Proceedings Paper:
Jaramillo-Avila, U., Aitken, J.M. orcid.org/0000-0003-4204-4020 and Anderson, S.R. orcid.org/0000-0002-7452-5681 (2020) Visual saliency with foveated images for fast object detection and recognition in mobile robots using low-power embedded GPUs. In: Proceedings of the 19th International Conference on Advanced Robotics (ICAR). 2019 19th International Conference on Advanced Robotics (ICAR), 02-06 Dec 2019, Belo Horizonte, Brazil. IEEE , pp. 773-778. ISBN 9781728124681

https://doi.org/10.1109/ICAR46387.2019.8981557

Visual saliency with foveated images for fast object detection and recognition in mobile robots using low-power embedded GPUs

Uziel Jaramillo-Avila[1], Jonathan M. Aitken[1], and Sean R. Anderson[1]

***Abstract*— This paper presents a visual saliency algorithm for fast object detection and recognition in mobile robots using low power graphics processing units (GPUs), based on human vision foveation. The step of image foveation enables the use of small images, which leads to a much reduced number of computations in deep convolutional neural networks and consequent increase in frame-rate. We demonstrate how using a simple foveated downsampling method, we can maintain a detection-recognition performance level similar to the level at larger image resolutions, even when transforming from 416x416 to 128x128 pixels, for a small high acuity region of the image, which can lead to a 4× speed up in frame rates, maintaining a relatively stable mean Average Precision. The visual saliency algorithm is evaluated on the Stanford drone dataset and our own experimental drone dataset.**

I. INTRODUCTION

Computer vision for robotics is a research area that has grown rapidly in the last few decades, tackling problems towards scene understanding from very diverse fronts. A key current challenge to overcome is how to leverage advanced vision processing algorithms based on computationally intensive deep neural networks in mobile robots with low power embedded GPUs. There are a number of solutions to this problem, e.g. more compact neural network design, and the general advancement of low power GPU hardware. But there is also potential to exploit insight from biological design, specifically in foveated image processing.

Foveated image processing is the focus of this paper for faster, more light-weight use of deep neural networks in mobile robots. It has long been understood that there are very valuable lessons to learn from biology, while the visual system is possibly the most studied brain mechanism. Two main insights from human vision often brought up with this goal are saliency and foveation. Often these principles become independent research areas with the rightful aim of testing competing algorithms, but a congruent integration of them is not always straight forward. Foveal vision is a well established principle, since machines and animals face a same basic dilemma: prioritizing areas of attention to faster process information with limiting computer power.

In general, mobile robots have the potential to benefit from more efficient detection-recognition algorithms. In particular, the focus chosen in this paper is Unmanned Aerial Vehicles (UAVs), since they have a particular need for power-efficient computing. More computing resources also means more weight to be carried, increased battery consumption and decreased flying time. Visual saliency is also naturally beneficial for aerial images given their inherent wide-view, an idea that has been explored in cases like [1], [2], [3].

Fig. 1. (upper left) Nvidia Jetson Nano board, weighting 136 grams and rated at 10 Watts in default conditions, this GPU was used to obtain performance frame-rates to better illustrate the drastic difference from using lower resolutions CNNs on an implementation ready platform; (upper right) DJI Phantom 4 Pro drone, as the one used to record some of the test images, as described in §II-D; (bottom) frame downsampled to 416x416 pixels from an original 848x480 frame, run through object detection network of the same size, where two persons standing up and one person lying down are correctly labeled, after been deemed salient and foveated on.

In this work we present a visual saliency algorithm, based on biological principles aimed at selecting main regions of interest on aerial images, so that they can be foveated into a down-sampled image. This foveated image will be processed by a Convolutional Neural Network (CNN), which in turn is required to be done at a relatively small scale, due to the constraint of computing power in UAVs. This paper builds on our previous work [4], which only studied the use of foveation to speed up CNN processing. The new visual saliency algorithm presented here enables the selection of regions of interest on which to foveate, which is crucial for developing a real-time system.

A. Visual saliency

Given how fast the human visual system allows us to interact with our environment (e.g. scan it to locate an object, find food, or detect a threat), a robust system is needed to regulate this behavior, by broadly answering the question *"Where to look next?"*. The stimuli that drives us to look at something are often classified as either *bottom-up* or *top-down*, the former makes reference to purely visual stimuli (such as a bright color, an odd shape or a sudden movement),

The authors would like to thank the EU for funding support through grant number 731593 (Dreams4Cars).

