Fast Unsupervised Object Localization with DeepLift

Sakif Khan, Haotian Lyu, Josh Valchar, Congcong Zhi

December 10, 2017

1 Introduction

Object localization refers to the task of automatically locating objects belonging to one of several pre-determined classes within a given image. While neural networks have achieved human-level performance on the related but simpler task of image classification, the same is not yet true for object localization. The paradigmatic deep learning approach suffers not only from the computational burden of training but also requires access to a set of images annotated with bounding boxes. Such annotation is a time-consuming task requiring scarce domain-specific human expertise.

To ameliorate these problems, the paper [DMM16] proposes a technique which circumvents the need for annotated data altogether and trains or tunes a single convolutional network on raw image data to produce localizations. Apart from the technical simplicity of the method, it has the virtue of being consistent with the deep learning philosophy of having an end-to-end model which requires minimal external input beyond an appropriate dataset. In this work, we explore a modified version of the technique presented in [DMM16].

2 Fast Unsupervised Object Localization Algorithm

We provide here a high-level description of the main algorithm in [DMM16]. For simplicity, we describe the algorithm in the case where we wish to compute a bounding box for a single object only. The algorithm is premised on the simple idea that pixels which *contribute most* to the ultimate object categorization result carry the most information about the object, i.e,. the object can be localized within these pixels. Three core technical ingredients underlie the algorithm: a convolutional neural network identifying different objects, a scoring criterion for the neurons and a backpropagation method to map neurons back onto images to produce heat maps for each object.

Suppose we have a picture which contains, for instance, an image of a dog and we would like to construct a bounding box around the parts of the picture containing the dog. At the very least, we should have a model which is able to classify the image as that of a dog (with high probability). We can employ a (pretrained) convolutional neural network towards this end. Heuristically, learned weights encode spatial data contained in natural images and the weights of the top layers represent high-level features in such images. The algorithm we describe exploits such information to construct a bounding box as follows. For a given input image, we look at the neurons in the l^{th} layer, where l is a suitably chosen integer fixed ahead of time, with non-zero activation. Among these neurons are the ones whose "firings" are attributable to the presence of the dog in the original image. Assume that we have access to a scoring criterion which is able to quantify the extent to which the firing of a neuron affects the final output of the network. We can use such a criterion to pick out the top k_max neurons, subsequent to which we can use a modified backpropagation technique to determine, for

each neuron, which pixels led to the firing in the first place. Hence, we automatically infer the parts of the image which (should) contain the dog and which neurons in the l^{th} layer are activated by it. The authors of [DMM16] use $l = 11$ in their implementation; additionally, to improve performance, they employ a technique which they call the **DAM heuristic** and they construct a sub-image for each neuron in the l^{th} layer.

3 Fast Unsupervised Object Localization Algorithm using DeepLIFT

We follow [DMM16] in using VGG-16 pre-trained on ImageNet for the neural network. However, we introduce some novelty into the algorithm by using what we believe is a richer scoring criterion for the neurons. Whereas [DMM16] uses a simple scoring criterion which computes the product of a neuron's activation and its effect on the output and uses guided backpropagation (see [ZF14]) to map neurons to images, we borrow DeepLIFT from [SGK17], which can assign scores to neurons based on a set of robust algebraic rules and which is believed to outperform a variety of backpropagation techniques (including guided backpropagation) in scoring (see [SGK17]). In short, DeepLIFT is an analytical method based on secants rather than the derivatives (tangents) used in (guided) backpropagation. We provide our algorithm in the listing below.

Algorithm 1 Fast Unsupervised Object Localization Algorithm using DeepLIFT

1: Pass the image through VGGNET-16 to identify the category of the image.
2: Use DeepLIFT and Guided Backpropagation to map the result back into the image and to produce a pair of "heat maps".
3: Smooth the two heat maps produced using a Gaussian kernel.
4: Apply thresholding and masking techniques to eliminate spurious heat pixels.
5: Mix the smoothed heat maps into a single final heat map.
6: Construct a bounding box encompassing the highlighted regions in the final heat map.

