Object Recognition and Object Counting using CNNs

L.S. Vishnu Sai Rao (12376), Saurabh Kataria (12637)
Department of Electrical Engineering

CS676A: Computer Vision Project Report
Mentor: Prof. Vinay Namboodari, IIT Kanpur

***Abstract*—Object recognition is a popular task in computer vision. The method usually requires the presence of a data-set annotated with location information of the objects, which is in the form of bounding boxes around the objects. In this project, we have implemented a method to carry out object recognition in a weakly supervised manner i.e., using partially annotated data-set. The data-set provides the information about what objects are present in the image but not where they are present. We have used a Convolutional Neural Network(CNN) based architecture to perform this task. We also validated by experimenting with different architectures that mere information of presence/ absence of objects in an image (weak labels) does provide their location information for free. We have further investigated the suitability of this idea to another application of object counting using supervised training i.e. by providing information of location of objects too (strong labels). We conclude the report by listing the various challenges and progress we made in our project.**

Keywords*—*Object Recognition, CNNs, Weakly Supervised Learning, Object Counting, Bounding boxes

I. Introduction

We have implemented the paper [1] for the application of object detection and localisation in scenery images (images with possible multiple objects of multiple instances). The method is weakly supervised because the labels have only the information about presence/ absence of objects and not their locations. This method is called weakly supervised as the training data-set contains images labelled only with lists of objects they contain and not their locations. Such a form of data is important because firstly, annotating locations in an image is an expensive process and secondly, ‘label-only’ annotations are often readily available in large amounts, e.g. in the form of text tags or full sentences even geographical meta-data. The whole analysis in this report is done on PASCAL VOC dataset [2]. The concept of transfer learning has been used to use first seven convolutional layers pre-trained on ImageNet dataset. Two adaptation layers (convolutional) are appended at the end of architecture to adapt to our problem of object detection and localisation. Precision results show that the object location information comes free with the weak labels provided the training set is large and model has been trained upto a sufficient number of epochs.

We have further extended the idea in [1] to another application of object count. This application aims at detecting the number of objects in a scenery irrespective of their labels. However, this idea can be extended further to counting objects of different classes. We got motivated by the patterns we observed in the activations of the last layer of architecture.

The organisation of report is as follows. Section II contains the details of method used in implementing [1]. Section III contains the motivation for pursuing the application of object count, clear description of the method and conclusion by the experimental results. Section IV describes the possible future work and finally we conclude in section V.

II. Weakly supervised object recognition

A. Method overview

This method is implemented as described in paper [1]. It uses the concept of transfer learning [3] in the implementation of the CNN architecture. In this form of learning, a pre-trained architecture is incorporated into the current model and training is done only on few layers in the current model. This saves huge amount of training time. We have used a pre-trained architecture trained on ImageNet data-set [4] in a manner used in the papers [1] [5]. The ImageNet database [4], consists of tightly cropped images of single objects which enables the pre-trained architecture to recognise individual objects. Two fully connected adaptation layers are added at the end of the pre-trained architecture, which adapts the new combined architecture to recognise individual objects in a cluttered image with multiple objects in it. Training for the combined architecture has been done on Pascal VOC2012 data-set [6]. A flow-chart describing the concept of transfer learning as used in this method is described in Fig.1 Two additional ideas can be incorporated into the CNN architecture which

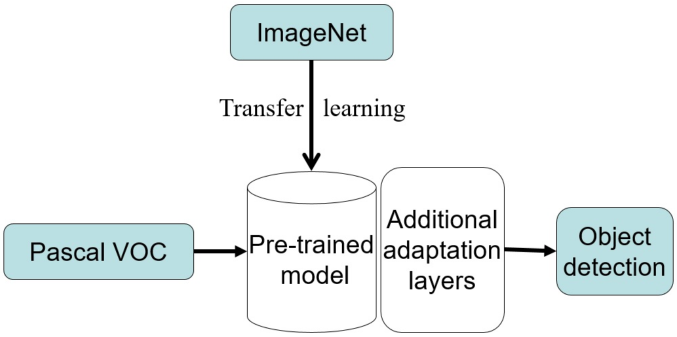

Fig. 1: Weak supervised object recognition flowchart

are firstly, the sliding window training and secondly, multi-scaling of input images. Sliding windows can be implemented using the convolutional layer of the CNN. Sliding window recognition would help us to recognise objects present at different locations in an image. Besides this, multi-scaling of the input images can be done to recognise objects irrespective of their sizes. These additional ideas have been implemented as described in [1].