[1]All authors are with the Department of Automatic Control and Systems Engineering, University of Sheffield. Sheffield, United Kingdom, S1 3JD. {ujaramilloavila1, jonathan.aitken, s.anderson} @sheffield.ac.uk

the latter is directed by our current task. In this work we use a bottom-up top-down method for visual saliency.

Itti and Koch [5] proposed a popular bottom-up engineering saliency map model, derived from the visual attention theory presented by [6], an updated version of which is used here - VOCUS2 [7]. This model is based on extracting color, intensity and orientation cues from an image and iteratively comparing them against each other. We fuse this bottom-up saliency with a top-down saliency path based on deep neural networks. Recently, deep neural networks have been used to learn visual saliency with foveated vision in an end-to-end scheme [8]. Here, we focus on a modular scheme, which more closely mimics biological structures [9, Fig. 9].

B. Foveal vision

Foveation refers to the mechanism in which the human eye has a small high acuity area of the retina, with a much denser presence of photo-receptor cells [10]. While the rest of the retina, the periphery, still allows to process a larger field of view, with considerable less detail. Full acuity would not be possible for the complete field of view without a much larger brain area dedicated to this purpose alone [11]. This problem is tackled by active vision, an approach that very directly extrapolates to embedded robotics.

A considerable amount of literature is available on foveal image transformation, here we build on previous feveation work, presented on [4].

C. Convolutional neural networks

Deep convolutional neural networks (CNNs) have received in recent years a large amount of attention in computer vision, a good overview is given in [12]. For the task of object detection and recognition there are two well established networks, which we use in this work: Faster-RCNN [13] and YOLOv3 [14], with different approaches and performance trade-offs. Faster-RCNN uses a Region Proposal Network to reduce the number of bounding boxes, and YOLOv3 is posed as a single shot regression-classification system (see Fig. 2).

The key problem with using these networks in mobile robots is the limited computational resources: GPUs are used for implementation and the GPUs for embedded systems have far fewer cores than for workstations. The solution proposed here is to use a visual saliency algorithm to detect where to look, then centre a foveated image transformation at this point to produce a small image that is fast to process in a CNN object detection system.

II. METHODS

A. Visual saliency algorithm

The main task addressed in this paper is the development of a visual saliency algorithm that uses bottom-up visual saliency fused with top-down information via the CNN detection, operating on foveated images - the main algorithm is illustrated in Fig. 3.

By normally down-sampling the current frame I_o, by a magnitude m (we use $m = 4$), into I_D, two advantages are obtained; (a) given the pyramidal saliency model structure

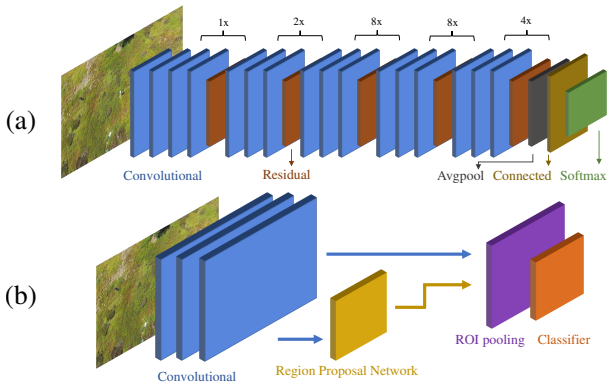

Fig. 2. (a) YoloV3, a fully convolutional model with 106 layers, that makes single shot detections at 3 different scales, (b) Faster-RCNN, a classification model using anchors and a dedicated network for region proposals.

(Fig. 4), a smaller image will be processed considerably faster, and (b) the chance is decreased that an one-off pixel with high saliency will be marked as the most salient one. With I_D as input, the bottom-up saliency map, S_{BU}, is obtained, providing the top salient location, which is used as the center of the fovea, f_c. Then the frame I_o is transformed into a squared foveal image I_f, of equal length and width, to match the input size of the CNN.

