

MASON: A Model AgnoStic ObjectNess Framework

K J Joseph¹, Rajiv Chunilal Patel², and Vineeth N Balasubramanian¹

¹ Indian Institute of Technology, Hyderabad, India

{cs17m18p100001,vineethnb}@iith.ac.in

² ANURAG, Defense Research and Development Organization, India

rc.patel@anurag.drdo.in

Abstract. This paper proposes a simple, yet very effective method to localize dominant foreground objects in an image, to pixel-level precision. The proposed method ‘MASON’ (Model-Agnostic ObjectNess) uses a deep convolutional network to generate category-independent and model-agnostic heat maps for any image. The network is not explicitly trained for the task, and hence, can be used off-the-shelf in tandem with any other network or task. We show that this framework scales to a wide variety of images, and illustrate the effectiveness of MASON in three varied application contexts.

Keywords: Object Localization, Deep Learning

1 Introduction

Identifying the pixels in an image that contribute towards the most distinct object(s) in it, is an important computer vision task. This forms a core component of many downstream tasks such as object detection, instance segmentation and object tracking to name a few. Traditional methods for foreground segmentation use texture information [1], edge information [2] or graph-cut based methods [3, 4] to infer the location of the foreground objects. Some of these methods enable users to interactively provide suggestions to modify the segmentation results, thus allowing for better results at the cost of manual intervention.

Convolutional Neural Networks (ConvNets) have shown superior performance in image classification [5–7], object detection [8, 9], semantic segmentation [10, 11] and many other computer vision tasks. It is widely understood now that the feature maps in consecutive layers of ConvNets capture a hierarchical nature of the representations that the network learns. This typically starts with detectors for edges with different orientations or blobs of different colors in earlier layers, followed by progressively learning more abstract features. An extensive analysis of these hierarchical representations was presented by Zeiler and Fergus in [12].

In this work, we show that the feature maps (activations) from any of the convolutional layers in a deep network model, that is trained for image classification, contain sufficient information to pinpoint where exactly in a given image lies the dominant foreground object(s). This information can be used to

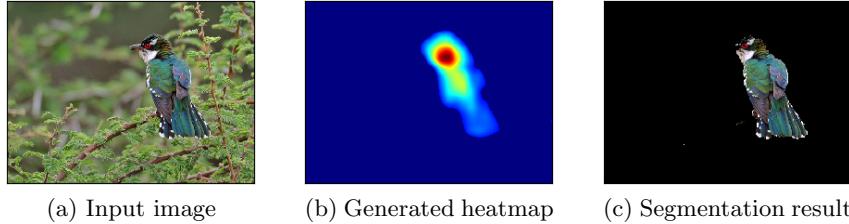


Fig. 1: An illustration of the proposed method. Subfigure (a) is a test image from Pascal VOC 2012 [13] dataset. Subfigure (b) is the objectness heat map that MASON generates for the image. Subfigure (c) shows the segmentation result generated by following the methodology described in Section 3.1 (best viewed in color).

construct a heat map, where the intensity at each location indicates the likelihood of the pixel representing a foreground object. We make use of this heat map in three different application settings. First, the heatmap acts as an effective alternative for the manual annotation required for interactive foreground segmentation algorithms to guide their segmentation. We find and show in this paper that this is a powerful combination that can provide impressive results even when the images contains multiple objects of different sizes. An illustration of the proposed method for foreground segmentation is shown in Figure 1. (We note that this result requires no additional training, given a pre-trained image classification model, and the method can be used as is for any input image.)

Next, we use the proposed framework to implement a generic method to extend existing object detectors to perform instance segmentation. Lastly, we make use of the proposed framework to improve the quality of poorly annotated object detection datasets by automatically filtering out false positive annotations and tightening bounding box annotations inscribing true ground truths. Our experiments validate that this pre-processing step helps the object detection algorithms to learn a more generic object detector on using the cleansed dataset.

The remainder of this paper is organized as follows: We discuss literature related to our work in Section 2. The proposed Objectness Framework (MASON) is introduced in Section 3, followed by detailed explanation of the aforementioned three application settings in Sections 3.1 - 3.3. We showcase two additional capability of the method in Section 4. We conclude the paper with pointers to future work in Section 5.

2 Related Work

We review the related literature from both recent work using ConvNets, as well as more traditional vision-based approaches that achieve a similar objective as our work.

ConvNet-based Object Localization: Different methods with various levels of supervision have been explored in literature to localize objects in images. Convolutional Neural Networks have been used to predict the salient regions in images. The parts of an image that stand out from the background is often annotated in the ground truth in such a task. This ground truth is used to train a ConvNet, which learns generic features to predict the saliency of unseen images. [14–17] work on this principle. These efforts show that they are able to generate better results when compared to traditional vision-based saliency prediction methods.

