

# Comparison of Machine Learning Techniques to Determine Bee Hive Mite Infestation Level from Mobile Phone Digital Images

John Futcher<sup>1</sup>, Islam Choudhury and Gordon Hunter

*School of Computer Science and Mathematics*

*Kingston University*

*London, UK, KT1 2EE*

**Abstract.** This paper describes the early phase of a framework being developed to identify infestation levels of pests and parasites of the Western Honey Bee. Image processing techniques and two classical machine learning algorithms have been used to examine close-up images of the material falling on a varroa board beneath the mesh floor of a bee hive and identify Varroa Destructor mites versus other debris to a high level of accuracy.

**Keywords.** Honeybee Health, Varroa Destructor, Machine Learning, Image Processing

## 1. Introduction

Western honey bees (*Apis Mellifera*) provide three main benefits to mankind and the world in general. Through pollination of commercial crops, they contribute to 30% of the food we eat [1]; by pollinating wild flowers they lay the basis of the wildlife ecological system [2], and they produce hive products such as honey and beeswax which can be sold, and this is promoted by the UN as an economic solution in impoverished communities [3].

The health of honey bees is under pressure from several stressors. These include chemicals e.g., Neonicotinoids [4] and Herbicides [5], bee viruses, lack of variety of forage and various pests and parasites. Probably the most significant parasite currently is the varroa mite (*Varroa Destructor*), see Figure 1. This mite jumped species from the Eastern honey bee (*Apis Cerana*) in the 1950s and has spread of its own accord and through human movement of honey bee stocks and is now present on all continents where honey bees exist. The varroa mite is harmful to honey bees in two ways. Firstly, they obtain their nutrition from the honey bee, hence weakening it, and secondly it acts as a vector for bee viruses, increasing the viral load suffered by the honey bee [6].

Within an integrated pest management framework, several chemicals and actions can be taken to manage the varroa infestation, but key to deciding which action to take is to understand the infestation rate. Several methods of estimating varroa levels are

---

<sup>1</sup> Corresponding Author, John Futcher, K1709382@kingston.ac.uk.

available, but most involve opening the hive to collect a sample of bees, and some even result in the death of the bee sample. The varroa mite drop method is non-invasive to the colony, has been shown [7] to be accurate at estimating infestation levels, and is also one of the methods supported by DEFRA (Department of Environment, Food and Rural Affairs) [8] in the UK.

This overall target of this project is to produce a framework of techniques to support the estimation of the level of infestation of honey bee colonies by significant pests and parasites. This will be of benefit to commercial and hobby bee keepers alike as it will aid appropriate levels of treatment leading to savings in time, money, and honey bees.

The purpose of the work covered in this paper is to develop and compare two machine learning approaches to classification – Decision Tree and Naïve Bayes – and to determine their accuracy against magnified varroa board images.



Figure 1 - Varroa Destructor, Photo by USGS Bee Inventory and Monitoring Lab, Public Domain.

## 2. Related Work

Much work has been done recognizing varroa on the bodies of honey bees as they enter and exit their colonies, e.g. [9,10,11] but the specialized resource requirements of these projects included items such as cameras, lighting equipment, network connections and custom-built entryways to the hive forcing the bees to walk over the camera focus area. This results in the cost of the set up being prohibitive to both amateur and commercial bee keepers.

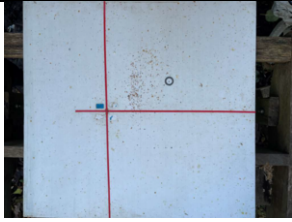

Based on work by [12,13] on identifying chemicals originating from varroa-infested brood cells, several projects have tried to identify these trace chemicals using sensors. The use of multiple semiconductor gas sensors is described in [14] using a mechanical mains-driven pump to push hive air through the sensors and claimed a significant difference in gas readings when there was a difference of varroa infestation level over 9.1%. However, this again requires expensive and specialized equipment.