The bounding box predictions of the CNN can then be transformed back into the coordinates at the same size to I_D, and used as a top-down saliency influence, S_{TD}, in the next frame, at $t + 1$, allowing to obtain an overall saliency,

$$S_O = \beta \cdot S_{BU} + (1 - \beta) \cdot \gamma \cdot S_{TD} \tag{1}$$

where β is an influencing factor that allows to fine-tune the magnitude in which the top-down information is considered. How the value of gamma affects the overall saliency is illustrated in Fig. 6, using conventional saliency evaluation metrics.

The factor γ enables to give a higher priority to any given class of the CNN detection. For example, with $c = 5$ categories present in the data-set, giving a priority to the third one, $\gamma = [0.5, 0.5, 1, 0.5, 0.5]$,

$$\gamma \cdot S_{TD} = \sum_{n=1}^{c} \gamma_{[n]} \cdot s_n \tag{2}$$

where s_n is the normalized predicted bounding boxes for the n^{th} category.

B. Bottom-up saliency

A well established bottom-up saliency algorithm was chosen to compute the bottom-up saliency, VOCUS2 [7], for several reasons: this model is closely structured to the original [5] proposal, using difference of Gaussian at different scales, as a representation of ganglion cells in the human retina [7]. It also provides a pixel level saliency map, which is necessary to establish the location of the fovea. Fig. 4 shows a diagram of this model. Given its pyramidal structure, a smaller image and limiting the number of layers, provides a considerable speed-up. One scale pyramid was used, with

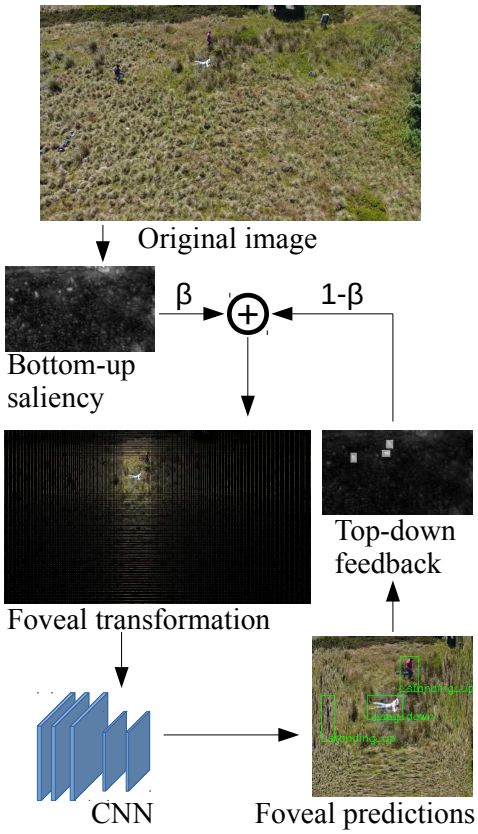

Fig. 3. Diagram of how the bottom-up saliency orientates the top location for the foveal transformation, in its turn feeding the deep neural network. The predictions of the CNN are used as feedback to supplement the saliency computation, with a variable magnitude β, for the frame at $t+1$.

arithmetic mean for feature and conspicuity fussing, two center surround pyramids and four layer levels.

C. Foveated image transformation

With the goal of obtaining foveal images of a smaller size than the original, and having determined the foveal location, f_c, the original image can be transformed following the principles described in previous work [4] where for an image of size $N_x \times N_x$, rows of pixels are selected at sample points x_k that are at a logarithmically increasing distance,

$$x_k = \exp(k\Delta_x) \text{ for } k = 0, \ldots, n_x/2 \quad (3)$$

where $\Delta_x = 2n_x^{-1}\log(N_x/2)$, and then applying the same process to the columns of the image.

The main difference is that here a fast foveal transformation is required, which can be efficiently achieved by creating a Look Up Table (LUT) with the coordinates of all the possible locations of the fovea and simply consulting the required rows and columns to create the foveal image, avoiding the need to do online computations to reach an exact number of pixels to match the shape of the CNN. The LUT contains the pixel coordinates required to transform the input image to a desired size. Here we start with the original frames at 480x848 pixels, a popular resolution with a ratio close to 16:9, to then test with foveal images at increasingly