We now offer a few observations pertaining to this algorithm. First, since DeepLIFT is a secant-based method, it tends to produce noisy heat maps, i.e., relatively high heat may be assigned to pixels which do not participate in the object categorization. Guided backpropagation is employed to eliminate such spurious pixels. Second, we see no particular reason to adhere to the DAM heuristic presented in [DMM16] and we simply score neurons in the output layer. Third, we pass the whole image through the network instead of breaking it into sub-images as done in [DMM16]. Lastly, the mask in Step 4 is designed to assign less heat to the borders of the image. The reason for this will become clear in the sequel.

4 Experimental results

The experimental set-up is mostly as already described above. We computed bounding boxes for images in the ImageNet test set and we curated a subset of the output images along with their respective heat maps, which are displayed below. The top panel represents a set of images on which the algorithm performs well as revealed by visual inspection. The bounding boxes for these images are tight around the objects which should be localized. We feel that these samples are representative of the overall performance of our algorithm. Nevertheless, we have included examples in the middle panels where the bounding boxes are not as ideal.

However, if the mask in Step 4 of Algorithm 1 is absent, the method has a strong tendency to overshoot by drawing a bounding box which covers the entire image. The bottom panel shows examples of where this occurs. Careful examination of the heat maps revealed that, in each case, there is a tiny cluster of pixels near the edges of the image which have high heat. These edge pixels then cause an expansion in the bounding box constructed since we effectively take unions of pixels with high heat. As a result, when the edge pixels are excluded by the mask, we expect good localization since the rest of the heat map is faithful to the objects to be localized.

5 Conclusion

We remark that the hyperparameters in Step 4 of Algorithm 1, including thresholds, the mask and the standard deviation of Gaussian kernels, all have apparent effects on how much the method will under- or overshoot in constructing bounding boxes. While this may seem like a disadvantage at first, we can turn it into an advantage for our method by realizing that these hyperparameters can be tuned. Such tuning provides a boost to localization for datasets where we have some prior information about general properties of the images. For example, if we know that most images in our dataset have objects in the foreground as opposed to the background, a larger amount of blur will help the algorithm focus less on the background. Moreover, tuning for the amount of blur and thresholds is generally much less expensive than tuning for the neural network architecture.

In closing, we would like to emphasize the purely computational nature of our method for object localization. As our results hopefully demonstrate, we can obtain decent object localization in natural images with nothing more than a pre-trained convolutional network. Thus, the only computational power we are required to expend goes towards calculating the "heat" induced in the input images and then performing a simple union of sets of pixels with sufficient heat. This represents significant computational savings while demonstrating the efficacy of transfer learning approaches in object localization. Indeed, it would be an interesting project to examine whether an approach similar to the one presented here could be competitive with real-time object localization methods such as those in [RHGS15]. Moreover, an extended analysis on object localization performance for a variety of pre-trained networks such as ResNet and Xception combined with DeepLIFT would not only be valuable for obtaining an optimal model for localization but perhaps lend insight into how and why pretrained neural networks provide solid performance on localization tasks.

