dependence on cloud computing.

4.2. Power Sustainability and Energy Harvesting

The concern of power sustainability is critical, yet it is inadequately addressed in the reviewed studies. Effective and continuous monitoring of beehives, especially in remote locations, requires IoT systems that are not only power-efficient but also capable of operating independently over extended periods without frequent maintenance or battery replacements. To ensure continuous monitoring, future research should focus on the following:

- Including a detailed analysis of power consumption, considering the major components that contribute to the power consumption and using the formula stated as

$$P_{total} = P_{Sensor} + P_{transmission} + P_{Processing}$$

where:

- i. P_{Sensor} is the power required to acquire data.
 - ii. $P_{transmission}$ is the power required for data transmission.
 - iii. $P_{Processing}$ is the power consumed during local data processing.
- Optimising sensor power usage through techniques such as dynamic sampling that adjusts the frequency of data collection based on environmental conditions or predefined thresholds. For instance, sensors can increase sampling rates during periods of high variability, such as fluctuating hive temperatures, and decrease them during stable conditions.

- Selection of energy-efficient sensors designed for precision monitoring, such as MEMS-based temperature or humidity sensors, which consume minimal power while maintaining accuracy.
- Reduction in communication energy costs through the use of low-power communication protocols, such as LoRaWAN, Zigbee, or Bluetooth Low Energy (BLE), for short-range, energy-efficient transmission. For long-range communication, protocols like Narrowband Internet of Things (NB-IoT) [65] or Long Term Evolution Machine Type Communication (LTE-M) [66], optimised for IoT applications, can reduce power usage and data compression before transmission to reduce the volume of data sent, significantly minimising energy costs associated with the communication.
- Performing data analysis locally on IoT devices with lightweight machine learning models can ensure computational tasks consume minimal energy and reduce the need for frequent data transmissions to cloud servers, saving energy.
- Leveraging hardware accelerators or application-specific integrated circuits (ASICs) tailored for IoT operations. These specialised chips are designed to handle IoT workloads while consuming significantly less energy compared to general-purpose processors.

4.3. Comprehensive Field Testing and Dataset Diversity

Most of the existing studies rely on limited datasets collected from a small number of beehives over short periods. These datasets often lack the diversity needed to determine the complex and dynamic nature of bee colonies and their environments. Consequently, the models trained on these datasets may not generalize well to different conditions, leading to suboptimal performance in real-world applications.

Many studies rely on limited datasets, which may not represent the full range of conditions encountered in real-world beekeeping. Future research should consider the following:

- Expand data collection across a range of environmental conditions, including varying lighting (from daylight to low light), temperature fluctuations, seasonal changes, and weather conditions (e.g., rain, humidity). For acoustic classification, models should be tested under different ambient noise levels, such as windy or rainy conditions, to assess their ability to distinguish hive sounds. Additionally, validation should consider hive densities, bee population sizes, and hive designs, as these factors can influence environmental variables like temperature gradients and sound patterns. Models should also be trained and tested across multiple apiaries with varying environmental conditions and management practices to ensure generalizability and minimize site-specific overfitting.
- Creating and publishing large, diverse datasets with the research community to facilitate the development of more generalisable machine learning models.

4.4. Challenges with Existing Modalities and the Need for Multimodal Approaches

The current research often relies on a single modality, such as video, gaseous composition, or acoustics, to monitor and detect various aspects of bee colony health. However, each of these modalities presents specific challenges: Video and image data; while video-based monitoring is effective for certain tasks, such as detecting pollen-bearing bees or monitoring the entrance of hives, it has limitations. Pests such as Varroa mites may hide in areas not visible to cameras, and video data are computationally intensive in storage and processing. Gaseous composition is useful in determining conditions like Varroosis, but the sensors require constant resetting with clean air, making them obtrusive and less practical for continuous monitoring. Additionally, the complexity of gaseous compositions influenced by varying environmental conditions poses challenges for accurate analysis. Acoustics and vibration provide valuable insights into the health and behaviour of bees

but are affected by low Signal-to-Noise Ratios (SNR), making data processing complex and potentially less reliable.

To overcome these challenges, future research should explore multimodal approaches that combine several parameters. This could be accomplished by the following:

- Developing multimodal monitoring systems: integrating multiple sensing modalities, such as combining video, acoustics, temperature, and gaseous elements, to create a more comprehensive and reliable system.
- Creating fusion algorithms: developing techniques that can effectively combine data from different sensors, enhancing the accuracy and reliability of the system.
- Optimising data processing techniques: addressing the computational challenges of processing multimodal data, ensuring the system remains efficient and scalable.
- Investigating specific use cases for each modality: determining the strengths and weaknesses of each modality and exploring task-specific applications to determine which combinations of modalities are best suited for specific tasks, such as pest detection, swarming prediction, or hive health monitoring.