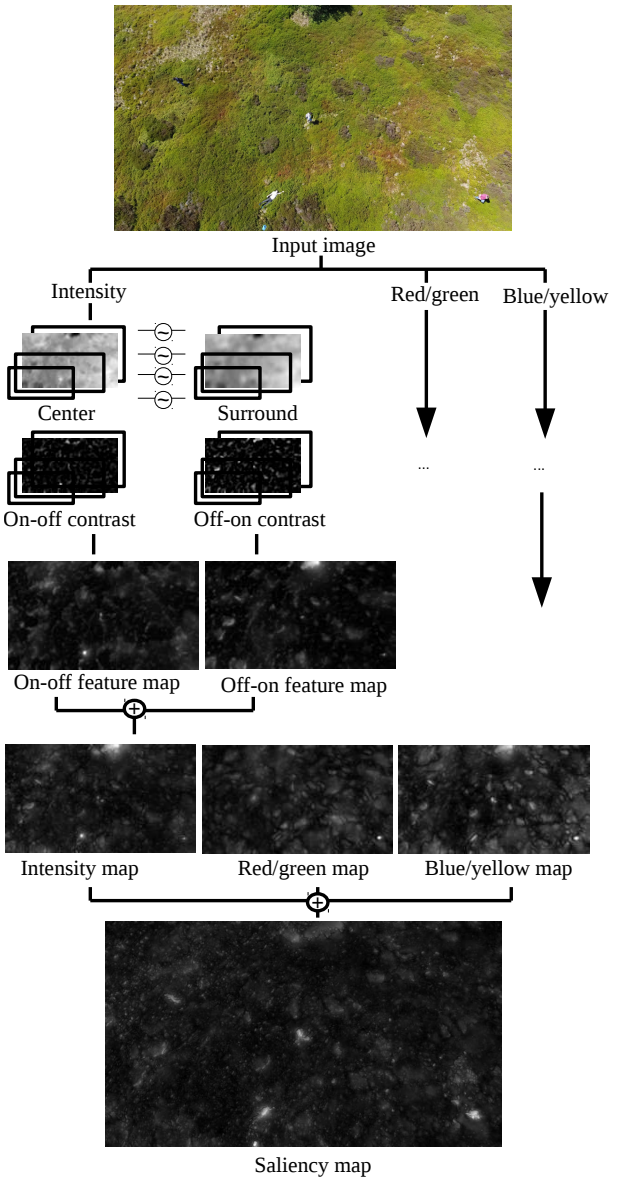

Fig. 4. Partial scheme of the Bottom-up VOCUS2 [7] Saliency model used, with one scale pyramid, feature and conspicuity fussing by arithmetic mean, two center surround pyramids and four layer levels.

smaller images, from 416x416 to 96x96 pixels, at 32 pixel intervals. The Neural Network size is made to match the one of the image.

D. Experimental data

It is considerably more beneficial, for a drone surveying a natural area, to be able to distinguish between actions than simpler categories, e.g. "person", specially for search and rescue operations; a drone could be monitoring a large area, with several dispersed people present, with only one requiring assistance or special attention. Here we hypothesize two relatively simple categories; "person standing up" and "person lying down", with the latter being a clue or representation that someone might be in need of assistance.

Video footage was taken using a *DJI Phantom 4 Pro* drone

Fig. 5. Frame from the Stanford drone dataset, at 848x480 pixels, and downsampled by the described foveation approach to 416x416, 288x288 and 160x160, from left to right, respectively. As the image size decreases, less objects are present in the fovea, but keeping a resolution similar to the original.

in a meadow region of the Lake District, United Kingdom. Using five participants either walking around in random directions or lying down on the floor, with them switching between the two categories previously described. For training, 4804 frames where manually labeled, from 4 different scenes, and extended to 24020 using the *Imgaug* image augmentation library [15], with transformations including Gaussian blur and noise, contrast normalization, rotation and flipping.

For a better generalization, the dataset was merged with a subset of the Stanford drone data-set [16], following a few conditions to balance the number of occurrence of each category, given the overwhelming number of appearances of "pedestrians", which was fused with our "person standing up" label. Similarly, the "golf cart" and "car" labels were considered as one. From 18 separate videos, of 6 different scenes, the frames that contained bus were considered, with a total count of "pedestrian/person standing up"; 84201, "biker"; 57280, "golf cart/car"; 16040, and "bus"; 12006. Fig. 5 shows an example frame at 848x480 pixels resolution, and foveated to 416x416, 288x288 and 160x160 pixels respectively, from left to right.

III. RESULTS

Two main approaches are taken to evaluate the performance of this implementation. First, conventional saliency metrics provide insight into the influence of the top-down feedback loop. Although this metrics are designed and normally compared to human eye fixations, given as ground truth. In this context the main interest is the objects that the CNN is trained to classify.

