1 Evaluation

1.1 Statistics

The annotation table obtained from the image labelling is in the form of rectangle location and dimensions of the bounding box dimensions. The datasets are divided into three sets

1. *Training Dataset*: annotation data obtained using the consumer drone and used for training the model.
2. *Testing Dataset*: annotation data obtained using the consumer drone and used for testing the model.
3. *GeoScience Dataset*: annotation data obtained using the data obtained from the GeoScience Department using a commercial drone.

Bounding box heights, widths and aspect ratios can be obtained from the annotation data. The testing data and GeoScience data will be tested against the training data to see whether they are from the same population. If any populations are the same, the analysis done on one population should have the same evaluation results. If the populations are different, the analysis on both will be different, and the algorithms would need to be adapted to cater for this difference.

1.1.1 Training Dataset

Training bound boxes have a similar average width and average height pixel sizes. The average size is around 76 pixels, with minimum pixel size of around 28 pixels. The smallest bounding box matches the $32x32$ VGG16 pixel field of view within the *conv5* output feature map. The highest number of bounding boxes fall under a $2x2$ VGG16 feature map.

Table 1: Consumer Drone Bounding Box Statistics - Train data

	Mean	Std. Dev.	Min	Max	Population
Width	73.848	28.359	25	206	524
Height	79.293	27.197	31	207	524

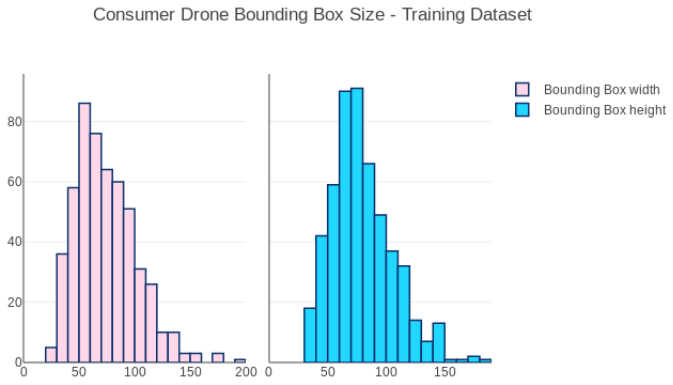

Figure 1: Consumer Drone Bounding Box Height and Width Histograms- Train data

As per qqplot show in figure 1.1.1 the data does not follow a normal distribution, so non-parametric tests will be performed to check that samples are obtained from the same population.

The aspect ratio of the training set as shown in the figure 1.1.1 shows that there is no preference between log(-0.5) to log(0.5). This shows that there is no orientation preference of the bounding boxes in the training data set. As can be expected from aerial imagery data.

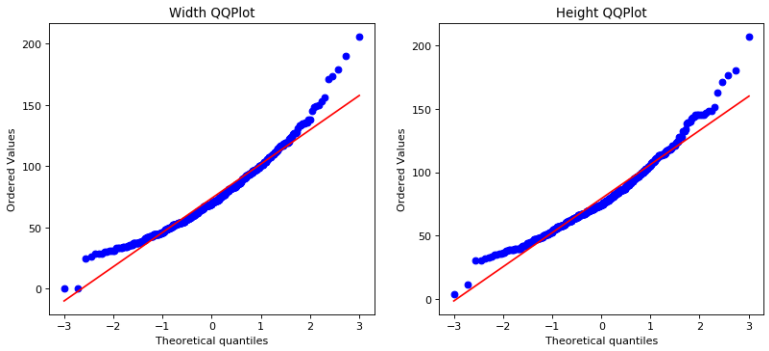

Figure 2: Consumer Drone Bounding Box Height and Width QQplot - Train data

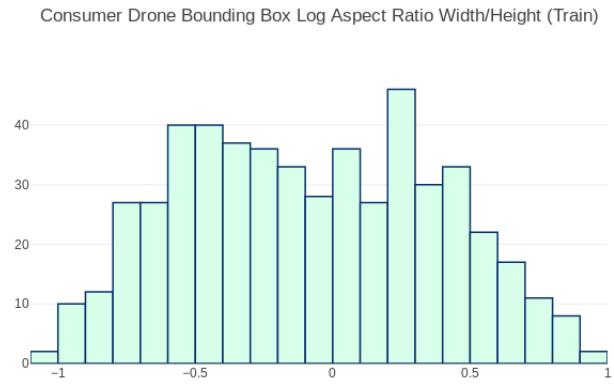

Figure 3: Consumer Drone Bounding Box Log Aspect Ratio Histogram - Train data

1.1.2 Testing Dataset

Testing bound boxes have a minor tendency to be larger in height than in width. The average size is also around 76 pixels, with minimum pixel size of around 29 pixels. Again this is within the field of view of the VGG16 pixel single feature vector of the *conv5* output feature map.

Table 2: Consumer Drone Bounding Box Statistics - Test data

	Mean	Std. Dev.	Min	Max	Population
Width	71.654	29.435	26	199	246
Height	81.232	34.203	30	272	246

As per qqplot show in figure 1.1.1 the data does not follow a normal distribution. This confirms the non-parametric tests to be performed on the population sampling.

The aspect ratio of the training set as shown in the figure 1.1.1 shows that there is no preference between log(-0.5) to log(0.5). This shows that there is no orientation preference of the bounding boxes in the training data set. The smaller values at the O for both the training set and the test set indicate that the objects bounding boxes are rarely a box shape (i.e. equal width and height) which indicates that the objects have an elongated shape. This is to be expected as the images labelled are of beverage bottles and containers, which are predominantly manufactured to be stacked side-by-side, and ergonomically designed to be handled by a single hand.

