Improving the Elephant Survey System Detection and Classification

Deon Joubert and Hannes Naude, Innoventix Consulting, 3 June 2015

The aim of the Elephant Survey System (ESS) is to provide a cost- and time effective estimate of the elephant population within a specified region. To achieve this aim, the system collects Infrared (IR) and visual band images from a suite of aircraft mounted cameras, as can be seen in the system overview diagram provided in Figure 1. IR images are processed to detect the possible location of elephants, after which these locations are projected onto the corresponding visual band images. Sub-images surrounding these locations are then processed by a classifier to determine whether the detection is an elephant or not. Thereafter a population estimation algorithm approximates the total number of elephants within the area where the images were collected

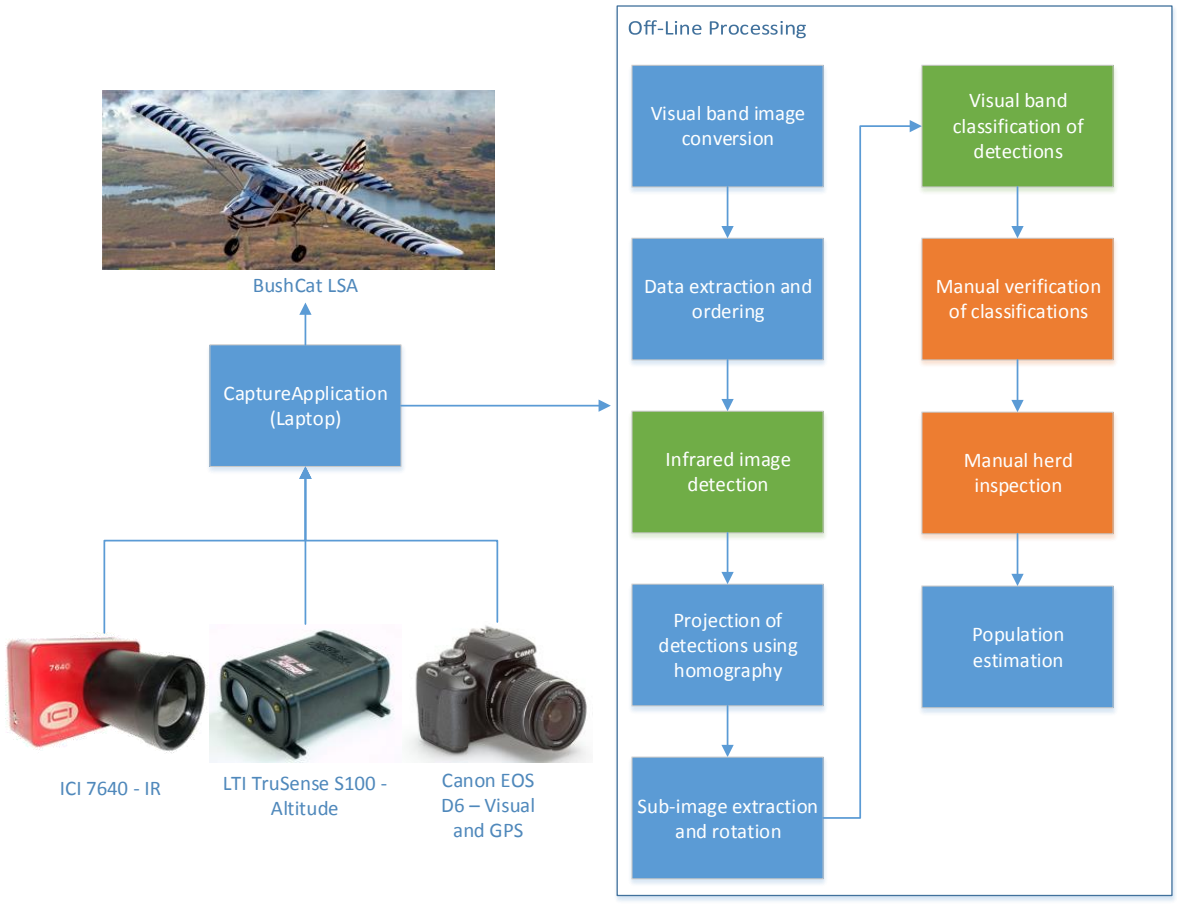

Figure 1. Overview of the Elephant Survey System. The components highlighted in green were investigated and improved, while those highlighted in orange will be introduced in future versions of the system.

Both the detector and classifier were found to perform unsatisfactorily during an evaluation of the ESS [1]. The detector was found to produce far too many detections, as can be seen in Figure 2, most of which would have been easily discounted by a human operator. As all of these detections had to be classified, the processing time became prohibitively long. The analysis and improvement of the detector so as to reduce the number of unnecessary detections is detailed in Section 1.

