**Project Report – 31/01/22**

AIML-MDBA: Species Counting using ML

# **Objective**:

As a biodiversity analyst, I want a demonstration of how machine learning (ML) can be used to count species, so that I can understand the trade-offs with a manual approach.

# **Dataset and training preparation instructions:**

Given Dataset**:** 205 overhead images of birds with resolution 8192x5460px

1. A total of 117 images contained at least 1 bird (according to manual data provided by MDBA). These were separated from those containing none (88 “empty” images). These “empty” images are randomly sampled and added to the training set in step 6.
2. Of the 117 non-empty images, 17 images were chosen at random to make up a hold-out test set to measure model performance. The remaining 100 images formed the training/validation set.
3. The 100 training/validation set images were split into 16x10 non-overlapping slices (resolution 512x546px), totalling 100x16x10 = 16000 slices.
4. The slices were filtered to exclude empty slices (i.e. those containing no birds), according to manual data provided by MDBA. The filtered training/validation set contained 2201 non-empty slices.
5. The training/validation set was partially labelled with bounding boxes using annotation software for a total 2142 images (59 were left unlabelled).
6. Slices from empty images were randomly sampled to add 238 empty images to the training set, for a total of 2380 images. This was done to reduce the False Positive detection bias of the model.
7. The 2380 images were split into a training set containing 1904 images and a validation set containing 476 images.
8. A training algorithm[[1]](#footnote-2) was used with a FasterRCNN base network to train an object detector on the training/validation set (see **Appendix A** for hyper-parameters used during training).
9. The selected model was that with the highest mAP on the validation set, and its performance was measured on the hold-out test set. See **Figure 1** for examples of model performance on individual slices.
10. Finally, two additional blind datasets were provided containing 52 and 207 full resolution images. These datasets were evaluated using the trained model (**APPENDIX C**), with counts excluding outer 5% of images due to overlapping regions between adjacent images.