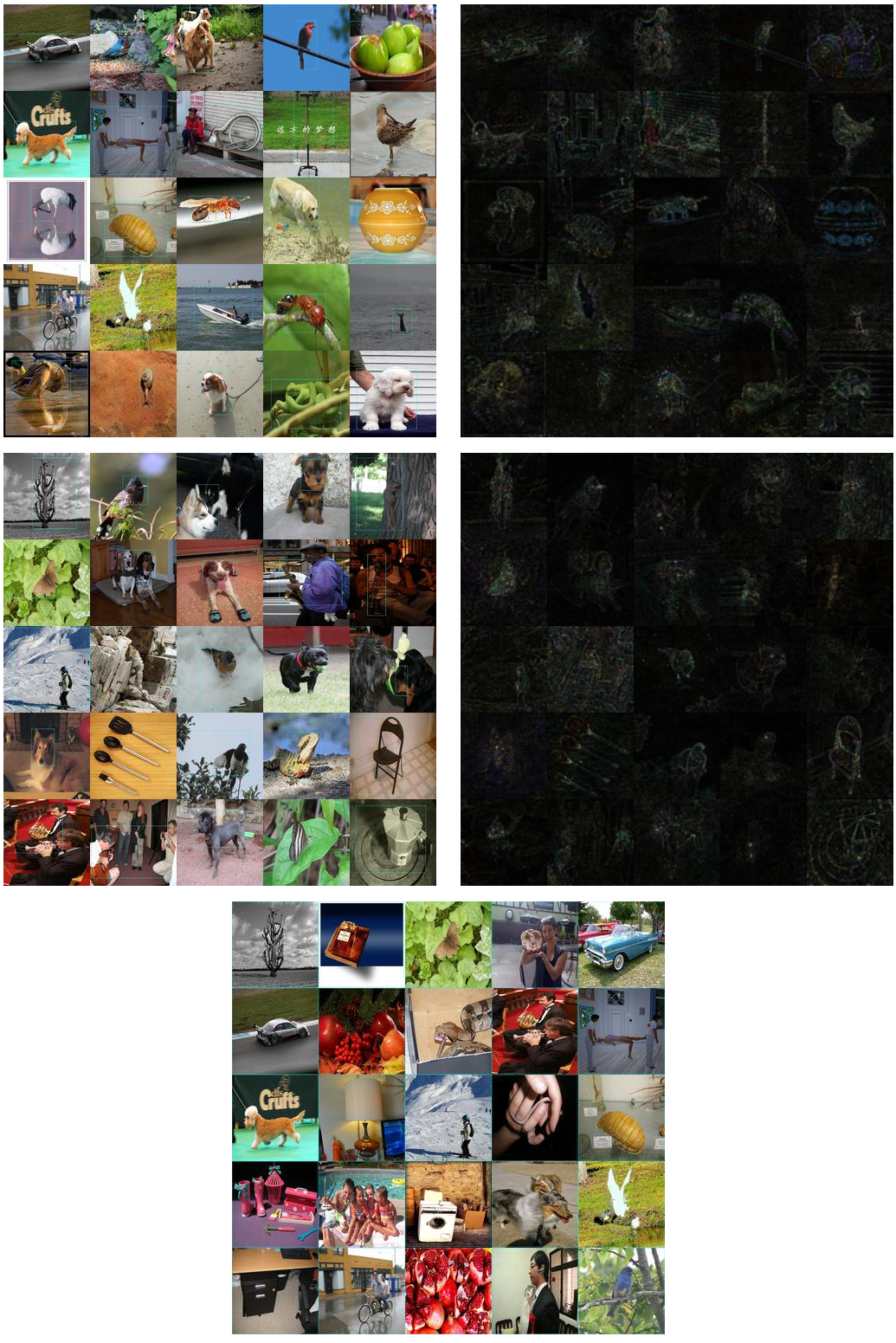

Figure 1: **Top left**: Images with tight object localization. **Top right**: Heat maps for images in panel on top left. **Middle left**: Images where the bounding boxes are not tight. **Middle right**: Heat maps for images in panel on bottom left. **Bottom**: Images where the bounding boxes overshoot when the mask is excluded.

References

[DMM16] Anjan Dwaraknath, Deepak Menghani, and Mihir Mongia. Fast unsupervised object localization. http://cs231n.stanford.edu/reports/2016/pdfs/285_Report.pdf, 2016.

[RHGS15] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 91–99, 2015.

[SGK17] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 3145–3153, 2017.

[ZF14] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I*, pages 818–833, 2014.