SINGLE SHOT LOGO: BRAND RECOGNITION AND DETECTION IN REALTIME

Saurabh Kumar, Huy Nguyen

Georgia Tech, Yahoo

ABSTRACT

Brand identification in photos and videos has a variety of important industrial applications. The detection of logos in images is perhaps the most widely recognized form of brand identification; proper localization and recognition of logos in images can for example be used in brand analytics and in trademark protection. In this paper, we propose a new approach for logo detection using the SSD object detection framework. We demonstrate through experimentation that our approach surpasses previously published state-of-the-art results on the FlickrLogos-32 dataset and runs significantly faster than prior approaches. Another key contribution of this work is the development of a methodology that optimizes the SSD network's priorbox and ground truth bounding box matching scheme for logo detection.

1. INTRODUCTION

Logo detection can be viewed as a subset of generalized object detection where the task is to recognize instances of one or more logos in a photo and to produce the associated bounding box locations for where these logos appear. This task has been shown to be useful across a wide range of industrial applications such as brand monitoring on social media platforms [1], contextual advertising [5], and for blurring of logos in videos for brand protection and displacement [14].

The main contributions in this paper include a new architecture for logo detection based on the Single Shot Multibox Detector (SSD) object detection framework [9]. Our final architecture is able to evaluate images at 23 frames per second on a K40 GPU while surpassing the previous state-of-the-art results on the FlickrLogos-32 dataset [5]. This represents a 46X improvement in computation time over the Fast R-CNN approach that Iandola *et al.* [5] uses in their *DeepLogo* paper. Secondly, we perform an analysis of errors in detections arising from the SSD framework and use the insights generated from our analysis to tune SSD hyper parameters for higher precision and recall.

2. RELATED WORK

In the past, logo detection and recognition research has focused on utilizing hand-crafted image descriptors such as SIFT to recognize and localize logos [15] [7]. As compared to more recent convolutional neural net based approaches, these early methods suffered from an inability to generalize well across domains and deformations. Another prominent weakness in these approaches often manifested in low recall rates below 50% [11] for logos like the Apple or Adidas logo whose geometrically simplistic shapes offered few keypoints for these techniques to extract local image descriptors from.

Following the success of convolutional neural networks in image classification tasks [8], Girshick *et al.* propose the R-CNN [4] architecture as a general object detection framework using convolutional features. At its core, the algorithm samples image windows from pre-determined regions proposals. It then extracts pre-trained convnet features (i.e. VGG16 [13]) from each of these sampled windows. Using these extracted features, it trains a set of linear SVM classifiers – one for each object category – to determine whether a given region proposal contains or does not contain the target object. While the approach is agnostic to how regions are proposed, selective search is a popular option for choosing regions of interest to feed into the SVM classifiers.

The main drawback with this approach is that it is slow. The computational cost is linear with the number of region proposals because each proposal requires a full feed-forward pass through the convnet architecture to extract features. It's been reported that for a single image, test times can take up to 47 seconds on a Nvidia K40 GPU [3].

To reduce the computational costs, Fast R-CNN[3] proposes using a spatial pooling layer to extract fixed-length feature vectors from regions of interest in the final convolutional feature map of VGG16 and using these feature vectors as the input to a softmax classifier and bounding-box adjustment regressor. This approach facilitates sharing of the feature computation between each region proposal and thereby more efficiently achieves localization.

In their *DeepLogo* paper Iandola *et al.* [5] builds upon the Fast R-CNN work and applies this technique to the task of logo detection. They evaluate their approach on the FlickrLogos-32 dataset [11] where they consider all logo variations belonging to the same brand as belonging to the same class. Using this technique they are able to achieve mAP of 74.4% starting from a pre-trained VGG network and following the PASCAL VOC object detection evaluation protocol [2]. Similarly Hoi *et al.* employ the same techniques on a custom dataset of on 160 logos mined from Flickr to achieve mAP of 65.8%.

Despite the improvements over its predecessor, Fast R-CNN still suffers from large computational bottlenecks. Because each region-of-interest requires a resampling of the underlying feature map, it does not benefit from the cache-locality of the highly tuned and efficient convolution operation present in modern hardware. And because Fast R-CNN still depends on an outside algorithm for region proposals, any detection scheme must take into account the computational performance of the region proposal method. Selective search, one of the most robust methods used alongside Fast R-CNN has been measured to take 2 seconds per image on a CPU [10] which means it now becomes the computational bottleneck for this detection framework.

