# Blurring Operation Of Brands Including Objects With Deep Learning Methods

Özgür KAN, Eray CINCI
Bilgisayar Mühendisliği Bölümü
Yıldız Teknik Üniversitesi, 34220 Istanbul, Türkiye
ozgur.kan@std.yildiz.edu.tr, cincieray@gmail.com

***Özetçe* —Günümüzde popüler markalar sürekli bir yarış halindedir ve sektörlerinde lider olmayı hedeflerler.Markaların, satış rakamlarını arttırmak için kullandığı yöntemlerin başında ürün yerleştirme gelir.Ürün yerleştirme basit,etkili ve insanlara ulaşma oranı çok yüksek olan bir yöntem olduğu için son dönemlerde popülerliği giderek artmıştır. Genellikle dizi veya film sektörlerinde kullanılan bu yöntem sayesinde yapım şirketleri önemli gelirler elde etmektedir. Ayrıca Ürünlerin dizilerde veya filmlerde gösterilmesi reklam veren markaların prestij ve popülerliğini arttırır. Bu nedenden dolayı dizi ve film sektöründe ürün yerleştirme yaygın olarak kullanılmaktadır.**

**Bu projede nesne tespiti ağlarından biri olan Faster R-CNN kullanılarak videolarda istenmeyen markaların tespit edilmesi ve tespit edilen bu markaların bulanıklaştırılmasını sağlayan bir sistem gerçekleştirilmiştir.**

***Anahtar Kelimeler—Marka, Ürün Yerleştirme, Bulanıklaştırma, Faster R-CNN***

***Abstract*—Most popular brands are in a constant race with each other in order to be the leader of their own sectors. Wishing to increase the number of product sales, brands use product placement as their primary method. Product placement is a highly simple and effective method for making their product seen. It also has a significant reach to anyone, making it one of the most popular method. Mostly used in movies and television series, brands make a essential revenue with product placement. Moreover, with making their product seen on television, brands increase their prestige and popularity. Thus, product placement is widely used in television series and movie sectors.**

**In this project, unwanted brands are located with a object detection network called Faster R-CNN and blurred.**

***Keywords—Product placement, brand, blur, Faster R-CNN***

## I. Introduction

In recent years, deep learning methods has been use for object detection. CNN has been used for detection of objects in a picture. Deep learning networks has been modified in different ways. These are R-CNN, Fast R-CNN, Faster R-CNN, YOLO etc. These methods are proven to be more efficient against Sliding Window Method. Sliding Windows Method dismantle picture into segments and check every part, on the other side deep learning methods gives every segment to a network once. This has proven to be way faster. We will use Faster R-CNN[1] in our project. Real-time efficiency and consistency made it an obvious choice.

In Faster R-CNN, both region proposal generation and objection detection tasks are all done by the same convolution networks. With such design, object detection is much faster.

## II. Related Work

First project to be examined was made by International Journal of Innovative Research in Computer and Communication Engineering in 2016. This project's aim was to detect arabic numbers(0-9). Data set contains 45000 pictures. Each of them labelled by humans. They have used CNN. Since a picture was obtaining only one number they did not need a regional proposal system.[2]

Secondly, we observed a project made by Franck FOTSO. Faster R-CNN was used in this project. Thus, it has common network structure as ours. Caffe library was used instead of Tensorflow(ours). This project has a data set containing 20 pictures. This project was tested after 30000 steps. Mean Average Precision is 75%.

## III. Data

Open Logo dataset was used to get acquire pictures[3]. 17 classes was determined to be detected in this project. We accomplished to gather 300 pictures for each class. Each of them were labelled by us. Classes include {Coca-Cola, Apple, BMW, DHL, Adidas etc.}. Every photos resolution was resized to 500x500 to make training faster. Also we managed to write every label into excel file with coordination of brands existing which will be used as an input for training.

## IV. Faster R-CNN Networks

Faster R-CNN is one of the state-of-art methods for detecting objects, and it was developed originally by CNN. CNN can classify images by identifying their visual features. Krichevsky et al. use CNN to solve the single image classification problem in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC)2012, and won the first place . However, the classification and localization of multiple objects in an image are still problems. To solve these problems, Girshick et al.[4] proposed a region-based convolutional neural network (R-CNN). R-CNN generates region proposals by using a selective search (SS) algorithm , and each one of the region proposals is resized to a fixed size and fed as input to the CNN for feature extraction. The CNN is followed by a support vector machine (classifier) to estimate the kind of image and by a regressor to refine the position of the bounding box. The disadvantages of RCNN are that multiple stages must be trained, the steps are cumbersome, the training is time consuming, and it takes up a lot of space on the disk. In addition, it has very slow training and detection speeds, e.g., the VGG16[5] model requires about 47s to deal with an image by using GPU. In 2015, Girshick et al. proposed an improved scheme for R-CNN, i.e., Fast R-CNN[6]. It processes the entire image with several convolutional and max pooling layers to produce a convolutional feature map. For each region proposal, a fixed-length feature vector is extracted from the feature map through a region of interest (RoI) pooling layer and fed into a sequence of fully connected (fc) layers. Instead of Support Vector Machine (SVM) and the regressor, two sibling output layers at the end of the network produce softmax probability estimates for K classes and refine the position of the bounding box. Fast R-CNN avoided repeating the convolution operation for each region proposal and unified the classification and the regression of the position of the bounding box. Therefore, it improved the speed of training deeper neural networks, such as VGG16. Compared to R-CNN, the speed for the Fast R-CNN training stage is nine times faster and the speed for the test is 213 times faster.

