An Approach Combined the Faster RCNN and Mobilenet for Logo Detection

Terry Mudumbi, Naizheng Bian*, Yiyi Zhang, and Florian Hazoume

College of Computer Science and Electronic Engineering, Hunan University, Changsha, China.

*Email: 741565912@qq.com

Abstract. Although deep learning object detection tools such as Faster Recurrent Convolution Neural Network (Faster R-CNN) has demonstrated good performances in object detection, they also have a limited success rate for some applications. It is due to the lack of refinedness of feature maps for accurate localization, the insensitivity for small scale objects and fixed-window feature extraction in Region Proposal Network (RPN). In this paper, we performed a meticulous examination of both the proposal and the classification process by evaluating the adequacy of feature representations from different stages of the feature sequencing. We presented an approach to improve the Regional Proposal Network (RPN) by appropriate anchors selection, and proposed a modification by combining Faster R-CNN and MobileNet which influences higher-resolution feature maps for mobile devices. The results demonstrate that Faster R-CNN architecture with MobileNet has the best detection accuracy. The experiment result showed that we managed to achieve a final accuracy of 92.4% on a NVIDIA GeForce Gtx 1070 compare to previous work that achieved 90.8% of accuracy and found that our models performed well at the detection, with very low false positive rates possible for a fairly reasonably.

1. Introduction
Brand recognition is a very challenging topic with many useful applications in localization recognition, advertisement and marketing. Traditionally, logo detection is solved by defining key points and texture descriptors combined with statistical classifiers, such as Nearest Neighbour (NN) or Support Vector Machines (SVM) [1, 2]. In [3, 4] the images can be damaged by many artifacts and image distortions. In order to handle the limitations of the above models, in [5] they solved the problem by proposing Recurrent Convolution Neural Network (RCNN) and apply it for the classification task. Recent advances in hardware has increased multiple methods for detection using CNNs have been presented [6]. Through the Region-Based Convolutional Neural Network (RCNN) came out very good results compared to previous methods. In RCNN [7], regional proposals submitted using the proposal methods are implemented on CNN network.

The disadvantage of RCNN is its expensive computational cost, since each area is treated separately. By solving the expensive computational the Fast RCNN [8] improves the RCNN defect by using region proposal as detectors for a common feature map. Method [9] proposed the complex for object detection called Fast R-CNN that combines R-CNN and Spatial pyramid pooling networks (SPPnet). The network is formed by a set of convolutional layers, fully-connected layers, an external region proposal method typically Selective Search (SS) [10, 11, 12] and a Region of Interest (RoI) pooling layer. However, their method is still computationally heavy for real-time processing on an embedded platform. In the work of
Huang et al. [13], they showed the Single Short Multibox Detector (SSD) approach together with SqueezeNet and MobileNet [14] as the backbone networks. Although SSD with SqueezeNet backbone results in a smaller model than MobileNet, [15, 16] says that the results are less accurate and its computation is slightly more expensive.

In this article, we want to improve the Faster R-CNN’s mAP (mean average precision) by processing higher resolution layers objects which makes the feature map larger. Therefore, we proposed an approach based on Faster R-CNN and MobileNet for Logo detection and created implementations for a Faster R-CNN and MobileNet architectures which influence higher-resolution feature maps for mobile devices. The approach described the core layers that MobileNet is built as a good feature-extractor. Finally, we explored the networks that are able to locate and classify multiple logos in real time.

The rest of this paper is organized as follows: Section II analysis model and then discusses the details of our approach. Section III analysis the proposed approach and evaluates the performance through experiments. Finally, Section IV concludes the paper and points out the future research directions.

2. Proposed Approach

Our proposed architecture consists of 2 key steps: Firstly, we mathematically design the suitable sizes of anchor boxes with 6 aspect ratios {0.48, 0.65, 1.09, 1.28, 1.48, and 1.70} and perform the effectiveness of their choice. The RPN has proposed a set of bounding boxes with a trusted rating associated with potential logo. Secondly, we described detailed analysis of these fully convolution architectures by using MobileNet as a feature-extractor. To every proposal, the corresponding feature maps use the resolution in a fixed-size representation, many layers that are fully associated with presentations are classified within specific bounding box regression. On the features of different feature maps, the deep layers are likely to be able to provide better properties, which means that single activation processes for input stimuli are more specific than earlier layers. We establish that, features of the previous layers can provide performance for small objects that matches or even surpasses the performance of features from the deep layers.

We rate our observations on the well-known FlickrLogos-32 dataset as an extension of the Faster R-CNN pipeline.