It is from these deficiencies in Fast R-CNN that we begin to explore using SSD proposed by Liu *et al* [9] as the foundation for our logo detection method.

3. SSD FOR LOGO DETECTION

In this section, we briefly review SSD, a convnet-based joint localization and recognition framework proposed by Liu *et al* [9]. Traditional object detection methods such as R-CNN[4], take upwards of several seconds even when computed on the GPU. The SSD architecture proposed consolidates object detection into one framework by eliminating region proposals and pooling present in other object detection techniques, such as Fast R-CNN[3]. Because SSD removes this expensive step, it is faster while achieving state-of-the-art performance, making it an optimal candidate for industrial logo detection.

The SSD network attaches a set of layers, called detection heads, to a pre-trained network. Specifically, for a base VGG16 network[13], the $conv4\_3$, $fc7$, $conv6\_2$, $conv7\_2$, $conv8\_2$, and $pool6$ layers of the network are used as detection heads. Each detection head is responsible for making classification and localization predictions at a particular scale. The network generates default priorboxes with predetermined scales and aspect ratios and matches the priorboxes at each detection head to the ground truth bounding boxes in the training images. This matching scheme allows the network to be receptive to target objects of various scales and sizes. The SSD framework then uses convolutional filters for both predicting the locations of bounding boxes in localization prediction layers and producing confidence scores for different target classes in confidence prediction layers.

With the original parameter settings as used by Liu et al. [9], the feature maps of the $fc7$, $conv6\_2$, $conv7\_2$, $conv8\_2$, and $pool6$ layers are responsive to scales that are regularly spaced from 0.2 to 0.95. For these layers, the default priorbox sizes are determined by a min scale value, s_{min}, and a max scale value, s_{max}. The default priorboxes are generated with aspect ratios 1, 2, 3, $\frac{1}{2}$, and $\frac{1}{3}$. For the aspect ratio 1, the size of the priorbox is $\sqrt{s_{min} * s_{max}}$ x $\sqrt{s_{min} * s_{max}}$. For all other aspect ratios a, the priorbox sizes are $s_{min} * \sqrt{a}$ x $(\frac{s_{min}}{\sqrt{a}})$. Thus, each layer other than the $conv4\_3$ layer has 6 default priorboxes. Since the 38x38 feature map of the $conv4\_3$ layer is large, it does not have a max scale value associated with it. It has 3 default priorboxes, one of aspect ratio 1 which has scale 0.1, and two of aspect ratios 2 and $\frac{1}{2}$. The candidate default priorboxes generated at each layer are tiled across the input image so that they are matched to the ground truth boxes. The default priorboxes generated in the manner described above are matched to the ground truth bounding boxes when training the SSD network. First, each ground truth bounding box is matched to the default box with which it has the highest Jaccard overlap. Each default box is then matched to any ground truth bounding box with which it has Jaccard overlap greater than a threshold, which is set at 0.5.

We apply the SSD framework for logo detection on the FlickrLogos-32 dataset. We modify the parameters that determine the distribution of priorbox sizes to improve the matching scheme between the SSD priorboxes and ground truth boxes in the FlickrLogos-32 *trainval* set. We use the VGG16 base network, which is pre-trained on the ILSVRC CLS-LOC dataset [12].

4. FLICKRLOGOS-32 DATASET

We perform all of our experiments using the FlickrLogos-32 dataset [11]. This dataset consists of 32 logo classes plus one additional class of images which contain no logos at all. Any image containing a logo also has associated with it a series of bounding box labels for that logo. We use these bounding box annotations in our experiments for training the SSD architectures for logo detection. In all of our experimental setups, we train SSD on the *trainval* set, which consists of 4280 images, 1280 of which contain logos, and evaluate the network on the *test* set, which consists of 3960 images, 960 of which contain logos. We perform our experiments using the Caffe deep learning framework [6].

5. SSD EXPERIMENTS

5.1. SSD 300x300 Experiments

We design three experiments with VGG16 as the base network for SSD 300x300. For all our experiments, we use the same training parameters as Liu et al. [9] except for the initial learning rate, which we set to 10^{-4} instead of 10^{-3} to prevent exploding gradients. We tune the priorbox scale distribution parameters to better fit the ground truth bounding box scales in the FlickrLogos-32 *trainval* set. As shown in Figure 1, this distribution is skewed towards ground truth bounding boxes of scales from 0 to 0.3.