A large number of successful semantic segmentation methods have been based on Fully Convolutional Networks (FCNs) proposed by Long *et al.* in [11]. SegNet [18] improved the performance of FCNs by using an encoder-decoder architecture followed by a pixel-wise softmax layer to predict the label at each pixel. U-Net [19] uses a fully convolutional architecture with a contraction and expansion phase, where the features from contraction phase is used in the expansion phase to recover the information lost while downsampling. DeepLab[10] introduces convolution with upsampled filters, atrous spatial pyramid pooling and a CRF based post processing to improve the segmentation benchmarks. The global context information available in the feature maps of different depths is utilized in PSPNet [20]. These methods need a large amount of data to train, which however can be offset to some extent by initializing the network using pretrained weights. A deeper issue in such methods is the need for large datasets which are densely annotated at the pixel-level, which is non-trivial and requires immense manual effort. In contrast, in this work, we propose to provide dense pixel-level segmentation of foreground objects with no explicit training of the network for the task.

Segmenting an object from an image using its ‘objectness’ has been explored over the years. Alexe *et al.* [21] explored ways to quantify objectness and used it as the key cue to infer saliency in images in the Pascal VOC [13] dataset. However, such methods have largely been based on handcrafted features, which are fast losing relevance in this era of learned feature representations. The method proposed in [22] generated an objectness map for foreground objects in images using a ConvNet that is trained on image-level and boundary-level annotations. Their method needs to be explicitly trained with pixel-level segmentation ground truth to achieve good performance, while we propose a framework that needs no explicit training for the segmentation task.

Recently techniques like Class Activation Maps (CAM)[23] and Grad-CAM [24] has been proposed for producing visual explanation for the decisions from a large class of CNN models. They also use the activations from pretrained networks to generate heat-maps to localize the object in an image. While these methods localize objects of a specific class, our proposed method is class agnostic. We compare ourself with Grad-CAM in section 4.2. The results reveal that despite being much simpler than Grad-CAM, the method is able to achieve competitive results.

Traditional Vision-based Object Localization: Historically, vision-based methods for foreground segmentation have been studied for many years. Color, contrast

and texture cues have been used often to segment dominant foreground objects. [1] and [2] work on this principle. However, the more successful group of methods for foreground segmentation have been based on graph-cut methods. [25] offers a holistic review on all graph-cut based algorithms for foreground segmentation. We leverage such methods to improve our object localization in this work.

Object proposal methods, that have been proposed in recent years as a pre-processing step for object detection, closely align with the motive of finding discriminative regions in an image. We compare our framework against methods of this kind, in particular: Edge Boxes [26], Selective Search [27] and Randomized Prim’s method [28] in Section 4.1.

3 MASON: Proposed Framework

We propose a simple but effective ConvNet-based method to find the location of those pixels that correspond to the most distinct object(s) in an image. We define this property of an image as its *Objectness*.

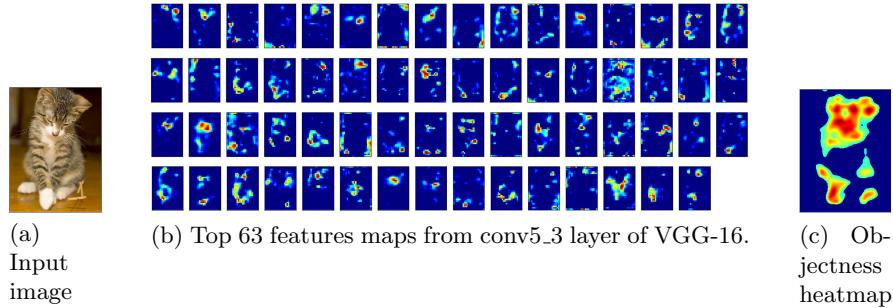


Fig. 2: Subfigure (b) shows feature maps from conv5_3 layer, when the Input image (Subfigure (a)) is passed through a VGG-16 [7], pretrained on ImageNet Dataset. Subfigure (c) is the generated Objectness heatmap. (best viewed in color).

ConvNet architectures like AlexNet [29], VGG-16 [7] and ResNet [6], which are trained for the image classification task on large datasets like ImageNet [30] and PascalVOC [13], learn very good features in their kernels. The activations obtained from the different layers of these image classification networks, yield very good information for localizing the distinct object(s) in an image. Figure 2(b) shows the top 63 feature maps (ranked by the intensity of pixels) from a VGG-16 network trained on ImageNet, when the cat image in Figure 2(a) is passed through it. It is evident that the activations provide strong cue about the location of the cat in the image, albeit distributed across the feature maps. We exploit this information to localize objects from any given image. We hypothesize

that a linear weighted combination of the activations from such networks (pre-trained on standard object recognition task), helps to generate a very useful heat map of the object in an image. In particular, in this work, we find that a simple sum of the activations provides useful results. We call this, the *objectness heatmap*, O , of the object in the scene, i.e:

$$O(I) = \sum_{i=1}^{|f^l|} f_i^l \quad (1)$$

where I refers to the input image, f^l refers to the feature maps at layer l and f_i^l refers to i^{th} feature map in layer l . The objectness heatmap, $O(I)$, that MASON generates, is of the same dimensions as the input image, i.e. $O : \mathbb{R}^{m \times n} \rightarrow [0, 1]^{mn}$. Each entry in $O(I)$ has a value which suggests the degree to which, the particular pixel contains an object or not. We observe that this sum of activations shows high likelihood at locations in the image with objects, and thus provides a useful objectness heatmap. (One could derive more complex variants by learning weights in a linear combination of the feature maps, but our experiments found the sum to work very well in practice.) An example is shown in Figure 1 (b).

