A Preliminary Study of Image Analysis for Parasite Detection on Honey Bees

Stefan Schurischuster[1], Beatriz Remeseiro[2], Petia Radeva[3], and Martin Kampel[1]

[1] Computer Vision Lab, Vienna University of Technology (Austria)
schurischuster@caa.tuwien.ac.at, martin.kampel@tuwien.ac.at
[2] Department of Computer Science, Universidad de Oviedo (Spain)
bremeseiro@uniovi.es
[3] Departament de Matemàtiques i Informàtica, Universitat de Barcelona (Spain)
petia.ivanova@ub.edu

Abstract. Varroa destructor is a parasite harming bee colonies. As the worldwide bee population is in danger, beekeepers as well as researchers are looking for methods to monitor the health of bee hives. In this context, we present a preliminary study to detect parasites on bee videos by means of image analysis and machine learning techniques. For this purpose, each video frame is analyzed individually to extract bee image patches, which are then processed to compute image descriptors and finally classified into mite and no mite bees. The experimental results demonstrated the adequacy of the proposed method, which will be a perfect stepping stone for a further bee monitoring system.

Keywords: honey bees, varroa destructor, bee localization, mite characterization, parasite detection

1 Introduction

For almost a decade now, it is a well-known fact that honey bee (*apis mellifera*) population is reducing in a global scale. Beekeepers and those in favor of honey bees are at the mercy of this phenomenon. At the time of this work, there are no known solutions to stopping this paradox. This is mainly due to the fact that the reasons are manifold and ambiguous [4].

Beekeepers and researchers are facing the problem by extending the health surveillance of their bee hives. This includes time consuming manual tests to control the great amount of parasites that bees are coping with [6]. These manual processes can be performed in different ways, such as sampling living bees using alcohol or powder sugar or opening brood cells to check for mites, and have a number of drawbacks. First, they rely completely on statistical projections to estimate the parasite load of a bee hive. To get statistically accurate results for each inspection, a minimum of 300 bees have to be tested per hive. Second, most accurate procedures are invasive in the sense that bees die during the monitoring procedures which adds additional strain to the colonies.

The idea of automatically monitoring the entrance of apiaries for flight estimations dates back over several decades [7]. This was long before computer enhanced monitoring became part of a beekeeper's life. First micro-processor monitoring was presented by Struye et al. [12], who used small tunnels equipped with infrared sensors to successfully count the number of bees flying in and out.

Chiron et al. [3] stated the need for observation of bees to monitor abnormal behavior, which can be used to diagnose the health state of a hive. In particular, they are interested in the number of bees and trajectories, and used a stereo camera system to capture video footage. The company Keltronixinc is marketing bee monitoring as a service [1], by using a system that consists of a camera and a computer, both setup at the bee hive. The camera is looking at the bee hive and monitors the bee activity with a tracking algorithm, allowing to distinguish between healthy and not healthy hives. A recent sensor study for visual observation of honey bees was presented in [10], in which the camera system proposed can be mounted at the entrance of a standard honey bee hive.

With the ultimate goal of automating the parasite monitoring, this paper presents a novelty approach for varroa mite detection on honey bees (see Fig. 1). The proposed framework makes three important contributions: (1) it includes a foreground detection approach to localize bees on video frames, (2) it provides a pipeline system susceptible to be applied to other types of bee parasites, and (3) it allows to reliably detect varroa mites with maximum accuracy over 80%.

The remainder of this manuscript is organized as follows: Section 2 presents the bee dataset and the proposed methods for parasite detection, Section 3 includes the experimentation carried out and the validation results, and Section 4 closes with the conclusions and future lines of research.

2 Materials and Methods

2.1 Data Collection

Videos of honey bees, when they are entering or leaving the hive, are used as input data in this research. The acquisition of the videos was carried out by a camera system that can be mounted at the entrance of standard bee hives [10]. Figure 1 (right) illustrates a representative frame of these videos.