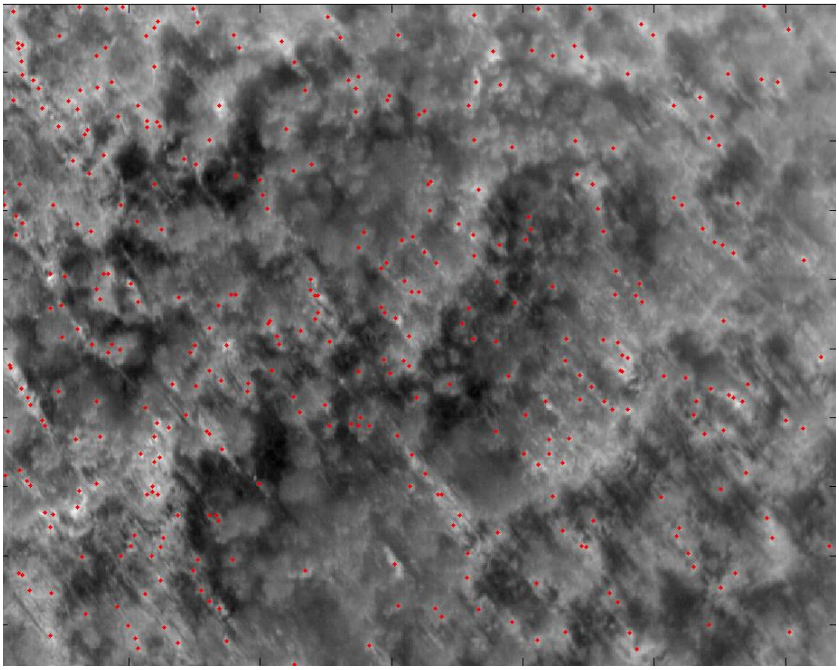

Figure 2. High number of detections on an Infrared image using the 2-stage CFAR detector.

A redesign of the classifier was made necessary for a number of reasons. Firstly, during the evaluation the classifier misclassified a number of detected elephants. Such misclassifications could severely affect the accuracy of the subsequent population estimation. Secondly, it was found that manual verification of the classifier output was viable and would serve to greatly increase the overall accuracy of the system. However, the classifier would have to be modified to produce more positive classifications, which would include both correctly and incorrectly labelled elephants, as manual rejection of false positives is far easier than finding an elephant among the expected high number of negative classifications. Thirdly, as new data had been collected, the classifier would have to be retrained to make use of this data. The redesign and retraining of the classifier is documented in Section 2.

The new detector and classifier were implemented as part of the ESS. The evaluation of this new ESS using collected transect data is described in Section 3. Conclusions and ideas for future work are presented in Section 4.

1 Improving the Detector

The ESS evaluated in [1] used a two stage constant false alarm (CFAR) detector to detect or that is to say find the possible locations of elephants within collected IR images. One of the first problems that was witnessed during the evaluation is the fact that the detector generated a vast number of detections in areas where there were no observable hot spots. Some images would generate thousands of detections. These would have to be individually extracted and classified. Obviously this has a major impact on computational load, and as such on runtime performance. Even more significantly, it means that even classifiers with low false alarm rates would still generate several false alarms per image.

The CFAR detector is intended to provide a constant false alarm rate. However, a quick glance at the detections generated in the images it was developed against versus the detections it produces on the images from the March 2015 Phinda transect survey [1], makes it abundantly clear that this design goal is not being met. The reason for this failure is the fact that the CFAR detector assumes that the background is Gaussian noise. Figure 3 shows an image from the September 2014 training set that the detector was initially developed and tested against [2], as well as its histogram. It is clear that the background is not Gaussian, but rather a bimodal distribution, with one mode corresponding to the visible hot soil and the other to the cooler vegetation. If the vegetation and the soil occupied different parts of the image, this would be acceptable, since the Gaussian assumption would hold locally within a single typical CFAR window and the CFAR detector would only perform poorly at the boundary between the two regions. However, the vegetation and soil are intermingled throughout the image and the Gaussian assumption never holds.

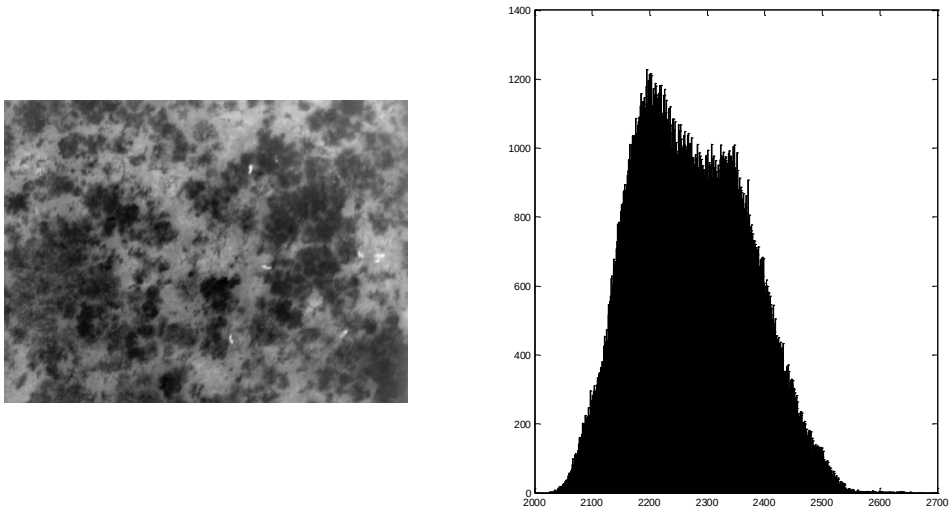

Figure 3. Example image showing bimodal background.