Despite its simplicity, this is an attractive option for finding the objectness of an image because this method can work out-of-the-box from the pretrained image classification models available in public repositories. The only change that needs to be done is to remove the final fully connected layers. Through our experiments, we find that the features thus obtained generalize well to object categories beyond the ones they were originally trained for. Hence, we name our idea **MASON**: Model AgnoStic ObjectNess framework. The ability to quantify the objectness in an image is generic across various ConvNet architectures that have been trained for image classification. In our evaluation with AlexNet [29], CaffeNet [31] and VGG-16 [7], the results consistently show the versatility of MASON. In a given detector, the deeper the layer from which the activation maps are considered, the more is the granularity of the heat map. These visual results are shown in Figure 3.

For all the results presented in this work, we use all 512 feature maps from conv5_3 layer of VGG-16 network trained for image classification on the ImageNet dataset. Conv5_3 layer is the last convolutional layer in the VGG-16 architecture. Owing to the high level features that are captured in these deeper layers [12], these heat maps are much denser than heat maps from previous layers. The four pooling layers in the VGG-16 will successively reduce the size of the input image by half. Hence the resulting heat map will be 16x smaller than the input image. Bicubic interpolation is used to scale up the heat map to the original image size. The values of the heat map is scaled between 0 and 255.

We illustrate the usefulness of the *objectness heatmap* in three application domains: (i) Task 1: Fine-grained object localization; (ii) Task 2: Extending object detectors to perform instance segmentation; and (iii) Task 3: Improving the quality of detection datasets. Algorithm 1 provides an overview on how the objectness heatmap is used in each task. Section 3.1 through 3.3 explain each of these in detail.

Algorithm 1: MASON Methodology

```

1 Input: CNN model pre-trained for image classification  $\Phi(\cdot)$ , Input Image  $\mathbf{I}$ ,
   Layer of interest  $l$ 
2 Output: Binary object localization mask  $O(\mathbf{I})$ 
3
4  $f \leftarrow \phi(\mathbf{I})$  {Forward pass the image to generate the activations}
5  $\tilde{O} \leftarrow \sum_{i=1}^{|f|} f_i^l$  {Applying Equation 1 on features from layer  $l$ }
6  $O(\mathbf{I}) \leftarrow \text{Bicubic\_Interpolation}(\tilde{O})$  {Interpolating to the original size}
7
   /* Task 1: Fine-grained object localization */ *
8 Stratify  $O(\mathbf{I})$  to obtain foreground and background
9 Use GrabCut [4] to generate fine-grained object localization
10
   /* Task 2: Extending Object Detectors to Instance Segmentation */
11 For each predicted bounding box  $\mathbf{b} \in B$ , stratify  $O(\mathbf{b})$  to obtain foreground and
      background
12 Use GrabCut [4] to generate instance segmentation in  $\mathbf{b}$ 
13
   /* Task 3: Improving Quality of Detection Datasets */ *
14 For each predicted bounding box  $\mathbf{b} \in B$ , compute  $O(\mathbf{b})$ 
15 Crop bounding box to box inscribing largest contour of  $O(\mathbf{b})$ 

```

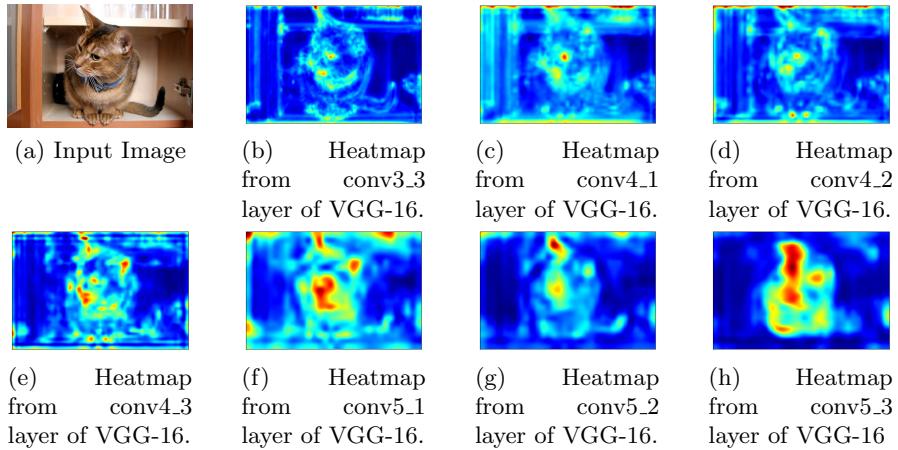


Fig. 3: The figure shows the different heatmaps generated by MASON when feature maps of different layers of VGG-16 are used. Feature maps from deeper layers give a better cue for the location of the object rather than the earlier layers.