B. Implementation

The method of weakly supervised object recognition has been implemented on an Nvidia GPU with $2GB$ RAM. The system ran CentOS and Torch was used to implement the CNN architectures. Training has been done on the train dataset provided by Pascal VOC2012. Testing on the test dataset of Pascal VOC2012 [6] could have been done by uploading the results on their server and getting accuracies after a week. In order to save time, testing in this project has been done on the test dataset provided by Pascal VOC2007 [7]. Memory constraints required us to cut down the pre-trained model network architecture from 7 convolutional layers to 5 convolutional layers.

The training data-set had 9232 images while the test data-set had 2308 images. There were twenty classes in the data-set, the true positive rates, the true negative rates and the precision values for each class is mentioned in the table I. All values are in percentage terms.

Class	TP Rate	TN Rate	Precision
aeroplane	60	98.67	66.13
bicycle	4.4	99.89	68.75
bird	12.8	99.74	75.51
boat	28.98	99.48	67.11
bottle	0.83	99.6	9.52
bus	20.77	98.51	34.86
car	62.19	94.64	68.28
cat	70.48	93.25	42.86
chair	37.06	95.05	48.1
cow	9.45	99.73	48
dining table	0.81	13.33	0.89
dog	7.39	99.07	43.24
horse	0.36	99.88	14.28
motorbike	1.72	13.79	0.89
person	81.45	76.36	71.67
potted plant	1.97	99.83	38.46
sheep	32.65	99.24	46.38
sofa	4.51	99.59	45.71
train	20.46	99.08	55.21
tv monitor	14.51	48.1	0.89

TABLE I: Results of object recognition

From the accuracies tableI, we observe that the scores are not at par with the paper [1] on which this method was based on. This may be because of the removal of two layers from the pre-trained architecture. Nonetheless, we observe satisfactory accuracies throughout the classes. We also observe that the person class is being predicted with considerably good rates than any other classes. The prediction capability of a class depends on the number of training images available for that class. A variation of class prediction accuracies is shown in Fig.2 which can reflect the ratio of the images per class present in the data-set.

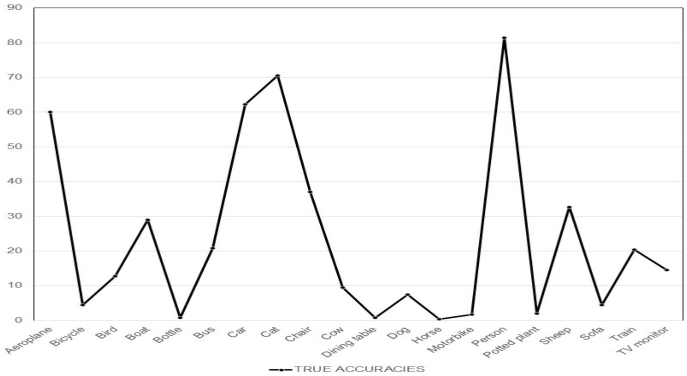

Fig. 2: Variation of class prediction accuracies

A variation of the described architecture has also been implemented with only three convolutional layers in the pre-trained architecture and four adaptation layers. With this form of architecture we observed that the accuracies reduced. This tells us that higher the number of layers in the pre-trained architecture better is the object recognition capability.

In the Fig.3, the activations of the final layer of the CNN architecture has been plotted for an image. It is observed from the figure that the three persons present on the original image on the left correspond to three blobs on the final layer activation image on the right. Thus, approximate locations of the objects can be inferred by looking at the activations of the layers of CNN. The final layer activations can be traced back through the layers and the approximate position of the object in the image can be inferred. However, in this process the information about the object count is lost. In the following section we investigate the applicability of this model to count objects.