2.1 Regional Proposal Network

The main point of the RPN is to propose a set of bounding boxes with a trusted rating associated with potential logo. We modified the RPN to detect a logo with this configuration \(B_a = 9\) anchors. Since each anchor box acts as a detector for sliding windows in a grouped image area, there are:

\[
B_p = B_a \times \frac{w}{f_s} \times \frac{H}{f_s}
\]  

Where \(f_s\) corresponds to an anchor. The RPN structure conv1, conv2…, conv5. The general observation has been confirmed in part by the fact that increasing the efficiency of highway classification on the ImageNet. This applies to at least two common phase detection devices, such as Faster-RCNN and MobileNet. We used a network of regional presentations (RPN), consisting of two layers to locate the regions that can contain objects in feature maps (image). The network uses the RoI pool layer to reduce and resize resource maps based on proposals from that region. The maps use the new features of each region to select frame into three fully connected layers. In this work, MobileNet which took the layers as learning functions was used as a convolutional network the original feature extraction contains several layers and the first convolution stack structures acquired through transfer learning by using MobileNet.

2.2. Faster RCNN_MobileNet

Our approach has two steps forming the current object detection such as: The first one consists on identifying ROI from images. These ROI can be considered as references in recommending some possible object location that are more carefully developed in the second step. As shown in figure 1 with 5 convolutional layers and 3 fully connected (FC) layers. While Faster R-CNN only uses the features of
the last convolutional layer to localize and classify. In the first two convolution layers, after each successive layer and one Max-pooling layer, respectively. In the next three levels, just after each convolution layer, there is only one level of ReLU. In particular, on three levels, 3, 4 and 5, their outputs are also used as input data for the three levels of pooling of the ROI and the corresponding normalization levels. For each RPN anchor constituting a fully convolutional network, a degree is predicted which makes it possible to measure the probability of this anchor which contains the element of interest. In addition, the RPN provides the acceleration and measurement coefficients for each anchor that is part of the peripheral regression mechanism, thereby improving the position of the object.

Figure 1. Our proposed Approach with 5 convolutional layers and 3 fully connected (FC) layers.

To properly illustrate this problem, considering the situation in figure 2a: we supposed that a second-order ground truth bounding box $A_2$ is delimited by a side length $x_2$ and a square anchor box $A_1$ of side length $x_1$.

$$t \leq IoU(A_2, A_1) = \frac{|A_2 \cap A_1|}{|A_2 \cup A_1|} = \frac{x_2^2}{x_1^2}$$

In general, an anchor is considered a positive example if it contains an IoU greater than 0.5 for a ground truth objects.

Figure 2. (2a) $IoU$ can be expressed as the ratio of the areas of the limit box in the case of limits of the same proportion. (2b) shows two bounding boxes of the same size when the anchors are selected in stride $y$.

To classify the anchor box as a positive example, ask the $IoU$ to exceed a certain threshold $t$. It follows that for $t > \sqrt{t^{-1}}$, the anchor cannot cover the field of truth sufficiently enough to be classified as a positive example. The same thing applies to non-quadrature anchors, provided that the ratio of groundtruth boxes and anchor boxes correspond to each other. For the above considerations, we suppose
that there is an attachment point where the corner of an anchor is perfectly aligned with the ground truth example. In practice, this is not the case, because the network performance map based on RPN is usually much smaller than the original image. The reduction factor $y=1$ between the source image and the object map effectively creates a network of anchors with stride $y$. To examine the effect of the characteristic resolution of the card on the potential RPN to determine the state of small objects, considering the situation in figure 2b, we suppose the case of quadratic atoms of phase $A_2$ and the existence of an anchor box $A_1$ of scale and the corresponding form factor. In the worst case, each box is moved a distance $\frac{t}{2}$. The IoU between these boxes can be represented by:

$$ IoU(A_2, A_1) = \frac{(x_2 - \frac{t}{2})^2}{(x_2 - \frac{t}{2})^2 + 2 \left(\frac{x_2}{2} - x_1\right)^2} $$

(3)

The initiative learning rate is $3e-3$ and the stride size is set to, $x = 16$. Assuming that $t = 0.5$, this gives the minimum size of the detectable object. This indicates that for a small fraction of our size distribution, we need an object map at a higher resolution.