4.5. Scalability and Practical Deployment

The scalability of IoT-based precision beekeeping systems remains a significant challenge. Many of the existing solutions are developed and tested on a small scale, with limited consideration for how they can be scaled up for larger operations. Moreover, the practical deployment of these systems in diverse environments, including rural and under-resourced areas, has not been sufficiently explored. Future research should focus on developing scalable solutions that are cost-effective and easy to deploy across multiple hives and apiaries. This could involve the following:

- Improving connectivity: developing systems that can operate in remote areas with limited internet connectivity, potentially through low-bandwidth communication protocols like LoRaWAN.
- Feedback loops for system improvement: establishing mechanisms for collecting user feedback to continuously improve the systems based on real-world experiences.

4.6. Recommendation for Comprehensive Evaluation

To advance precision beekeeping and ensure the practical applicability of machine learning models, future research should adopt a more comprehensive evaluation framework that includes the following:

- Balanced Accuracy and Matthews Correlation Coefficient (MCC): these metrics should provide a more nuanced assessment of model performance, particularly in handling imbalanced datasets, which are common in real-world beekeeping scenarios.
- Area Under the Curve (AUC)–ROC and precision–recall curves: these tools should be employed to evaluate the trade-offs between different types of errors (e.g., false positives vs. false negatives) and to assess the model's sensitivity to various thresholds.
- Computational efficiency: it is crucial to report metrics such as inference time, memory usage, and overall resource utilization to assess the feasibility of deploying these models in real-world, resource-constrained settings like remote apiaries.

5. Conclusions

The integration of IoT technologies and machine learning techniques has shown significant potential in enhancing beekeeping practices. These advancements have enabled the detection of queen bee presence, pests, swarming, pollen-bearing bees, and the monitoring of ambient colony conditions, which are crucial indicators of colony health. However, the current state of research predominantly focuses on the application of machine learning

algorithms for the analysis of apiculture-related datasets, with less emphasis on the development of robust IoT systems. The complexity and computational demands of machine learning models directly influence data acquisition strategies, making the integration of these systems a significant challenge. Future research areas could focus on the development of intelligent IoT-based edge devices, power sustainability solutions, investigation of the effect of long-term deployment of sensor modules in beehives, comprehensive evaluation using diverse datasets comprised of data collected from different apiaries over different seasons, exploration of scalable and practical systems, development of a multimodal approach to mitigate the limitations of the singular modalities, and obtaining more accurate and precise results to enhance apiary management and contribute to the sustainability of bee populations.

Author Contributions: Writing—original draft preparation, A.T.; writing—review and editing, A.T., R.A. and M.N.; supervision, R.A., M.N., A.K. and M.E.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by NORAD through the NORHED2 program under grant agreement QZA21/0159 as part of the AdEMNEA project.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hristov, P.; Shumkova, R.; Palova, N.; Neov, B. Factors Associated with Honey Bee Colony Losses: A Mini-Review. *Vet. Sci.* **2020**, *7*, 166. [CrossRef] [PubMed]
2. Orr, M.C.; Hughes, A.C.; Chesters, D.; Pickering, J.; Zhu, C.D.; Ascher, J.S. Global Patterns and Drivers of Bee Distribution. *Curr. Biol.* **2021**, *31*, 451–458.e4. [CrossRef] [PubMed]
3. Hadjur, H.; Ammar, D.; Lefèvre, L. Toward an intelligent and efficient beehive: A survey of precision beekeeping systems and services. *Comput. Electron. Agric.* **2022**, *192*, 106604. [CrossRef]
4. Meikle, W.G.; Holst, N. Application of continuous monitoring of honeybee colonies. *Apidologie* **2015**, *46*, 10–22. [CrossRef]
5. Meikle, W.G.; Rector, B.G.; Mercadier, G.; Holst, N. Within-day variation in continuous hive weight data as a measure of honey bee colony activity. *Apidologie* **2008**, *39*, 694–707. [CrossRef]
6. Thoenes, S.C.; Buchmann, S.L. Colony Abandonment by Adult Honey Bees: A Behavioural Response to High Tracheal Mite Infestation? *J. Apic. Res.* **2015**, *31*, 167–168. [CrossRef]
7. Meikle, W.G.; Holst, N.; Colin, T.; Weiss, M.; Carroll, M.J.; McFrederick, Q.S.; Barron, A.B. Using within-day hive weight changes to measure environmental effects on honey bee colonies. *PLoS ONE* **2018**, *13*, e0197589. [CrossRef] [PubMed]
8. Cecchi, S.; Spinsante, S.; Terenzi, A.; Orcioni, S. A Smart Sensor-Based Measurement System for Advanced Bee Hive Monitoring. *Sensors* **2020**, *20*, 2726. [CrossRef] [PubMed]
9. Mezquida, D.A.; Martínez, J.L. Platform for beehives monitoring based on sound analysis. A perpetual warehouse for swarm 's daily activity. *Span. J. Agric. Res.* **2009**, *7*, 824–828. [CrossRef]
10. Pham-Delègue, M.H.; Decourtye, A.; Kaiser, L.; Devillers, J. Behavioural methods to assess the effects of pesticides on honey bees. *Apidologie* **2002**, *33*, 425–432. [CrossRef]
11. Goodfellow, I.J.; Bengio, Y.; Courville, A.; Goodfellow, I.J.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
12. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2006; ISBN 0387310738. Available online: <https://link.springer.com/book/9780387310732> (accessed on 23 July 2024).
13. Schölkopf, B.; Smola, A.J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond Adaptive Computation and Machine Learning*. 2001. Available online: <https://ebookcentral-proquest-com.tudelft.idm.oclc.org/lib/delft/detail.action?docID=3338886> (accessed on 13 August 2024). [CrossRef]
14. Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern Classification*. 2001. 654. Available online: <https://www.wiley.com/en-us/Pattern+Classification,+2nd+Edition-p-9781118586006> (accessed on 13 August 2024).
15. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*; Springer: Berlin/Heidelberg, Germany, 2021. [CrossRef]
16. Awad, M.; Khanna, R. Machine Learning. In *Efficient Learning Machines*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 1–18. [CrossRef]