A Pascal programme was developed by Imler-Kiel[15] to estimate varroa infestation level from a processed image, though this work does not appear to have been progressed since 2010. The use of Python and Scikit [16] was reported to produce good results but only when the varroa board was sparsely populated, as part of König's IndusBee4.0 bee monitoring programme. A smartphone app [17] is available on Google Play, however, this performed poorly with a response of 456 when used with a varroa board whose actual manual count was 15 (in our own experiment).

### 3. Examining the Varroa Board

A varroa board sits beneath a mesh floor in the bee hive and collects the debris that falls out of the bee hive. Table 1 shows images of a full varroa board, and a magnified image of the board taken from a closer distance. The long-term plan is for a mite count to be performed on a single image of a full board. However, in this early phase, the project captured magnified images at close distance where the varroa can also be identified with the naked eye.

**Table 1** - Full and close-up images of varroa board.

| <b>Whole Varroa Board</b>                   |   |
|---|---|
| Size of board area captured in image        |  |
| 414 x 443mm                                 |   |
| Approximate Distance photo taken from board |   |
| 400mm                                       |   |
| <b>Closeup</b>                              |   |
| Size of board area captured in image        |  |
| 50 x 37mm                                   |   |
| Approximate Distance photo taken from board |   |
| 100mm                                       |   |

### 4. Image Processing

Using Microsoft Visual Studio 2019 IDE and OpenCVSharp [18] library for image processing, a C# program was developed to allow the user to select an image and then perform the processing as detailed below.

- Images are read and converted to 8-bit grayscale using the ITU standard [19].
- The image is then thresholded to produce a binary image. A threshold value of 70 was used, whereby a pixel’s intensity value was changed to 255 (white) if its grayscale intensity value was greater than this threshold value and to 0 (black) otherwise. The usage of the value of 70 was obtained from experiments using the Fiji [20] program. See Figure 2.
- The resulting image is then processed to identify blobs using the function “SimpleBlobDetector”.
- For each blob the OpenCV function “FindContours” (based on[21]) derives the contours in the image, from this contour, the best fit ellipse is found using “fitEllipse” via a least squares approach.
- The ellipse and the contour are compared to calculate the IOU (Intersection over Union) measurement (formula 2) as well as an “Exclusive Or” (XOR) error (formula 5). See Figure 3.

- Flood filling the contour allows the blob to be isolated from the background and then the mean and standard deviation of the different colour intensities for the blob are calculated.
- Each blob image is then saved to the file system and the corresponding data to a SQL database.

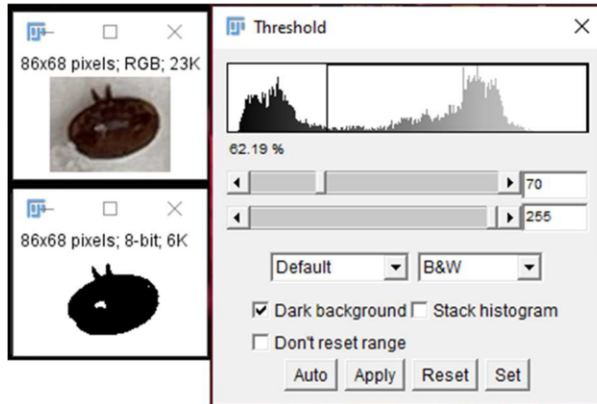


Figure 2 - Effect of thresholding an image of a varroa mite using Fiji

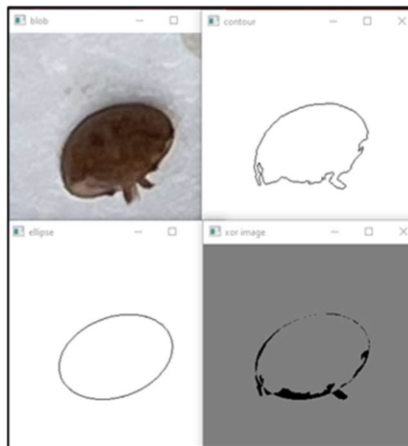


Figure 3 – Contour, ellipse and XOR visualization for a varroa mite.