As the Gaussian assumption is invalid, the estimated variance of the background is higher than it should be and in order to still get some detections, one needs to apply a small gain factor on this variance when calculating the threshold. This is exactly what was done.

However, with such a gain, the system becomes overly sensitive when it encounters an image where the Gaussian background assumption does hold (or is violated less). An example of such an image, together with its histogram is shown in Figure 4.

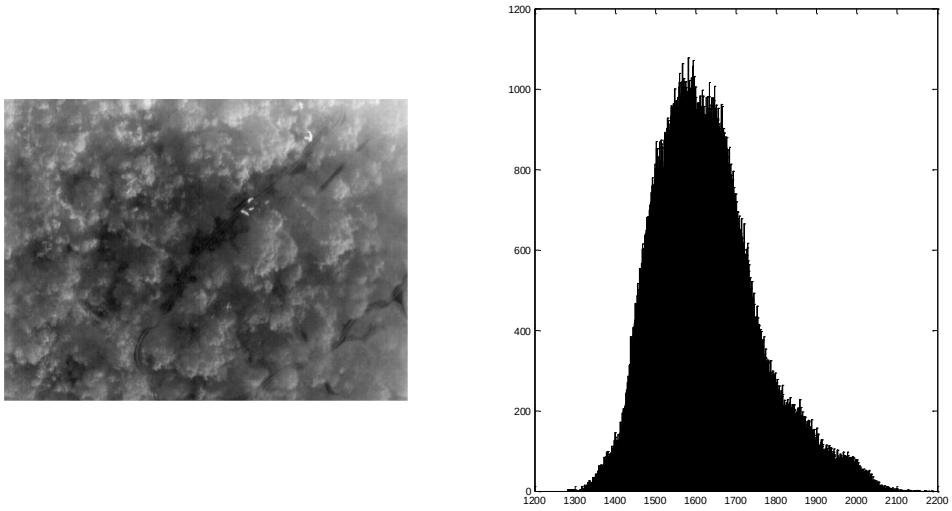

Figure 4. Image from the Phinda transect set, together with its intensity histogram.

Given the unknown and minimally constrained nature of the backgrounds the system will have to perform against, it seems unlikely that a truly constant false alarm rate can be achieved. Moreover, it is not entirely clear that a constant false alarm rate is actually required. On very uniform backgrounds we can realistically aim for a zero false alarm rate while still maintaining 100% probability of elephant detections at the design range, while most heterogeneous backgrounds will cause justifiable false alarms due to rocks, puddles of water or patches of dirt that are hotter than their surroundings and roughly in the shape and size of an elephant.

Perhaps a more realistic goal would be to design a detector that creates detections which correspond with areas where a human observer sees significant bright (warm) blobs at roughly the expected scale for elephants. This is a well-studied problem in image processing and can be solved by utilising any of a number of blob detection algorithms. Here we have chosen to use the Laplacian-of-Gaussian, followed by a fixed threshold, as described in [3]. Essentially we blur the image with a Gaussian filter with a variance chosen such that the scale matches roughly to the expected scale of an elephant detection. Then we calculate the gradient of this blurred image, and finally we calculate the divergence of this vector field. Locations with large negative divergence, correspond to the centres of blobs of the corresponding scale in the original image.

The new blob detection method provides vastly improved performance compared to the CFAR as can be seen from the results described in Section 3, but the following points need to be investigated in future work:

- It was found that better results could be obtained by utilising normalised IR images (i.e. images are scaled between 0 and 1) thereby adjusting the contrast of low-contrast images upwards and vice versa. This is difficult to justify from fundamentals.

- It is expected that performance can be significantly improved by utilising a scale invariant Laplacian-of-Gaussian transform with extrema selection. This approach was considered but not followed because we are only interested in blobs at a given scale. However, it was only realised later that even if only a single scale is of interest, multi-scale detection is still superior because it allows explicit rejection of blobs at the wrong scales rather than simply relying on the weak suppression provided by the Gaussian filter.

2 Improving the Classifier

In the implementation of the ESS described in [2], two Viola-Jones classifiers were used to classify or that is to say label IR detections as elephants or not. The detections referred to here are the extracted visual band sub-images. A detection was only accepted as an elephant if it was marked as positive by both classifiers. This classification system was developed and trained to both increase the number of elephants correctly identified (the true positive rate) and reduce the number of false positives.