We attempt to modify the priorbox distribution to improve the matching between the priorboxes and ground truth

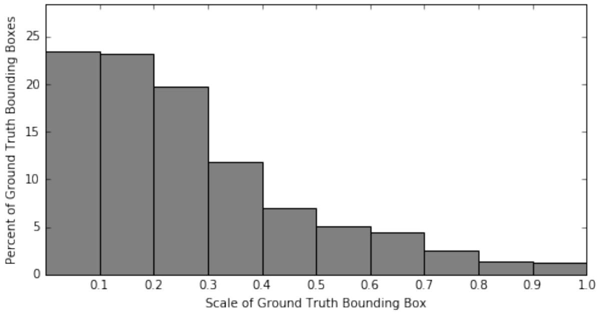

Fig. 1. The distribution of the scales of the ground truth bounding boxes in the FlickrLogos-32 dataset. The scale of a bounding box is calculated as the average of its width divided by the image width and its height divided by the image height.

bounding boxes. In our analysis, we measure three types of bounding box matches: (1) the number of matches where the Jaccard overlap between the ground truth bounding box and the best matching priorbox is at most 0.5, (2) the number of matches where the jaccard overlap between the ground truth bounding box and the best matching priorbox is greater than 0.5, and (3) the number of overall matches between the priorboxes and ground truth bounding boxes. The matches of type 1 indicate that, for a particular ground truth bounding box, there is no priorbox that matches with it well. We modify the priorbox parameters to minimize the number of such matches between the priorboxes and ground truth bounding boxes while retaining a large number of overall matches.

In Experiment 1, we use the priorbox parameter configuration described above. As shown in Figure 2, over 90% of the type 1 matches occur in the $conv4\_3$ layer, and there are 273 such matches. This means that 15 percent of the ground truth bounding boxes have best matches of this type with the priorboxes. The total number of overall matches with this configuration is 20,013 and the total number of candidate priorboxes is 7308. We hypothesize that shifting the distribution of priorboxes towards those of smaller scales will reduce the number of type 1 matches, which will contribute to an increase in performance. This is because the ground truth bounding box scale distribution in the FlickrLogos-32 dataset is skewed towards ground truth bounding boxes of small scale.

In Experiment 2, we alter the priorbox distribution so that the $conv4\_3$ layer uses default priorboxes with scale 0.09, and the remaining detection heads use default priorboxes with scales regularly spaced between 0.17 and 0.95. This modification keeps the total number of priorboxes used fixed, but it shifts the scale distribution to better match that of the ground truth bounding boxes in the FlickrLogos-32 dataset. The total number of type 1 matches decreases by 40 to become 12 percent of the number of ground truth boxes, but the number of type 3 matches decreases to 17,141.

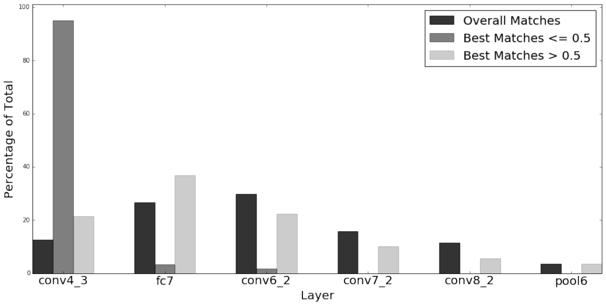

Fig. 2. The distribution of priorbox and ground truth bounding box matches among the detection layers for Experiment 1. The best matches ≤ 0.5 distribution depicts matches where the best matching priorbox for a ground truth box has a Jaccard overlap of at most 0.5. The best matches > 0.5 distribution depicts matches where the best matching priorbox for a ground truth box has a Jaccard overlap of greater than 0.5. The overall matches include all the best matches as well as any match between a priorbox and ground truth box with Jaccard overlap greater than 0.5.

To further improve the matches between priorboxes and ground truth boxes with small scales, we introduce additional priorboxes of small scale in Experiment 3 while keeping the same scale distribution used in Experiment 2. This modification increases the total number of priorboxes. In the $conv4\_3$ layer, we include a max scale equivalent to the min scale of the $fc7$ layer. We also introduce aspect ratios of 3 and $\frac{1}{3}$ so that the $conv4\_3$ detection head also has 6 default priorboxes. This decreases the number of type 1 matches to 227 while increasing the number type 3 matches to 19,045. By introducing additional aspect ratios, the added priorboxes have greater Jaccard overlap with the small scale ground truth bounding boxes.