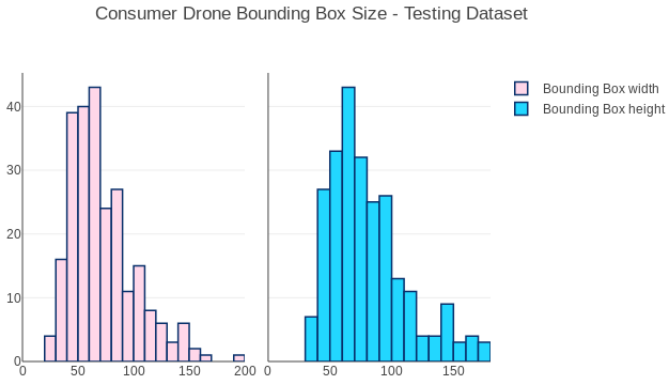

Figure 4: Consumer Drone Bounding Box Height and Width Histograms- Test data

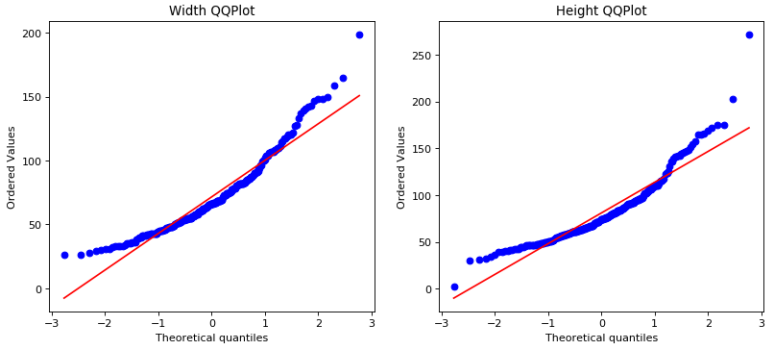

Figure 5: Consumer Drone Bounding Box Height and Width QQplot - Test data

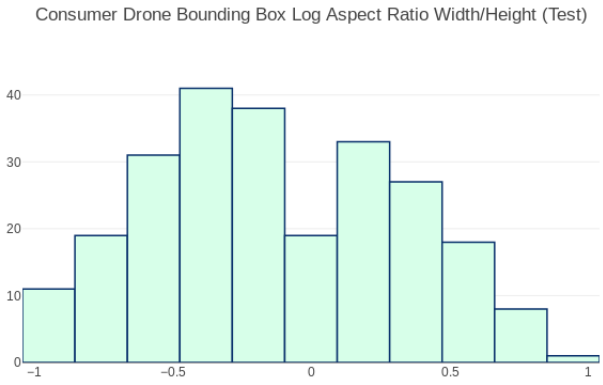

Figure 6: Consumer Drone Bounding Box Log Aspect Ratio Histogram - Test data

1.1.3 GeoScience Dataset

The Geoscience annotation also show no orientation prefernce with equal average width and height. The average size is also around 37 pixels, with minimum pixel size of around 11 pixels. This is a marked distinction from the previous datasets. The smaller bounding box sizes within this dataset might cause difficulties in the VGG16 field of view.

Table 3: GeoScience Drone Bounding Box Statistics - Data

	Mean	Std. Dev.	Min	Max	Population
Width	36.038	11.921	11	120	793
Height	37.318	10.420	10	82	793

The average size of the bounding boxes of the GeoSciences is half that of the two other datasets.

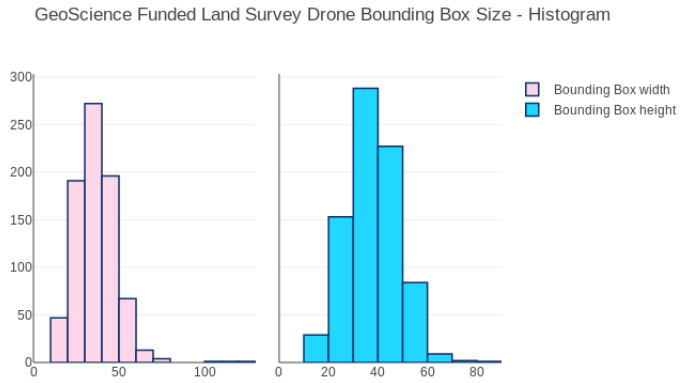

Figure 7: GeoScience Bounding Box Height and Width Histograms

As per qqplot show in figure 1.1.3 the data does not follow a normal distribution. This confirms the non-parametric tests to be performed on the population sampling.

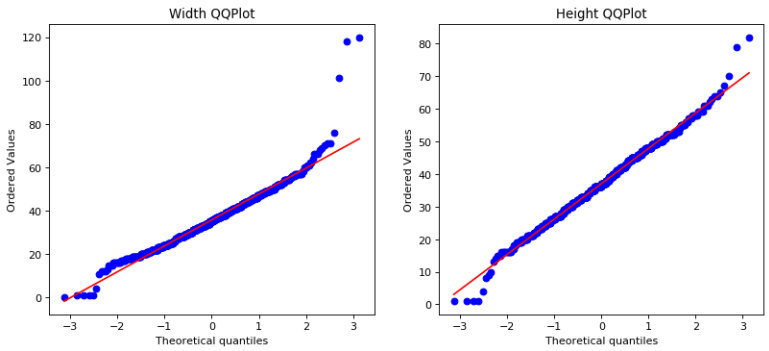

Figure 8: GeoScience Bounding Box Height and Width QQplot

The aspect ratio figure 1.1.3 has a similar shape as the train and data set, confirming the nature of the images being labelled.

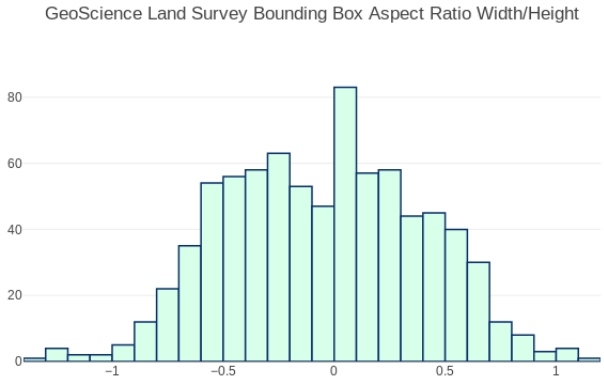

Figure 9: GeoScience Bounding Box Log Aspect Ratio Histogram

1.1.4 Population Testing

The train, test and Geoscience population will be tested for independance. Since the populations are not normally distrbuted, a non-parametric test will be used. A two-tail Mann Whitney test will be performed. The whole data in the datasets will be used. The scale will be in pixel size. A confidence interval $\alpha = 0.05$ will be used.