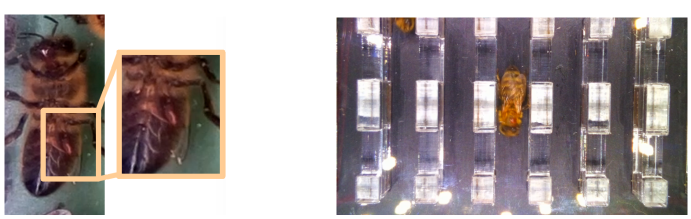

Fig. 1. Representative honey bee in which a varroa mite appears highlighted (left), and video frame of bees entering or leaving the hive (right).

The main properties of these videos are: (1) the camera position and focus are fixed creating a static environment, (2) the background is also static due to the tunnel setup that includes artificial lightning to ensure constant filming conditions, (3) the tunnel is covered with a dark adhesive foil to minimize reflections and create a homogeneous background.

2.2 Video Frame Processing

Once the videos are recorded, the next step entails processing them frame by frame. The target here is to detect the foreground first, and then to extract image patches containing individual bees. Figure 2 shows the different steps for video frame processing, which are following explained in depth.

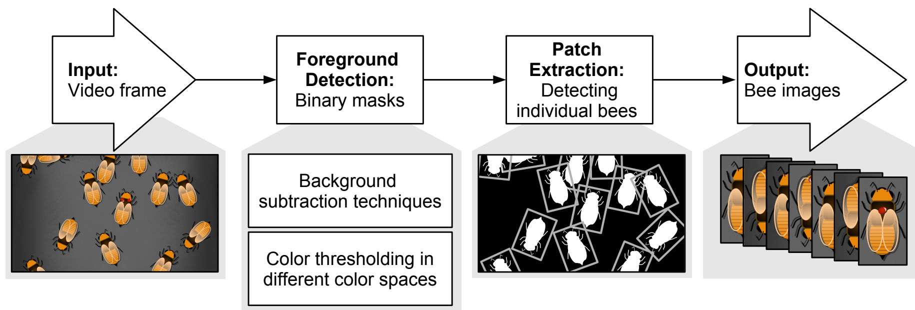

Fig. 2. Workflow for video frame processing with the different proposed methods.

Foreground Detection. This task consists in separating the foreground from the background in each single video frame, getting a trade-off between noise and honey bee details. The foreground is defined as all non-static *regions of interest* (ROIs) in the image, being the rest of the regions part of the background. In the problem at hand, the ROIs are the honey bees passing through the tunnels. From a technical point of view, the result of this step is a binary image serving as a mask. This mask assigns to each pixel in the original image (video frame) one of the binary values depending if the pixel belongs to a ROI or not.

Taking into account the properties of the input data, especially the static camera setup, the following methods were considered:

- **Background subtraction** [11]. The idea is to subtract the intensity values of the image with the intensities of the background, and then classify each pixel as foreground if it is greater than a proposed threshold, t. Therefore, the binary image, M_i of the i-th frame, F_i is computed as: $M_i = |F_i - B| > t$, where B is the background image. Two different approaches were considered to calculate the background image, B. The first one uses the median image over the complete video as a model for the background. The second one calculates the background using Gaussian Mixture Models (GMMs), being the idea to model each pixel intensity value as a linear combination of Gaussian distributions calculated over the histogram of the image.

– **Color thresholding.** This approach uses a simple thresholding in different color spaces, both individually and combined. The threshold defines a region in a given color space, in such a way that all the pixels that fall into this region are treated as foreground, whereas the rest is background. Raw image data is organized in the RGB format, which addresses the color of each pixel as a mix of the three base colors: red, green, and blue. When looking at color as a potential feature, the RGB format is not always optimal, mainly because the brightness information of each pixel is directly linked to the color information. For this reason, two color models that separate color from brightness have been also considered: CIELab and HSV.

Patch Extraction. Since the input of the subsequently image classification step is one single foreground object, bounding boxes with individual bees need to be extracted from the foreground previously detected. To this end, the foreground image is processed using connected component analysis to determine the biggest component and its direction, whilst all other components are removed. The procedure is as follows: (1) find the boundaries of segmented bees, (2) calculate the bounding boxes of individual bees and extract images from them, and (3) post-process the patch images to remove other bees and rotate if needed.

