Ecological Informatics 64 (2021) 101353

Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf

Deep learning-based classification models for beehive monitoring

Selcan Kaplan Berkaya [a,*], Efnan Sora Gunal [b], Serkan Gunal [a]

[a] Dept. of Computer Engineering, Eskisehir Technical University, Eskisehir, Turkiye
[b] Dept. of Computer Engineering, Eskisehir Osmangazi University, Eskisehir, Turkiye

ARTICLE INFO

Keywords:
Beehive monitoring
Deep learning
Honey bee
Smart agriculture
Varroa detection

ABSTRACT

Honey bees are not only the fundamental producers of honey but also the leading pollinators in nature. While honey bees play such a vital role in the ecosystem, they also face a variety of threats, including parasites, ants, hive beetles, and hive robberies, some of which could even lead to the collapse of colonies. Therefore, early and accurate detection of abnormalities at a beehive is crucial to take appropriate countermeasures promptly. In this paper, deep learning-based image classification models are proposed for beehive monitoring. The proposed models particularly classify honey bee images captured at beehives and recognize different conditions, such as healthy bees, pollen-bearing bees, and certain abnormalities, such as Varroa parasites, ant problems, hive robberies, and small hive beetles. The models utilize transfer learning with pre-trained deep neural networks (DNNs) and also a support vector machine classifier with deep features, shallow features, and both deep and shallow features extracted from these DNNs. Three benchmark datasets, consisting of a total of 19,393 honey bee images for different conditions, were used to train and evaluate the models. The results of the extensive experimental work revealed that the proposed models can recognize different conditions as well as abnormalities with an accuracy of up to 99.07% and stand out as good candidates for smart beekeeping and beehive monitoring.

1. Introduction

Honey bees are eusocial flying insects within the genus *Apis*. They are the main actors in honey production and pollination. Though 7 to 11 species of honey bees are recognized historically, only two of these species have been truly domesticated for honey production: the western (or European) honey bee (*Apis mellifera*) and the eastern honey bee (*Apis cerana*) (Engel, 1999). The western honey bee is more common worldwide, while the eastern honey bee is found mainly in South Asia (Beaurepaire et al., 2020).

Unfortunately, these creatures, which are of great importance to nature and humanity, face various threats such as pests, diseases, honey robberies, and changing landscapes (Andrews, 2019; Beaurepaire et al., 2020). Among all these threats, *Varroa destructor*, an ectoparasitic mite, is one of the most significant ones that can lead to the collapse of bee colonies if left untreated (Mondet et al., 2020; Ramsey et al., 2019; Traynor et al., 2020). Varroa attaches to the body of the honey bee and weakens it by sucking the fat body tissue (Ramsey et al., 2019). Varroa also transmits several viruses, such as acute bee paralysis virus and the deformed wing virus, that may cause severe health issues in beehives (Beaurepaire et al., 2020; Ogihara et al., 2020b). Robber bees pose another threat as they gain their food by plundering the hives of the other bees instead of visiting the flowers (von Zuben et al., 2016). Also, small hive beetles may cause severe damage to the colonies and endanger beekeeping (Schafer et al., 2019). Beehives also attract other insects, such as ants (Nouvian et al., 2016).

Because of the abovementioned problems, beehive monitoring is vital for beekeepers to take appropriate countermeasures promptly. In traditional beekeeping, there are several methods for beehive monitoring. For example, a roll test with powdered sugar or roasted soybean flour is one of the common methods used to detect Varroa destructor infestations in beehives (Noël et al., 2020; Ogihara et al., 2020a). Visual observation is another traditional method to detect unintended beetles. However, these methods are time-consuming and not very efficient at all. Using signal processing, computer vision, and machine learning techniques, rather than the conventional methods, can be faster and much more effective in beehive monitoring.

Hence, in our work, deep learning-based classification models are proposed for beehive monitoring. The proposed models particularly classify honey bee images captured at beehives and can recognize different conditions, such as healthy bees, pollen-bearing bees, and certain abnormalities, such as Varroa parasites, ant problems, hive

* Corresponding author.
E-mail address: skb@eskisehir.edu.tr (S. Kaplan Berkaya).

https://doi.org/10.1016/j.ecoinf.2021.101353
Received 12 April 2021; Received in revised form 8 June 2021; Accepted 8 June 2021
Available online 18 June 2021

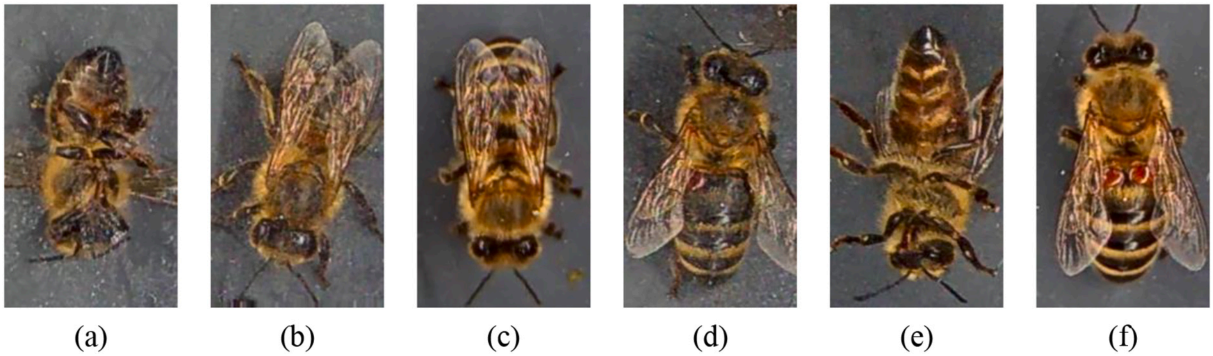

Fig. 1. Sample images from the Varroa dataset: (a-c) healthy, and (d-f) infected honey bees.

Table 1
The content of the Varroa dataset.

Class no	Class label	# Images
1	Healthy	9,561
2	Infected	3,946

robberies, and small hive beetles. The models utilize transfer learning with seven different pre-trained deep neural networks (DNNs) and also a support vector machine classifier with deep features, shallow features, and both deep and shallow (hereinafter referred to as deep+shallow) features extracted from these DNNs. Three benchmark datasets, consisting of a total of 19,393 honey bee images for different conditions, were used to train and evaluate the models. A thorough experimental analysis was carried out to assess the models in terms of classification performance and processing time. An extensive experimental work revealed that the proposed models can recognize different conditions as well as abnormalities with significantly high performance and surpass the previous works.

The rest of the paper is organized as follows: Section 2 expresses the related work, Section 3 describes the honey bee image datasets and introduces the proposed classification models. In Section 4, the experimental work and related results are presented and discussed. Also, a comparison of this work against the literature is provided in terms of various aspects, including dataset size, number of classes, features, classification method, application field, classification performance, and processing time. Finally, some concluding remarks and future directions are given in Section 5.