However, very few elephants were detected during the evaluation described in [1] and of these elephants there were several that were misclassified. If the system classification is to feed directly into a population estimation algorithm then false positives and false negatives are roughly equally harmful. However, if we anticipate using human operators as a second filter to remove false positives, then the cost of producing a false positive is vastly reduced (since a human can reject any given false positive with roughly a man-second worth of labour). Unfortunately there is no analogous way to cheaply correct false negatives, since false negatives make up a very small part of the total negative set.

Therefore, the anticipated presence of a human-aided final filtering stage affects our classifier design in that we should bias the classifier towards positive (elephant) classifications in order to lower our false negative rate as much as possible. The false positive rate (which will rise as a result) need only be limited so as to not overwhelm the human operators. As the desired outcome of the classifier had changed a redesign of the classifier was required, which in turn required additional research.

The research conducted is summarised in Section 2.1. Thereafter the evaluation method with which developed classifiers were compared to one another is explained in Section 2.2. The various classifiers developed are compared in Section 2.3.

2.1 Research Highlights

The first research goal was to better understand the Viola-Jones algorithm and how it should be implemented. Thereafter the role of colour spaces in classification was investigated, initially to inform the choice of single-channel image to use as input to the Viola-Jones algorithm but soon as a classification tool in itself. The Random Forest classifier was then studied as a replacement to the Viola-Jones algorithm as it can utilise a broader and more varied feature space. Finally, the MPEG7 descriptors were studied as possible features for use in a Random Forest classifier.

2.1.1 The Viola-Jones Classifier

A Viola-Jones classifier consists of a cascade of weak classifiers which use increasingly complex features to reject incorrect detections [4]. Those detections which remain after having been processed by all stages of the cascade are classified as positive. The weak classifiers are trained to have a fixed false positive rate by selecting a small subset from a set of features. A set of Haar-like features are specified in [4], while the MATLAB *CascadeObjectDetector*, which uses the Viola-Jones

algorithm, can also use either the Local Binary Pattern (LBP) or Histogram of Gradient (HOG) feature sets [5].

The MATLAB object "detector" uses a sliding window to classify each part of a query image to find that part which contains a desired object. The size of the sliding window is varied so as to find the object at different scales. However, the aspect ratio of the sliding window must remain constant. Therefore, the implementation is sensitive to the angle of the object. In the ESS, this is compensated for by first rotating all IR detections so that their longest axes are vertical.

An alternative method to overcome the rotational sensitivity is found in [6]. Several Viola-Jones classifiers are trained to find faces rotated at fixed angles. When new sub-images are to be classified, a decision tree is used to determine the most likely angle and the appropriate classifier is then used. Another method is to train a single Viola-Jones classifier with images rotated in a range of angles while also extending the Haar-like feature set with additional diagonal features. Invariance to rotations of ±15° have been achieved using this method [7].

The Haar-like, LBP and HOG features are all extracted from single-channel images, typically grey-scale images. Such images are not guaranteed to contain the necessary information to correctly classify a detection. To overcome the problems posed by variations in illumination and shadows, the difference between the red and green colour channels is used in [8] as input to a Viola-Jones classifier trained to find the faces of lions. From [8] it was seen that the use of colour with the Viola-Jones algorithm could influence the success of the classifier and required further investigation.

2.1.2 Colour-Based Classification

A combination of colour modelling and morphological filters are used in [9] to find marine animals from aerial images. The ratio between the red channel and the sum of the green and blue channels was used as part of a thresholding mechanism to find dugongs underneath the surface of the ocean. The approach was improved in [10] by using a neural network classifier to build a more sophisticated colour model of the dugongs.

The automated classification of elephants in videos taken at ground level was investigated in [11]. Texture and shape based classification methods were not used as elephants do not have noticeable skin patterns and the natural bush environment often obscures viewed elephants. Colour-based methods were considered as elephants are easily identified by their characteristic grey colour. However, the perceived colour can vary greatly due to illumination, shadows, dust and mud covering the elephants. A colour model was generated to overcome the problem posed by these factors by training a support vector machine (SVM) with images of elephants and natural backgrounds.

Colour modelling has the potential to improve the classifier of the ESS. From [10] and [11] it can be seen that an effective colour model must be trained using a classification model. An interesting application of colour modelling is presented in [12], where a skin-detection algorithm for online videos was developed by training a colour model. Here it was found that using a Random Forest to train the model resulted in better performance than when other methods were used, such as neural networks or SVMs.

2.1.3 Random Forest

A decision tree is a simple classification mechanism in which an input sample is classified into one of several categories based on the path followed through a series of thresholding nodes. A decision tree can be automatically generated by recursively creating nodes until a desired performance has

been reached. Each node is created by randomly selecting a subset of features as well as a threshold value for those features. Random forests consist of an ensemble of such decision trees and classify samples based on the majority vote from the trees [13]. It is important to note that each tree is trained using only a randomly selected subset of the features and training data so as to prevent correlation between the trees.