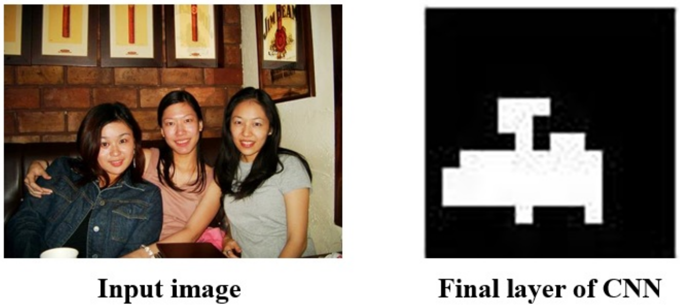

Fig. 3: Visualization of the activations of the final layer

III. Supervised Object Counting

A. Motivation

In the previous method, we were able to see how object recognition can be carried out with just the information about the list of objects present in the image. Further, paper [1] tried to predict the approximate locations of the object using the weak labels. In an attempt to extract as much information possible from an image using only the weak lables, we seeked to predict the count of the object. Before, carrying out the weak supervised procedure, we first investigated the suitability of the model for the fully supervised variant.

B. Method Description

A labelled data was prepared for the PascalVOC training and testing sets. For every image present in the data-set, another image was prepared such that the new image contained a gaussian at the position of the object. Let us call this new image as target-image. The number of gaussians present in the target-image corresponded with the number of objects present in the image. The object number could be found by integrating the target-image. A pictorial representation of this technique is shown in Fig.4.

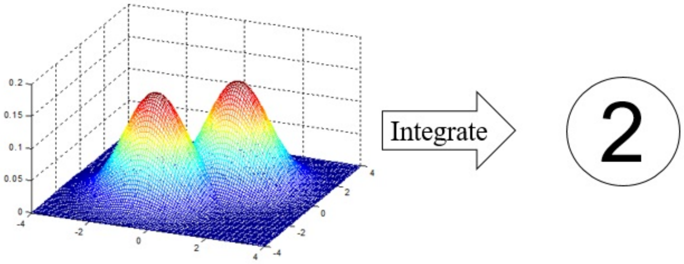

Fig. 4: Idea behind object count technique

The target-image which is a two-dimensional matrix was transformed into one-dimensional vector. Let us call it target-vector. This was done so as to incorporate target-vector as a layer in the CNN architecture. Input images are fed into the model described in section II or Fig.1 and upon training, a vector output is expected whose integral would give the count of the object. Currently, the object count method was carried out irrespective of the class of the object. A flow-chart describing the procedure of object counting can be found in Fig.5.

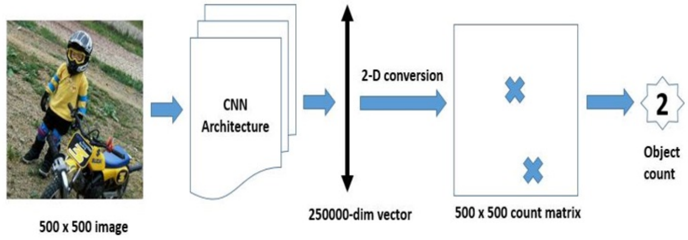

Fig. 5: Object Counting flow-chart

C. Experiments

The system descriptions are the same as that used in section II. All the images were padded with zeros to bring them to a dimension of 500×500 to suite the CNN model. The target-images, which were formed for every data-set image, were of the same dimension which is 500×500. This resulted in the dimension of the target-vector to be of dimension 250000×1. Such a large dimensional vector could not fit into the CNN architecture because of memory constraints. Two alternative measurements have been taken to address this problem

1) Inherent Scaling between the inputs and the targets: The initial target-image which was of 500×500 dimension was scaled down to 100×150 resulting in the target-vector to be of dimension 15000×1. Thus the model has to now account also for the scale difference between the inputs and the targets. Several other scaling down factors were also considered.