3.1 Fine-grained Object Localization using MASON

The objectness heatmap $O(I)$ (Eqn 1) obtained using MASON can be combined with GrabCut [4], a graph cut based foreground extraction method, to generate fine-grained object localization in an image. GrabCut lets the user specify one bounding box for the object under consideration as well as pixel-wise ‘strokes’ which explicitly specifies those regions in the image which is surely a foreground or a background. A Gaussian Mixture Model (GMM) is used to model the foreground and the background, whose initialization is dependent on the user specified cues. A graph is built from this pixel distribution where pixels are considered as vertices. Each foreground pixel is connected to a pseudo-vertex called Source and the background pixels to another pseudo-vertex called Sink. The weight of the edges that connects the source/sink nodes to their respective pixels is defined by the probability of that pixel being foreground or background. The weight of edges between pixels is proportional to inter pixel similarity. Then a min-cut algorithm is used to segment the graph. The process is iterated until the GMM converges.

In this work, we stratify the pixel intensities from the objectness heatmap to automatically generate the foreground and background ‘strokes’ for GrabCut, and show that the results thereby generated are very informative. In particular, we specify the ‘strokes’ as a two-dimensional mask on a given image with each entry in the mask as one of four values; 0 (sure background), 1 (sure foreground), 2 (probable background), and 3 (probable foreground). All regions of the objectness heatmap generated by MASON with pixel intensity greater than the mean intensity are labeled as 1 (sure foreground) in the mask. Those regions whose intensity is between zero and the mean intensity is labeled 3 (probable foreground) and those with zero intensity value are labeled 2 (probable background). The additional information that MASON provides, enables GrabCut to perform foreground extraction well, even when there are multiple objects in the scene (Figure 4), which is beyond the scope of the vanilla GrabCut algorithm.

This methodology helps generate good foreground segmentation for any image without any explicit training. The ConvNet that powers MASON has been trained for image classification on the ImageNet [30] dataset, and doesn’t need finetuning to generate the objectness map. Besides, GrabCut does not need training too. We evaluate the segmentation result on Pascal VOC 2012 [13] test set. Each image from the test set is passed to MASON and the segmentation that is generated is compared with the ground truth. Mean IoU is used as the evaluation metric. Our proposed foreground segmentation method is able to achieve 0.623532 mean IoU with the ground truth. Figure 4 shows example qualitative results of MASON’s performance.

3.2 From Object Detection to Instance Segmentation using MASON

ConvNet based Object Detectors like Faster-RCNN [8], R-FCN [9], YOLO [32] and SSD [33] have shown competent performance in detecting objects from real-world images in MS COCO [34] and Pascal VOC [13] challenges. We show how