2. Related work

In recent years, a limited number of efforts have been made to monitor beehives and bees using various image processing, audio processing, and machine learning techniques on still images, video streams, and audio recordings. In (Santana et al., 2014), a reference process was proposed to automate the identification of bee species based on wing images and digital image processing techniques. In (Babic et al., 2016), a system was proposed to detect pollen-bearing honey bees from surveillance videos recorded at the entrance of a beehive. Background subtraction, color segmentation, and morphological operations were used for the segmentation of honey bees. Classification of segmented honey bees as "pollen-bearing or not" was then carried out using the nearest mean classifier, with a simple descriptor consisting of color variance and eccentricity features. In (Martineau et al., 2017), over forty studies on image-based insect (including honey bees) classification were reviewed based on several aspects such as image capture, feature extraction, classifier, and dataset. The detection and measurement of pollen on bees entering a beehive were addressed in (Yang, 2018) using video recordings. For this purpose, object detection and tracking methods based on low-level image processing techniques and deep learning were employed. In (Schurischuster et al., 2018), an approach for detecting varroa destructor was proposed using image processing and traditional machine learning techniques on honey bee videos. In (Kulyukin et al., 2018), the authors designed several convolutional neural networks and compared their performance with traditional machine learning methods in classifying audio samples from microphones deployed above landing pads of Langstroth beehives. In (Bjerge et al., 2019), a portable computer vision system based on deep learning was presented to monitor the infestation level of the Varroa mite in a beehive by recording a video sequence of honey bees. In (Yang and Collins, 2019), a deep learning-based model was proposed to detect and measure pollen sac on honeybee monitoring video. In (Uzen et al., 2019), the health status of honey bees was classified using deep learning on bee images. For this purpose, various convolutional neural network architectures were studied. In (Dembski and Szymański, 2019), an approach based on a neural network classifier was proposed to detect bees in video streams. In that work, different color models such as RGB and HSV were compared as the input format for feedforward convolutional architecture applied to bee detection. In (Mukherjee, 2020), acquisition, processing, and analysis of audio, video, and meteorological data for electronic beehive monitoring were studied. The author presented an electronic beehive monitoring system requiring no structural modifications to a standard beehive so that natural beehive cycles were not interrupted. Also, the design of a convolutional neural network architecture processing raw audio waveforms was discussed and the performances of several deep learning models were compared. In (Alves et al., 2020), the authors developed a deep learning-based model that can automatically detect cells in comb images and classify their contents into several classes including eggs, larvae, capped brood, pollen, nectar, honey, and others. In (Buschbacher et al., 2020), wild bee species in the field were identified using convolutional neural networks on digital images. The study enabled participatory sensing using mobile phones and a cloud-based platform. In (Braga et al., 2020), a new method was proposed for classifying seasonal honey bee patterns. The method aimed to assist the beekeepers in the management and maintenance of their hives. In (Margapuri et al., 2020), the identification of species of bumblebees was studied using deep transfer learning on bee images. In (Schurischuster and Kampel, 2020a), the authors utilized deep learning techniques and classified bee images as 'healthy' and 'infected' based on the presence of Varroa destructor on bees. Finally, a sound analysis of beehives was carried out using machine learning methods to detect air pollutants (Zhao et al., 2021). For this purpose, the beehives were exposed to the common air pollution chemicals while acquiring the sounds of the honey bees. It was found that these sounds have a certain relationship with pollution.

3. Materials and methods

In this section, the utilized image datasets are first described. Then, the proposed classification models are elaborated.

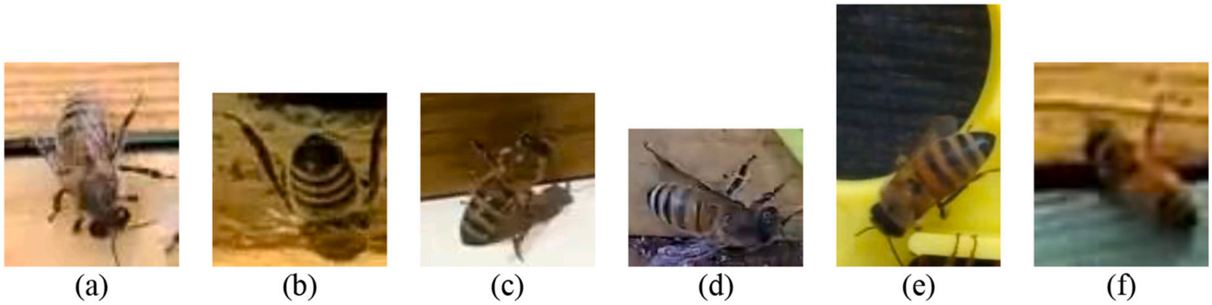

Fig. 2. Sample images from the BeeImage dataset. (a) Ant problems, (b) Few Varroa, hive beetles, (c) Healthy, (d) Hive being robbed, (e) Missing queen, (f) Varroa, small hive beetles.

Table 2
The content of the BeeImage dataset.

Class no	Class label	# Images
1	Ant problems	457
2	Few Varroa, hive beetles	579
3	Healthy	3,384
4	Hive being robbed	251
5	Missing queen	29
6	Varroa, small hive beetles	472

3.1. Datasets

The first dataset employed in this work is the Varroa dataset (Schurischuster and Kampel, 2020b) consisting of a total of 13,507 images, including healthy and Varroa-infected honey bees. The images are true-color (RGB) and have a resolution of (160 × 280). Sample images from the dataset are shown in Fig. 1. The dataset was divided into three subsections, namely training (61%), validation (14%), and test (25%). The distribution of the dataset is listed in Table 1.

The second dataset is the BeeImage dataset, which was previously proposed in (Yang, 2019). This dataset consists of a total of 5,172 honey bee images with 6 categories, including ant problems, few Varroa-hive beetles, healthy, hive being robbed, missing queen, and Varroa-small hive beetles. Sample images from the dataset are shown in Fig. 2. Unlike the first dataset, the images in this dataset have varying sizes. The dataset was divided into three subsections, namely training (70%), validation (15%), and test (15%). The content of the dataset is summarized in Table 2.

The third dataset is the Pollen dataset, which was previously proposed in (Rodriguez et al., 2018). This dataset consists of a total of 714 images (180 × 300), including pollen-bearing and non-pollen-bearing honey bees. Sample images from the dataset are shown in Fig. 3. The dataset was divided into three subsections, namely training (70%), validation (15%), and test (15%). The content of the dataset is summarized in Table 3.

3.2. Classification models

In this work, four different deep learning-based classification models are proposed to classify honey bee images captured at beehives and recognize different conditions as well as abnormalities.

Deep learning is a branch of machine learning based on DNNs with representation learning (LeCun et al., 2015; Schmidhuber, 2015). A DNN is a type of artificial neural network with many hidden layers between the input and output layers, whereas a shallow neural network has only one or few hidden layers. On the other hand, deep convolutional neural networks (CNNs) are a specific type of DNNs, which are particularly useful in image recognition (Rawat and Wang, 2017). Although there are many different CNN architectures, their general structure is formed by adding several convolutional and pooling layers one after the other, after the input layer from which the data is fed. One or more fully connected layers come after these layers and finally, the output layer takes place. The convolutional layer, containing a set of convolution filters, is responsible for feature extraction from the images fed from the input layer. The output of the convolution operation is processed by an activation function such as sigmoid, tanh, and ReLU to form a feature map. Pooling layer is then used to downsample, or reduce, the spatial dimension of the feature map of the convolutional layer. Maximum and average functions are widely used for pooling. Finally, fully connected layers are placed before the output layer and used to flatten the results prior to classification. A typical deep CNN architecture is illustrated in Fig. 4.

The first of the proposed classification models utilizes transfer learning with various pre-trained DNNs (referring to deep CNNs). Transfer learning allows us to use pre-trained DNNs instead of training a new network from scratch (Chen et al., 2020; Espejo-Garcia et al., 2020; Han et al., 2018; Kaplan Berkaya et al., 2020; Nguyen et al., 2018; Sarkar et al., 2018). In this way, rich feature representations for a wide range of images obtained by pre-trained networks (previously trained with over a million images) can be rapidly transferred to classify bee images using a relatively smaller number of training images. Therefore, compared to a new network trained from scratch, not only improved classification performance is achieved, but also overall training time is reduced. For this purpose, a pre-trained network, which has already learned to classify particular images, is used as a starting stage of learning a new classification task. Then, the last fully connected layer and the final classification layer of the pre-trained DNN are replaced in accordance with the total number of classes in this work. Finally, the modified pre-trained DNN is re-trained with the training and validation images in our dataset so that bee images can be appropriately classified by the re-trained DNN. The actual performance of this final DNN is then measured with the test images which are not used during the training phase.

The other three classification models employ an SVM classifier (Theodoridis and Koutroumbas, 2009) with deep features, shallow features (Kaplan Berkaya et al., 2020; Sun and Lv, 2019), and deep+shallow features extracted from the pre-trained DNNs, respectively. A DNN constructs a hierarchical representation of input images through layers. In the deeper layers, higher-level features are constructed using lower-level features from earlier layers. The shallower features are extracted from earlier layers and have a higher spatial resolution and a larger total number of activations. With the help of these deep, shallow, and deep+shallow features fed to an SVM classifier, various conditions at beehives can be recognized accurately.