One of the advantages of the Random Forest classifier is that training and classification is fast [14]. Another advantage is that a variety of feature types can be used to build a classifier. An image classifier is trained in [14] to differentiate between 256 classes using scale-invariant feature transform (SIFT) and local shape features. A combination of colour, edge response and height information is used in [15] for semantic classification of aerial images. The raw pixel value from an image is used to train a Random Forest variant in [16].

Although the Random Forest classifier can accept and combine any types of features, it was decided to not attempt to use local features, such as SIFT descriptors, as these would most likely not be effective within the constraints of the ESS. Global features, as represented by the suite of Moving Picture Experts Group (MPEG) descriptors, were investigated instead.

2.1.4 MPEG-7 descriptors

The aim of the MPEG-7 visual standard was the formulation of a standardised non-text method for describing the content of images and video so that applications could be built on this standard to identify, categorize and filter such multimedia [17]. As part of the standard a number of motion, shape, texture and colour descriptors were formulated. For a full listing please refer to [17].

A content-based image retrieval system was developed in [18] that used the MPEG-7 scalable colour, colour layout and Edge Histogram (EHD) descriptors. In [19] a cartoon image classification system was developed and it was found that the Colour Structure Descriptor (CSD) produced the best results. The CSD is a type of histogram which encodes both the frequency of occurrence and spatial structure of colours within an image [20]. A metal part defect classification system was built using the MPEG-7 descriptors and a nearest-neighbour classifier in [21]. Here the CSD and the Homogeneous Texture Descriptor (HTD) were found to be the most effective.

2.2 Evaluation Methodology

From Figure 1 it can be seen that the ESS projects the detections in the IR image to the visual band image and classifies an image block surrounding the detection. A dataset of such blocks was assembled from the September 2014 data used to train the original classifier [2] as well as the March 2015 transect and non-transect data described in [1]. The blocks were identified as positive if it contained at least one elephant and negative if there was no elephant present. Negative blocks were selected from the raw data so as to present typical bush, veld and forest backgrounds as well as the known problem cases such as man-made structures and ant hills. All possible positive blocks were included in the dataset. The positive blocks were rotated so that the major axis of the most central elephant was vertically aligned.

The dataset was then divided into training, validation and testing subsets. The composition of the dataset is summarised in Table 1. From the table it can be seen that the number of negative images is far greater than the number of positive images, which is due to the fact that training the Viola-Jones classifier requires far more negative than positive images [5].

Subset of data	Percentage of all images	Number of positive images	Number of negative images
Training	50%	304	5034
Validation	25%	152	2517
Testing	25%	152	2517

Table 1. Composition of dataset used for classifier training and evaluation.

The training subset was used to train the classifiers and the validation subset was used to evaluate the resultant classifier and tune the training parameters. The testing subset was then used to evaluate the classifier again so as to assess the generality of the classifier.

The following metrics were used to evaluate each classifier:

- TP - number of true positives detected
- FP - number false positives detected
- Recall - fraction of true positives detected
- F1 Score- a combined measure indicating the overall performance of the classifier

The following formulas were used to compute the metrics:

$$Recall = \frac{TP}{TP + FN}$$
$$Precision = \frac{TP}{TP + FP}$$
$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

where FN indicates the number of false negatives classified. Note that a classification was deemed to be positive if at least one elephant was detected in a block, regardless of the location of the elephant in the image. As it is expected that the positive classifications will be reviewed by a human operator, it was only necessary to identify whether an image contained a possible elephant so as to flag the image for review.

2.3 Implemented Classifiers

As it is expected that the IR detector will produce fewer detections and a user verification step will be introduced, the purpose of the training has changed from producing a good overall classifier to one that has a high true positive detection rate and an acceptable false alarm rate. This approach was also followed in [11] where the classifier was also biased in this manner.

Various Viola-Jones and Random Forest classifiers were trained using the database described in Section 2.2. As the classifiers were trained using the same dataset, their performance could be directly compared to one another to determine the best classifier for use in the ESS.

2.3.1 Viola Jones classifiers

A series of Viola-Jones classifiers were trained using the MATLAB *CascadeObjectDetector* object [5]. It was observed that the Viola-Jones classifiers could only be trained to a maximum of eight cascade levels as too many negative images were rejected after having been processed by the previous cascade levels for further cascade level training.

The results of the evaluation of several of the newly trained and the original ESS classifiers [2] can be seen in Table 2. Only the results from those classifiers with a high recall or true positive rate and a

reasonable false positive rate have been included in the table. From the table it can be seen that the combination of the original classifiers has a good overall F1 score but a low recall.

The *Grey LBP* classifiers were trained using the grey-scale versions of the coloured image blocks and the LBP feature set. The LBP feature set was used as it trained classifiers faster than the Haar-like feature set. The *Hue LBP* classifier was trained by first converting the colour image to the Hue-Saturation-Value (HSV) colour space and then using the Hue-channel to train the classifier.