2.3 Image Classification

In this stage, image patches previously extracted from video frames are used as input data. A set of features is extracted from each single patch to create an image descriptor that finally feeds a classifier in order to categorize it into *mite* or *no mite*. Figure 3 shows the different steps, which are subsequently described.

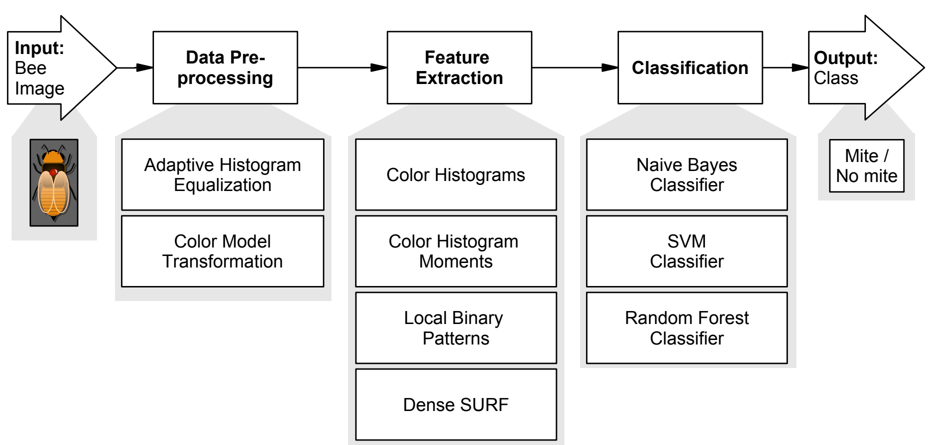

Fig. 3. Workflow for image classification with the different proposed methods.

Data pre-processing. In order to improve the contrast of input images, Contrast Limited Adaptive Equalization (CLAHE) [14] has been considered. This method operates on small regions of the image, by applying the contrast transform function to each region individually, rather than to the entire image.

Taking into account the distinctive color of varroa mites, other color spaces apart from RGB have been analyzed. So, the pre-processing step also includes the transformation of input images from RGB to CIELab and HSV.

Feature extraction. Varroa mites are parasites with a button shape and a reddish-brown color (see Fig. 1). In order to detect them in bee images, different feature extraction methods have been considered:

- **Color histograms.** Binned color histograms of the input images in the different color spaces have been computed. Note that histograms for each color component have up to 256 bins.
- **Color histogram moments.** In addition to using color histograms as feature vectors, a few statistical measures have been computed from them and used as features: mean, standard deviation, skewness, and kurtosis.
- **Local binary patterns** [8]. This technique allows to compute a local representation of texture. It partitions input images into non-overlapping regions, and then it calculates the LBP value for the center pixel of each region.
- **Dense SURF** [2]. Inspired in the popular scale-invariant feature transform (SIFT) descriptor, it also allows to compute local features but being faster and more robust against image transformations.

Classification. The last step consists in classifying the input image, by means of its feature vector, into one of the two classes considered: *mite* or *no mite*. Three popular classifiers were selected aiming to provide different approaches of the learning process [5]: naive Bayes, based on the Bayesian theorem; support vector machine (SVM), based on the statistical learning theory; and random forest, a combination of decision trees.

3 Experimental Results

This section presents the evaluation of the proposed method for parasite detection on honey bees, including the results for video processing and image classification. Note that the source code was implemented in Python, making use of the OpenCV[4] library for image analysis and Scikit-learn[5] for machine learning.

3.1 Results for Video Frame Processing

The first experiments were designed to qualitatively evaluate the detection of individual bees on video frames. For this purpose, a total of 12 videos that

[4] https://opencv.org/
[5] http://scikit-learn.org/

includes confirmed mites were acquired as described in Section 2.1. The videos have a resolution of 1920×1080 pixels in RGB, with a frame rate of 30 fps.