Specifically, seven different pre-trained DNNs, including AlexNet (Krizhevsky et al., 2017), DenseNet-201 (Huang et al., 2017), GoogLeNet (Szegedy et al., 2015), ResNet-101 (He et al., 2016), ResNet-18 (He et al., 2016), VGG-16 (Simonyan and Zisserman, 2014), and VGG-19 (Simonyan and Zisserman, 2014) are preferred for this work. These networks are previously trained on ImageNet (Deng et al., 2009) consisting of over a million images for a wide range of objects. Although the first use of CNNs began with LeNET (LeCun et al., 1998) for handwritten

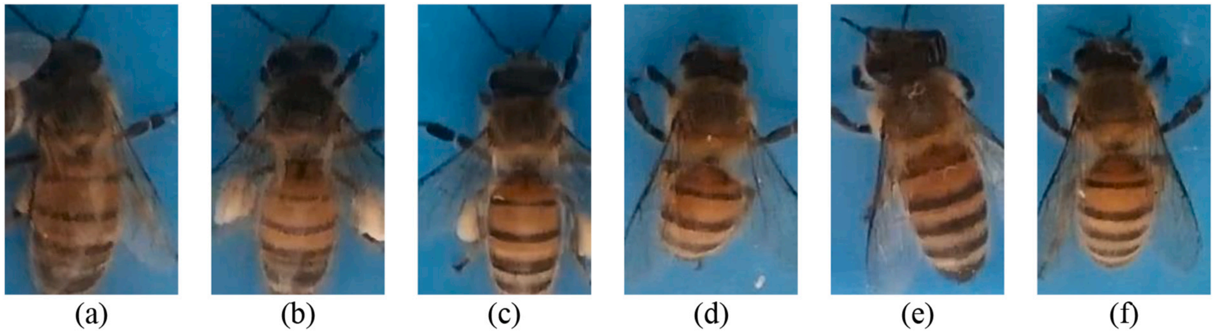

Fig. 3. Sample images from the Pollen dataset: (a-c) pollen-bearing, and (d-f) non-pollen-bearing honey bees.

digit recognition, AlexNet, the winner of the ILSVRC-2012 competition, can be considered as the first deep CNN for image classification (Krizhevsky et al., 2017). It has five convolutional layers, three fully connected layers, and 60 million parameters. In the network, stochastic gradient descent is used to optimize the loss function. VGG, which stands for Visual Geometry Group (of Oxford University), is another DNN architecture and proposed by (Simonyan and Zisserman, 2014). It was improved over AlexNet by replacing large-sized filters with multiple 3 × 3 filters one after another. VGG-16 network has 13 convolutional layers and 3 fully connected layers, while VGG-19 has 16 convolutional layers and 3 fully connected layers. These two networks have 138 and 144 million parameters, respectively. GoogLeNet, also called Inception-V1, is the CNN structure developed by the Google team that won the ILSVRC-2014 competition (Szegedy et al., 2015). It introduced the new concept of inception block in CNN, whereby it incorporates multi-scale convolutional transformations using split, transform, and merge. This architecture is 22 layers deep and has 7 million parameters. ResNet, developed by (He et al., 2016), won the ILSVRC-2015 competition. It was designed as an ultra-deep network that prevents vanishing gradients, unlike previous networks. Several ResNets have been designed with different numbers of layers. ResNet provides the ability to create shortcut links within layers with cross-layer linking, independent of parameter and data. The numbers of layers for ResNet-18 and ResNet-101 are 18 and 101, respectively. Although ResNet has more depth than VGG architecture, it has less computational complexity. Densely connected CNNs, DenseNets, are the next step on the way to keep increasing the depth of DNNs. The DenseNet structure proposed by (Huang et al., 2017) connects each layer to another layer in a feed-forward manner. It simplifies the connectivity pattern between layers introduced in other network architectures. It enables the reuse of the features in the progressive layers and a smaller number of parameters. The depth of the DenseNet-201 network is 201.

The attributes of the DNNs are summarized in Table 4. In this table, the network depth indicates the largest number of sequential convolutional or fully connected layers on a path from the input layer to the output layer. The inputs to all networks are true-color images. Input images to the pre-trained networks are resized to the input image size of each network. The mini-batch size is 32 for all the networks and datasets utilized. For the Varroa dataset, the maximum number of epochs is 3 for the ResNet-18 and VGG-19 networks, 10 for AlexNet, and 5 for the other networks. For the BeeImage dataset, the maximum number of epochs is 5 for DenseNet-201, 10 for AlexNet, and 15 for the other networks. For the Pollen dataset, the maximum number of epochs is 5 for the VGG-16 and VGG-19 networks, 7 for ResNet-18, 10 for AlexNet and DenseNet-201, and 20 for the other networks. Early stopping is used to prevent overfitting during the training. Also, the decrease in the validation loss in parallel with the training loss indicates the prevention of overfitting, too.

4. Results and discussions

The performances of the proposed classification models were evaluated by a comprehensive experimental work. As mentioned earlier, the test parts of the datasets introduced in the previous section were used for the performance evaluation. The experiments were carried out on a workstation equipped with an Intel(R) Xeon(R) E5–2643 v2 3.50 GHz CPU and 128 GB of RAM. The results of the experimental work for each dataset and the comparison against the related works are given in the following subsections.

Table 3
The content of the Pollen dataset.

Class no	Class label	# Images
1	Non-pollen-bearing	345
2	Pollen-bearing	369

4.1. The results for the Varroa dataset

The classification results (accuracy, precision, recall, F1-score, and average classification time per image) of the proposed models for the Varroa dataset are comparatively shown in Table 5, where the best value for each performance metric is indicated in bold.

The models reached accuracies ranging from 0.7066 to 0.9322. Among all models, the highest accuracy was achieved by the transfer learning with the VGG-19 network, whereas SVM with the shallow features of the same network offered the lowest accuracy. The precision values of the models were between 0.7236 and 0.9499. The highest precision was achieved by the transfer learning with the VGG-16 network, while SVM with the shallow features of DenseNet-201 had the lowest precision. The recall values of the models were between 0.8463 and 1.0000. The highest recall was achieved by SVM with the shallow features of DenseNet-201 or ResNet-101 networks, whereas SVM with the deep features of VGG-19 had the lowest precision. The models reached F1-scores ranging from 0.8085 to 0.9534. The highest F1-score was achieved by the transfer learning with the VGG-19 network, while SVM with the shallow features of the same network offered the lowest F1-score.

Among all models, the shortest classification time was 7 ms and achieved by AlexNet, which is the smallest of the utilized pre-trained networks considering the number of layers, and the connections between the layers. Also, the minimum classification time attained by the model utilizing the SVM classifier with the deep features was 22 ms when the ResNet-18 network was preferred. On the other hand, the minimum classification time attained by the model utilizing the SVM classifier and the shallow features was 8 ms when the ResNet-101 was used. Finally, the minimum classification time attained by the model utilizing the SVM classifier with deep+shallow features was 19 ms when AlexNet was used.

An analysis was also made for the classification times of the four models (with each DNN) against their F1-scores in the Varroa dataset. The analysis is illustrated with the plot shown in Fig. 5, where the highest F1-score and the shortest classification time for each model are

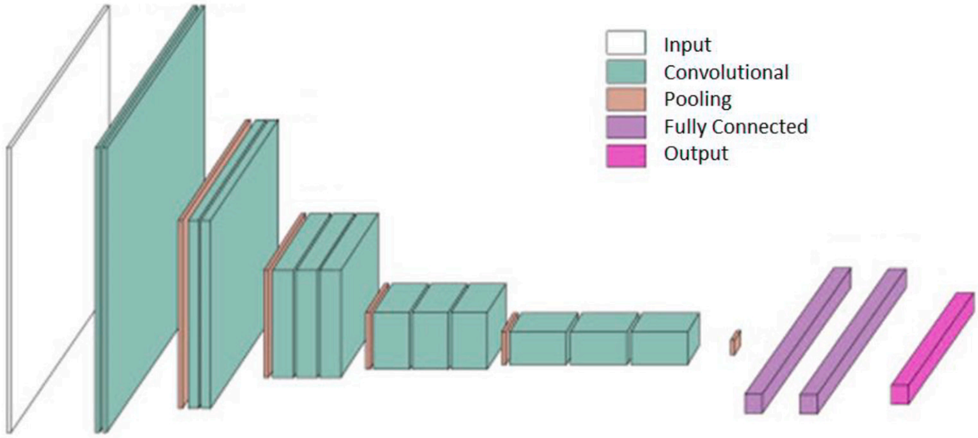

Fig. 4. A typical deep CNN architecture.