See Table 1 for the mAP scores on the FlickrLogos-32 *test* set after training for 60k iterations with the priorbox parameter configurations in each experiment. Both the modification of the priorbox distribution and the increase in the number of priorboxes of small scale improve detection performance. Altering the priorbox distribution improves performance by 0.5% and increasing the number of small scale priorboxes improves performance by an additional 1%. These results support the intuition that better matching the priorboxes to ground truth boxes of small scale in the FlickrLogos-32 *trainval* set allows the SSD framework to detect logos of small scale in the *test* set.

5.2. Individual Detection Head Performance

In addition to measuring the overall mAP, we measure the mAP scores of the individual detection heads. The results show that modifying the priorbox distribution and increasing the number of priorboxes with small scale improves the performance of the detection heads that match priorboxes with

Experiment	Priorbox distribution modified?	# of small Priorboxes increased?	mAP%
1	No	No	69.1
2	Yes	No	69.6
3	Yes	Yes	70.6

Table 1. mAP% scores on FlickrLogos-32 *test* set after 60k iterations of training using SSD 300x300.

Parameters Modified?	$conv4\_3$	$fc7$	$conv6\_2$	$conv7\_2$	$conv8\_2$	$pool6$
No	21.2	42.4	33.1	25.2	16.6	8.8
Yes	29.6	40.8	34.2	24.6	16.7	9.5

Table 2. mAP% scores on FlickrLogos-32 *test* set after 60k iterations of training for each individual detection head.

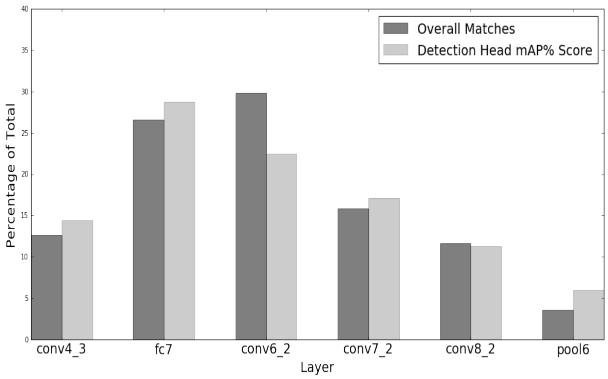

Fig. 3. The distribution of priorbox and ground truth box matches plotted against the mAP scores for the individual detection layers in Experiment 1. For each layer, the percent mAP score is computed as 100 times the mAP score for that layer divided by the sum of all mAP scores for the individual detection heads.

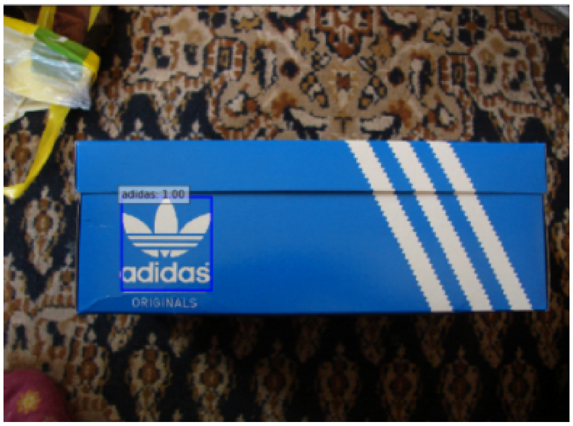

Fig. 4. Sample detection of the Adidas logo using our SSD 500x500 network. The network outputs predicted localization coordinates and a confidence score for the predicted class.

small ground truth boxes and detect small logos in the *test* set. To perform this analysis, we isolate each detection head from the remaining detection layers and test the network using only the multibox priorbox, confidence, and localization layers attached to that head.

As shown in Table 2, the $conv4\_3$ detection head performance increases by 8% from the SSD 300x300 network with the original priorbox distribution to the SSD network with the modified priorbox distribution and increased number of small scale priorboxes. The remaining detection head performances have marginal differences between the two parameter settings. Since the distribution of the ground truth bounding boxes in the FlickrLogos-32 dataset is concentrated at small scale bounding boxes (see Figure 1), the relatively large increase in performance of the $conv4\_3$ detection head, which matches the smallest scale priorboxes with the ground truth bounding boxes, contributes to improvement in overall performance (see Table 1).

The distribution of the individual detection head mAP scores follows the same general trend as the distribution of overall matches between the priorboxes and ground truth boxes (see Figure 3). This analysis suggests that the distribution of matches between priorboxes and ground truth boxes is an indicator of the distribution of the individual detection head performances, given that the ground truth bounding boxe sizes in the training set have a similar distribution as the ground truth bounding box sizes in the test set. The result provides further support for the intuition that well-matched priorboxes and ground truth bounding boxes at train time contribute to better performance at test time.