## 5. Machine Learning Component

### 5.1. Feature Selection

At first sight the varroa appears to be elliptical in shape, reddish brown in colour and consistent in shape and size. So, rather than use every feature captured in the Image Processing phase in the machine learning process, we decided to utilise the following four features: ellipse axis ratio, IOU, blob area and normalised mean red intensity as derived in formulas 1-4. The low number of images of varroa resulted in a situation whereby addition features resulted in lower accuracy. Mean red intensity was considered as an alternative to normalised red, but the normalised red value was deemed to segment the data set more effectively.

The ellipse axis ratio as per formula 1.

$$\text{ellipse axis ratio} = \frac{\text{ellipse short axis}}{\text{ellipse long axis}} \tag{1}$$

The IOU (Intersection over Union) using formula 2,

$$IOU = \frac{\sum \text{Pixels in bot contour and ellipse}}{\sum \text{Pixels in either contour or ellipse}} \tag{2}$$

The area of the blob,

$$\text{blob area} = \sum \text{Pixels within blob contour} \tag{3}$$

The normalised red mean intensity,

$$\text{Normalised Red Mean Intensity} = \frac{\text{Red Mean Intensity}}{\text{Red Mean} + \text{Bl Mean} + \text{Gree Mean}} \tag{4}$$

Where the red, blue, green means are the mean intensity values in those colour channels averaged over the pixels in the blob.

And XOR Error.

$$XOR\ Error = \frac{\sum \text{Pixels in contour not in ellipse} + \sum \text{Pixels in ellipse not in contour}}{\text{Blob Area}} \tag{5}$$

### 5.2. Data Augmentation

Due to the good health and low varroa count of the first author’s bee colonies, the project has had to deal with an unbalanced data set with far fewer images of varroa mites than those of other general debris. Therefore, augmentation was applied to the training data set to balance the varroa to debris ratio and increase the number of data points used to build the models.

Augmentation was achieved by multiplying each of the values of the four selected features by 0.99, 1 and 1.01, resulting in three data points for each observed data point,

a cartesian join (where all data points of each feature are combined together) was applied when extracting the data resulting in the training data containing approximately 9000 each of varroa and debris records.

### 5.3. First Algorithm – Decision Tree

The first algorithm used was the FastTree classifier from Microsoft’s ML.NET machine learning suite. This classifier is described [22] as an efficient implementation [23] of the MART (Multiple Additive Regression Trees) algorithm[24]. This algorithm uses boosting to combine an ensemble of weaker prediction models to create a strong model. The code developed and used was informed by [25]. One disadvantage to running the data under the Microsoft ML.Net umbrella is that the details of the decisions being made within the model are not accessible to the user and, as such, we are unable to analyse the decisions being made in the classification process.

To facilitate cross-validation, the dataset was separated randomly into folds of approximately equal size as shown in Table 2.

**Table 2** - Count of data points in each fold.

| <b>Fold</b>       | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|-------------------|----------|----------|----------|----------|----------|
| Actual Debris     | 368      | 369      | 379      | 418      | 417      |
| Actual Varroa     | 29       | 30       | 22       | 32       | 26       |
| Total Blob Images | 397      | 399      | 401      | 450      | 443      |

The ML.NET Decision Tree algorithm is driven by three key parameters, namely the learning rate (0.2), the maximum number of trees in the ensemble (20) and the maximum number of leaves (21). The detailed results of testing the model against Fold 4 (chosen as it had the largest test data set) are shown in Table 3. The precision of predicting varroa is negatively impacted by the class imbalance, where a small proportion of False Positive results make up a significant proportion of the overall Positive results. However, the accuracy rate is high, and the model is performing well.

**Table 3** - Decision Tree results using fold 4 as test data.

|               | <b>Predicted Debris</b>    | <b>Predicted Varroa</b> |
|---------------|----------------------------|-------------------------|
| Actual Debris | 408                        | 10                      |
| Actual Varroa | 0                          | 32                      |
| Total         | 408                        | 42                      |
| Precision     | 0.761905                   |                         |
| Recall        | 1.000 (no false negatives) |                         |
| Accuracy      | 0.977778                   |                         |

This algorithm was then used to train models against the five different folds of the data points (training used four folds and testing the indicated remaining fold) and the summary results were as shown in Table 4. The cross-validation results show consistent values across the folds, which means that the model is robust for the dataset.