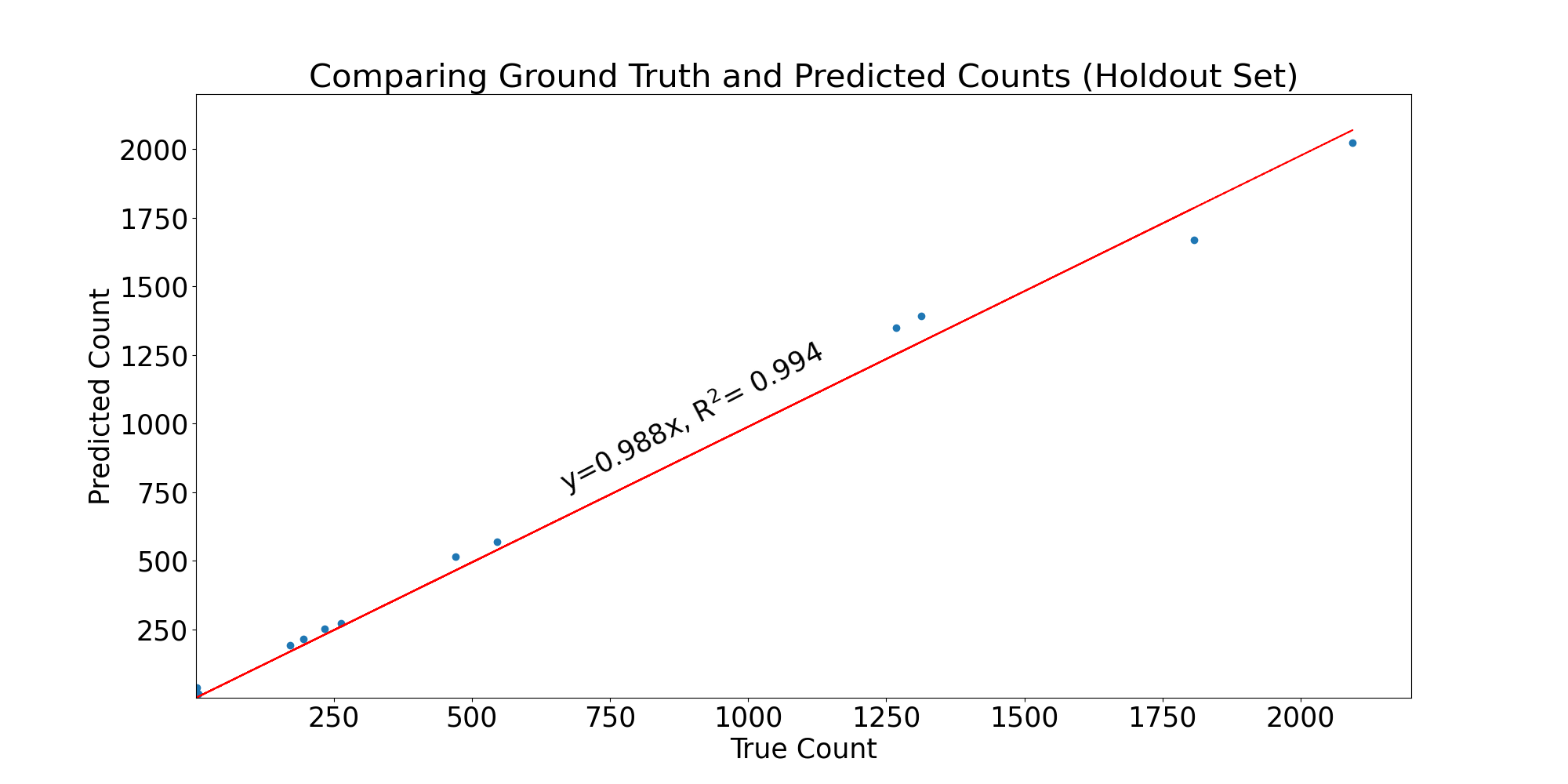
# **Results on the hold-out test set:**

To properly assess the performance of the model on unseen data, the model was used to generate predictions of total bird counts on whole images in the hold-out test set. This was done as follows:

1. Slicing the 17 hold-out test set images into 16x10 slices.
2. Feeding all slices into the trained model and extracting bird counts for each slice.
3. Adding up bird counts from all slices for each of the 17 whole images.
4. Comparing predicted vs. true bird counts for each image (see **Figure 2**)

**Figure 1 –** Examples of model predictions on randomly selected slices from hold-out test set.

The result in **Figure 2** shows a clear linearity between true and predicted counts for the hold-out test set, with less than 15% error on any image containing 50 birds or more, and less than 4% error on the total bird count (see **Appendix B**). The model confidence threshold can be calibrated to optimise the performance of the model.

**Figure** **2 –** Predicted (y-axis) vs true (x-axis) counts for images in the hold-out test set, showing strong linearity and less than 5% error on the overall count.

# Discussion

Three issues were improved compared to the previous iteration:

1. **False Positive Bias:** Randomly sampling empty slices and adding them to the training set to teach the new model what “empty” slices look like seemed to explain the reduction in over-counting over certain terrains (compare new vs. old model predictions in **APPENDIX B** for images 0047 and 0109).
2. **False Negatives Bias:** For some images (such as image 137), the above behaviour is reversed, where the model predicts more birds than are manually counted. In these cases it was observed that the model seems to be counting birds that are difficult to count manually (e.g. hidden under thick grass or above trees). Therefore, in these cases it seems the model is counting birds that were missed by manual annotators.
3. **Simplified Model Calibration:** Previously a parameter called non-max suppression was suggested for calibration. Now, a more intuitive calibration method is proposed; thresholding the model confidence itself. This way, if the model tends to under- or over-count, the confidence threshold can be tuned up or down to calibrate/correct for the bias until it performs adequately across various datasets.

# Conclusions

1. The results above clearly demonstrate the feasibility of using even relatively painless ML pipelines to count species.
2. Model confidence can be used to calibrate model performance.
3. Further labelling of datasets captured on days with different lighting conditions or with different camera settings, and also camera height above ground is recommended to improve the generalisation of the model’s performance.
4. Randomly sampling empty images from varied backgrounds and including them in the training data will help teach the model how to correctly identify empty backgrounds and improve model performance.
5. Careful inspection of images that contained very few manual annotations (such as image 0137) showed that the new model is capable of locating birds in images that were missed by human annotators. This is an important advantage of a well-trained model over manual annotation.
6. Other ML methods/procedures can be considered, including but not limited to:
   1. Relabelling using polygons to reflect the oblong and affine shapes of birds in the dataset.
   2. Using crowd density modelling to enhance performance especially in crowded areas.