The null hypothesis will be formulated :

Null Hypothesis 1 *H_0: the distribution of the pixel sizes of the two datasets are equal*

Hypothesis 1 *H_A: the distribution of the pixel sizes is not equal*

The two comparisons will be made between train vs test and train vs geoscience dataset.

1.1.5 Train vs Testing Population

Testing the two datasets using Mann Whitney results with U = 69299.00 and a p-value of 0.092 (for width) and U = 66246.50 and a p-value of 0.533 (for height) . Therefore we fail to reject the Null Hypothesis H_0. The train and testing data set come from the same population. This is to be expected as the data comes from the same sensor and using the same land-survey techniques.

1.1.6 Train vs GeoScience Population

Testing the two datasets using Mann Whitney results with U = 383029.00 and a p-value of 0.0 (for width) and U = 400391.50 and a p-value of 0.0 (for height) . Therefore we reject the Null Hypothesis H_0. The train and geoscience dataset come from the different populations.

1.1.7 Statistical Results

The statistical tests show that the geoscience and the consumer drone data have different image sizes. This shows that the surveying techniques and parameters are different between geoscience data and consumer drone data. Comparing the geoscience images to that of the consumer drone, it can be deduced that the altitude of the surveys are different. This can also be the result of the drone operator knowledge of higher camera resolution.

An algorithm which is dependant on the size of the picture for successful recognition, will have difficulty correctly infering the smaller sized dataset. The ability of the human labeller to correctly label the objects, indicates that the shape information is present. This has to be exploited by the CNN. The algorithm must

be scale invariant or specifically tuned to the zoom levels of the geoscience dataset.

1.2 CNN Algorithm Testing

Four CNN architectures will be testing for maximum validation accuracy results. The one with highest validation accuracy and highest Litter recall will be chosen. Each algorithm will be tested pre-trained with Imagenet weights. The fully connected layer at the end of the convolutional layers will be stripped off, and replaced by

1. A connected layer with softmax activation,
2. A connected layer with relu activation and a connected layer with softmax activation,
3. Two connected layers with relu activation and a connected layer with softmax activation

The connected layer with softmax activation will two class output. The connection layer between the Convolutional Layer and the Fully Connected Layer will be chosen from a Flatten layer, a Max Pooling Layer and a Global Average Pooling Layer. The testing will use the default layer used in the algorithm Imagenet training. If the training procedure does not converge, other layers will be tested.

At this stage the trainable layers will be reserved only for the new fully connected layers. The best algorithm setup will be chosen. Further algorithm tuning will be performed by allowing more layers inside the convolutional blocks to be trained. Incremental training will be performed on previous trained models. An epoch length of 60 will be used for testing.

1.2.1 Dataset Preparation

One dataset will be compiled for all tests.

The data will be fed using Keras ImageGenerator with inbuilt data augmentation algorithms. The options used for testing are as follows:

1. Preprocessing Function: Keras provides algorithm specific preprocessing functions that transform an image to an output tensor of the same shape,
2. Shear range 0.2 : Angle of shear in degrees,
3. Horizontal Flip : Random horizontal flip of image,
4. Verticial Flip: Random vertical flip of image,
5. Zoom Range 0.2: Random zoom between 0.8 and 1.2 of the image size.

All algorithms will have a default input tensor size of $224x224$. The fill-mode parameter is set to true so that no black areas appear within the image. Later RCNN tuning will test more data augment parameters and parameter ranges.

Validation will only be processed. No augmentation of this set is performed.

Training Batch size : 150 Validation Batch size: 50

1.2.2 Gradient Descent Optimizers

RMSProp will be used for all the tests. The learning rate will be tuned according to the training and validation convergence speed. Convergence rate varies between 1e-4 to 1e-6.

1.3 Algorithm Test Results

1.3.1 Inceptionv3

1. : Inception with Max Pooling Layer, FCN 1-layer 1024 and Softmax Classifier

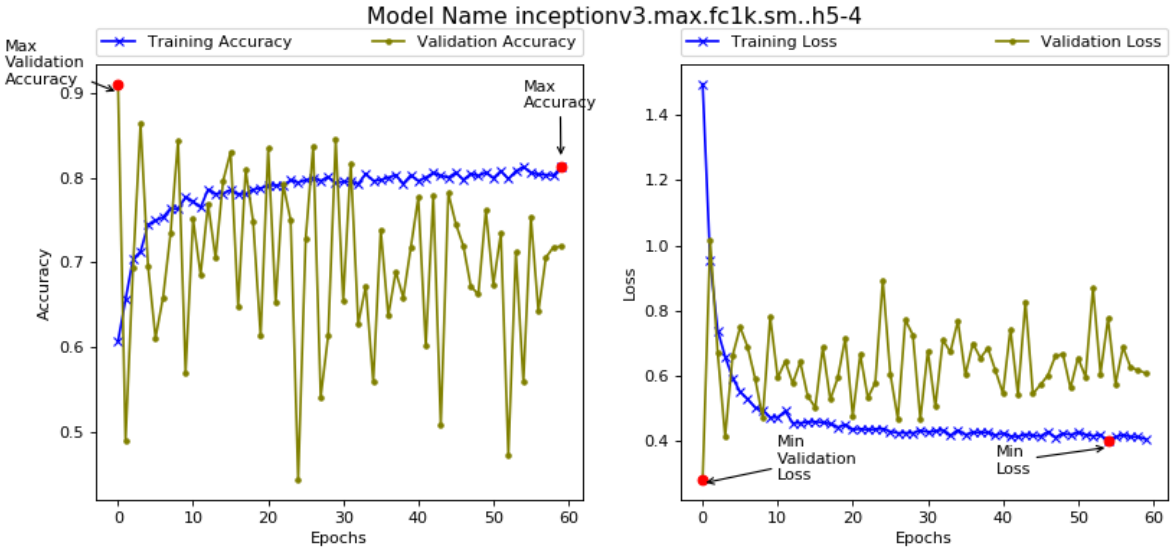

Figure 10: Inception Training Tasks Max Pooling Layer, FCN 1-layer 1024 and Softmax Classifier