Table 4
The attributes of the pre-trained DNNs.

Pre-trained DNN	Depth	Input Image Size
AlexNet	8	227 × 227
DenseNet-201	201	224 × 224
GoogLeNet	22	224 × 224
ResNet-101	101	224 × 224
ResNet-18	18	224 × 224
VGG-16	16	224 × 224
VGG-19	19	224 × 224

explicitly labeled with the corresponding DNN. It should be noted that the models located in the upper left corner of this plot are superior to the others due to relatively higher F1-scores and shorter classification times. In this sense, transfer learning surpassed the other models by achieving F1-scores over 0.92 and classification times less than 75 ms for all DNNs except DenseNet-201. Though SVM with shallow features required short classification times, this model achieved mostly the lowest F1-scores.

4.2. The results for the BeeImage dataset

The classification results (accuracy, precision, recall, F1-score, and average classification time per image) of the proposed models for the BeeImage dataset are comparatively shown in Table 6, where the best value for each performance metric is indicated in bold.

The models reached accuracies ranging from 0.8610 to 0.9820. Among all models, the highest accuracy was achieved by SVM with the shallow or deep+shallow features of the GoogLeNet network, whereas SVM with the shallow features of ResNet-101 offered the lowest accuracy. The precision values of the models were between 0.6482 and 0.9756. The highest precision was achieved by SVM with the shallow features of the VGG-16 network, while SVM with the shallow features of DenseNet-201 had the lowest precision. The recall values of the models were between 0.6925 and 0.9731. The highest recall was achieved by SVM with the deep+shallow features of the GoogLeNet network, whereas SVM with the shallow features of DenseNet-201 had the lowest precision. The models reached F1-scores ranging from 0.6696 to 0.9728.

Table 5
The classification results of the proposed models for the Varroa dataset.

Pre-trained DNN	Method	Accuracy	Precision	Recall	F1-score	Time (ms)
AlexNet	Transfer learning	0.9061	0.9049	0.9724	0.9375	**7**
	SVM with deep features	0.8266	0.8591	0.9096	0.8836	28
	SVM with shallow features	0.7908	0.8279	0.8974	0.8613	14
	SVM with deep+shallow features	0.8283	0.8591	0.9124	0.8850	19
DenseNet-201	Transfer learning	0.9090	0.9393	0.9347	0.9370	216
	SVM with deep features	0.8081	0.8587	0.8796	0.8690	148
	SVM with shallow features	0.7236	0.7236	**1.0000**	0.8396	32
	SVM with deep+shallow features	0.8116	0.8596	0.8840	0.8717	144
GoogLeNet	Transfer learning	0.8976	0.8987	0.9676	0.9318	33
	SVM with deep features	0.7430	0.7986	0.8621	0.8292	35
	SVM with shallow features	0.7256	0.7305	0.9838	0.8384	18
	SVM with deep+shallow features	0.7746	0.7979	0.9221	0.8555	36
ResNet-101	Transfer learning	0.8964	0.9300	0.9266	0.9283	57
	SVM with deep features	0.8058	0.8765	0.8516	0.8638	92
	SVM with shallow features	0.7236	0.7236	**1.0000**	0.8396	8
	SVM with deep+shallow features	0.8099	0.8747	0.8605	0.8675	76
ResNet-18	Transfer learning	0.8888	0.9027	0.9485	0.9251	30
	SVM with deep features	0.7925	0.8374	0.8852	0.8606	22
	SVM with shallow features	0.7708	0.7659	0.9842	0.8614	17
	SVM with deep+shallow features	0.8093	0.8343	0.9189	0.8746	37
VGG-16	Transfer learning	0.9252	**0.9499**	0.9465	0.9482	61
	SVM with deep features	0.8148	0.8768	0.8658	0.8713	141
	SVM with shallow features	0.7336	0.7327	0.9947	0.8438	77
	SVM with deep+shallow features	0.8348	0.8710	0.9059	0.8881	218
VGG-19	Transfer learning	**0.9322**	0.9493	0.9574	**0.9534**	72
	SVM with deep features	0.7972	0.8699	0.8463	0.8580	185
	SVM with shallow features	0.7066	0.7660	0.8560	0.8085	128
	SVM with deep+shallow features	0.8192	0.8730	0.8779	0.8755	275

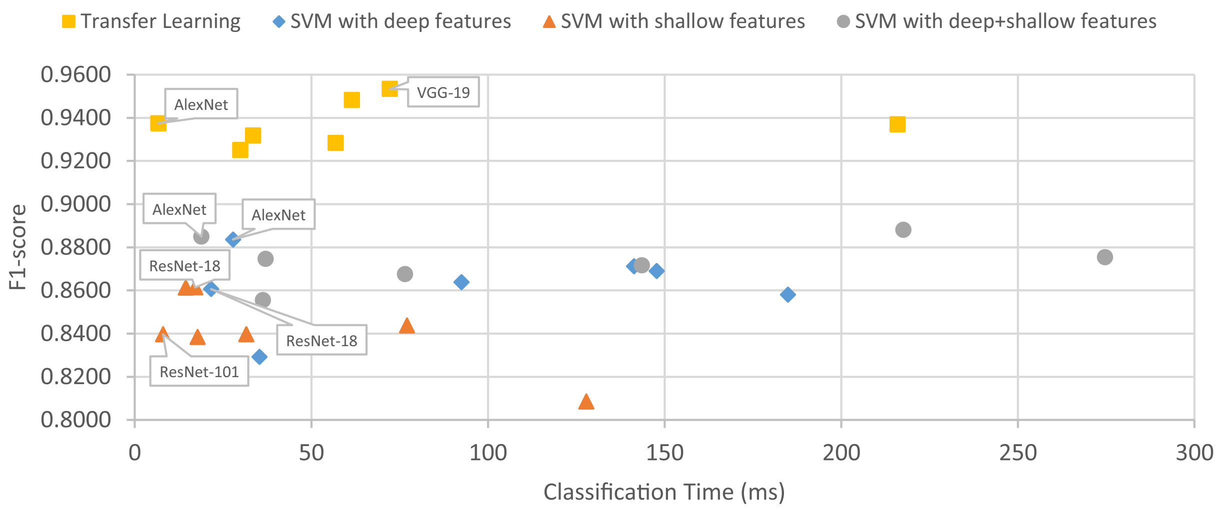

Fig. 5. The average classification times vs. F1-scores of the proposed models for the Varroa dataset.

Table 6
The classification results of the proposed models for the BeeImage dataset.

Pre-trained DNN	Method	Accuracy	Precision	Recall	F1-score	Time (ms)
AlexNet	Transfer learning	0.9781	0.9615	0.9644	0.9630	8
	SVM with deep features	0.9640	0.8115	0.9531	0.8766	44
	SVM with shallow features	0.9768	0.9624	0.9653	0.9638	6
	SVM with deep+shallow features	0.9730	0.8694	0.9610	0.9129	46
DenseNet-201	Transfer learning	0.9691	0.8610	0.9536	0.9049	122
	SVM with deep features	0.9678	0.9046	0.9556	0.9294	141
	SVM with shallow features	0.8906	0.6482	0.6925	0.6696	32
	SVM with deep+shallow features	0.9678	0.9046	0.9556	0.9294	169
GoogLeNet	Transfer learning	0.9640	0.9356	0.9493	0.9424	18
	SVM with deep features	0.9254	0.8751	0.8963	0.8856	27
	SVM with shallow features	**0.9820**	0.9737	0.9719	**0.9728**	10
	SVM with deep+shallow features	**0.9820**	0.9677	**0.9731**	0.9704	36
ResNet-101	Transfer learning	0.9717	0.8718	0.9558	0.9119	94
	SVM with deep features	0.9665	0.8671	0.9498	0.9065	86
	SVM with shallow features	0.8610	0.7048	0.8304	0.7624	**5**
	SVM with deep+shallow features	0.9665	0.8671	0.9502	0.9067	86
ResNet-18	Transfer Learning	0.9743	0.8734	0.9587	0.9140	16
	SVM with deep features	0.9511	0.8808	0.9219	0.9009	22
	SVM with shallow features	0.9653	0.7897	0.7785	0.7840	17
	SVM with deep+shallow features	0.9511	0.8808	0.9219	0.9009	38
VGG-16	Transfer learning	0.9704	0.8741	0.9465	0.9089	68
	SVM with deep features	0.9447	0.7677	0.8639	0.8130	141
	SVM with shallow features	0.9794	**0.9756**	0.9683	0.9719	76
	SVM with deep+shallow features	0.9768	0.9591	0.9678	0.9635	280
VGG-19	Transfer Learning	0.9807	0.9649	0.9201	0.9419	83
	SVM with deep features	0.9472	0.7792	0.8670	0.8207	225
	SVM with shallow features	0.9755	0.9617	0.9309	0.9461	84
	SVM with deep+shallow features	0.9755	0.9181	0.9636	0.9403	373

The highest F1-score was achieved by SVM with the shallow features of the GoogLeNet network, while SVM with the shallow features of DenseNet-201 offered the lowest F1-score.