Second, the mean Average Precision (mAP), a measure commonly used in machine learning, helps to validate how for the objects that are within the vicinity of the fovea, the performance rate can be kept at a similar level than with bigger network/image sizes. While doing so, a considerable increase in frame-rate is obtained, for which we test using a portable GPU, with relatively low energy consumption requirements, the Nvidia Jetson Nano, that at 10 Watts and 136 grams, is well suited for drone implementations.

A. Visual saliency

The Area under ROC Curve (AUC), Pearson's Correlation Coefficient (CC), Kullback-Leibler divergence (KLdiv) and the Normalized Scanpath Saliency (NSS) are four commonly used saliency metrics. For the latter,

$$\mathrm{NSS} = \frac{1}{N}\sum_{i} S_i \times G_i \tag{4}$$

where S_i is the overall saliency, N is the total number of pixels with a fixation and G_i is the binary ground truth map.

To study the influence of top-down versus bottom-up visual saliency, Fig. 6 shows the behaviour of overall saliency when varying β from Eq. 1. Given that both the ground truth and the top down predictions are rectangular boxes, AUC and NSS are most relevant here, as location-based metrics [17].

For the same reason, a performance increase for these saliency metrics can be expected when the top-down information has a larger weight than the bottom-up. However the AUC reaches an almost steady level around $\beta = 0.5$. Of these four metrics, the KL-divergence is the only one for dissimilarity, instead of similarity, meaning that a lower value signifies a better prediction of saliency [17]. In this case, a key point to remark is that the best performance is obtained with approximately a similar influence for top-down and bottom-up information.

A second relevant behavior to consider is the effect of γ, from Eq. 1, which can be used to give priority to any selected label. In Fig. 7, this effect is illustrated using the NSS metric (Eq. 4), by making $\gamma = [a, b, c, d, e]$, where any of $\gamma_{\{a,\ldots,d\}} = 1$ when the corresponding label and $\gamma_n = 0.5$ for all the rest, for example $\gamma = [1, 0.5, 0.5, 0.5, 0.5]$ to give priority to the first label (person lying down). Even when this effect is not drastic, it can help to prioritize information.

B. Object detection and recognition

The most common method for detection-recognition evaluation is to obtain the Intersection over Union (IoU) between the ground truth bounding box and the prediction box,

$$\mathrm{IoU} = \frac{\text{Area of overlap}}{\text{Area of union}}$$

and then use it as a threshold to determine if a predicted box can be considerate positive. The mean Average Precision (mAP) is then calculated using the metrics of the PASCAL VOC 2012 competition [18], with an IoU of at least 50%. We measure the performance of the instances where the objects are at least 30% into the foveal region.

$1-\beta$	AUC	CC	KLdiv	NSS
0 *	0.551	0.161	3.680	1.420
0	0.546	0.084	3.49	0.820
0.1	0.591	0.158	3.620	1.374
0.3	0.730	0.289	3.306	2.301
0.5	0.796	0.392	3.035	2.982
1.0	0.813	0.481	8.507	3.573

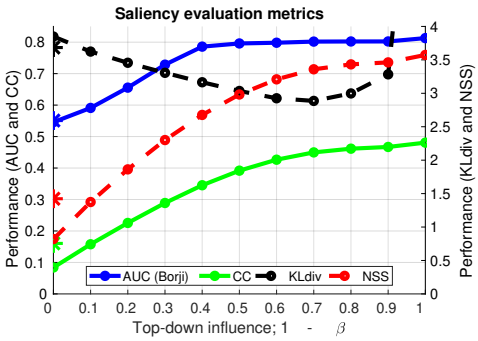

Fig. 6. (left) Table with key values for performance changes using some of the conventional saliency metrics; AUC, CC, KLdiv and NSS, as presented in [17], while varying the weight β of the top-down influence. (right) The graph shows the behaviour for values of $\beta \in \{0, 1\}$. Given the nature of the ground truth (binary bounding boxes with the top-down object locations), it its expected that a larger effect of the top-down information, $1-\beta$, will give a better result. However, for most metrics, the performance flattens around $\beta = 0.3$ to $\beta = 0.6$, supporting that hypothesis that a good balance is obtained giving equal weight to the bottom-up and top-down information.