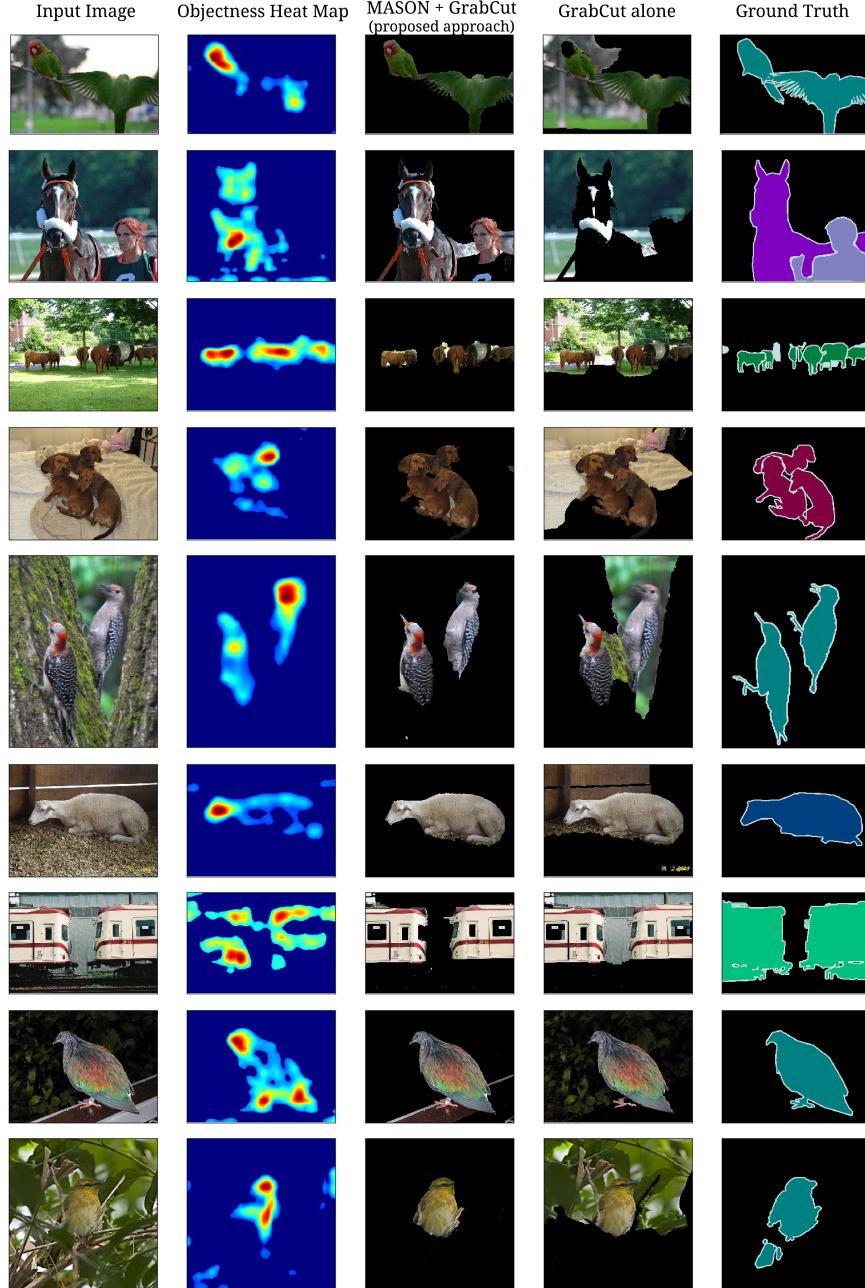


Fig. 4: The effectiveness of MASON in fine-grained object localization is captured in the figure (Column 3). The proposed method (Column 3) is more effective than just using Grabcut [4] (Column 4) in foreground segmentation.

MASON can be used to extend the capability of an existing object detector model to perform instance segmentation of the detected objects, without additional training.

Any object detector takes an image as input and produces a set of bounding boxes with the associated class label as its output. Each of those areas that are inscribed by the corresponding bounding boxes can be passed through the MASON + GrabCut method as explained in Section 3.1, to obtain the foreground segmentation for that small area. This is combined with the class label information that the object detector predicts to generate instance segmentation for the objects that are detected by the object detector. For those detections that overlap each other, the segmentation with lesser area is overlaid on top of the others.

While the methodology that is proposed above can work for any object detectors, region based object detectors like Faster-RCNN [8], R-FCN [9] etc. can be modified without any retraining to integrate MASON into its architecture. This will enable such object detectors to produce instance segmentations in a nearly cost-free manner. The family of region based object detectors has a classification head and a regression head, which shares most of the computation with a backbone CNN. Image classification networks architectures like VGG [7] or ResNet [6] is usually used as the backbone CNN. Its weights are initialized with that of ImageNet [30] training. During the joint training of the object detector, the weights are altered. Still, we find that the backbone CNN retains the property to generate Objectness heatmap with significant detail. This heat map can be used to generate foreground segmentation following the further steps explained in Section 3.1, which in-turn can be combined to generate instance segmentation for the image.

In order to showcase the fact that MASON generates instance segmentation results for datasets that contains just bounding box annotations and not any instance segmentation ground truths, we choose Stanford Drone Dataset (SDD) [35] and UAV123 Dataset [36] to carry out our experimentation. SDD contains videos shot from a drone, where each frame is annotated with objects of six categories. The objects in SDD are very small as the dataset contains aerial view of objects from high altitude. The UAV123 dataset contains objects of standard sizes. A bounding box is drawn inscribing each of the object in the frames. We train the R-FCN [9] object detector on the datasets. Each of the detections of R-FCN is passed through the methodology described above to generate instance segmentation. Qualitative results for the instance segmentation from UAV123 and SDD datasets are shown in Figure 5. No quantitative results could be presented since these datasets do not have segmentation ground truth.

3.3 Improving Quality of Detection Datasets using MASON

The availability of large amounts of high quality annotation is a prerequisite for any deep learning model to work. The success of deep learning models in image classification, object detection, segmentation and many other computer vision task has achieved impressive performance just because of the availability