Among all models, the shortest classification time was 5 ms and achieved by the model utilizing the SVM classifier with the shallow features of the ResNet-101 network. Also, the minimum classification time attained by the model utilizing the SVM classifier and the deep features (of the ResNet-18 network) was 22 ms. Meanwhile, the minimum classification time attained by the transfer learning with the AlexNet network was 8 ms. Finally, the minimum classification time attained by the model utilizing the SVM classifier with deep+shallow features was 36 ms when the GoogLeNet was used.

Similar to the previous subsection, an analysis was also made for the classification times of the four models (with each DNN) against their F1-scores in the BeeImage dataset. The analysis is illustrated with the plot given in Fig. 6, where the highest F1-score and the shortest classification time for each model are explicitly labeled with the corresponding DNN. Here, unlike the Varroa dataset, top-performing models (with certain DNNs) were transfer learning, SVM with shallow features, and SVM with deep+shallow features achieving F1-scores over 0.95 and classification times less than 50 ms. Though SVM with shallow features required relatively shorter classification times (less than 100 ms) than those of the other models, it offered the lowest F1-scores (even less than 0.80) with particular DNNs.

4.3. The results for the Pollen dataset

The classification results (accuracy, precision, recall, F1-score, and average classification time per image) of the proposed models for the Pollen dataset are comparatively shown in Table 7, where the best value for each performance metric is indicated in bold.

The models reached accuracies ranging from 0.8598 to 0.9907.

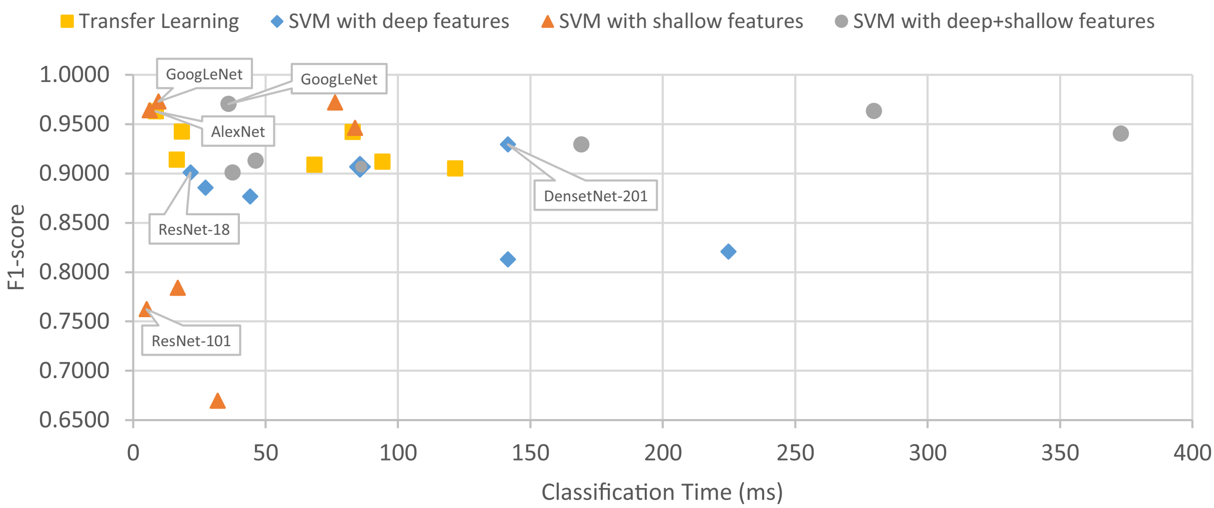

Fig. 6. The average classification times vs. F1-scores of the proposed models for the BeeImage dataset.

Among all models, the highest accuracy was achieved by the transfer learning with the GoogLeNet network, whereas SVM with the shallow features of ResNet-101 offered the lowest accuracy. The precision values of the models were between 0.8364 and 1.0000. The highest precision was achieved by transfer learning with the AlexNet or VGG-16 network, while SVM with the shallow features of ResNet-101 had the lowest precision. The recall values of the models were between 0.7500 and 1.0000. The highest recall was achieved by the transfer learning with the GoogLeNet network, whereas SVM with the shallow features of DenseNet-201 had the lowest precision. The models reached F1-scores ranging from 0.8478 to 0.9905. The highest F1-score was achieved by the transfer learning with the GoogLeNet network, while SVM with the shallow features of DenseNet-201 offered the lowest F1-score.

Among all models, the shortest classification time was 5 ms and attained by the transfer learning with the ResNet-18 network. Also, the minimum classification time attained by the model utilizing the SVM classifier with the deep features was 26 ms when the ResNet-18 network was used. On the other hand, the minimum classification time attained by the models utilizing the SVM classifier and the shallow features was 12 ms when the GoogLeNet and ResNet-101 networks were used. Finally, the minimum classification time attained by the model utilizing the SVM classifier and the deep+shallow features was 28 ms when AlexNet was used.

Similar to the previous subsections, an analysis was also made for the classification times of the four models (with each DNN) against their F1-scores in the Pollen dataset. The analysis is illustrated with the plot given in Fig. 7, where the highest F1-score and the shortest classification time for each model are explicitly labeled with the corresponding DNN. Here, unlike the other two datasets, all four models (with certain DNNs) performed well by achieving F1-scores over 0.95 and classification times less than 50 ms. Though SVM with shallow features required relatively shorter classification times (less than 100 ms) than those of the other

Table 7
The classification results of the proposed models for the Pollen dataset.

Pre-trained DNN	Method	Accuracy	Precision	Recall	F1-score	Time (ms)
AlexNet	Transfer learning	0.9813	**1.0000**	0.9615	0.9804	14
	SVM with deep features	0.9626	0.9800	0.9423	0.9608	30
	SVM with shallow features	0.9720	0.9623	0.9808	0.9714	20
	SVM with deep+shallow features	0.9626	0.9800	0.9423	0.9608	28
DenseNet-201	Transfer learning	0.9439	0.9600	0.9231	0.9412	71
	SVM with deep features	0.9533	0.9608	0.9423	0.9515	175
	SVM with shallow features	0.8692	0.9750	0.7500	0.8478	41
	SVM with deep+shallow features	0.9533	0.9608	0.9423	0.9515	193
GoogLeNet	Transfer learning	**0.9907**	0.9811	**1.0000**	**0.9905**	6
	SVM with deep features	0.9346	0.9245	0.9423	0.9333	33
	SVM with shallow features	0.9439	0.9259	0.9615	0.9434	12
	SVM with deep+shallow features	0.9439	0.9107	0.9808	0.9444	40
ResNet-101	Transfer learning	0.9252	0.9074	0.9423	0.9245	29
	SVM with deep features	0.9346	0.9245	0.9423	0.9333	94
	SVM with shallow features	0.8598	0.8364	0.8846	0.8598	12
	SVM with deep+shallow features	0.9252	0.9074	0.9423	0.9245	101
ResNet-18	Transfer learning	0.9346	0.9412	0.9231	0.9320	**5**
	SVM with deep features	0.9065	0.8750	0.9423	0.9074	26
	SVM with shallow features	0.9159	0.9388	0.8846	0.9109	18
	SVM with deep+shallow features	0.9065	0.8750	0.9423	0.9074	41
VGG-16	Transfer learning	0.9813	**1.0000**	0.9615	0.9804	14
	SVM with deep features	0.9626	0.9800	0.9423	0.9608	144
	SVM with shallow features	0.9720	0.9623	0.9808	0.9714	135
	SVM with deep+shallow features	0.9626	0.9800	0.9423	0.9608	289
VGG-19	Transfer learning	0.9813	0.9808	0.9808	0.9808	16
	SVM with deep features	0.9252	0.9583	0.8846	0.9200	177
	SVM with shallow features	0.9626	0.9444	0.9808	0.9623	158
	SVM with deep+shallow features	0.9533	0.9608	0.9423	0.9515	333

models, it offered the lowest F1-scores (even less than 0.80) with particular DNNs.