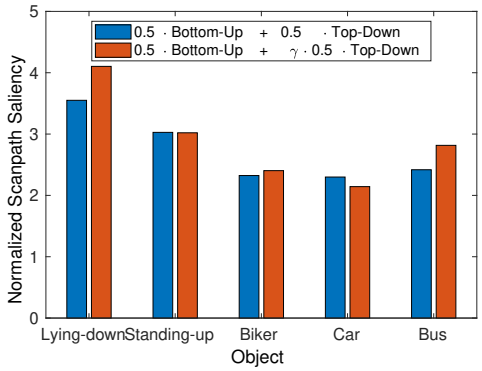

Fig. 7. The γ influencing factor allows to give priority to any of the top-down detection categories, to make it more likely for the fovea to stay centered on it, in case it is required by the task. In this case, $\gamma_i = 1$ for every of the plotted objects, and $\gamma_i = 0.5$ for the rest of them. The effect of γ is determined by the difficulty and frequency in which each object appears, but in most cases at least a slight increase is obtained, compared to treating all categories equally (blue bars), here using the NSS metric.

Network size	**416**	**256**	**192**	**160**	**128**
Frame-rate	8.89	20.31	26.92	31.48	37.54
Normal	45.12	26.12	8.07	1.3	0.08
Foveal	38.88	49.15	44.18	46.76	46.48
Normal (S)	21.71	12.37	5.17	1.57	0.43
Foveal (S)	17.55	17.52	17.72	28.43	30.79

Fig. 8. Key values for mean Average Precision performance using the tinyYolo V3 neural network. The second row exemplifies the frame-rate averaged by all the test images on the Jetson Nano, going from 8.89, at a resolution of 416x416, to 37.54 frames/second, at 128x128 pixels. In the foveal images, performance can be seen to maintain a steady level, although considering less objects as the scale decreases, only those that are at least 30% present in the rows and columns selected for the foveal transformation. The last two rows, marked by a *(S)*, give the performance when only considering the Stanford dataset images, which proved to be considerably more difficult, but where the effect of the foveation remained.

As the network sizes are made smaller, fewer objects are considered. But those that are can be taken with a higher confidence as true positives. While with a conventional linear downsample the performance is affected near linearly by network size. Additionally, with smaller networks, the slowdown of adding the bottom-up saliency and foveal transformation becomes more noticeable.

Fig. 8 gives some of the key mean Average Precision values while using the tinyYolo network. And Fig. 9 does the equivalent while using the complete YoloV3 model, together with their respective frame-rates. While the values vary considerably depending on the network, or when only using the more difficult subsection of the dataset, the behaviour is consistent, where the performance is considerably more steady for the detections that appear in the fovea, as easily seen in the right side graph of Fig. 9. A key result from Fig. 8 is that for the foveal images (last row), the performance can even be seen to increase for the objects in the fovea.

When using the Faster-RCNN network (Fig. 10), only trained at full 416x416 resolution, it is clear that it does not generalize as well for smaller resolutions (marked by the steep performance decline in the normal downsample). The foveated images show a behaviour similar to the one described in the previous cases. This implementation also did not show a considerable speed-up, staying around 1.1 frames/second on the Jetson Nano, possible due to having a bottleneck on the Region Proposal Network (Fig. 2).

IV. CONCLUSIONS

This paper has proposed a novel visual saliency algorithm with foveated vision that enables fast object detection and recognition using low power GPUs. It was shown how down-sampling an image, while keeping a small high resolution region, allows to maintain confidence in the CNN predictions comparable to that at higher resolutions, with the trade-off of the performance on the low-resolution areas, the periphery. While keeping the periphery still allows to have information to influence saliency estimation and the location of the fovea, in contrast to simply cropping the images.