Second one consists of a depthwise decomposable approach to integrate the local context from each selected scale of the feature maps and then add it again. To factorize the convolution, a depthwise convolution a removable winding is used, as this can help to significantly reduce calculations and parameters. In CNN, convolution filters extract objects from input property maps through a sliding window. Different size of pixels will be extracted by the other size of filters, so that they can be utilized as context extractor’s tools. By using these tools, we propose an end-to-end convolution approach with these context extractors that can be subdivided to give context to a local context. With a separable convolution depthwise, the computation is reduced by a factor of 8-9 times and our detection speed rate can be increased effectively. We define the convolution depth ratio in percentage $M = (0.25, 0.50, 0.75, 1)$ and adjust $t = 0.5$. If Object is in a group of $C$ classes, $L$ is the group of basic objects $L_c$ ($c \in C \in \text{CandN}$ and there is a set of sentences of objects, then we can estimate the performance of this class's RPN, that its average value is $Avg(c)$ expressed by:

$$ Avg(c) = \frac{1}{|L_c|} \sum_{l \in L_c} \max_{n \in N} IoU(l, n) $$

(4)

3. Experiment and Result

To investigate the effects of RPN performance on the object size, different versions of the FlickrLogo-V2 dataset have been designed, and applied the following algorithm to each image. We began by choosing a point where the maximum distance between the two unbounded square boxes. This point specifies two axes that can divide the image between them into four parts. We ensure that the axes of division do not break any other point of truth. If no separation is found, the image will be discarded. For each resulting layer, which contains more than one main element, this process is applied repeatedly. After applying this algorithm, there is only one instance of the object in each image, which is measured to match the desired target size. To track the performance of RPNs at different levels, we create 3 RPN which are: RPN conv3, RPN conv4 and conv5 RPN. These networks use functions of level 2, 2 and 4 levels conv3 to predict the proposals of objects. Objects are transmitted through the normalization level, which normalizes the activation to change the mean and unit of measurement. We normalized the activation processes with respect to the training package then we put a standard RPN on the top of this normalization function, which is materialized by a $3 \times 3$ convolution using a similar number of channels as the previous layer.

Our approach improved by appropriate anchors selection in the Regional Proposal Network (RPN), had negative mining and non-maximum suppression. Our approach integrates a process that can be described as follows:

- **Regional proposal network** uses a RPN network that incorporates the regional generation network into a framework
- **Feature extraction**: includes many layers of conv+relu, and the first few convolution batch structures can be obtained through transfer learning. Faster R-CNN with mobilenet feature
extractor for 450k iteration, batch 8, learning rate 3e-03. A supplemental conv+relu layer is added to export the feature. According to each image position, 9 possible filter windows are considered, which are called anchors.

- **Classification and box regression:** At each point, the cls_score output keys have 9 anchor points for the forward and backward probabilities, the window regression prediction layer (bbox pred) displays each position and the 9 anchor matching window must move the hyper-parameters. Previous initializing hyper-parameters can be established as follows:
  - The selected appropriate scales and aspect ratios of the anchors, making it more efficient to detect object. For the normal Faster R-CNN (with inception_v1, inception_V2, etc...) used 9 anchors with 4 scales and 4 aspect ratios. These aspect ratios (height/width) were { 0.25, 0.5, 1.0, 2.0}
  - RPN network has been used to extract area proposal by changing the anchors from 4 scales {0.25, 0.5, 1.0, and 2.0} to 6 scales {0.125, 0.25, 1.0, 1.5, and 2.0} and 6 aspect ratios {0.48, 0.65, 1.09, 1.28, 1.48, and 1.70}. According to the experiment, the appropriate aspect ratios of anchors obtained higher accuracy and lower loss function than the others ones.
  - Some valuable negative samples, such as false positives, were selected from the negative ones. At the time of the prediction, we use a maxima deletion algorithm to filter the multiple boxes per object that can appear.

As the training process progresses, the expectation are that total loss (errors) gets reduced to its possible minimum (That means we have to get less than 1 or lowest one). We ran our training job for 500k steps (took about one day) and stopped at a total Loss (errors) value of 0.07413.

The result of The Function loss of each part after 450 k steps is shown in figure 3a that is the combination of: the proportion of bounding boxes produced by RPN that are correctly classified (as the correct object class) and some distance measure between the predicted and target regression coefficients.

![Figure 3](image)

**Figure 3.** Show us the result of The Function loss of each part after 450 k steps (3a) classification_loss function. (3b)Localization loss function. (3c)Total loss function.