4.4. Comparison with the related work

In addition to analyzing its performance, the proposed models were also compared with the previous works on beehive monitoring, considering several aspects, including dataset type, dataset size, number of classes, types of features, monitoring method, application field, classification performance, and processing time. The comparison is summarized in Table 8. In the table, the highest values of each performance metric were given for the proposed models.

While one of the previous works (Kulyukin et al., 2018) employed audio signals for beehive monitoring, the others used images or videos for this purpose. In the meantime, our work made use of images to monitor beehives. Besides, most of the previous works studied on just a single dataset whereas the proposed work used three different datasets with different content. Also, those datasets were among the largest ones.

Most of the previous works (Babic et al., 2016; Bjerge et al., 2019; Kulyukin et al., 2018; Schurischuster and Kampel, 2020a; Yang and Collins, 2019) considered just 2 or 3 classes, only a few of them (Alves et al., 2020; Uzen et al., 2019) analyzed the cases with more than 3 classes that makes the problem harder to solve. On the other hand, the proposed work considered both 2-class and 6-class problems so that the performances of the proposed models were tested, and validated under different conditions.

As shown in Table 8, some of the previous works (Babic et al., 2016; Bjerge et al., 2019; Kulyukin et al., 2018; Yang and Collins, 2019) used traditional image processing, feature extraction, and/or classification methods, the others (Alves et al., 2020; Schurischuster and Kampel, 2020a; Uzen et al., 2019) only made use of deep learning algorithms. Meanwhile, the proposed work utilized deep learning, transfer learning, and traditional classification algorithms together.

One of the previous works (Babic et al., 2016) offered a solution to detect, and classify pollen-bearing honey bees. Another work (Kulyukin et al., 2018) tried to identify stressors from audio recordings at beehives. Pollen sac detection and measurement were the focus of another work (Yang and Collins, 2019) whereas the detection and classification of honey bee comb cells were handled in (Alves et al., 2020). One of the works classified several conditions at beehives such as ant issues, Varroa, small hive beetles, and hive robberies (Uzen et al., 2019). A few works (Bjerge et al., 2019; Schurischuster and Kampel, 2020a) addressed the detection of parasites at beehives. Meanwhile, the proposed work addressed the recognition of different conditions, such as healthy bees, pollen-bearing bees, and certain abnormalities, such as Varroa parasites, ant problems, hive robberies, and small hive beetles.

Considering the performance metrics, most of the previous works provided only accuracy values, whereas the proposed work offered several metrics, including not only accuracy but also precision, recall, and F1-score. Based on all these metrics, the proposed models offered a significant performance. Moreover, the performance is even superior to those of the previous works using the same datasets.

Also, as described earlier, our work offered a detailed classification time vs. F1-score analysis of the proposed models so that the pros and cons of the models can be observed considering both classification performance and processing time at the same time. On the other hand, the previous works offered either no or just a limited timing analysis of their models.

5. Conclusions

While honey bees have an important role in the ecosystem, they face many problems, including parasites, ants, hive beetles, and robberies. Some of these problems could even lead to the collapse of colonies. Unfortunately, most of the traditional methods for solving these problems are very time-consuming and not effective at all. Therefore, there is a need for methods that would enable fast and effective beehive monitoring.

In this paper, four new image classification models were proposed to recognize different conditions, as well as abnormalities in beehives, fast and accurately. The models mainly employ transfer learning with pre-trained DNNs and also an SVM classifier with deep, shallow, and deep+shallow features extracted from these DNNs. Three benchmark datasets, consisting of honey bee images for various conditions, were used to train and evaluate the models. The results of a thorough experimental work revealed that the proposed models can recognize different conditions as well as abnormalities in beehives with significantly high accuracy and short classification time.

Considering all three datasets together, there is no specific DNN that outperforms the others for all of the proposed models in terms of different performance metrics. In the Varroa dataset, transfer learning with all DNNs surpassed the other three models. In the BeeImage dataset, top-performing models were transfer learning, SVM with shallow features, and SVM with deep+shallow features with certain DNNs such as GoogleNet, AlexNet, and VGG-16. In the Pollen dataset, all four models performed well by achieving F1-scores over 0.95 with all DNNs except the ResNet networks. Since the pre-trained networks were trained with certain images in particular domains, it is obvious that the

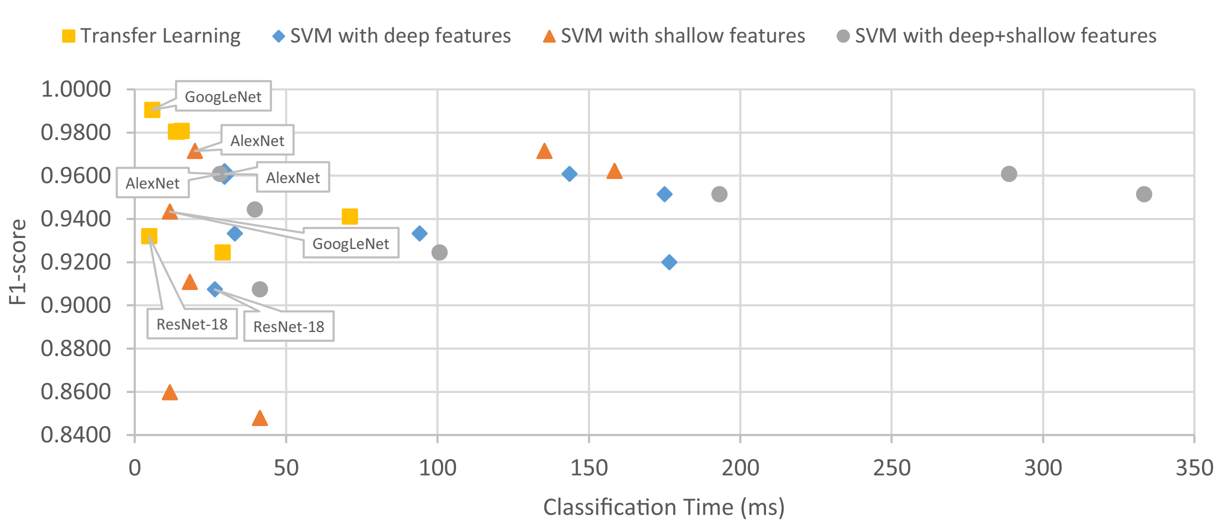

Fig. 7. The average classification times vs. F1-scores of the proposed models for the Pollen dataset.

Table 8
The comparison of the proposed work with the literature.