As explained in Section 2.2, different methods for foreground detection have been considered. Figure 4 illustrates the behavior of each one as applied to a representative video frame. As can be observed, the background subtraction with median provides poorer results, since it allows to detect the bee with very low precision. As opposite, background subtraction with GMMs as well as color thresholding (using CIELab or HSV) provide much better results. They allow to detect the whole bee, although some noise appears as foreground. The combination of these three methods to calculate the foreground leads to a more robust approach, with a good trade-off between noise and honey bee details.

Fig. 4. Comparing different foreground detection methods applied to the video frame of Fig. 1. From left to right: background subtraction with median, background subtraction with GMMs, color thresholding with HSV, color thresholding with CIELab, and HSV-CIELab color thresholding combined with GMM background subtraction.

3.2 Results for Image Classification

After processing the 12 videos acquired from bee hives, a total of 1300 patches were obtained, each one containing a single bee. These images were manually labeled with one of the two classes considered: *mite* vs. *no mite*. Only 103 of these images correspond to bees with mites, resulting in a high imbalanced dataset. For this reason, a balanced dataset was created by randomly selecting 103 negative samples to match the available positive ones. Based on a 80-20 percent split, two distinct subsets were selected: a 163-sample train set and a 43-sample test set.

The experimental results were analyzed in terms of the two following metrics: *accuracy*, the percentage of correctly classified instances; and *F1-measure*, the harmonic mean of precision and recall. Due to the size of the train set, a 3-fold cross-validation [9] was used to evaluate the hyper-parameters of each feature extraction method, such as the number of bins used for color histograms (30-256 bins), or the grid-size of SURF features (30-80 px).

The goal of this second experiment was to find the feature vector that best describes the underlaying data. For this purpose, the three different color models considered, as well as their individual color components, were combined with the four feature extraction methods. Given the great number of combinations obtained, only the most relevant ones are included in Table 1.

As can be observed, R from RGB, L from Lab, and H from HSV are the individual components that provide better results, similar to the three components of each one. Regarding feature extraction, the color histogram moments and SURF are the most competitive ones when combined with other methods.

With respect to the classifiers, random forest provides the highest values for the two performance measures considered, followed closely by SVM. And finally, the use of CLAHE as a pre-processing step seems not to be necessary, in general, since the best results for most of the combinations are obtained when no image equalization is applied. However, the best results for both accuracy and F1-score, 0.81 and 0.83 respectively, are obtained when using the following configuration: (1) CLAHE and HSV color space as pre-processing, (2) LBP (radius: 1 px, 8-neighborhood) and SURF (grid-size: 80 px) feature extraction on all three color channels, and (3) random forest classifier for final classification.

Table 1. Image classification results obtained with different combinations of colors models and feature extraction methods, with no equalization (top) and after applying CLAHE (bottom). Results are shown in terms of accuracy and F1-measure using three different classifiers. The best result per column, top and bottom, appears in bold.

Color model	Features applied	Naive Bayes		SVM		Random Forest	
		Acc	F1	Acc	F1	Acc	F1
RGB	Moments, SURF	0.72	0.74	**0.79**	**0.82**	0.67	0.67
Lab	Moments, SURF	0.61	0.71	0.65	0.65	0.72	**0.79**
HSV	Histog., Moments, SURF	0.56	0.58	0.51	0.53	**0.81**	**0.79**
R	Hist., Momen., LBP, SURF	0.70	0.70	**0.79**	**0.82**	0.70	0.67
L	Hist., Momen., LBP, SURF	0.72	0.74	0.65	0.69	**0.81**	**0.79**
H	Histog., Moments, SURF	**0.77**	**0.78**	0.56	0.56	0.65	0.61
RGB	Histog., moments, SURF	0.56	0.63	0.61	0.64	0.77	0.77
Lab	Moments, SURF	0.67	**0.76**	0.51	0.68	0.77	0.81
HSV	LBP, SURF	**0.70**	0.74	**0.67**	0.68	**0.81**	**0.83**
R	Histog., moments, SURF	0.51	0.59	0.58	0.59	0.77	0.78
L	Moments, SURF	0.56	0.64	0.63	**0.70**	0.77	0.79
H	Histograms, moments	0.49	0.56	0.56	0.58	0.72	0.74