	Validation				Testing			
Classifier Description	**TP**	**FP**	**Recall**	**F1**	**TP**	**FP**	**Recall**	**F1**
Original LBP	132	161	0.868	0.593	136	179	0.895	0.582
Original Haar	83	264	0.546	0.333	82	308	0.539	0.303
Original Combination	76	77	0.500	0.498	78	104	0.513	0.467
Grey LBP v1	147	783	0.967	0.272	143	810	0.941	0.259
Hue LBP v0	150	694	0.988	0.301	143	710	0.941	0.285

Table 2. Evaluation results for the Viola-Jones classifiers.

From Table 2 it can be seen that the *Hue LBP* classifier has a higher recall and F1 score than the *Grey LBP* classifier on the validation subset while having fewer false positives on the testing subset. Therefore the Hue-channel of a block can be stated as producing a somewhat better classifier than the grey-scale image.

Both of these classifiers have better recall rates than the original classifiers but far more false positives. Whether the false positive rate is acceptable would be dependent on the number of detections generated, although the initial impression is that it is too high.

2.3.2 Random Forest

A number of Random Forest classifiers were trained using the dataset described in Section 2.2. During training it was perceived that the number of negative images used during training overwhelmed the algorithm in that the classifier would label all evaluation image blocks as negative. After some experimentation, the number of negative training images used was reduced to approximately three times the number of positive images available to prevent this from occurring.

The classifiers were initially trained used the raw pixel values, converted to grey-scale, Hue channel and the LAB red-green opponency channel, as inputs. However, these classifiers did not produce satisfactory results. Thereafter various methods were used to reduce the number of pixels in the image block while using all available colour channels in both the HSV and RGB colour spaces as inputs to the classifier. These methods included sub-sampling the block image to various sizes and developing a sliding-window system similar to the MATALB *CascadeObjectDetector.* Although the performance of the classifiers improved slightly, the results were still worse than those reported for the Viola-Jones classifiers.

After the unsatisfactory performance of the raw pixel classifiers it was decided to train classifiers using the MPEG-7 colour and texture features described in Section 2.1.4. The C++ implementation of the MPEG-7 feature extraction methods from [22] was adapted for use with MATLAB. Numerous classifiers were trained using the various MPEG descriptors, for which the evaluation results of the selected classifiers can be seen in Table 3.

	Validation				Testing			
Classifier Description	**TP**	**FP**	**Recall**	**F1**	**TP**	**FP**	**Recall**	**F1**
CSD	147	461	0.967	0.387	145	436	0.954	0.396
HTD	111	478	0.730	0.300	99	489	0.651	0.268
CSD and EHD	147	516	0.967	0.361	144	595	0.947	0.364
CSD Rotated	152	879	1.000	0.257	152	857	1.000	0.262

Table 3. Evaluation results for the Random-Forest classifiers.

Of the colour-based features used during the training of classifiers, the CSD was found to produce the best results. The HTD produced the best results for the texture-based Random Forest classifiers. These results are in accordance with those reported in [19] and [21]. Various grouping of features were also used as input to classifiers, and of these the combination of the CSD and EHD features generated the most successful classifier. However, overall the use of the CSD on its own produced the most promising classifier, competing very favourably with the Viola-Jones classifiers.

In order to address the imbalance between the positive and negative training images as well as to make the classifier more robust, the training dataset was augmented with several rotations of all the positive images. From Table 3 it can be seen that the *CSD Rotated* classifier has a perfect recall score, however the false positive rate is quite high.

3 Evaluation on Recorded Data

The newly developed blob detector and Random Forest classifier were implemented and integrated into the ESS. The new ESS was evaluated by having the system process the first hour of the third transect recorded in March 2015. This dataset, *03_04_2015_06h*, contained 1026 images, with 3 of these images containing 6 elephants each.

Three detectors were evaluated: The original two stage CFAR detector, the new blob detection based method described in Section 1 and a version of this detector where the input image is first normalised. The results of the evaluation are displayed in Table 4.

Detector	Total number of detections	Elephants detected	Elephants missed
Two Stage CFAR	65664	14	4
Blob Detection	10508	18	0
Blob Detection Normalised	2898	14	4

Table 4. Comparison of detector performances on the 03_04_2015_06h dataset.

From the table it can be seen that the blob detector has a lower total number of detections than the two stage CFAR detector while also successfully detecting all elephants in the test data. This indicates that the new detector is more accurate than the original. In Figure 5 a comparison of the blob detector and the two stage CFAR detector is shown, where it can clearly be seen that the blob detector has more accurate detection capabilities. The normalised blob detector has even fewer detections than the other two detectors but misses a number of elephants, which would negatively impact the accuracy of the final population estimate and therefore precludes its use in the ESS.