17. Dimitrijevic, S.; Zogovic, N.; Dimitrijević, S.; Zogović, N. Machine Learning Advances in Beekeeping. In Proceedings of the 12th International Conference on Information Society and Technology—ICIST, Kopaonik, Serbia, 13–16 March 2022. Available online: <https://www.researchgate.net/publication/363924735> (accessed on 16 December 2022).
18. Bilik, S.; Bostik, O.; Kratochvila, L.; Ligocki, A.; Poncak, M.; Zemcik, T.; Richter, M.; Janakova, I.; Honec, P.; Horak, K. Machine Learning and Computer Vision Techniques in Bee Monitoring Applications. *Comput. Electron. Agric.* **2022**, *217*, 108560. [\[CrossRef\]](#)
19. Robles-Guerrero, A.; Saucedo-Anaya, T.; Guerrero-Mendez, C.A.; Gómez-Jiménez, S.; Navarro-Solís, D.J. Comparative Study of Machine Learning Models for Bee Colony Acoustic Pattern Classification on Low Computational Resources. *Sensors* **2023**, *23*, 460. [\[CrossRef\]](#)
20. Alleri, M.; Amoroso, S.; Catania, P.; Lo Verde, G.; Orlando, S.; Ragusa, E.; Sinacori, M.; Vallone, M.; Vella, A. Recent developments on precision beekeeping: A systematic literature review. *J. Agric. Food Res.* **2023**, *14*, 100726. [\[CrossRef\]](#)
21. Jewani, K.; Abimannan, S. Edge Intelligence in IoT: Architecture and Applications. In Proceedings of the 2023 8th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 1–3 June 2023; pp. 337–345. [\[CrossRef\]](#)
22. Kong, X.; Wu, Y.; Wang, H.; Xia, F. Edge Computing for Internet of Everything: A Survey. *IEEE Internet Things J.* **2022**, *9*, 23472–23485. [\[CrossRef\]](#)
23. Ramsey, M.T.; Bencsik, M.; Newton, M.I.; Reyes, M.; Pioz, M.; Crauser, D.; Delso, N.S.; Le Conte, Y. The prediction of swarming in honeybee colonies using vibrational spectra. *Sci. Rep.* **2020**, *10*, 9798. [\[CrossRef\]](#) [\[PubMed\]](#)
24. Bjerger, K.; Frigaard, C.E.; Mikkelsen, P.H.; Nielsen, T.H.; Misbih, 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. [\[CrossRef\]](#)
25. Tu, G.J.; Hansen, M.K.; Kryger, P.; Ahrendt, P. Automatic behaviour analysis system for honeybees using computer vision. *Comput. Electron. Agric.* **2016**, *122*, 10–18. [\[CrossRef\]](#)
26. Szczurek, A.; Maciejewska, M.; Bąk, B.; Wilk, J.; Wilde, J.; Siuda, M. Gas Sensor Array and Classifiers as a Means of Varroosis Detection. *Sensors* **2019**, *20*, 117. [\[CrossRef\]](#)
27. Andrijević, N.; Urošević, V.; Arsić, B.; Herceg, D.; Savić, B. IoT Monitoring and Prediction Modeling of Honeybee Activity with Alarm. *Electronics* **2022**, *11*, 783. [\[CrossRef\]](#)
28. Ngo, T.N.; Wu, K.C.; Yang, E.C.; Lin, T. Te A real-time imaging system for multiple honey bee tracking and activity monitoring. *Comput. Electron. Agric.* **2019**, *163*, 104841. [\[CrossRef\]](#)
29. Voudiotis, G.; Kontogiannis, S.; Pikridas, C. Proposed Smart Monitoring System for the Detection of Bee Swarming. *Inventions* **2021**, *6*, 87. [\[CrossRef\]](#)