5.3. SSD 500x500 Experiments

We replicate the SSD 300x300 experiments with SSD 500x500. With SSD 500x500, we resize the input image to 500x500 instead of 300x300. We hypothesize that using a larger input size will further boost mAP by improving the detection of logos of small scale in the input. The runtime per-

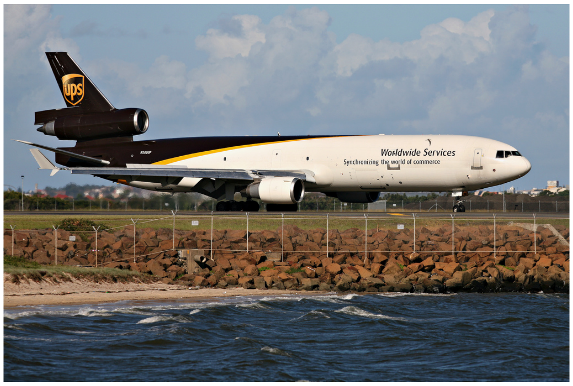

Fig. 5. Sample Ups brand image in the FlickrLogos-32 dataset.

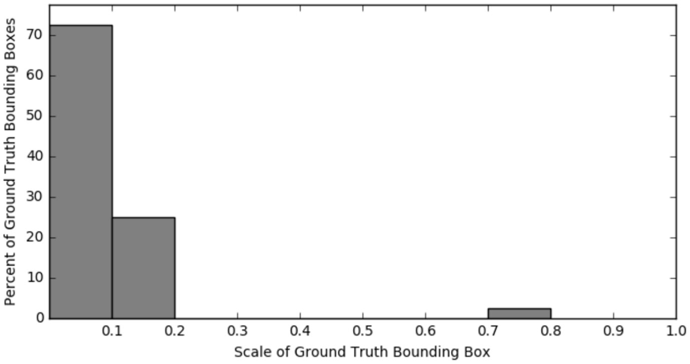

Fig. 6. Ground Truth Bounding Box Distribution for the Ups brand.

formance is still better than Faster RCNN [10] (7FPS Vs 23 FPS on Titan X GPU) [9], and our experiments demonstrate that we surpass the mAP achieved in the DeepLogo paper [5].

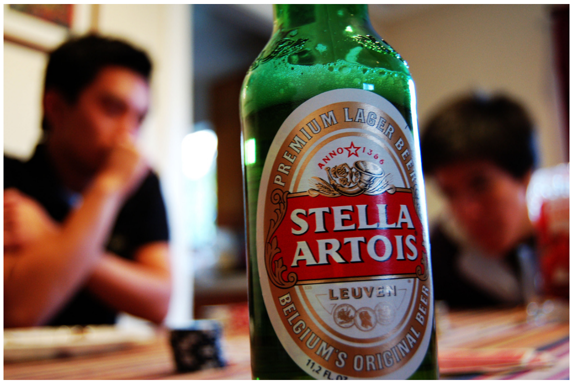

Fig. 7. Sample Stellaartois brand image in the FlickrLogos-32 dataset.

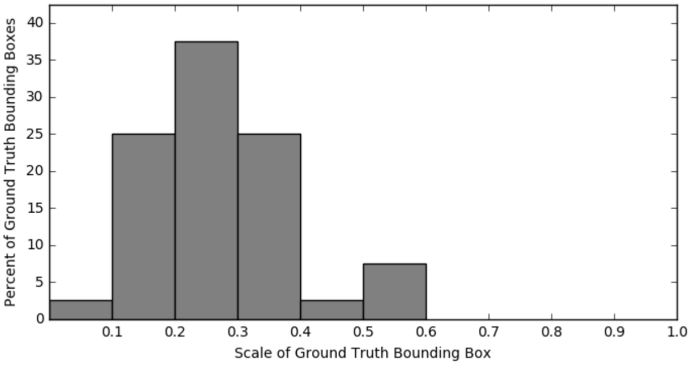

Fig. 8. Ground Truth Bounding Box Distribution for the Stellaartois brand.