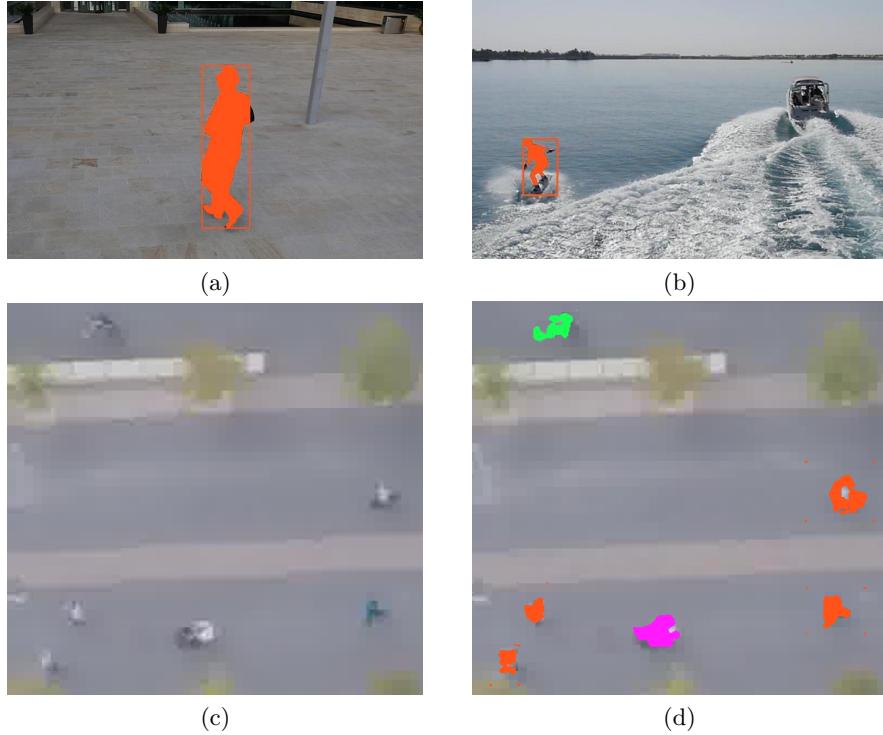


Fig. 5: Instance Segmentation results from UAV123 Dataset [36] and Stanford Drone Dataset[35]. Subfigure (a) is from ‘person18’ video and Subfigure (b) is from the ‘wakeboard1’ video of UAV123 Dataset. The object detection bounding box is also shown for reference. The bottom row contains the input frame from the Bookstore Scene of SDD (left) and its corresponding segmentation result (right). Red color annotates pedestrians, while violet and green colors annotate cart and biker respectively.

of quality annotations from datasets like ImageNet [30], Pascal VOC [13], MS COCO [34], Cityscapes [37], etc. Creating such datasets with quality annotations is a costly and time consuming activity. It limits the applicability of proven deep learning techniques to a new domain. An example would be to adapt a semantic segmentation model for understanding urban scenes, trained on CityScapes [37] to some place outside Europe, or detecting objects from aerial footages using standard object detection techniques like Faster-RCNN [8] or R-FCN [9].

One common way to alleviate the efforts to create annotated datasets, especially in videos, is to manually annotate certain key frames and then interpolate the annotation to few successive frames. This can lead to spurious annotations especially when the target under observation becomes occluded or exits the frame. Given the volume of the data that needs to be annotated, such false positives often creeps into the final dataset due to manual error. MASON can provide an

efficient and automated solution to filter out annotations that actually contains no object inside its annotated area, or refine the annotations.

Most of the annotations for object detection involve the coordinates of rectangular bounding boxes that enclose the object. Such rectangular annotations inherently gives scope for the background of the object under consideration to be part of the annotated area. This concern gets amplified when we consider labeling small objects, which occupies significantly small amount of pixels in an image. Practical application that demands such small objects to be annotated would be to study migration statistics of birds by detection them from videos footages by ornithologists, detecting ground objects from drone footages for surveillance activities, etc. Creating precise bounding boxes that wraps around just the object under consideration is an extremely difficult task and at most times we might have to settle down for lower quality annotations. MASON can be used to enhance such annotations by making them wrap around the object under consideration much closely.

MASON can be used effectively for solving both the issues enumerated above by using the following methodology. For each of the low quality annotation in such datasets, MASON can generate the Objectness heatmap as explained earlier in Section 3. If the pixel intensities in those heat maps are below a specific threshold, it essentially means that there is no dominant foreground object in the annotated region under consideration. Hence, those annotations can be removed from the dataset. This helps to reduce false positive annotations. For those heatmaps with significant intensity values, we can obtain tighter bounding boxes by considering the box inscribing the largest contour of the heatmap.

We study the usefulness of MASON on this task using the Stanford Drone Dataset (SDD) [35], which contains annotated UAV footages from eight different parts of Stanford University Campus. Each of the objects are very small, relative to the size of the image. Bicyclists, pedestrian, skateboarders, carts, cars and buses are annotated with rectangular bounding boxes. This dataset was introduced in ECCV 2016, with the primary focus on accelerating research in trajectory prediction of objects. Figure 6 (a) contains a sample frame from the dataset.

Human annotators have annotated some of the frames and the successive frames were automatically extrapolated. Hence these annotations are not very accurate as is evident from Figure 6 (a). The authors of the dataset included an explicit meta-data on which all annotations were manually labeled. Out of 361129 annotated objects in Video 0 of ‘Bookstore’ scene, 6036 objects were manually annotated and 355093 objects were interpolated. The MASON based methodology that is explained above can help filter out bad annotations and make the existing bounding box tighter, adding value to any task that the dataset is intended for. Figure 6 (b) shows the result after applying MASON based methodology to sub-figure 6 (a). After cleaning up the annotations from the Bookstore Scene of Stanford Drone Dataset, an R-FCN [9] object detector is tasked to learn a model for object detection. The RFCN is trained with frames from the ‘Bookstore’ scene on an NVIDIA P-100 GPU with two images per



(a) Ground Truth annotation from Stanford Drone Dataset [35]



(b) Results after using MASON to remove false positive annotations and making the bounding boxes tight.