30. Wei, H.-L.; Liu, Z.-H.; Voudiotis, G.; Moraiti, A.; Kontogiannis, S. Deep Learning Beehive Monitoring System for Early Detection of the Varroa Mite. *Signals* **2022**, *3*, 506–523. [\[CrossRef\]](#)
31. 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. [\[CrossRef\]](#)
32. Ngo, T.N.; Rustia, D.J.A.; Yang, E.C.; Lin, T. Te Automated monitoring and analyses of honey bee pollen foraging behavior using a deep learning-based imaging system. *Comput. Electron. Agric.* **2021**, *187*, 106239. [\[CrossRef\]](#)
33. Danieli, P.P.; Addeo, N.F.; Lazzari, F.; Manganello, F.; Bovera, F. Precision Beekeeping Systems: State of the Art, Pros and Cons, and Their Application as Tools for Advancing the Beekeeping Sector. *Animals* **2024**, *14*, 70. [\[CrossRef\]](#)
34. Bencsik, M.; Newton, M.I. Honey Bee Vibration Monitoring Using the 805M1 Accelerometer. *Proceedings* **2018**, *4*, 42. [\[CrossRef\]](#)
35. Mariani, S.; Falcone, F.; Bosse, S.; Kulyukin, V.; Laheurte, J.-M.; Bencsik, M.; Mcveigh, A.; Tsakonas, C.; Kumar, T.; Chamberlain, L.; et al. A Monitoring System for Carbon Dioxide in Honeybee Hives: An Indicator of Colony Health. *Sensors* **2023**, *23*, 3588. [\[CrossRef\]](#) [\[PubMed\]](#)
36. Szczurek, A.; Maciejewska, M.; Bąk, B.; Wilk, J.; Wilde, J.; Siuda, M. Detecting varroosis using a gas sensor system as a way to face the environmental threat. *Sci. Total Environ.* **2020**, *722*, 137866. [\[CrossRef\]](#)
37. Zacepins, A.; Kviesis, A.; Komasilovs, V.; Rido Muhammad, F. Monitoring system for remote bee colony state detection. *Balt. J. Mod. Comput.* **2020**, *8*, 461–470. [\[CrossRef\]](#)
38. Centenaro, M.; Vangelista, L.; Zanella, A.; Zorzi, M. Long-range communications in unlicensed bands. *IEEE Wirel. Commun.* **2016**, *23*, 60–67. [\[CrossRef\]](#)
39. Zhu, J.; Tao, Z.; Lv, C. Performance evaluation of IEEE 802.15.4 CSMA/CA scheme adopting a modified LIB model. *Wirel. Pers. Commun.* **2012**, *65*, 25–51. [\[CrossRef\]](#)
40. Zgank, A. Bee Swarm Activity Acoustic Classification for an IoT-Based Farm Service. *Sensors* **2019**, *20*, 21. [\[CrossRef\]](#)
41. Zgank, A. IoT-Based Bee Swarm Activity Acoustic Classification Using Deep Neural Networks. *Sensors* **2021**, *21*, 676. [\[CrossRef\]](#)
42. Cecchi, S.; Terenzi, A.; Orcioni, S.; Spinsante, S.; Mariani Primiani, V.; Moglie, F.; Ruschioni, S.; Mattei, C.; Riolo, P.; Isidoro, N. Multi-sensor platform for real time measurements of honey bee hive parameters. In *IOP Conference Series: Earth and Environmental Science*; IOP Publishing: Bristol, UK, 2019; Volume 275, p. 012016. [\[CrossRef\]](#)
43. Lin, J.; Zhu, L.; Chen, W.-M.; Wang, W.-C.; Han, S. Tiny Machine Learning: Progress and Futures. *IEEE Circuits Syst. Mag.* **2024**, *23*, 8–34. [\[CrossRef\]](#)