The classification loss $l_{\text{cls}}$ is the log loss function over two classes, as we can easily translate a multi-class classification into a binary classification by predicting a sample being a target object versus not $L_{\text{smooth}}^1$ is the smooth L1 loss?

$$l_{\text{cls}}(p_i, p_i^*) = -p_i^* \log p_i^* - (1 - p_i^*) \log(1 - p_i)$$

(5)

The result of localization loss in figure 3b uses feature maps that belong to the proposed RoIs:

$$l([p_i], \{t_i\}) = \frac{1}{N_{\text{cls}}} \sum_i l_{\text{cls}}(p_i, p_i^*) + \frac{\lambda}{N_{\text{box}}} \sum_i p_i^* L_{\text{smooth}}^1(t_i - t_i^*)$$

(6)

The multi-task loss function combines the losses of classification and bounding or regression, as shown in figure 3c:

$$l = l_{\text{cls}} + l_{\text{box}}$$

(7)
The main interest here is about our approach with area proposals. In training and in testing, our approach collects raw images and area proposals having important features in the images with the shape of bounding boxes. This stage is about localizing and classifying every bounding box area proposal as a kind of logo or "background". It means that the region in the area proposal is not a logo at all. In case the area proposal includes a logo, it also returns a bounding box regression as output, modifying the area proposal to better highlight the area with the logo. Our area proposals were made from very deep search. After that, these areas on the images are given for training and testing purposes. The AP values per class are shown in Table 1.

Table 1. Experimental result of each classes using FlickrLogos-32. Detection exhibits fairly good detection performance, especially on distinctive logos such as that of Coca-Cola with 98.1%.

| method             | Adidas | Aldi | Apple | Becks | Bmw | Cars | Chim | Cocacola |
|--------------------|--------|------|-------|-------|-----|------|------|----------|
| Corona             | 89.7   | 94.9 | 94    | 90.7  | 88.4| 93.5 | 89.4 | 98.1     |
| Dhl                | 87.3   | 96.6 | 95.1  | 93.9  | 94.8| 90.2 | 89.9 | 91.3     |
| Erdi               | 97.2   | N/A  | 97.4  | 92.1  | 95.3| 91.8 | 92.3 | 95.7     |
| Esso               | 85.3   | 95.0 | 89.9  | 89.8  | 90.7| 94.9 | 88.9 | 90.3     |
| Fedex              |        |      |       |       |     |      |      |          |
| Ferrari            |        |      |       |       |     |      |      |          |
| Ford               |        |      |       |       |     |      |      |          |
| Fosters            |        |      |       |       |     |      |      |          |
| Google             |        |      |       |       |     |      |      |          |
| Guiness            |        |      |       |       |     |      |      |          |
| Heineken           |        |      |       |       |     |      |      |          |
| Hp                 |        |      |       |       |     |      |      |          |
| Milka              |        |      |       |       |     |      |      |          |
| Nvidia             |        |      |       |       |     |      |      |          |
| Paul               |        |      |       |       |     |      |      |          |
| Pepsi              |        |      |       |       |     |      |      |          |
| Ritt               |        |      |       |       |     |      |      |          |
| Shell              |        |      |       |       |     |      |      |          |
| Sing               |        |      |       |       |     |      |      |          |
| Starb              |        |      |       |       |     |      |      |          |
| Stel               |        |      |       |       |     |      |      |          |
| Texas              |        |      |       |       |     |      |      |          |
| Tsins              |        |      |       |       |     |      |      |          |
| ups                |        |      |       |       |     |      |      |          |
| mAP                |        |      |       |       |     |      |      |          |
| 92.40              |        |      |       |       |     |      |      |          |

A commonly used metric for performance is mean average precision (mAP) which is single number used to summarize the area under the precision-recall curve. mAP is a measure of how well the model generates a bounding box that has at least @0.5IOU overlap with the groundtruth bonding box in our test dataset and obtained an average accuracy of 92.4% as process in figure 4.

![Figure 4](image_url)  
*Figure 4. Show us the result of mean average precision (mAP) after 100k steps.*

In this work, the experimental environment: NVIDIA GeForce Gtx 1070, CUDA9.0, Ubuntu16.04, memory 8GB, Tensorflow 1.8, Android Sdk 3.2, Android Ndk-r16b and Huawei Mate 10(EMUI 9.0.0181). TensorFlow provides integrated support for calculating the mAP metric via TensorBoard during training and evaluation. By conducting a series of experiments, we collected enough data to evaluate the performance of the models. We have made the process result as shown in figure 5 by conducting a series of experiments, the performance measure of logo detection, we collected FlickrLogo-V2 dataset and added 6400 images to evaluate the performance of the Average Precision (AP) for each individual logo class, and the mean Average Precision (mAP) for all classes. A detection is considered being correct when the Intersection over Union (IoU) between the predicted and groundtruth exceeds 50%. If predicted and true classes match, and the Intersection over Union (IoU) between the predicted bounding box $A_2$ and the ground truth bounding box $A_1$ is larger than 50%, the prediction is a True Positive ($TP$), otherwise it is a False Positive ($FP$).
Figure 5. Shows the logo detection through Huawei mate 10.