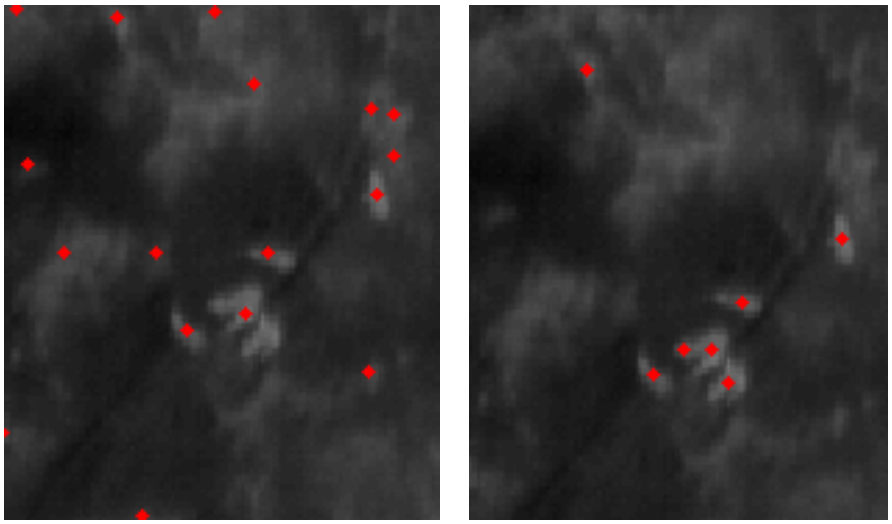

Figure 5. Comparison of detections made by the original 2-stage CFAR detector (left) and the blob detector (right) on image 06_49_27. Note the reduced number of detections and higher accuracy of the blob detector.

The classification results for various configurations of the ESS are shown in Table 5. Here it can be seen that the combination of the blob detector with the Random Forest *CSD Rotated* classifier (configuration *F*) produced extraordinary results. Configuration *G*, where the detected sub-images were classified without rotation, also produced very good results. However, this should be expected as the *CSD Rotated* classifier was trained on images taken from *03_04_2015_06h*. Overfitting cannot be discounted as an explanation for the good results and therefore these results cannot be used on their own to evaluate the classifier[1].

Configurations	A	B	C	D	E	F	G
Detector	Two stage CFAR	Two stage CFAR	Blob detector	Blob detector	Blob detector	Blob detector	Blob detector
Classifier	Original VJ	CSD OOB	Original VJ	CSD OOB	CSD OOB 0.9	CSD Rotated	CSD Rotated[1]
Detections	65664	65664	10508	10508	10508	10508	10508
Classifications	213	54	90	16	264	198	201
Total Detected Elephants	14	14	18	18	18	18	18
Total Identified Elephants	5	5	9	7	13	17	14
Total Missed Elephants	9	9	9	11	5	1	4

Table 5. Comparison of classifier performances on the 03_04_2015_06h dataset. Original VJ indicates the Viola-Jones classifier trained in [2]. [1]Configuration F uses the CSD Rotated classifier but without pre-alignment of the extracted sub-images.

To produce a valid comparison to the original ESS implementation, it was decided to create configuration *B*, which used the two stage CFAR detector together with an out of bag (OOB) Random Forest *CSD* classifier (*CSD OOB)* which was designed to only use those decision trees that were not

[1] The results presented in Section 2 can however be taken at face value as there was proper separation between training, validation and testing data.

trained on an extracted sub-image to classify that particular sub-image. The *CSD* classifier was modified instead of the *CSD Rotated* classifier as determining which trees were trained with what rotated version of a sub-image would have been prohibitively difficult.

From the results it can be seen that the *CSD OOB* classifier in configuration *B* produced fewer false positives but an equal amount of true positives to the original Viola-Jones classifier (configuration *A)*. However, upon investigation of the results it was found that the Random Forest classifier identified an elephant on the *06_49_27* image, which the Viola-Jones classifier of configuration *A* had missed. As a manual verification step is to be included in the ESS, it is very important to at least detect one elephant on an image so that the human operator can identify the other elephants in the herd.

Both the original Viola-Jones and the *CSD OOB* classifiers were combined with the blob detector (configurations *C* and *D*). From the table it can be seen that for both of these configurations more elephants were successfully classified. Both configurations detected elephants in all three images. Interestingly, the Viola-Jones classifier correctly classified more elephants than the *CSD OOB* classifier.

As the *CSD OOB* classifier made very few positive classifications, it was decided to modify the classifier so a sub-image is classified as true if the average of the tree scores for the negative class was less than 0.9 (the classifier had to be very sure that the sub-image was not an elephant). Configuration *E* shows that more elephants were correctly detected. While the number of false positives did rise, it is still within a manageable amount.

4 Conclusion

The detector and classifier of the ESS were redesigned to improve the overall performance of the system as well as to adapt to the new paradigm of manual verification of classification results. The original two stage CFAR detector was replaced with a blob detector, while a Random Forest classifier was developed to replace the original Viola-Jones classifier.

From the results it can be seen that the blob detector is a great improvement. It detects the location of elephants more accurately and with fewer missed detections which leads to the improvement in results for whichever classifier is used.