The training procedure shows erratic validation training. Validation Accuracy immediately falls from the first epoch validation accuracy of 0.91. Learning rate is 1e-4, which could be too high for convergence. An average accuracy of value of 0.7 is acheived. The MaxPool layer shows a very erratic validation curve, whilst training curve increases gradually to a limit of 0.8. The MaxPooling layer will be dismissed.

2. : Inception with Global AveragePooling Layer and SoftMax Classifier

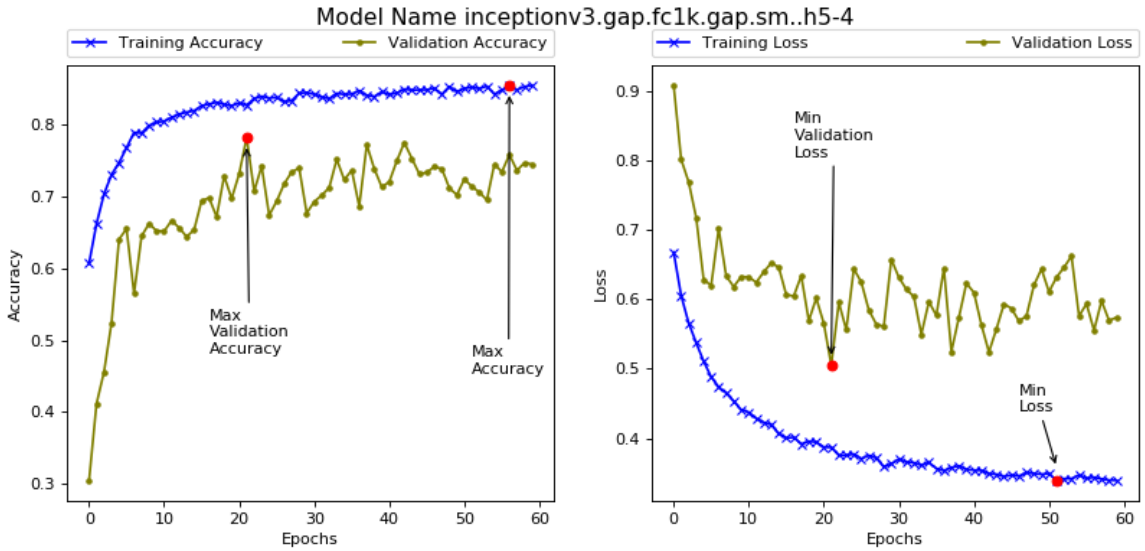

Figure 11: Inception Training Tasks GAP and Softmax Classifier - training Softmax layer

The InceptionV3 GAP layer maximum validation accuracy acheived when training on 4 layers, is of 0.781 accuracy at epoch 21. The number of trainable layers are increased to 37 layers, whilst using the same weights acheived earlier on.

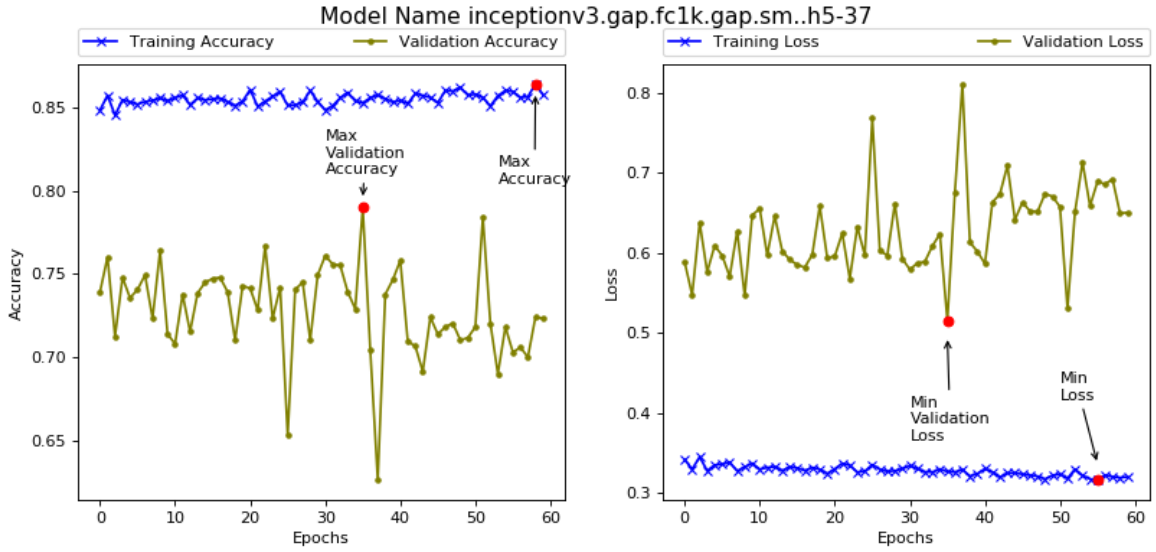

Figure 12: Inception Training Tasks GAP and Softmax Classifier - training last 37 layers

Validation accuracy increased to 0.79 at epoch 58. The number of trainable layers are increased to 37 layers.