44. Yang, C.; Collins, J. Deep Learning for Pollen Sac Detection and Measurement on Honeybee Monitoring Video. In Proceedings of the 2019 International Conference on Image and Vision Computing New Zealand (IVCNZ), Dunedin, New Zealand, 2–4 December 2019. [\[CrossRef\]](#)
45. Kim, J.; Oh, J.; Heo, T.-Y. Acoustic Scene Classification and Visualization of Beehive Sounds Using Machine Learning Algorithms and Grad-CAM. *Math. Probl. Eng.* **2021**, 2021, 5594498. [\[CrossRef\]](#)
46. Várkonyi, D.T.; Seixas, J.L.; Horváth, T. Dynamic noise filtering for multi-class classification of beehive audio data. *Expert Syst. Appl.* **2023**, 213, 118850. [\[CrossRef\]](#)
47. Chazette, L.; Becker, M.; Szczerbicka, H. Basic algorithms for bee hive monitoring and laser-based mite control. In Proceedings of the 2016 IEEE symposium series on computational intelligence (SSCI), Athens, Greece, 6–9 December 2017. [\[CrossRef\]](#)
48. Stojnić, V.; Risojević, V.; Pilipović, R. Detection of pollen bearing honey bees in hive entrance images. In Proceedings of the 2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH), East Sarajevo, Republic of Srpska, 21–23 March 2018; pp. 1–4. [\[CrossRef\]](#)
49. Nolasco, I.; Benetos, E. To bee or not to bee: Investigating machine learning approaches for beehive sound recognition. *arXiv* **2018**, arXiv:1811.06016.
50. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, 9, 1735–1780. [\[CrossRef\]](#)
51. Taylor, S.J.; Letham, B. Forecasting at Scale. *Am. Stat.* **2018**, 72, 37–45. [\[CrossRef\]](#)
52. Wulff, S.S. Time Series Analysis: Forecasting and Control, 5th edition. *J. Qual. Technol.* **2017**, 49, 418–419. [\[CrossRef\]](#)
53. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556. Available online: <https://arxiv.org/abs/1409.1556v6> (accessed on 13 August 2024).
54. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269. [\[CrossRef\]](#)
55. Jolliffe, I.T.; Cadima, J. Principal component analysis: A review and recent developments. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2016**, 374, 20150202. [\[CrossRef\]](#) [\[PubMed\]](#)
56. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*; Springer: Berlin/Heidelberg, Germany, 2009. [\[CrossRef\]](#)
57. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016*; Springer: Berlin/Heidelberg, Germany, 2015; Volume 9905, pp. 21–37. [\[CrossRef\]](#)
58. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, 39, 1137–1149. [\[CrossRef\]](#)
59. Shanmugamani, R. *Deep Learning for Computer Vision: Expert Techniques to Train Advanced Neural Networks Using TensorFlow and Keras*; Packt Publishing Ltd.: Birmingham, UK, 2018; Volume 302. Available online: <https://www.oreilly.com/library/view/deep-learning-for/9781788295628/> (accessed on 25 July 2024).
60. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2818–2826. [\[CrossRef\]](#)
61. Muda, L.; Begam, M.; Elamvazuthi, I. Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques. *Int. J. Comput. Appl.* **2010**, 116, 34–41. Available online: <https://arxiv.org/abs/1003.4083v1> (accessed on 13 August 2024).
62. Rabiner, L.R.; Schafer, R.W. *Theory and Applications of Digital Speech Processing*; Pearson: London, UK, 2011.
63. Redmon, J.; Farhadi, A. *YOLOv3: An Incremental Improvement*; Springer: Berlin/Heidelberg, Germany, 2018. Available online: <https://arxiv.org/abs/1804.02767v1> (accessed on 25 July 2024). [\[CrossRef\]](#)
64. Arandjelović, R.; Zisserman, A.A. All About VLAD. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1578–1585. [\[CrossRef\]](#)
65. Migabo, E.M.; Djouani, K.D.; Kurien, A.M. The Narrowband Internet of Things (NB-IoT) Resources Management Performance State of Art, Challenges, and Opportunities. *IEEE Access* **2020**, 8, 97658–97675. [\[CrossRef\]](#)
66. Dawaliby, S.; Bradai, A.; Pousset, Y. In depth performance evaluation of LTE-M for M2M communications. In Proceedings of the 2016 IEEE 12th international conference on wireless and mobile computing, networking and communications (WiMob), New York, NY, USA, 17–19 October 2016. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.