**Table 4 - Decision Tree model cross validation results**

| Test Fold | 1     | 2     | 3     | 4                      | 5     | Mean  |
|-----------|-------|-------|-------|------------------------|-------|-------|
| Precision | 0.778 | 0.867 | 0.808 | 0.762                  | 0.862 | 0.815 |
| Recall    | 0.966 | 0.867 | 0.955 | 1 (no false negatives) | 0.962 | 0.950 |
| Accuracy  | 0.977 | 0.980 | 0.985 | 0.977                  | 0.989 | 0.982 |

*5.4. Analysis of Incorrectly Classified Data*

One of the 10 mis-classified images reported in Table 3 is shown in Figure 4. The outline of this piece of debris is approximately elliptical, but it is quite rough, especially when compared with the contour shown in Figure 3 for a varroa mite which is quite smooth for at least half of the contour. Establishing and including a boundary roughness score would most likely improve the accuracy achieved to date.



Figure 4 - Mis-classified Debris.

*5.5. Second Algorithm – Naïve Bayes*

The second algorithm used was Naïve Bayes. Initial attempts were made to make use of the ML.NET Naïve Bayes framework, but we were unable to produce varroa predictions on any of the data folds. Therefore, it was necessary to implement it from first principles. Laplacian smoothing [26] was used to avoid singularities due to any counts being zero.

Detailed results using fold 4 as the test dataset are given in Table 5. This is a direct comparison with the decision tree results shown in Table 3. The cross-validation process was then performed on the same data folds as for decision tree, and the results are shown in Table 6. As can be seen, the Naïve Bayes model underperformed compared with the Decision Tree results, with each fold performing less well on virtually all measurements.

**Table 5** - Naïve Bayes model results using fold 4 as test data.

|               | <b>Predicted Debris</b>    | <b>Predicted Varroa</b> |
|---------------|----------------------------|-------------------------|
| Actual Debris | 392                        | 26                      |
| Actual Varroa | 0                          | 32                      |
| Total         | 392                        | 58                      |
| Precision     | 0.551724                   |                         |
| Recall        | 1.000 (no false negatives) |                         |
| Accuracy      | 0.942222                   |                         |

The Naïve Bayes model identifies the varroa correctly, matching the Decision Tree Model, the big difference is a large increase in the number of images of debris being classified as varroa, which has a significant effect on the precision score.

**Table 6** - Naïve Bayes model cross validation results

| <b>Test Fold</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> | <b>Mean</b> |
|------------------|----------|----------|----------|----------|----------|-------------|
| Precision        | 0.581    | 0.676    | 0.576    | 0.552    | 0.571    | 0.591       |
| Recall           | 0.862    | 0.767    | 0.864    | 1.000    | 0.923    | 0.883       |
| Accuracy         | 0.945    | 0.958    | 0.958    | 0.942    | 0.955    | 0.952       |

## 6. Discussion

### 6.1. Data Set Size

The volume of data images of varroa was small, which had to be mitigated by augmentation. Action is already in place to capture images from other apiaries, and to collect actual varroa mites to generate images on demand.

### 6.2. Improvements to the Image Processing Used

It should be possible to improve performance of the classifiers firstly by analysing the contour for its smoothness (or roughness) and determining a score for each contour's relative smoothness, and secondly by capturing the minimum and maximum values of each of the colour channel intensities.

### 6.3. Algorithm Performance

Both machine learning algorithms worked well, though the Decision Tree model performed demonstrably better than the Naïve Bayes model. The Decision Tree model resulting from being trained on all but Fold 4 data, was then incorporated into the image processing program to help set initial ground truth values. The precision value from both models was negatively affected by the relatively small number of images of varroa in the testing sets, and more so for the Naïve Bayes model.

### 6.4. Blob Area Consistency

For each image (of debris or varroa) the main object in the image is used to generate the feature data for the image, this object being called a "blob". Because the images are being captured by a hand-held smartphone, the distance from phone to varroa board is



not fixed, and so the area of the blob (as calculated by formula 3) will vary according to this distance as much as to the actual size of the physical object being photographed. This variation has a detrimental effect on the models being trained, though the computed blob area is still able to correctly classify particularly large or small blobs as not being varroa mites.