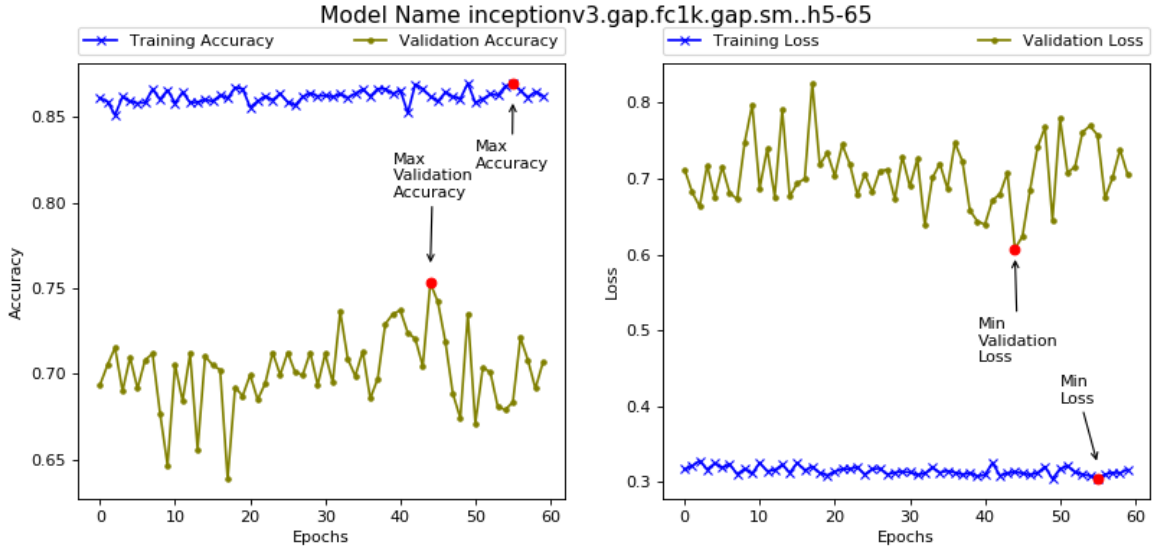

Figure 13: Inception Training Tasks GAP and Softmax Classifier - training last 65 layers

No further increase in accuracy has been noted, and the validation accuracy levels off at 0.7 mark average. To note the gradual improvement in training accuracy levels, during the training procedure , however the validation is more erratic. This shows a direct neural network to the convolutional output is unable to obtain the classifier features directly. The training accuracy levels off at a 0.87 mark,

showing no increase during the last 120 epochs of the total 180 epochs run using a Global Average Pooling Layer.

Incrementing the FCN to two layers produced an neural network without high value validation accuracy convergence.

1.3.2 Resnet-50

1.3.3 MobileNetv2

A successful mobilenetv2 implementation will help us increase the speed of all subsequent inferencing. The default depth multiplier of 1 and default number of filters in the layer will be used as per paper.

Two connection layers will be used Flatten and GlobalAveragePooling layer.

1. Mobilenet with Flatten Layer and Softmax Classifier

   Maximum Validation accuracy of 0.944 acheived at epoch 53. Training curve is gradual, whilst validation curve is constant with single degredatation points.

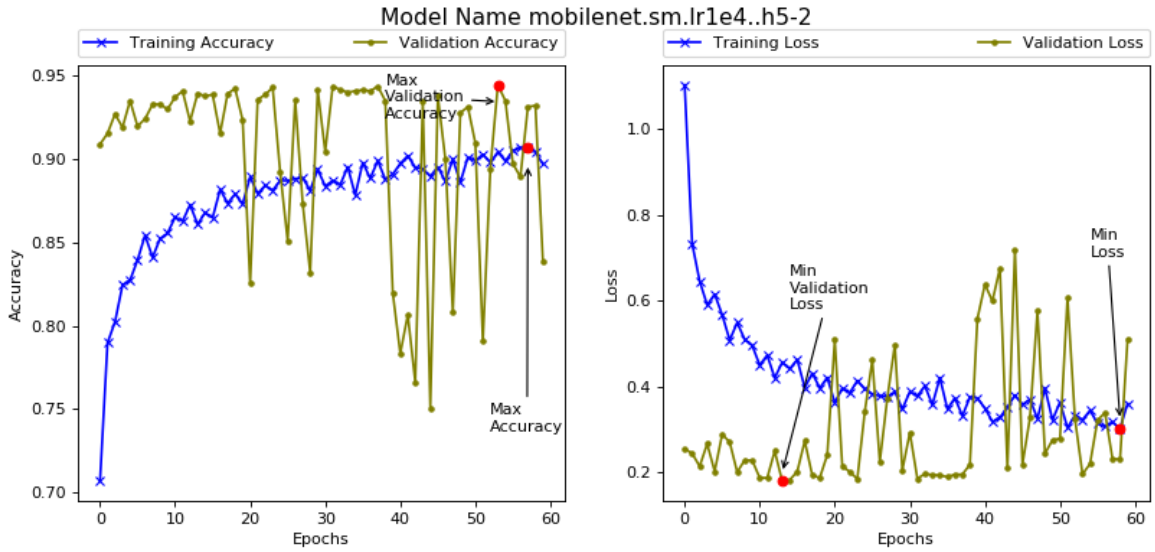

Figure 14: MobileNetv3 Training Flatten and Softmax Classifier - training Softmax only

Incremental training on previous architecture using 13 trainable layers. The maximum validation accuracy increased to 0.966. The validation accuracy is not incrementing, showing an overfitting of the training data. Training accuracy has a gradual increase from training epoch 40 onwards.