Fig. 6: The picture captures the effectiveness of MASON to enhance bad annotations in a dataset. False positive annotations are removed and remaining annotations are made tighter by MASON. The image is the 4005th frame from Video 0 of ‘Bookstore’ scene in the Stanford Drone Dataset [35]. Red color bounding boxes annotate Pedestrians while blue color annotate Biker.

mini-batch. A learning rate of 0.001 with a weight decay of 0.0005 is used. The momentum is fixed to 0.9. Frames from ‘DeathCircle’ and ‘Hyang’ scenes are used for validating the trained model. Visual characteristics like lighting and background is strikingly different in the scene used for training and testing. Hence, the generalization capability of the model is put to test.

Table 1 shows the mean average precision (mAP) of the R-FCN object detector which was trained using original annotations from the dataset and the annotations that were cleansed by removing the bad annotations. MASON based pre-processing is able to improve the detection accuracy by two folds as is captured by the last two columns of Table 1. To the best of our knowledge, no other work has been published in the research community, providing baseline for

object detection on Stanford Drone Dataset. Hence this result will stand as the current baseline.

| Class | Without Enhancement | | With Enhanced Dataset | |
|------------|---------------------|----------|-----------------------|----------|
| | AP @ 0.5 | AP @ 0.7 | AP @ 0.5 | AP @ 0.7 |
| Pedestrian | 0.302 | 0.093 | 0.763 | 0.542 |
| Biker | 0.303 | 0.124 | 0.711 | 0.521 |
| Skater | 0.144 | 0.092 | 0.589 | 0.436 |
| Car | 0.330 | 0.228 | 0.554 | 0.440 |
| Bus | 0.241 | 0.282 | 0.591 | 0.553 |
| Cart | 0.577 | 0.420 | 0.750 | 0.684 |
| mAP | 0.316 | 0.207 | 0.659 | 0.529 |

Table 1: Table shows the mAP values of object detection on SDD using R-FCN object detector , with and without using enhanced annotations.

4 Discussions

4.1 Object Proposal Generation using MASON

As the objectness heatmap that is generated by MASON is class agnostic, it can be used to generate region proposals. Multi scale boxes around blobs in heatmap is considered as proposals from MASON. In order to validate its capability, we compare against three best-in-class object proposal methods: Edge Boxes [26], Selective Search [27] and Randomized Prim’s [28]. All these methods are evaluated on 4952 images from the Pascal VOC 2007 [13] test set. We used the framework proposed by Chavali *et al.* [38] to compare the performance of each of the object detectors. The average recall @ 0.9 of each of the detector is captured in Figure 7, where MASON outperforms others. MASON goes a step further to predict the pixel location of the object, which the conventional Object Proposal methods does not address.

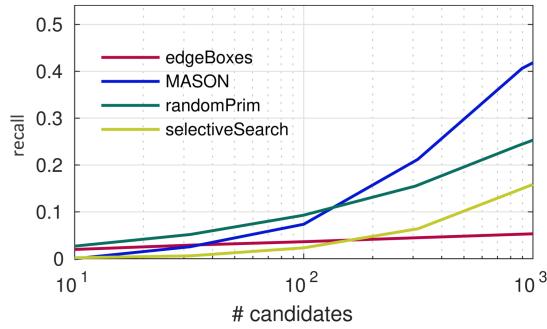


Fig. 7: Comparison of Recall of MASON with the other object proposal methods.

4.2 Comparison with CAM and Grad-CAM

Recently, Zhou *et al.* [23] proposed Class Activation Maps (CAM) using global average pooling of the activation maps to generate discriminative localization of objects in an image. Their method require changes to the architecture of standard image classification networks like VGG-16. Hence the network has to be retrained to achieve good localization results. Grad-CAM [24] is a modification to CAM, that can produce localization heat maps by making use of off-the-shelf image classification networks. Hence it closely matches with MASON.

Both CAM and Grad-CAM weighs the feature map from the last convolutional layer by a score, which is proportional to the importance of that feature map for a class of interest. Hence these methods are tightly coupled to a specific class, while MASON is class agnostic. This helps our method to propose object regions, for those object classes that it has been never trained for. In another way, MASON weighs all the feature maps at a specific convolutional layer equally. Despite this simplification, the method is able to give competitive results, when compared to Grad-CAM. Figure 8 shows an example.

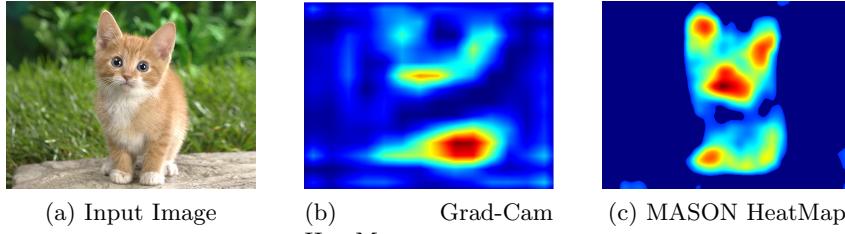


Fig. 8: A comparison of the heat maps generated by Grad-CAM [24] and MASON. Sub figure (a) is the input image and (b) and (c) are the heatmaps generated by Grad-CAM and MASON respectively. Grad-CAM HeatMap is generated for the TabbyCat class from ImageNet Dataset. [30]. (best viewed in color).