## **7. Conclusion and Future Work**

In conclusion, both classifier models gave a good level of accuracy for the test data, and the consistency of the results across the folds showed that the models were robust for the available dataset. The Decision Tree classifier produced models which gave a higher level of accuracy, precision and recall sufficient for the fold 4 model to be back-implemented into the image processing program to predict ground truth values for subsequent images processed. Our implementation only required the use of images taken using a commonly available smartphone rather than specialised camera equipment and only a crude guide to object-camera distance. These attributes should make our approach relatively straightforward for other beekeepers to use. A future version could implement an image processing app on the smartphone with varroa assessment executed in the cloud.

To improve the decision tree performance, a measure of blob contour roughness would need to be identified and implemented through re-processing the available images. The performance of the model with respect to blob area could also be improved if the phone to varroa board distance could be kept more consistent.

The planned next step of the project is to evaluate the application of deep learning techniques including Convolutional Neural Networks (CNN) [27] to the available image set and compare the results against the Decision Tree model. Potential candidates would include YOLO (“You Only Look Once”) [28], originally this technique struggled with small target objects (occupying relatively few pixels) but improvements have been made more recently [29]. Our current techniques have the advantage of yielding which image features are most influential for identifying varroa mites, but deep learning approaches usually prove highly effective.

## **References**

- [1] Klein A-M et al. Importance of pollinators in changing landscapes for world crops. *Proceedings of the Royal Society B: Biological Sciences*, 2007 Feb; 274, no. 1608, pp. 303–313.
- [2] Carreck NL, Ball BV, Martin SJ. Honey bee colony collapse and changes in viral prevalence associated with Varroa destructor. *Journal of Apicultural Research*, 2010 Jan, 49, no. 1, pp. 93–94.
- [3] Khan SLA, Poverty reduction strategy papers: Failing minorities and indigenous peoples. *Minority Rights Group International*, 2010.
- [4] Woodcock BA et al. Country-specific effects of neonicotinoid pesticides on honey bees and wild bees. *Science*. 2017 Jun;356, no. 6345, pp. 1393–1395.
- [5] Farina WM, Balbuena MS, Herbert LT, Mengoni Goñalons C, Vázquez DE. Effects of the Herbicide Glyphosate on Honey Bee Sensory and Cognitive Abilities: Individual Impairments with Implications for the Hive. *Insects*, 2019 Oct 10, no. 10, p. 354.
- [6] Sumpter DJT, Martin SJ, The dynamics of virus epidemics in Varroa-infested honey bee colonies. *Journal of Animal Ecology*, 2004 Jan; 73, no. 1, pp. 51–63.