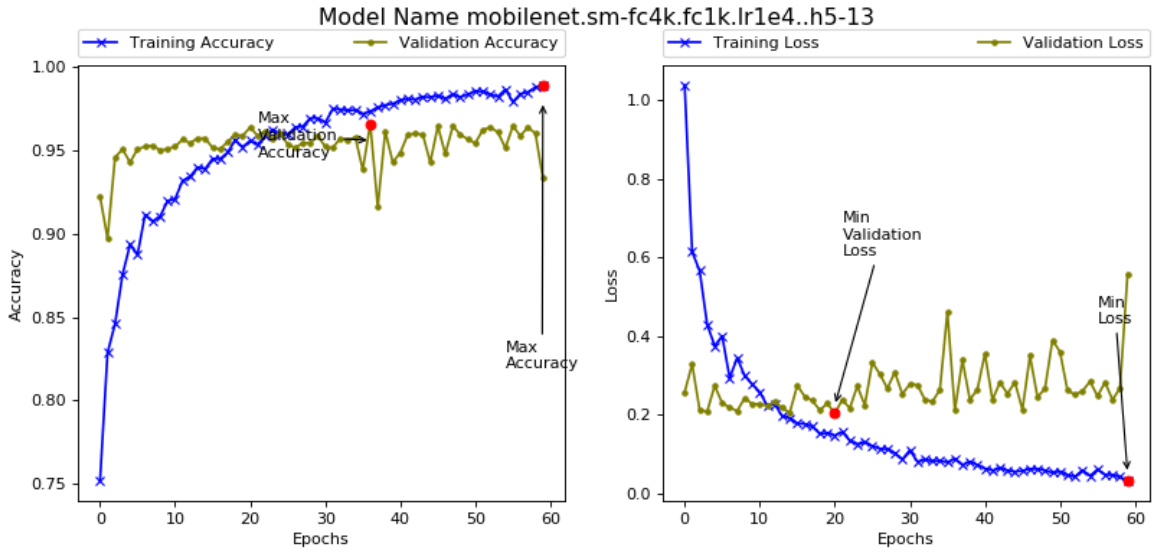

Figure 15: MobileNetv3 Training Flatten and Softmax Classifier - training 13 layers

A maximum validation accuracy of 0.987 is seen, where the curve is seen to level off.

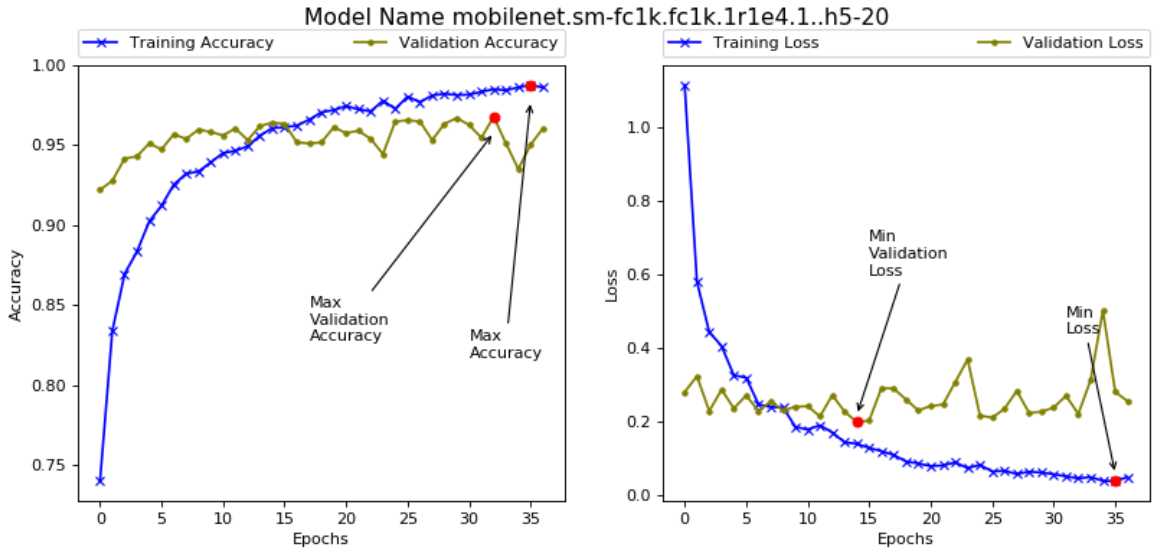

Figure 16: MobileNetv3 Training Flatten and Softmax Classifier - training 20 layers

1.3.4 VGG-16

1.4 R-CNN Evaluation

1.4.1 Test 1: Sliding Window - Baseline

Sliding Windows inferencing over whole image. Tile of 224x224 is cropped from original image from a 224-32 grid, with 32 being the overlap allowed between the two tiles to allow for detection of objects at the edge of the tile. The last row and last column are recalculated to allow for mismatching image height and width to grid.

The tile passed to the VGG16 inferencing engine, which outputs a 2 unit tensor having values for [nolitter,litter]. The value of litter is stored in the detection grid. Bounding Box co-ordinates are obtained using tile co-ordinates for the tiles which inference results is higher than a preset confidence level. The bounding box predictions evaluated against the ground-truth values.

VGG16 Training: A new VGG16 architecture will be trained on the dataset :dataset.v1. *Conv5*, *conv4* and *conv4* will all have trainable layers. Two fully connected layers with relu activation and a softmax layer (2 classes) will be appended to the VGG16 convolutional layer output. The CNN will be trained using checkpoint per epoch, early stopping based on Validation Accuracy metric with waiting for 10 epochs for validation accuracy improvement.

Table 4: Dataset.v1

	Litter	**Background**
Training	5174	2672
Validation	208	1172

Litter has been obtained from annotation data, whilst terrain obtained by sampling 10 images per scene, and vetted for presence of plastic or glass containers. Litter subdivided into 70% training and 30% validation. However, 70% training data has been augmented manually to 9 shifted positions.

VGG16 architecture training: The validation accuracy obtained was of **0.9833**.

Table 5: Validation Class Report

	Litter	**Background**
Precision	0.97	0.99
Recall	0.95	0.99

The precision/recall values are quite high for all the dataset. This maybe due to the limited number of cases present in the dataset. One has to note however that in RCNN the number of times the inference is done, where terrain is to be expected has a probability of 2000-3000 more than litter instances. Therefore the precision/recall obtained in the terrain 0.99/0.99 will have much more impact than the lower but less likely litter detection precision/recall of 0.97/0.95.