As shown in Table 3, the priorbox distribution modifications that we make result in an improvement in detection performance as with the corresponding changes made for SSD 300x300. In addition, changing the input image size to 500x500 from 300x300 increases mAP by 5.3%. Table 4 depicts a comparison between the average precisions of the individual logo classes using the Fast RCNN (FRCN) method described in the DeepLogo paper [5] and the SSD 500x500 method with both modified priorbox distribution and increased number of small priorboxes. For 17 of the 32 brands, the AP increases from FRCN to SSD 500x500. The average increase in AP for these 17 brands is 11.9%, while the average decrease in AP for the remaining brands is 5.6%. The overall increase in mAP between the two methods is 1.5%.

The FlickrLogos-32 test set is relatively small compared to other object detection datasets in the computer vision literature, which results in fluctuation of the mAP scores when using both methods. However, given this dataset as a means of comparison, our approach performs better at detecting logos in images, particularly those that are more difficult to detect, which include logos of small scale and different aspect ratios. We analyze the ground truth box distribution of the brands for which AP increases versus that of the brands for which AP decreases. We find that with brands for which the AP increases, the ground truth bounding box distribution for that logo class is concentrated at bounding boxes with small scale, whereas for several of the brands for which AP increases, the corresponding ground truth bounding box distribution is less concentrated at bounding boxes of small scale. This result is shown in Figure 5, which is an example of a logo class for which AP increased between the two methods, and Figure 7, which is an example of a logo class for which AP decreased between the two methods.

6. CONCLUSION

In this paper we propose a new architecture for logo detection based on the SSD detection framework. We show that this de-

Experiment	Priorbox distribution modified?	# of small Priorboxes increased?	mAP%
1	No	No	74.3
2	Yes	No	74.9
3	Yes	Yes	75.9

Table 3. mAP% scores on FlickrLogos-32 *test* set after 60k iterations of training using SSD 500x500.

Method	adidas corona google ritt	aldi dhl guin shell	apple erdi hein sing	becks esso hp starb	bmw fedex milka stel	carls ferra nvid texa	chim ford paul tsin	coke fost pepsi ups	mAP%
FRCN + VGG16 (Iandola et al.)	61.6 92.9 85.2 63.0	67.2 53.5 89.4 57.4	84.9 80.1 57.8 94.2	72.5 88.8 N/A 95.9	70.0 61.3 34.6 82.2	49.6 90.0 50.3 87.4	71.9 84.2 98.6 84.3	33.0 79.7 34.2 81.5	74.4
SSD 500x500 + VGG16 + Modified Params (ours)	63.1 98.8 89.8 78.7	64.6 73.7 87.9 50.5	69.5 76.9 69.1 91.1	66.7 89.2 59.4 99.5	80.4 78.1 57.6 71.9	67.7 87.7 46.4 81.8	67.3 89.5 96.6 86.6	76.6 87.2 52.9 70.3	75.9

Table 4. Fast R-CNN vs SSD 500x500 (Experiment 3): FlickrLogos-32 localized detection APs.

sign is able to surpass prior state of the art on the FlickrLogos-32 dataset [5] while increasing computational performance by 46X. We also introduce an error analysis approach to help tune SSD hyper-parameters for improved mean average precision. We validate this analysis by introducing a method to measure the individual detection head performance of SSD. Our final trained SSD 500x500 network is able to perform the detection of logos in images in realtime. Collectively, we refer to this work as *Single Shot Logo*.

7. REFERENCES

[1] Ditto labs inc.

[2] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan. 2015.

[3] R. Girshick. Fast r-cnn. In *International Conference on Computer Vision (ICCV)*, 2015.

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition*, 2014.

[5] F. N. Iandola, A. Shen, P. Gao, and K. Keutzer. Deeplogo: Hitting logo recognition with the deep neural network hammer. *CoRR*, abs/1510.02131, 2015.

[6] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. 2014.

[7] Y. Kalantidis, L. G. Pueyo, M. Trevisiol, R. van Zwol, and Y. Avrithis. Scalable triangulation-based logo recognition. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, page 20. ACM, 2011.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. SSD: Single shot multibox detector. *arXiv preprint arXiv:1512.02325*, 2015.

[10] 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 (NIPS)*, 2015.

[11] S. Romberg, L. G. Pueyo, R. Lienhart, and R. van Zwol. Scalable logo recognition in real-world images. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, pages 25:1–25:8, New York, NY, USA, 2011. ACM.

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[13] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[14] W.-Q. Yan, J. Wang, , and M. S. Kankanhalli. Automatic video logo detection and removal. 2005.

[15] G. Zhu and D. Doermann. Automatic document logo detection. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 2, pages 864–868. IEEE, 2007.