**APPENDIX A**

**Hyper-parameters used during training.**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value (Old)** | **Value (New)** |
| Base network | ResNet-50 | FasterRCNN |
| Early stopping patience | 30 | 3 |
| Early stopping tolerance | 0 | 0 |
| Epochs | 1000 | 10 |
| Image size | 512x512px | 512x546px |
| Learning rate | 0.00015 | 0.001 |
| Mini batch size | 16 | 8 |
| Number of classes | 1 | 1 |
| Pretrained | ImageNet | ImageNet |

**APPENDIX B**

**Predicted vs. True counts for Old vs. New models (confidence threshold = 0.96).**

| **Image** | **True** | **Predicted (Old)** | **Predicted (New)** | **Accuracy (Old)** | **Accuracy (New)** |
| --- | --- | --- | --- | --- | --- |
| 0005 | 545 | 508 | 570 | 93.2**%** | 104.6% |
| 0007 | 470 | 460 | 514 | 97.9**%** | 109.4% |
| 0011 | 171 | 155 | 191 | 90.6**%** | 111.7% |
| 0015 | 263 | 241 | 272 | 91.6**%** | 103.4% |
| 0021 | 233 | 239 | 253 | 102.6**%** | 108.6% |
| 0034 | 1313 | 1324 | 1391 | 100.8**%** | 105.9% |
| 0037 | 1268 | 1340 | 1350 | 105.7**%** | 106.5% |
| 0040 | 195 | 195 | 216 | 100.0**%** | 110.8% |
| 0047 | 1 | 44 | 10 | -- | -- |
| 0078 | 1807 | 1725 | 1669 | 95.5**%** | 92.40% |
| 0109 | 1 | 36 | 1 | -- | -- |
| 0113 | 2094 | 1835 | 2024 | 87.6**%** | 96.70% |
| 0124 | 1 | 3 | 11 | -- | -- |
| 0137 | 3 | 4 | 38 | -- | -- |
| 0190 | 1 | 3 | 6 | -- | -- |
| 0200 | 3 | 4 | 18 | -- | -- |
| 0201 | 5 | 1 | 15 | -- | -- |
| **TOTAL** | **8374** | **8117** | **8549** | **96.9%** | **102.1%** |

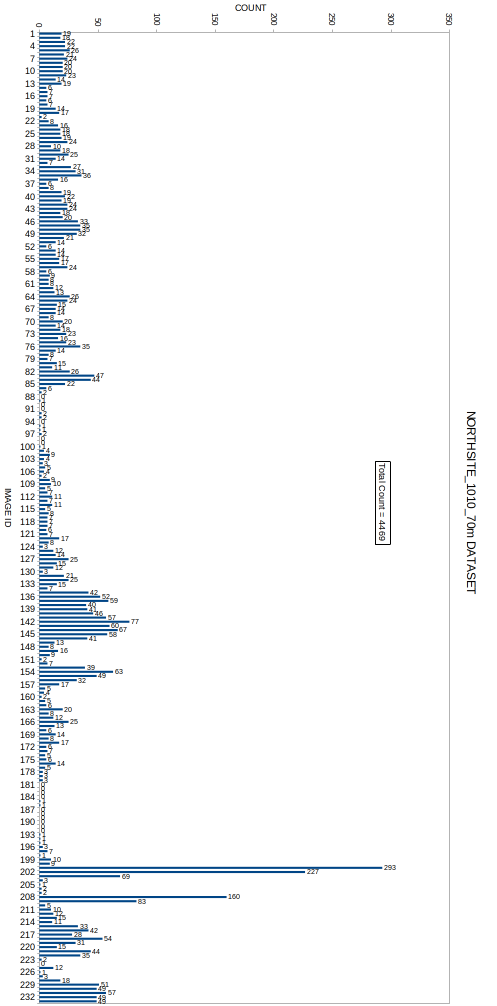
**APPENDIX C**

**Predicted counts for blind datasets (confidence threshold = 0.96) excluding overlap regions\***

\*Data to reproduce figures can be found in:

*Midlake\_1010\_100m\_35mm\_detections.csv and NorthSite\_1010\_70m\_50mm\_detections.csv*

NORTHSITE\_1010\_70m DATASET ON NEXT PAGE

****

1. https://arxiv.org/pdf/1506.01497.pdf [↑](#footnote-ref-2)