In a 4000 tile inference, 1% of incorrect predictions would provide 40 False Positive predictions. In a typical scene 3 litter objects, would give us an R-CNN precision value of $\frac{3}{3+40} = 0.069$

Consumer Drone Testing Results: **Average Precision 0.14** at IoU of 0.1. The Recall rate is 0.922 (objects located within a bounding box) or 0.859 (bounding boxes predicting objects). Due to the large size of the bounding box with respect to the size of the detected object there are occasions of a bounding box covering two objects. Two recall rates are being reported to denote the descrepancy between the two possible calculations.

This is a high recall rate indicating that litter is being detected successfully, and is very close to the validation litter rate detection. However the precision rate is very small at **0.0586**, as per example above.

1. Correct Predictions

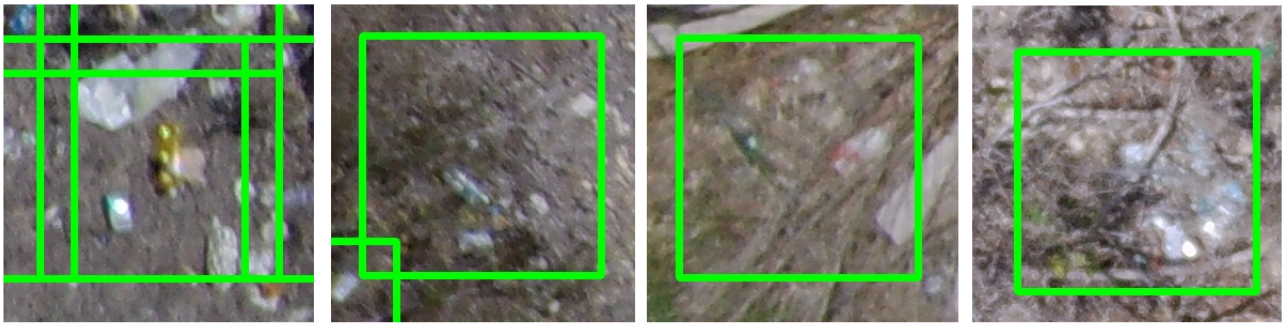

Figure 17: Sliding Window Correct Predictions

Figure 1 shows successful predictions on the Consumer drone test set. The large sliding window size and stride do not hinder the correct inferencing. The ratio of the size of the detected object is comparable to the inferencing window. Figure 1c and 1d show occlusion and deformations being detected adequately. To denote the specular aspect of the objects being detected.

2. Incorrect Predictions But Plausible

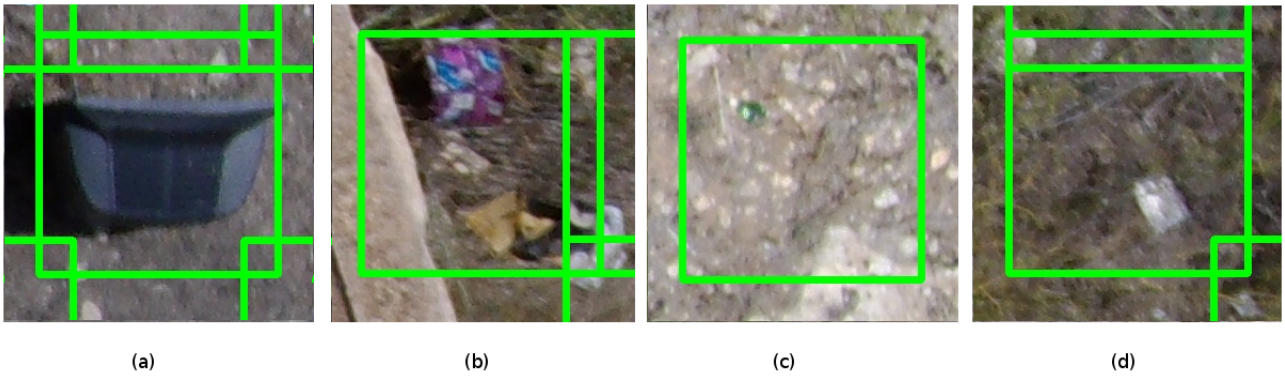

Figure 18: Sliding Window Incorrect put PlausiblePredictions

Figure 2 shows a number of predictions that have not been annotated as beverage bottles, but which the algorithm has marked them as litter. These objects are frequent in the detected dataset. These artefacts found in outdoor imagery degrade the precision value, even though they could be successfully tagged as litter, though not beverage bottles or containers. The common aspect of these artefacts is that they are man-made objects, distinct from the background. Objects small as in figure 2c are many. Human annotations of these small objects is very difficult.

3. Incorrect Predictions that can be ammended

Figure 19: Sliding Window Incorrect Predictions that could possibly be corrected

Figure 3 shows example of uncorrect predictions that could possible by corrected. Figure 3a shows a sky scene with branches that have been mistaken by the algorithm. This picture is evidently not an aerial image because sky can never be shown from a UAV with its camera shooting downwards. The imagery can easily be removed by filtering *sky* image tiles within the pictures. Figure 3 (b) and (c) are scenes depicting vegetation and branches that also have been incorrectly misdetected. The regular features of the twigs could be affecting the CNN. However, different to the *sky* images, plastic and litter can be found within *vegetation* imagery, therefore handling vegetation must be done by the RCNN. Figure 3d is an image with no clear plastic or man-made features. There is no clear explanation why the RCNN mistook this for litter. However, like *vegetation* RCNN can be made to detect this type of imagery as litter.