The *CSD Rotated* Random Forest classifier was found to be better classifier in terms of true positive identification within a valid testing environment. Unfortunately, proper comparison to the original Viola-Jones implementation was hampered by the lack of collected data containing elephants. Therefore it cannot be definitively stated that the Random Forest classifier is the better one until more data has been collected. However, given the positive results using the threshold adjusted out-of-bag classifier *CSD OOB 0.9*, it is believed that the Random Forest classifier does show significant promise. The Random Forest classifiers were also able to correctly classify elephants in each of the transect image containing elephants which is exactly what is needed for a manual herd inspection step.

In future more care should be taken in constructing the training dataset. From the training it was found that the Viola-Jones classifier required far more negative than positive images, while for the Random Forest classifier the more equal the better. Furthermore, difficulty was experienced in training a Random Forest classifier from the non-transect data to work on the transect data. This can be attributed to the fact that the original training images and the non-transect images collected in December of 2014 all have the same appearance, which prohibits the generation of a general colour model. It is believed that with more data, the Random Forest classifier can be made even more accurate and can be properly evaluated with a greater variety of data.

5 Bibliography

[1] D. Joubert, “Evaluation of the Elephant Survey System,” Innoventix Consulting, Centurion, 2015.

[2] S. Joseph-Malherbe and J. Kahlon, “System Design Improvements Phase 2,” iKubu, 2014.

[3] T. Lindberg, “Detecting Salient Blob-Like Image Structures and Their Scales with a Scale-Space Primal Sketch: A Method for Focus-of-Attention,” *International Journal of Computer Vision,* vol. 11, no. 3, pp. 283-318, 1993.

[4] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001.

[5] Mathworks, “Train a Cascade Object Detector,” Mathwords, [Online]. Available: http://www.mathworks.com/help/vision/ug/train-a-cascade-object-detector.html. [Accessed 11 May 2015].

[6] M. Jones and P. Viola, “Fast multi-view face detection TR-20003-96,” Mitsubishi Electric Research Laboratory, 2003.

[7] M. Kolsch and M. Turk, “Analysis of rotational robustness of hand detection with a viola-jones detector,” in *Proceedings of the 17th International Conference on Pattern Recognition*, 2004.

[8] T. Burghardt and C. Janko, “Real-time face detection and tracking of animals,” in *8th Seminar on Neural Network Applications in Electrical Engineering*, 2006.

[9] F. Maire, L. Mejias, A. Hodgson and G. Duclos, “Detection of dugongs from unmanned aerial vehicles,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, 2013.

[10] F. Maire, L. Mejias and A. Hodgson, “A convolutional neural network for automatic analysis of aerial imagery,” in *International conference on Digital Image Computing: Techniques and Applications*, Wollongong, 2014.

[11] M. Zeppelzauer, “Automated detection of elephants in wildlife video,” *EURASIP Journal on Image and Video Processing,* vol. 46, 2013.

[12] R. Khan, A. Hanbury and J. Stoettinger, “Skin detection: A random forest approach,” in *17th IEEE International Conference on Image Processing (ICIP)*, 2010.

[13] L. Breiman, “Random Forests,” *Machine Learning,* vol. 45, no. 1, pp. 5-32, 2001.

[14] A. Bosch, A. Zisserman and X. Muoz, “Image classification using random forests and ferns,” in *IEEE 11th International Conference on Computer Vision*, 2007.

[15] S. Kluckner, T. Mauthner, P. M. Roth and H. Bischof, “Semantic classification in aerial imagery by integrating appearance and height information,” in *Computer Vision–ACCV*, 2009.

[16] R. Maree, P. Geurts, J. Piater and L. Wehenkel, “A generic approach for image classification based on decision tree ensembles and local sub-windows,” in *ACCV*, 2004.

[17] T. Sikora, “The MPEG-7 visual standard for content description - an overview,” *IEEE transactions on circuits and systems for video technology,* vol. 11, no. 6, pp. 696-702, 2001.

[18] C. Iakovidou, N. Anagnostopoulos, A. C. Kapoutsis,, Y. Boutalis and S. A. Chatzichristofis, “Searching images with MPEG-7 (& mpeg-7-like) powered localized descriptors: the SIMPLE answer to effective content based image retrieval,” in *12th International Workshop on Content-Based Multimedia Indexing*, 2014.

[19] J. Kim, S. W. Baik, K. Kim, C. Jung and W. Kim, “A cartoon image classification system using MPEG-7 descriptors,” in *Third International Conference on Artificial Intelligence and Computational Intelligence*, Taiyuan, 2011.

[20] D. Messing, P. Van Beek and J. H. Errico, “The MPEG-7 Colour Structure Descriptor: Image description using colour and local spatial information,” in *IEEE International Conference on Image Processing*, Thessaloniki, 2001.

[21] J. Pakkanen, A. Ilvesmäki and J. Iivarinen, “Defect Image Classification and,” in *Proceedings of the 13th Scandinavian Conference on Image Analysis*, Göteborg, 2003.

[22] M. Bacstan, H. Cam, U. Gudukbay and O. Ulusoy, “BilVideo-7: An MPEG-7-Compatible Video Indexing and Retrieval System,” *IEEE MultiMedia,* vol. 17, no. 3, pp. 62-73, 2009.