\[
\text{IoU}(A_2, A_1) = \frac{|A_2 \cap A_1|}{|A_2 \cup A_1|}
\] (8)

Precision values are computed to calculate the mAP:

\[
\text{Precision} = \frac{TP}{TP + FP}
\] (9)

4. Conclusion
In this paper, we improved the Regional Proposal Network (RPN) by adopting appropriate anchors selection, and integrated Faster R-CNN with MobileNet to get higher-resolution feature maps for mobile devices. At the first, we initialize the network hyper-parameters using the selected appropriate scales and aspect ratios of the anchors. Secondly, we evaluated our approach in detail for the motion phase and the classification phase using FlickrLogo-V2 that the system achieved object accuracy of 92.4%. Throughout experiments performed on different feature maps using MobileNet as backbone which influences higher-resolution feature maps for mobile devices and we notice that the resolution of the feature map has an important role in detecting reliable logo detection. Our approach can usually detect more precisely in earlier feature maps, even if the logo are not so expressive from deeper layers. We have also shown that the choice of anchoring measures for detecting objects is important and specified a criterion for selecting an anchor scale based on the accuracy of the required translation. Finally, we documented our observations as a simple Faster R-CNN extension that can improve the overall performance of a true global corporate dataset for logo detection. In future work, we would like to think about how to increase productivity in classifying objects.

In future work we would like to integrate into our approach using an improved network architecture based on Feature Pyramid Networks by selecting a suitable resolution for generating proposals that may improve the performance of the RPN considerably.

References
[1] Yifei Zhang #1,2, MingMing Zhu#1, Daling Wang #1,2, and Shi Feng #1, Logo Detection and Recognition Based on Classification Springer International Publishing Switzerland 2014
[2] Hang Su, Shaogang Gong, Xiatian Zhu Queen Mary University of London Vision Semantics Ltd. Scalable Deep Learning Logo Detection
[3] Alireza Alaei, Mathieu Delalandre Laboratoire d’Informatique (LI EA6300, A Complete Logo
Detection/Recognition System for Document Images 2014 11th IAPR International Workshop on Document Analysis Systems 12 June 2014

[4] Yuanyuan Li #1, Qiuyue Shi #2 Jiangfan Deng #3, Fei Su #4. A logo is a graphic representation of a brand or company that is used on different communication media

[5] The Anh Pham #1, Mathieu Delalandre #2, and Sabine Barrat #3. A Contour-based method for logo detection. In Document Analysis and Recognition (ICDAR), 2011 International Conference on, pages 718–722. IEEE, 2011

[6] Hongye Wang #1 and Youbin Chen #2. Logo detection in document images based on boundary extension of feature rectangles. In Document Analysis and Recognition, 2009. ICDAR ’09. 10th International Conference on, pages 1335–1339. IEEE, 2009.

[7] Girshick. In: CoRR abs/1504.08083 (2015). Fast R-CNN. ArXiv: 1504.08083

[8] Ross B. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 39, NO. 6, JUNE 2017

[9] Zibo Meng, Xiaochuan Fan‡, Xin Chen‡, Min Chen‡ and Yan Tong. Detecting Small Signs from Large Images arXiv:1706.08574v1 [cs.CV] 26 Jun 2017

[10] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation Int. J. Comput. Vision, vol. 59, no. 2, pp. 167–181, 2004

[11] P. Arbelaez, M. Maire, C. C. Fowlkes, and J. Malik #4 From contours to regions: An empirical evaluation in CVPR, pp. 2294-2301, IEEE, 2009

[12] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders #4. Selective Search for Object Recognition. International Journal of Computer Vision (2013). DOI: 10.1007/s11263-013-0620-5

[13] Zhuoyao Zhong, Lei Sun, Qiang Huo # School of EIE. An Anchor-Free Region Proposal Network for Faster R-CNN based Text Detection Approaches

[14] Fei Xia and HuiZhou Li. Fast Detection of Airport on Remote Sensing Images with Single Shot MultiBox Detector

[15] Wei Liu (B), Dragomir Anguelov2, Dumitru Erhan3. SSD: Single Shot MultiBox Detector ECCV 2016: Computer Vision – ECCV 2016 pp 21-37

[16] Karanbir Chahal and Kuntal Dey. A Survey of Modern Object Detection Literature using Deep Learning arXiv:1808.07256v1 [cs.CV] 22 Aug 2018