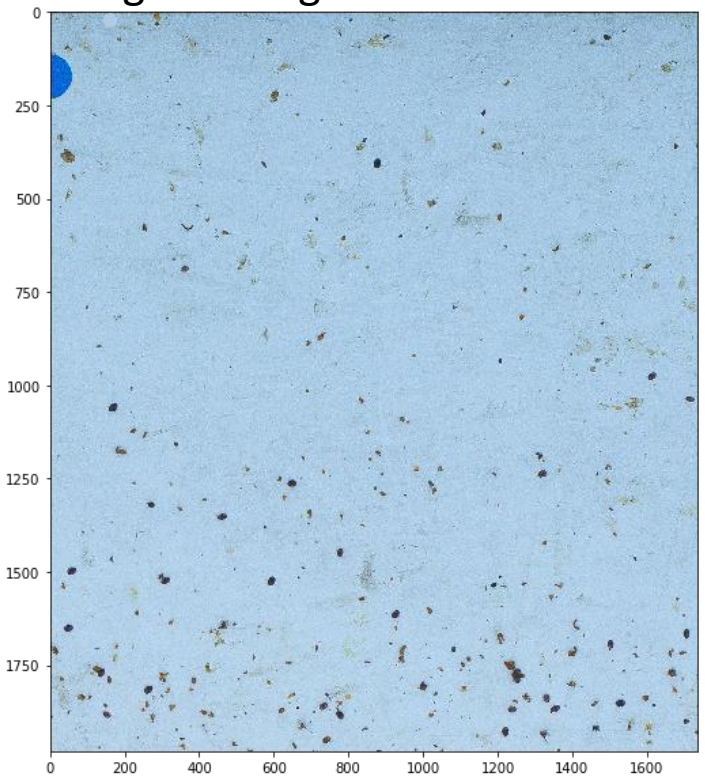
Special project

SAVE THE BEES

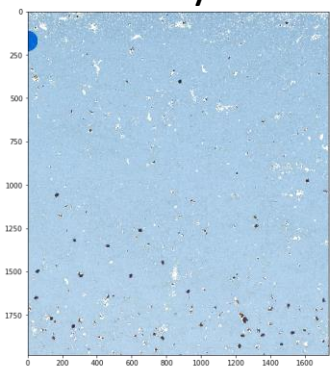
Emma Nilsson, Anna Fredriksson Häägg, Herman Dieset

Pre-processing

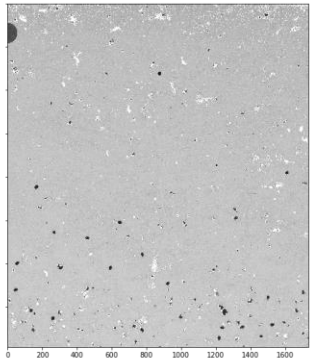
Original image



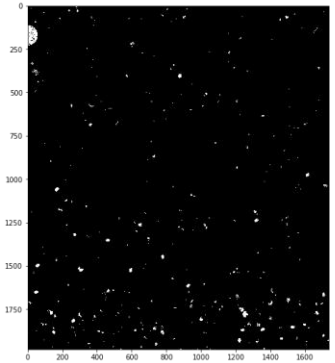
Remove yellow



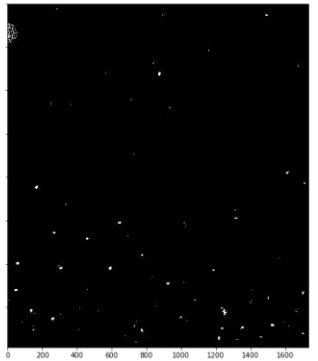
Grayscale



Thresholding



Erosion



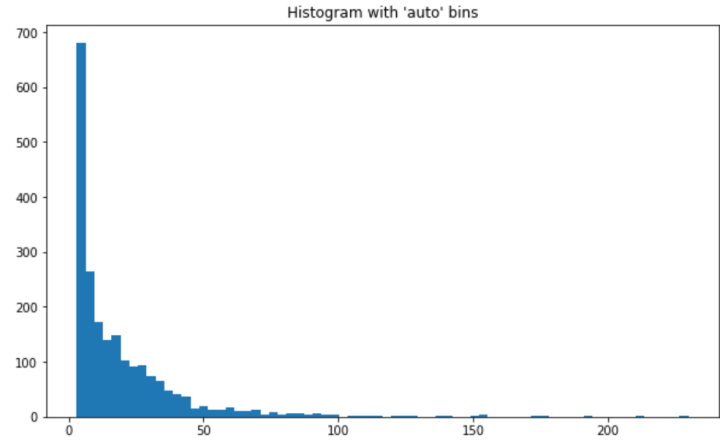
Finding varroas by segmentation

Method

- Do preprocessing
- Find all contours in the image
- Threshold them by looking at the length
  - Remove contour if it is shorter than 40 or longer than 80
- Find the bounding-boxes to the corresponding contours

Result

- Precision : 0.042
- Recall: 0.986
- F1: 0.081



First detector – using contours

Method

Build a classifier

- Using the train annotations, find contours in all train-bounding-boxes
- Convert to 6 Fourier descriptors for each contour
- Train an SVM one class classifier

Detection method

- Slide over the image
- For each window
  - Find contours
  - Convert to 6 Fourier descriptors
  - Predict probability of belonging to the positive class
  - If the contour belong to the positive class
    - Create bounding-box and add to the output list

Result

- Precision : 0.103
- Recall: 0.953
- F1: 0.185

Possible improvements

- Linear interpolation instead of padding
- Other features

Creating data for deep-learning

Positive class

- For each image, find the bounding box of each varroa in the image
- Take the starting point for each box and add the sliding window height and width
- Label as 1

Negative class

- Find all images with no varroas
- For each image, slide the window over the image
- For each window, save the part it covers in the image
- Label as 0

Second detector – using MLP

Method

Build the network

- **Five layers, decreasing size. ReLU.**
- **Binary crossentropy loss**
- **Optimize with Adam**

Detection method

- Slide over the image
- For each window
  - Predict probability of window being a varroa
  - If probability of belonging to class 1 is higher than 0.95, classify it as a varroa

Result

- Precision : 0.13
- Recall: 0.95
- F1: 0.23

Possible improvements

- Increase number of hidden units in the layers. ~8% error decrease with 6x decrease
- More layers?
- Keras' build-in randomizer functions; vflip, hflip, rotate, scale, order, …

Final detector – using CNN

Method

Building the network

- Used 3 convolution layers with the activation ReLU
- After each convolution layer, add a pooling layer
- One linear layer and then use softmax to finally get the output

Detection method

- Slide over the image
- For each window
  - Predict probability of window being a varroa
  - If probaability of belonging to class 1 is higher than 0.95, classify it as a varroa

Result

- Precision : 0.21
- Recall: 0.98
- F1: 0.35

Possible improvements

- More layers using smaller kernels
- Creation of the negative class

Conclusion

- Better pre-processing
- Deeper networks?
- Using all data
- Improving creation of negative class
- Numeric estimation instead of boolean.