References	Dataset Type and Size	# Classes (Class Labels)	Application	Features	Classification Method	Performance	Timing Analysis
(Babic et al., 2016)	454 images	2 (pollen-bearing honey bees, honey bees that do not have pollen load)	Detection and classification of pollen-bearing honey bees	Color variance, eccentricity features	Nearest Mean Classifier	Acc: 0.887	Available
(Kulyukin et al., 2018)	BUZZ1: 10,260 audio samples	3 (bee buzzing, cricket chirping, ambient noise)	Classification of the audio samples from the microphones deployed at beehives	CNN features 40 MFCCs, 12 chroma coefficients, 128 melspectrogram coefficients, 7 spectral contrast coefficients, 6 tonnetz coefficients	CNN Logistic Regression k-NN SVM Random Forest	Acc: 0.952 Acc: 0.946 Acc: 0.856 Acc: 0.839 Acc: 0.931	Available
	BUZZ2: 12,914 audio samples	3 (bee buzzing, cricket chirping, ambient noise)	Classification of the audio samples from the microphones deployed at beehives	CNN features 40 MFCCs, 12 chroma coefficients, 128 melspectrogram coefficients, 7 spectral contrast coefficients, 6 tonnetz coefficients	CNN Logistic Regression k-NN SVM Random Forest	Acc: 0.965 Acc: 0.685 Acc: 0.374 Acc: 0.566 Acc: 0.658	
(Rodriguez et al., 2018)	Pollen dataset: 710 images	2 (pollen-bearing honey bees, non-pollen-bearing honey bees)	Classification of pollen-bearing honey bees	Three feature map images: RGB images, Color and Gaussian CNN features	k-NN Naive Bayes SVM CNN VGG-16 VGG-19 ResNet-50	Acc: 0.874 Acc: 0.795 Acc: 0.912 Acc: 0.964 Acc: 0.872 Acc: 0.902 Acc: 0.617	Available
(Bjerge et al., 2019)	1,775 images	2 (healthy, infected)	Classification of honey bee status	SIFT, SURF, ORB for bee detection	CNN	F1: 0.910	N/A
(Yang and Collins, 2019)	1,400 images	2 (pollen bees, non-pollen bees)	Pollen sac detection and measurement	CNN features Conventional image processing and statistical analysis	Faster RCNN with VGG-16 Two Lines Model	Measurement Error: 0.07 Measurement Error: 0.33	N/A
(Uzen et al., 2019)	BeeImage dataset: 5,172 images	6 (ant problems, few Varroa-hive beetles, healthy, hive being robbed, missing queen, Varroa-small hive beetles)	Classification of honey bee status	CNN features	Custom CNN architectures	Acc: 0.924	Available
(Alves et al., 2020)	1,202 images	7 (eggs, larvae, capped brood, pollen, nectar, honey, and others)	Detection and classification of honey bee comb cells	CNN features	DenseNet, InceptionResNetV2, InceptionV3; MobileNet; MobileNetV2; NasNet; NasNetMobile; ResNet-50; VGG-16, VGG-19, Xception	F1: 0.943	Available
(Schurischuster and Kampel, 2020a)	Varroa dataset: 13,509 images	2 (healthy, infected)	Classification of honey bee status	CNN features	AlexNet, ResNet-101, DeepLabv3 (ResNet-50, ResNet-101)	F1: 0.950 Acc: 0.908	N/A
Proposed work	Varroa dataset: 13,507 images	2 (healthy, infected)	Classification of honey bee status	Deep features	SVM	Acc: 0.8266 Pre: 0.8768 Rec: 0.9096 F1: 0.8836	Available
				Shallow features	SVM	Acc: 0.7908 Pre: 0.8279 Rec: 1.0000 F1: 0.8614	
				Deep+Shallow features	SVM	Acc: 0.8348 Pre: 0.8747 Rec: 0.9221 F1: 0.8881	
				DNN features	Transfer Learning with pre-trained DNN	Acc: 0.9322 Pre: 0.9499 Rec: 0.9724 F1: 0.9534	
	BeeImage dataset: 5,172 images	6 (ant problems, few Varroa-hive beetles, healthy, hive being robbed, missing queen, Varroa-small hive beetles)	Classification of honey bee status	Deep features	SVM	Acc: 0.9678 Pre: 0.9046 Rec: 0.9556 F1: 0.9294	
				Shallow features	SVM	Acc: 0.9820 Pre: 0.9756 Rec: 0.9719 F1: 0.9728	

(*continued on next page*)

Table 8 (*continued*)

References	Dataset Type and Size	# Classes (Class Labels)	Application	Features	Classification Method	Performance	Timing Analysis
				Deep+Shallow features	SVM	Acc: 0.9820 Pre: 0.9677 Rec: 0.9731 F1: 0.9704	
				DNN features	Transfer Learning with pre-trained DNN	Acc: 0.9807 Pre: 0.9649 Rec: 0.9644 F1: 0.9630	
	Pollen dataset: 714 images	2 (pollen-bearing, non pollen-bearing)	Classification of honey bee status	Deep features	SVM	Acc: 0.9626 Pre: 0.9800 Rec: 0.9423 F1: 0.9608	
				Shallow features	SVM	Acc: 0.9720 Pre: 0.9750 Rec: 0.9808 F1: 0.9714	
				Deep+Shallow features	SVM	Acc: 0.9626 Pre: 0.9800 Rec: 0.9808 F1: 0.9608	
				DNN features	Transfer Learning with pre-trained DNN	Acc: 0.9907 Pre: 1.0000 Rec: 1.0000 F1: 0.9905	

architectures of the networks, as well as the similarities or dissimilarities of images from new domains (in terms of different aspects such as shape, color, statistical features, and so on) to the ones in those particular domains, have direct impact (either positive or negative) on the classification performance of the new images via the pre-trained networks. This is why different networks offer varying performances for different datasets.

Thanks to their recognition performance, the proposed models can contribute to the safety of honey bees and efficient beekeeping. Also, the models can be easily adapted to real-time smart agriculture and beekeeping applications because of their significantly short classification times. Adaptation of these models for different conditions in beehives and analyzing the performances of different DNNs remain interesting future works.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Declaration of Competing Interest

None.

References

Alves, T.S., Pinto, M.A., Ventura, P., Neves, C.J., Biron, D.G., Junior, A.C., De Paula Filho, P.L., Rodrigues, P.J., 2020. Automatic detection and classification of honey bee comb cells using deep learning. Comput. Electron. Agric. 170, 105244. https://doi.org/10.1016/j.compag.2020.105244.

Andrews, E., 2019. To save the bees or not to save the bees: honey bee health in the Anthropocene. Agric. Hum. Values 36, 891–902. https://doi.org/10.1007/s10460-019-09946-x.

Babic, Z., Pilipovic, R., Risojevic, V., Mirjanic, G., 2016. Pollen bearing honey bee detection in hive entrance video recorded by remote embedded system for pollination monitoring. ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci. 3, 51–57. https://doi.org/10.5194/isprsannals-III-7-51-2016.

Beaurepaire, A., Piot, N., Doublet, V., Antunez, K., Campbell, E., Chantawannakul, P., Chejanovsky, N., Gajda, A., Heerman, M., Panziera, D., Smagghe, G., Yanez, O., de Miranda, J.R., Dalmon, A., 2020. Diversity and global distribution of viruses of the western honey bee, Apis mellifera. Insects 11, 239. https://doi.org/10.3390/insects11040239.

Bjerge, K., Frigaard, C.E., Mikkelsen, P.H., Nielsen, T.H., Misbih, M., Kryger, P., 2019. A computer vision system to monitor the infestation level of Varroa destructor in a honeybee colony. Comput. Electron. Agric. 164, 104898. https://doi.org/10.1016/j.compag.2019.104898.

Braga, A.R., Gomes, D.G., Freitas, B.M., Cazier, J.A., 2020. A cluster-classification method for accurate mining of seasonal honey bee patterns. Ecol. Inform. 59, 101107. https://doi.org/10.1016/j.ecoinf.2020.101107.

Buschbacher, K., Ahrens, D., Espeland, M., Steinhage, V., 2020. Image-based species identification of wild bees using convolutional neural networks. Ecol. Inform. 55, 101017. https://doi.org/10.1016/j.ecoinf.2019.101017.

Chen, J., Chen, J., Zhang, D., Sun, Y., Nanehkaran, Y.A., 2020. Using deep transfer learning for image-based plant disease identification. Comput. Electron. Agric. 173, 105393. https://doi.org/10.1016/j.compag.2020.105393.

Dembski, J., Szymański, J., 2019. Bees Detection on Images: Study of Different Color Models for Neural Networks, International Conference On Distributed Computing and Internet Technology. Springer, pp. 295–308. https://doi.org/10.1007/978-3-030-05366-6_25.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Li, F.F., 2009. ImageNet: a large-scale hierarchical image database. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 2009, 248–255. https://doi.org/10.1109/cvpr.2009.5206848.

Engel, M.S., 1999. The taxonomy of recent and fossil honey bees (Hymenoptera: Apidae; Apis). J. Hymenopt. Res. 8, 165–196.

Espejo-Garcia, B., Mylonas, N., Athanasakos, L., Fountas, S., Vasilakoglou, I., 2020. Towards weeds identification assistance through transfer learning. Comput. Electron. Agric. 171, 105306. https://doi.org/10.1016/j.compag.2020.105306.

Han, D., Liu, Q., Fan, W., 2018. A new image classification method using CNN transfer learning and web data augmentation. Expert Syst. Appl. 95, 43–56.

He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J., 2016. Deep residual learning for image recognition. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 2016, 770–778. https://doi.org/10.1016/j.eswa.2017.11.028.

Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q., 2017. Densely connected convolutional networks. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 2017, 2261–2269. https://doi.org/10.1109/Cvpr.2017.243.

Kaplan Berkaya, S., Ak Sivrikoz, I., Gunal, S., 2020. Classification models for SPECT myocardial perfusion imaging. Comput. Biol. Med. 123, 103893. https://doi.org/10.1016/j.compbiomed.2020.103893.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2017. ImageNet classification with deep convolutional neural networks. Commun. ACM 60, 84–90. https://doi.org/10.1145/3065386.

Kulyukin, V., Mukherjee, S., Amlathe, P., 2018. Toward audio beehive monitoring: Deep learning vs. standard machine learning in classifying beehive audio samples. Appl. Sci. 8, 1573. https://doi.org/10.3390/app8091573.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86, 2278–2324. https://doi.org/10.1109/5.726791.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444. https://doi.org/10.1038/nature14539.

Margapuri, V., Lavezzi, G., Stewart, R., Wagner, D., 2020. Bombus Species Image Classification. arXiv preprint. arXiv:2006.11374.

Martineau, M., Conte, D., Raveaux, R., Arnault, I., Munier, D., Venturini, G., 2017. A survey on image-based insect classification. Pattern Recogn. 65, 273–284. https://doi.org/10.1016/j.patcog.2016.12.020.

Mondet, F., Beaurepaire, A., McAfee, A., Locke, B., Alaux, C., Blanchard, S., Danka, B., Yves, L.C., 2020. Honey bee survival mechanisms against the parasite Varroa

destructor: a systematic review of phenotypic and genomic research efforts. Int. J. Parasitol. 50, 433–447. https://doi.org/10.1016/j.ijpara.2020.03.005.
Mukherjee, S., 2020. Acquisition, Processing, and Analysis of Video, Audio and Meteorological Data in Multi-Sensor Electronic Beehive Monitoring. Utah State University.
Nguyen, L.D., Lin, D., Lin, Z., Cao, J., 2018. Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation. IEEE Int. Sympos. Circuits Syst. (ISCAS) 2018, 1–5. https://doi.org/10.1109/ISCAS.2018.8351550.
Noël, A., Le Conte, Y., Mondet, F., 2020. Varroa destructor: how does it harm Apis mellifera honey bees and what can be done about it? Emerg. Topics Life Sci. 4, 45–57. https://doi.org/10.1042/ETLS20190125.
Nouvian, M., Reinhard, J., Giurfa, M., 2016. The defensive response of the honeybee Apis mellifera. J. Exp. Biol. 219, 3505–3517. https://doi.org/10.1242/jeb.143016.
Ogihara, M.H., Stoic, M., Morimoto, N., Yoshiyama, M., Kimura, K., 2020a. A convenient method for detection of Varroa destructor (Acari: Varroidae) using roasted soybean flour. Appl. Entomol. Zool. 55, 429–433. https://doi.org/10.1007/s13355-020-00698-3.
Ogihara, M.H., Yoshiyama, M., Morimoto, N., Kimura, K., 2020b. Dominant honeybee colony infestation by Varroa destructor (Acari: Varroidae) K haplotype in Japan. Appl. Entomol. Zool. 1–9. https://doi.org/10.1007/s13355-020-00667-w.
Ramsey, S.D., Ochoa, R., Bauchan, G., Gulbronson, C., Mowery, J.D., Cohen, A., Lim, D., Joklik, J., Cicero, J.M., Ellis, J.D., 2019. Varroa destructor feeds primarily on honey bee fat body tissue and not hemolymph. Proc. Natl. Acad. Sci. 116, 1792–1801. https://doi.org/10.1073/pnas.1818371116.
Rawat, W., Wang, Z., 2017. Deep convolutional neural networks for image classification: a comprehensive review. Neural Comput. 29, 2352–2449. https://doi.org/10.1162/neco_a_00990.
Rodriguez, I.F., Megret, R., Acuna, E., Agosto-Rivera, J.L., Giray, T., 2018. Recognition of pollen-bearing bees from video using convolutional neural network. IEEE Winter Conf. Appl. Comput. Vis. (WACV) 2018, 314–322. https://doi.org/10.1109/WACV.2018.00041.
Santana, F.S., Costa, A.H.R., Truzzi, F.S., Silva, F.L., Santos, S.L., Francoy, T.M., Saraiva, A.M., 2014. A reference process for automating bee species identification based on wing images and digital image processing. Ecol. Inform. 24, 248–260. https://doi.org/10.1016/j.ecoinf.2013.12.001.
Sarkar, D., Bali, R., Ghosh, T., 2018. Hands-on Transfer Learning with Python: Implement Advanced Deep Learning and Neural Network Models Using TensorFlow and Keras. Packt Publishing Ltd.
Schafer, M.O., Cardaio, I., Cilia, G., Cornelissen, B., Crailsheim, K., Formato, G., Lawrence, A.K., Le Conte, Y., Mutinelli, F., Nanetti, A., Rivera-Gomis, J., Teepe, A., Neumann, P., 2019. How to slow the global spread of small hive beetles, Aethina tumida. Biol. Invasions 21, 1451–1459. https://doi.org/10.1007/s10530-019-01917-x.
Schmidhuber, J., 2015. Deep learning in neural networks: an overview. Neural Netw. 61, 85–117. https://doi.org/10.1016/j.neunet.2014.09.003.
Schurischuster, S., Kampel, M., 2020a. Image-Based Classification of Honeybees, 2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA), pp. 1–6. https://doi.org/10.1109/IPTA50016.2020.9286673.
Schurischuster, S., Kampel, M., 2020b. VarroaDataset Zenodo. https://doi.org/10.5281/zenodo.4085043.
Schurischuster, S., Remeseiro, B., Radeva, P., Kampel, M., 2018. A Preliminary Study of Image Analysis for Parasite Detection on Honey Bees, International Conference Image Analysis and Recognition. Springer, pp. 465–473. https://doi.org/10.1007/978-3-319-93000-8_52.
Simonyan, K., Zisserman, A., 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint. arXiv:1409.1556.
Sun, X., Lv, M., 2019. Facial expression recognition based on a hybrid model combining deep and shallow features. Cogn. Comput. 11, 587–597. https://doi.org/10.1007/s12559-019-09654-y.
Szegedy, C., Liu, W., Jia, Y.Q., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 2015, 1–9.
Theodoridis, S., Koutroumbas, K., 2009. Pattern Recognition, 4th ed. Academic Press.
Traynor, K.S., Mondet, F., de Miranda, J.R., Techer, M., Kowallik, V., Oddie, M.A., Chantawannakul, P., McAfee, A., 2020. Varroa destructor: a complex parasite, crippling honey bees worldwide. Trends Parasitol. 36, 592–606. https://doi.org/10.1016/j.pt.2020.04.004.
Uzen, H., Yeroglu, C., Hanbay, D., 2019. Development of CNN architecture for honey bees disease condition. Int. Artif. Intell. Data Process. Sympos. (IDAP) 2019, 1–5. https://doi.org/10.1109/IDAP.2019.8875886.
von Zuben, L.G., Schorkopf, D.L.P., Elias, L.G., Vaz, A.L.L., Favaris, A.P., Clososki, G.C., Bento, J.M.S., Nunes, T.M., 2016. Interspecific chemical communication in raids of the robber bee Lestrimelitta limao. Insect. Soc. 63, 339–347. https://doi.org/10.1007/s00040-016-0474-2.
Yang, C.R., 2018. The Use of Video to Detect and Measure Pollen on Bees Entering a Hive. Auckland University of Technology.
Yang, J., 2019. The Beeimage Dataset: Annotated Honey Bee Images. https://www.kaggle.com/jenny18/honey-bee-annotated-images.
Yang, C., Collins, J., 2019. Deep learning for pollen sac detection and measurement on honeybee monitoring video. Int. Conf. Image Vis. Comput. New Zealand (IVCNZ) 2019, 1–6. https://doi.org/10.1109/IVCNZ48456.2019.8961011.
Zhao, Y., Deng, G., Zhang, L., Di, N., Jiang, X., Li, Z., 2021. Based investigate of beehive sound to detect air pollutants by machine learning. Ecol. Inform. 61, 101246. https://doi.org/10.1016/j.ecoinf.2021.101246.