Although Fast R-CNN resolves many of the disadvantages associated with R-CNN, the first step of detecting candidate regions using the SS algorithm still makes the whole network slow. To solve this bottleneck, Faster R-CNN adds a regional proposal network (RPN) to Fast R-CNN, which is an alternative to the SS algorithm and can share convolutional layers. Fig. 1 shows the architecture of the Faster R-CNN. For an input image, first, feature maps are computed by the shared convolutional layers, and the RPN predicts a set of object proposals and corresponding scores indicating which are objects and which are not based on these feature maps. Then, a RoI pooling layer extracts a fixed-length feature vector for each proposal from the above feature maps. After that, the feature vector is fed into fully-connected layers. Finally, the classification layers calculate which category each proposal belongs to and outputs the probability of each category, after which the bounding box regressor refines the spatial location for the proposals.

The RPN is a key part of Faster R-CNN, which is constructed by adding an n x n (typically n = 3) convolutional layer and two sibling 1 x 1 convolutional layers on the top of shared convolutional layers. The first convolutional layer maps each 3 x 3 sliding window in the feature maps to a lower-dimensional feature vector (512-d for VGG16). The two following layers are the classification layer and the regression layer, which output the object-ness score and the bounding box coordinates for each window, respectively. An anchor-based method is proposed to address the problem of multiple scales and sizes of objects. Each anchor is at the center of a spatial window and has a specific scale and aspect ratio. For each spatial window, three scales (1282 , 2562 , and 5122 pixels) and three aspect ratios (1:1, 1:2, and 2:1) are used, i.e., k = 9 anchors are generated. Thus, the RPN classifies and regresses k proposals of different sizes that correspond to k anchor boxes in each window position. Consequently, the classification layer outputs 2k possibility scores of being an object, and the regression layer outputs 4k box coordinates.

## V. Detection Process

It involves the following steps for detecting brands in the image: (1) Taking the image to be tested as the input picture, and the feature map was obtained through 13 layers of the convolutional layer. (2) Using the convolution feature map as the input of RPN network, a large number of proposal regions were generated. (3) The non-maximum suppression (NMS) operation was performed on the proposal region box, and the top 300 proposal regions with the highest scores were reserved. (4) Take the features in the proposal regions of the feature map to form high dimensional feature vectors, and calculate the score of each class from the Fast R-CNN detection network and predict the position of brands. Through the above steps, the identification and localization of the brands in picture images were determined.

## VI. Experimental Results

Every classes mean average precision and overall mean average precision is stored. System is trained for 17 classes only in 110k steps long. Thanks to Tensorboard, results are visualized. Precision is the success of detection of a brand in a frame. Results of a each class is shown below.

**Table 1** mAP of classes

| | |
|---|---|
| Adidas - 66% | Apple - 69% |
| BMW - 62% | Carlsberg - 49% |
| Chimay - 66% | Corona - 42% |
| Coca Cola - 54% | DHL - 56% |
| Erdinger - 69% | Fedex - 58% |
| Ferrari - 70% | Guinness - 80% |
| HP - 53% | Shell - 47% |
| Starbucks - 84% | UPS - 77% |
| Volkswagen - 46% | |

## VII. Performance Analysis

The model created in this study will proceed with the processing of photographs under normal conditions. Therefore, the accuracy of the model is more important than the speed of processing photographs, but it should not be too slow. The model was tested on 3 different graphics cards. The processing time of a photograph is 1.1 second for the Nvidia GT-660M(ours), 0.58 seconds for the Nvidia GTX-950M, and 0.13 seconds for the Nvidia GTX-1660Ti. Data can be diversied with data-augmentation to improve the performance of the model. There are many brands and types of products in market shelves. Considering the similarity of these products to each other, all of them can be easily separated from each other. By increasing the diversity in classes, false-positive values can be reduced and success can be increased.

## VIII. Conclusion

In this project, it was tried to produce a solution for the unwanted brands logos appearance in the scenes. First, the object-detection infrastructure was coded, and then data collection for 17 classes and their diversication were carried out. Then, the suitability of the researched methods to solve the problem was tested on the problem. In the coded system, Faster R-CNN architecture and Inception-V2 are used as network architecture. The Faster R-CNN architecture consists of two networks, the rst of which is the region proposal network(RPN) that estimates where the object is in the photograph, and the second is the network trained to classify the estimated areas. "Custom sized anchor boxes"approach is inspired by the YOLO object detection architecture. This approach allows the region proposal network of the Faster R-CNN architecture to produce better results and better region estimates, specically for the trained data set. From those 17 classes we measured 62% of mean average precision.

## References

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

[2] H. A. Alwzwazy, H. M. Albehadili, Y. S. Alwan, and N. E. Islam, “Handwritten digit recognition using convolutional neural networks,” *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 4, no. 2, pp. 1101–1106, 2016.

[3] H. Su, X. Zhu, and S. Gong, “Open logo detection challenge,” in *British Machine Vision Conference*.

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[5] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[6] R. Girshick, “Fast r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.