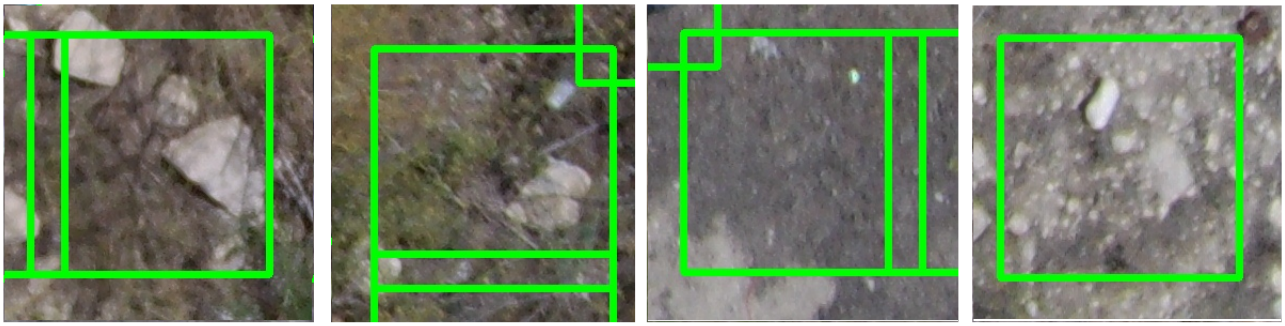

Figure 20: Sliding Window Incorrect Predictions

Figure **??**(a),(b) and (d) are images of stone incorrectly labelled as litter. Figure **??**(d) particularly depict stones with a specular features and shapes very similar to plastic or glass bottles. Surprisingly some scenes contain a lot of these stones, to the detriment of the algorithm precision. Figure **??**(c) show a very small feature affecting the classification of the CNN. This is the high sensitivity to the litter that has been designed within the dataset. This sensitivity might be the cause of the high False Positive number of the RCNN. Unfortunately regular features can be detected anywhere within the outdoor imagery but not be result of manufacture but by stone formations or shadows.

GeoScience Drone Testing Results: **Average Precision 0.02** at IoU of 0.1. The recall rate is 0.896

(objects located within bounding box) or 0.8225 (bounding boxes predicting objects. The average precision is much lower than the consumer drone test results. The recall rate is lower but one can still say that out of the annotated number of bottles a good percentage is being detected. This indicates that the algorithm has a good degree of scale invariancy, since the statistics have shown that the size of the bounding boxes are different. The precision of the R-CNN is lowering the average precision value. At a precision rate of 0.0091 one can say that only 1 in 100 predictions is correct. This partly stems from the larger number of tiles to be predicted, and partly due to the new terrain which is causing the algorithm to wrongly detect litter.

1. Correct Predictions

Figure 21: Sliding Window Correct Predictions from GeoScience Survey

The red underline is to highlight the position of the object

Figure 22: Normal text

Figure 1 shows the correct predictions by the algorithm that have been detected. As expected the size of the objects are much smaller from the objects found in the Consumer database set. To note that the image quality is superior than the drone imagery, and the definitition of both the objects and the terrain is better. The quality difference might have an impact becuase of the regular features learnt by the CNN algorithm. The CNN is invariant of the position of the object, and the overlap in figure 1(b) shows two nearby tiles detecting the same object.

2. Incorrect Predictions but plausible

Figure 23: Sliding Window Incorrect but plausible predictions from GeoScience Survey

Figure 24: Normal text

Here again we see that the algorithm is detecting very linear features and man-made shapes as being litter. Although one can correctly define them as litter the CNN algorithm was trained on beverage bottles and containers. The features found within these objects are being extrapolated by the CNN. The CNN, as in case of 2(b) has managed to detect them even under occlusion.

3. Incorrect Predictions that can be ammended

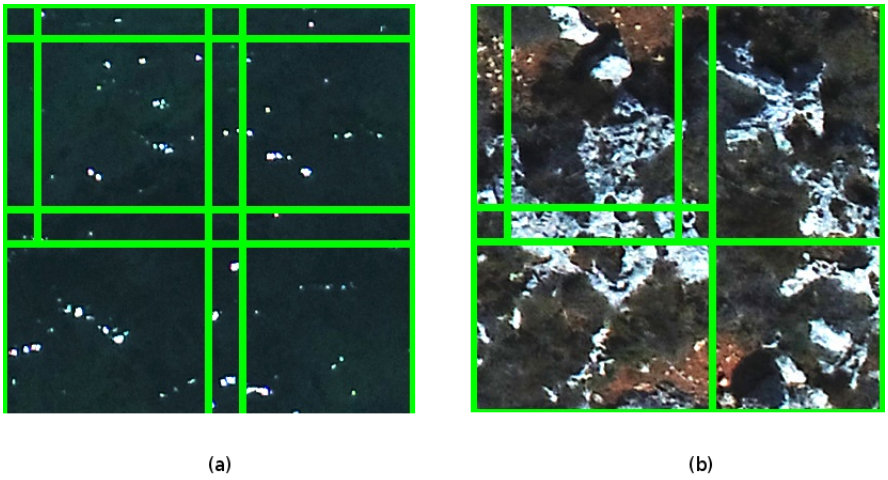

Figure 25: Sliding Window Incorrect predictions from GeoScience Survey (a) sea cover (b) rocky, grass and soil terrain

Figure 26: Normal text

The geosciences dataset has a number of terrains not present in the consumer drone dataset. This has affected majorly the CNN inferencing ability on this dataset, why the precision rate for the GeoScience dataset is so low. The terrains in figure 3 are an example of this imagery that are shown to be detected litter. Figure 3a is sea and the reflections on the ocean are being mistaken as specular reflections of bottles and glass. A large amount of incorrect predictions are being made, since the around 20% of the survey is carried out over water. Since the application is for land-detection of litter, we can easily omit these by filtering these images out. With regards to 3b the CNN also is having a very large

number of misdetections caused by unknown terrain. Removing such a terrain should be performed by modifying the litter detecting CNN, since plastic within this terrain should also be detected, and cannot be filtered out.

1.4.2 Comparison to other Small Object R-CNN

1.5 Test 2: Class Activation Mapping

Once an annotator starts reviewing the images from aerial shots he would deduce the size the objects he should expect. The RCNN by nature is designed to be shift invariant. However, through other means, one can determine the object size from CNN inference.

1.6 Test 3: Pre-Filter

1.7 Test 4: Dataset Engineering

1.8 Test 5: Field of View

1.9 Test 6: Further Data Augmentation

1.10 Test 7: Reduce Litter Sensitivity