- [7] Flores JM, Gil S, Padilla F. Reliability of the main field diagnostic methods of Varroa in honey bee colonies. *Archivos de Zootecnia*, 2015;64, no. 246, Accessed: Jan. 22, 2023. [Online]. Available: <https://www.redalyc.org/articulo.oa?id=49545650010>
- [8] BeeBase, Varroa Calculator. [nationalbeeunit.com](http://nationalbeeunit.com), 2022. <https://nationalbeeunit.com/public/BeeDiseases/varroaCalculator.cfm> (accessed Jan. 22, 2023).
- [9] Chen C, Yang E, Jiang J, Lin T. An imaging system for monitoring the in-and-out activity of honey bees. *Computers and Electronics in Agriculture*, 2012 vol. 89, pp. 100–109. doi: <https://doi.org/10.1016/j.compag.2012.08.006>.
- [10] Bjerge K, Frigaard CE, Mikkelsen PH, Nielsen TH, Misbih M, Kryger P. A computer vision system to monitor the infestation level of Varroa destructor in a honeybee colony. *Computers and Electronics in Agriculture*. 2019 Sep; 164, p. 104898.
- [11] Sevin S, Tutun H, Mutlu S. Detection of Varroa mites from honey bee hives by smart technology VarGor: a hive monitoring and image processing device. *Turkish Journal of Veterinary And Animal Sciences*. 2021 vol. 45, no. 3.
- [12] Schöning C et al. Evidence for damage-dependent hygienic behaviour towards Varroa destructor-parasitised brood in the western honey bee, *Apis mellifera*. *Journal of Experimental Biology*. 2012 Jan; 215, no. 2, pp. 264–271. doi: 10.1242/jeb.062562.
- [13] Nazzi F, Della Vedova G, D'Agaro M. A semiochemical from brood cells infested by Varroa destructor triggers hygienic behaviour in *Apis mellifera*. *Apidologie*. 2004 Jan; 35, no. 1, pp. 65–70.
- [14] Szczurek A, Maciejewska M, Bąk B, Wilde J, Siuda M, Semiconductor gas sensor as a detector of Varroa destructor infestation of honey bee colonies – Statistical evaluation. *Computers and Electronics in Agriculture*. 2019 Jul; 162, pp. 405–411.
- [15] Imler-Kiel. VarroaCount [www.imker-kiel.de](http://www.imker-kiel.de). 2010. <http://www.imker-kiel.de/news/molfssee-mit-horst/67varroacount.html> (accessed Jan. 21, 2023).
- [16] König A. “VarroaCounter–Towards Automating the Varroa Screening for Alleviated Bee Hive Treatment,” in SEIA’ 2019 Conference Proceedings. 2020 Jan, pp. 244–247.
- [17] Toplab, “VarroaCounter | Mite Detection | by TopLab - Toplak Laboratory E.U.,” varroa-counter-en, 2018. <https://www.varroacounter.com> (accessed Feb. 01, 2023).
- [18] Shimat, OpenCV wrapper for .net. GitHub, Nov. 27, 2022. <https://github.com/shimat/opencvsharp> (accessed Feb. 01, 2023).
- [19] ITU. Recommendation ITU-R BT.601-7. [www.itu.int](http://www.itu.int), 2017. [https://www.itu.int/dms\\_pubrec/itu-r/rec/bt/R-REC-BT-601-7-201103-I!!PDF-E.pdf](https://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT-601-7-201103-I!!PDF-E.pdf) (accessed Feb. 01, 2023).
- [20] FIJI. Fiji. ImageJ Wiki, 2022. <https://imagej.net/software/fiji/>
- [21] Suzuki S and Abe K. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 1985 Apr; 30, no. 1, pp. 32–46.
- [22] Microsoft. FastTreeBinaryTrainer Class (Microsoft.ML.Trainers.FastTree). [docs.microsoft.com](https://docs.microsoft.com), 2022. <https://dotnet/api/microsoft.ml.trainers.fasttree.fasttreebinarytrainer?view=ml-dotnet> (accessed Feb. 01, 2023).
- [23] Rashmi KV and Gilad-Bachrach R. DART: Dropouts meet Multiple Additive Regression Trees. arXiv:1505.01866 [cs, stat], May 2015, (Accessed: Feb 01, 2023. [Online]. Available: <https://arxiv.org/abs/1505.01866>
- [24] Friedman JH. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 2001 Oct; 29, no. 5, pp. 1189–1232.
- [25] Zivkovic NM. Machine Learning with ML.NET - Guide to Decision Trees. Rubik’s Code, Feb. 22, 2021. <https://rubikscodene.net/2021/02/22/machine-learning-with-ml-net-guide-to-decision-trees/> (accessed Feb. 01, 2023).
- [26] Liu AY and Martin CE. Smoothing Multinomial Naïve Bayes in the Presence of Imbalance. *Machine Learning and Data Mining in Pattern Recognition*, 2011, vol. 6871. [Online]. Available: [https://link.springer.com/chapter/10.1007/978-3-642-23199-5\\_4](https://link.springer.com/chapter/10.1007/978-3-642-23199-5_4)
- [27] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*. 2017 60(6), pp 84-90.
- [28] Redmon J, Divvala S, Girshick R and Farhadi A, You Only Look Once: Unified, Real-Time Object Detection, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.
- [29] Wang ZZ, Xie K, Zhang XY, Chen HQ, Wen C, He JB. Small-object detection based on yolo and dense block via image super-resolution. *IEEE Access*. 2021 Apr 9;9:56416-29.