Note that, in addition to the pipeline illustrated in Fig. 3, an object detection pipeline similar to the Viola/Jones face detector [13] was applied to the problem at hand. The main performance advantage of this approach comes from efficient computation of Haar-like features, using integral images in combination with a cascading classifier. Nonetheless, this alternative pipeline provides a maximum accuracy of 0.65 and F1-score of 0.71, mainly because an object detection approach of this type needs a greater amount of data samples for training.

4 Conclusion

Varroa mites are harmful parasites that affect bee colonies, causing the destruction of hives. Automatic monitoring systems play an important role to help beekeepers that traditionally perform a time consuming manual sampling.

This manuscript presents a first, novel approach to detect parasites in honey bees recorded when entering and leaving their hive. Video frames are individually processed in order to extract single images of each bee, which are further analyzed to classify them as *mite* or *no mite*. Experimentation results demonstrated the adequacy of the proposed pipeline, which includes both image analysis and machine learning techniques. The proposed methods are able to classify bee images with maximum accuracy and F1-measure over 80%.

Our future research is focused on the use of deep learning approaches for image classification and techniques to handle the class imbalance problem.

Acknowledgments

This work was partly supported by Vienna Business Agency under grant Innovation 2016 - 1583681, Ministerio de Ciencia e Innovación of the Gobierno de España (project TIN2015-66951-C2), SGR 1219, CERCA, *ICREA Academia 2014* and Marató TV3 (grant 20141510).

References

1. EYESONHIVES, http://www.keltronixinc.com/ Last accessed: April 2018
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). Computer Vision and Image Understanding 110(3), 346–359 (2008)
3. Chiron, G., Gomez-Krämer, P., Ménard, M.: Detecting and tracking honeybees in 3D at the beehive entrance using stereo vision. EURASIP Journal on Image and Video Processing 2013(1), 1–17 (2013)
4. Evans, H.: ARNIA: Using Remote Hive Monitoring Data, http://www.beeculture.com/arnia-using-remote-hive-monitoring-data/ Last accessed: April 2018
5. Kantardzic, M.: Data mining: concepts, models, methods, and algorithms. John Wiley & Sons (2011)
6. Lee, K.V., Moon, R.D., Burkness, E.C., Hutchison, W.D., Spivak, M.: Practical sampling plans for Varroa destructor in Apis mellifera colonies and apiaries. Journal of Economic Entomology 103(4), 1039–1050 (2010)
7. Lundie, A.E.: Flight Activities of the Honey Be. US Department of Agriculture Bulletin 1328 (1925)
8. Pietikäinen, M.: Local binary patterns. Scholarpedia 5(3), 9775 (2010)
9. Refaeilzadeh, P., Tang, L., Liu, H.: Cross-validation. In: Encyclopedia of Database Systems, pp. 532–538 (2009)
10. Schurischuster, S., Zambanini, S., Kampel, M., Lamp, B.: Sensor Study for Monitoring Varroa Mites on Honey bees (Apis Mellifera). In: Visual Observation and Analysis of Vertebrate and Insect Behavior Workshop (2016)
11. Sobral, A., Vacavant, A.: A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. Computer Vision and Image Understanding 122, 4–21 (2014)
12. Struye, M., Mortier, H., Arnold, G., Miniggio, C., Borneck, R.: Microprocessor-controlled monitoring of honeybee flight activity at the hive entrance. Apidologie 25(4), 384–395 (1994)
13. Viola, P., Jones, M.J.: Robust real-time face detection. International Journal of Computer Vision 57(2), 137–154 (2004)
14. Zuiderveld, K.: Contrast limited adaptive histogram equalization. In: Graphics gems IV. pp. 474–485 (1994)