The visual saliency system was demonstrated on two datasets: the Stanford drone dataset and our own UAV test

Network size	416	192	160	128
Frame-rate	1.44	4.57	5.11	6.69
Normal	72.56	21.3	20.65	10.31
Foveal	69.16	58.49	40.61	30.89
Normal (S)	35.51	15.78	12.25	5.53
Foveal (S)	30.79	28.79	25.84	21.41

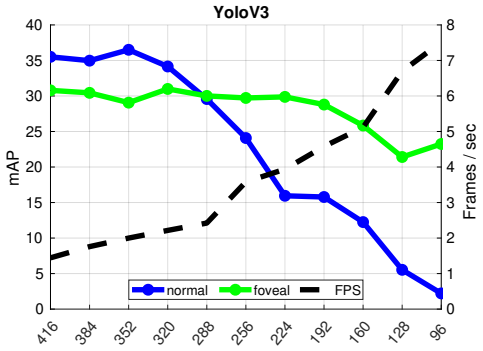

Fig. 9. Mean Average Precision using the YoloV3 neural network. The second row shows averaged frame-rate for all the test images on the Jetson Nano development board (running at high priority). The last two rows give the metrics when only evaluating the Stanford drone dataset (same as the graph on the right), which proved to be considerably more difficult. But both of them show a similar performance trend.

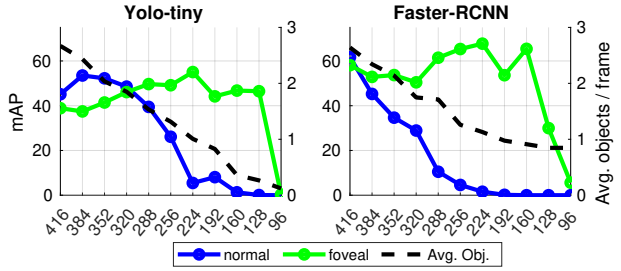

Fig. 10. Evaluating for only the images taken from our drone, the performance is considerably better (as shown in this two graphs), supporting the view that this section of the test images is easier to learn for the CNN, with around 3 object appearances per frame. The right axis shows the average number of objects taken into account for evaluation in each case, selected by being at least 30% in the fovea.

set. The results showed the benefit of the visual saliency algorithm in the applications domain of UAVs, where objects of interest (persons, vehicles, animals, etc.) are often small and naturally different from the rest of the scene (and hence more salient).

REFERENCES

[1] R. Roberts, D.-N. Ta, J. Straub, K. Ok, and F. Dellaert, "Saliency detection and model-based tracking: a two part vision system for small robot navigation in forested environment," in *Unmanned Systems Technology XIV*, vol. 8387, 2012.

[2] J. Sokalski, T. P. Breckon, and I. Cowling, "Automatic salient object detection in uav imagery," *Proc. of the 25th Int. Unmanned Air Vehicle Systems*, pp. 1–12, 2010.

[3] P. Doherty and P. Rudol, "A uav search and rescue scenario with human body detection and geolocalization," in *Australasian Joint Conference on Artificial Intelligence*. Springer, 2007, pp. 1–13.

[4] U. Jaramillo-Avila and S. R. Anderson, "Foveated image processing for faster object detection and recognition in embedded systems using deep convolutional neural networks," in *Lecture Notes in Computer Science*. Springer, 2019.

[5] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 11, pp. 1254–1259, 1998.

[6] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," in *Matters of intelligence*. Springer, 1987, pp. 115–141.

[7] S. Frintrop, T. Werner, and G. Martin Garcia, "Traditional saliency reloaded: A good old model in new shape," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 82–90.

[8] A. F. Almeida, R. Figueiredo, A. Bernardino, and J. Santos-Victor, "Deep networks for human visual attention: A hybrid model using foveal vision," in *Third Iberian Robotics Conference*, 2017.

[9] A. Kimura, R. Yonetani, and T. Hirayama, "Computational models of human visual attention and their implementations: A survey," *IEICE TRANSACTIONS on Information and Systems*, vol. 96, no. 3, pp. 562–578, 2013.

[10] D. Purves, R. Cabeza, S. A. Huettel, K. S. LaBar, M. L. Platt, M. G. Woldorff, and E. M. Brannon, *Neuroscience*. Sunderland: Sinauer Associates, Inc, 2004.

[11] L. Itti, "Automatic foveation for video compression using a neurobiological model of visual attention," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1304–1318, 2004.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[14] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[15] A. Jung, "imgaug," *URL: https://github. com/aleju/imgaug (visited on 26/02/2019)*, 2017.

[16] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *European conference on computer vision*. Springer, 2016.

[17] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand, "What do different evaluation metrics tell us about saliency models?" *arXiv preprint arXiv:1604.03605*, 2016.

[18] M. Everingham and J. Winn, "The pascal visual object classes challenge 2012 (voc2012) development kit," *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep*, 2011.