5 Conclusion

We have proposed a straightforward way to capitalize on the object localization information, that is available in activation map of ConvNets trained for image classification. The effectiveness of MASON has been demonstrated on three significant use cases. The spatial resolution loss of the image that is incurred during the pooling operation of the ConvNet is currently compensated by simple bilinear interpolation. This can be replaced by the ‘hole’ algorithm proposed in [10]. This method is parameter-free. We open-source the code here ³.

Acknowledgement

This work is funded by ANURAG, Defence Research and Development Organisation (DRDO), Government of India, through the CARS project. (Project Number: ANURAG/CSE/F121/2016-17/S16)

³ <https://github.com/JosephKJ/MASON>

References

1. Karoui, I., Fablet, R., Boucher, J.M., Augustin, J.M.: Variational region-based segmentation using multiple texture statistics. *IEEE Transactions on Image Processing* **19**(12) (2010) 3146–3156
2. Mortensen, E.N., Barrett, W.A.: Intelligent scissors for image composition. In: 22nd annual conference on Computer graphics and interactive techniques, ACM (1995) 191–198
3. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on. Volume 1., IEEE (2001)
4. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. In: ACM transactions on graphics (TOG). Volume 23., ACM (2004) 309–314
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (2015)
7. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
8. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems (NIPS). (2015)
9. Dai, J., Li, Y., He, K., Sun, J.: R-fcn: Object detection via region-based fully convolutional networks. In: Advances in neural information processing systems. (2016)
10. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv:1606.00915 (2016)
11. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 3431–3440
12. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: European conference on computer vision, Springer (2014) 818–833
13. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results
14. Pan, J., Sayrol, E., Giro-i Nieto, X., McGuinness, K., O'Connor, N.E.: Shallow and deep convolutional networks for saliency prediction. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (June 2016)
15. Liu, N., Han, J., Zhang, D., Wen, S., Liu, T.: Predicting eye fixations using convolutional neural networks. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (June 2015) 362–370
16. Li, X., Zhao, L., Wei, L., Yang, M.H., Wu, F., Zhuang, Y., Ling, H., Wang, J.: Deepsaliency: Multi-task deep neural network model for salient object detection. *IEEE Transactions on Image Processing* **25**(8) (Aug 2016) 3919–3930
17. Kruthiventi, S.S.S., Ayush, K., Babu, R.V.: Deepfix: A fully convolutional neural network for predicting human eye fixations. *IEEE Transactions on Image Processing* **26**(9) (Sept 2017) 4446–4456

18. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR* **abs/1511.00561** (2015)
19. Ronneberger, O., Fischer, P., Brox, T. In: U-Net: Convolutional Networks for Biomedical Image Segmentation. Springer International Publishing, Cham (2015) 234–241
20. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)
21. Alexe, B., Deselaers, T., Ferrari, V.: Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**(11) (Nov 2012)
22. Jain, S., Xiong, B., Grauman, K.: Pixel objectness. arXiv preprint arXiv:1701.05349 (2017)
23. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 2921–2929
24. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization
25. Yi, F., Moon, I.: Image segmentation: A survey of graph-cut methods. In: Systems and Informatics (ICSAI), 2012 International Conference on, IEEE (2012) 1936–1941
26. Zitnick, C.L., Dollar, P.: Edge boxes: Locating object proposals from edges. In: ECCV. (2014)
27. Uijlings, J., van de Sande, K., Gevers, T., Smeulders, A.: Selective search for object recognition. *International Journal of Computer Vision* (2013)
28. Manen, S., Guillaumin, M., Gool, L.V.: Prime object proposals with randomized prim’s algorithm. In: ICCV. (2013)
29. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q., eds.: Advances in Neural Information Processing Systems 25. Curran Associates, Inc. (2012) 1097–1105
30. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09. (2009)
31. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the 22Nd ACM International Conference on Multimedia. MM ’14, New York, NY, USA, ACM (2014) 675–678
32. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 779–788
33. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision, Springer (2016) 21–37
34. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision, Springer (2014)
35. Robicquet, A., Sadeghian, A., Alahi, A., Savarese, S.: Learning social etiquette: Human trajectory understanding in crowded scenes. In: European conference on computer vision, Springer (2016) 549–565

36. Mueller, M., Smith, N., Ghanem, B.: A benchmark and simulator for uav tracking. In: European Conference on Computer Vision, Springer (2016) 445–461
37. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 3213–3223
38. Chavali, N., Agrawal, H., Mahendru, A., Batra, D.: Object-proposal evaluation protocol is' gameable'. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 835–844