2) Inputs and Targets of similar dimension: In this case, the input and the outputs were brought to same dimension by scaling down the images in the target data-set. The training and testing images which were initially around 500×500 dimension were scaled down to 100×150 dimension. Suitable target dataset was prepared. Further, as it was observed that the CNN model used a sliding window of 224×224 dimension, the inputs and the targets were also brought to the same dimension, however, memory constraints prevented us from executing this case.

When the data-sets were trained using the above variations random patterns were obtained on the desired target data-sets. These patterns were varying slightly for every image. The integral of the resulting target-images resulted in some random number which did not correspond with the number of objects actually present in the image. A possible transformation to these patters can be done for them to depict the object counts. Fig.6 shows a sample desire target-image and observed target-image.

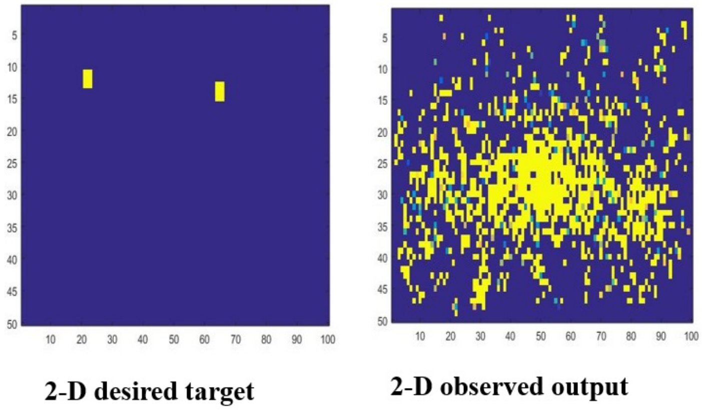

Fig. 6: Object Counting results

IV. Future work

The framework used in this project for the application of object count can be further extended and improvised through various means. Firstly, we can make simple modifications like increasing the training size, more epochs in training, and bigger size images for improving the accuracies. Secondly, one crucial change we can introduce is the choice of new cost function. In our work we have used L_2 norm as the cost in training. Maximum Excess over SubArrays (MESA) based cost function as described in paper [8] can be used instead. This cost function seems to give better performance as per [8]. If the target-images are formed for the images in the dataset as per section III-B, then the MESA distance between two target images may be defined as the largest absolute difference between the sums over all box sub-arrays. This form of distance seems to incorporate the positional information in its cost function which was not present in the L_2 norm cost used in our project.

V. Conclusions

Through experimenting with the CNN architecture by I) Varying the dataset used for training and testing (PASCAL VOC 2007 and 2012), II) Doubling the number of adaption layers and reducing the number of pre-trained layers, we have established the fact that weak labels (mere information of list of objects present in an image), can be used for object recognition. The location of the objects can further be extracted from activations of the layers of the CNN. Using the model for object counting, we found out that the given model may be used for object counting however proper interpretation of the results produced by the model are needed. The observations made from object count experiments leads us to believe that by using better cost function for training (for instance, MESA function), can help achieve good results on this task of object counting.

Acknowledgment

First of all, we would like to thank the authors of the the paper [1] for their wonderful work which motivated us to explore object recognition in much more detail. We would like to thank the tutors of CS676A course for their valuable inputs throughout the project. Finally, we would like to thank Prof. Vinay Namboodari for his constant guidance and motivation throughout the project.

References

[1] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Is object localization for free? – weakly-supervised learning with convolutional neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[2] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (voc) challenge,” *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[3] S. J. Pan and Q. Yang, “A survey on transfer learning,” *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, no. 10, pp. 1345–1359, 2010.

[4] 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)*, vol. 115, no. 3, pp. 211–252, 2015.

[5] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Learning and transferring mid-level image representations using convolutional neural networks,” in *CVPR*, 2014.

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results,” http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[7] ——, “The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results,” http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[8] V. Lempitsky and A. Zisserman, “Learning to count objects in images,” in *Advances in Neural Information Processing Systems*, 2010, pp. 1324–1332.