ISSN Print: 2394-7500
ISSN Online: 2394-5869
Impact Factor: 5.2
IJAR 2020; 6(9): 156-159
www.allresearchjournal.com
Received: 16-07-2020
Accepted: 22-08-2020

Utkarsh Pundir
College of Technology, SVPUA&T, Meerut, Uttar Pradesh, India

Corresponding Author:
Utkarsh Pundir
College of Technology, SVPUA&T, Meerut, Uttar Pradesh, India

An approach for object detection: Based on image AI

Utkarsh Pundir

Abstract
Object Detection is a domain that has witnessed great success in the field of computer visions. It deals with images to identify and locate objects. In the real world this is of great use and is used to identify objects of class humans, trees, cars and so on. The initial tasks of object detection are started by the category detection which further leads to category recognition. The state-of-art techniques are used to enhance the accuracy with real time performance. This paper presents a review of techniques for object detection that is self-contained with deep learning and computer vision capabilities. By the use of certain models and python libraries for object detection this technique overcomes certain drawbacks of deep learning. The data sets used for training the model are most challenging and publicly available. After the study of certain relevant literature about object detection all the necessary comments are drawn.

Keywords: Object detection; Deep learning; Computer vision; Image AI; Category recognition

Introduction
Real time object detection is a need of the world as their are so many advancements in the technology. It is a computer vision problem dealing with images and videos for the detection as well as identification of instances of semantic objects of certain class (humans, buildings, cars) and further deals with image processing. This has been possible because of the contribution from various technologies like deep learning, python, computer vision which have made object detection algorithms to be easily programmed in a machine.
Object detection problem can be termed as the labelling problem which is based on several models called models. In this scenario an image which is containing one or more objects of interest and a set of labels that corresponds to the set of models is known to the system and according to this system is expected to assign desired labels to various objects in image. For the proper implementation of the object detection problem image should be segmented and must be without partial segmentation. In the above given context the term detection is used to refer a combination of visual abilities such as identification, categorization and discrimination. Object detection was studied before the CNN's (Convolutional Neural Networks) in computer vision. These were automated and were capable of extracting more complex and better features by analyzing the conventional methods. The task of object detection before the emergence of deep learning took a series of steps which were started from edge detection followed by feature extraction using techniques like SIFT, HOG. Further these were compared with existing object templates at multi scale levels after which detection and localization of objects were done in image.
In this paper we have mainly discussed about the basic framework of the object detection, how the concepts of deep learning and computer vision were being used for object detection. All these contexts will be followed by the discussion on the main topic of the paper that is Image AI which is a great advancement in the field of object detection that is empowering developers to easily integrate the state-of-the-art.

Object detection
Object detection is the procedure that determines the instances of class to which the object belongs and also estimates the location of the objects by outputting bounding box around object. Image localization refers to the detection of image from the image having only a single object in the image. Partial or full occlusion, varying illuminations conditions, poses,

scales are some of the important points that need to be handled while performing object detection.

While detecting the objects in an image firstly category detection is done that distinguishes objects from the background. Then comes into picture is category recognition that classifies object into one of predefined categories and thus helps in identifying specific objects in digital images or videos. Object detection is not a single step process but it has some basic procedures that helps in the detection of the image. Preprocessing improves the intensity of the images by suppressing the unwanted features or enhancing them for further processing. This can be simply done by subtracting the mean of intensities and dividing them by standard deviation. It is followed by feature extraction that is helpful in simplifying images by properly functioning edge detection. This further helps to reduced the part of feature extraction. Object classification or object categorization are designed to achieve their goal to evaluate the presence of objects from a given set of object classes and then assigning them accordingly without the need of location. Semantic image segmentation is the topic of interest that basically aims at assigning each pixel in an image to a semantic class label. On the other hand object instance segmentation aims to distinguish different instances of the same object class completely opposite to the semantic segmentation which does not.

While this approach of object detection is a good way of detecting objects from various instances but at the end it's not that much ideal and comes with some of the challenges. When there is change in size of the image or the image is cropped out from the background then this might lead to the change in accuracy of the detection of objects. In certain conditions when there is change in the brightness or contrast then it also makes it difficult. Some problem arises when there are similar type of objects in the image and in these cases the system fails irrespective of the other object instances being present in the system. These also depend on the factors like weather conditions of image, illuminations, occlusion, all these factors are termed as imaging conditions and these significantly affect the detection of the objects. Some of the challenges that can be also added are poor resolution and filtering distortions. Not only this but the execution of these types of codes the implementation was too much hard because of the length of the codes but this would be resolved by the discussion on some of the technology that are mentioned in the paper below.

Deep Learning: Existing Models

Convolutional Neural Network

Convolutional neural network also called CNN's are somewhat different than other networks because it is a connective branch of machine learning that does the work of image recognition, object detection. This network is focused on processing image data and also extensions of other signals like 2-D grid pixels. It does the work of image processing frequently with the help of various layers. Deep learning CNN models have been used for training and testing in which each input image passes through a series of filtering convolutional layers.

CNN's were used for performing the pooling for the deduction of the dimensionality size. Moreover these CNN models were automated that is they did some of the initial methods without human intervention. These models showed further showed success in fields like image segmentation, similarity searching in images. Although having a number of applications these models found it difficult to detect small objects because of limited discriminativeness of the global visual features. Also the prediction of these models included only labels and segmentation boxes were not included in these predictions. That's why the convolutional neural networks were later on replaced by the new advancements in the technology.

Faster R-CNN

Faster R-CNN replaced the slowest part of the Fast R-CNN that is selective search or edge boxes with a very small convolutional network called RPN (Regional Proposal Network). It is a single layer model designed to run and output regions and do the classifications directly. It ranks boxes called anchors and provides the one with maximum frequency of objects. It provides user with the category wise distribution of anchor as background or foreground and later on refines anchors. Faster R-CNN are helpful in solving complex computer vision problems. But it required lots of computational power and was time consuming to provision. For such a complex network the management of the training datasets as well as the output datasets is also too much complex.

Single Shot Detector (SSD)

Single shot detector was the first one stage detector that is it processed the whole at one time rather than doing the same thing in two steps and at the same time worked in real time environment. This was the first of its kind whose accuracy was closest to the two stage detectors at that time. This achieves a good balance between speed and accuracy. Single shot detector works by running a convolutional network on input image only once and then calculating a feature map through which it calculates bounding box and classification probability. By using the anchor boxes it learns the offset rather than learning the whole box. As it predicts the bounding boxes after multiple layers therefore it is able to detect objects at various scales. Single shot detectors does not work well with small objects all the earlier layers are having smaller receptive fields that's why SSD does not work well with smaller objects as compared to larger objects. As there is need of complex data augmentation in single shot detectors that further leads to the demand of a large number of data sets to train. Single shot detectors showed some of the great advancements from the existing technology but also lacked from other technologies in various aspects which were later on fulfilled by the upcoming technologies.

Image AI

Image AI was developed by Moes Olafenwa and John Olafenwa from the deepquest–AI team. It is a python library self-contained with deep learning and computer vision capabilities which is at present used by developers and researchers for the development of applications and systems. It supports number of machine learning algorithms for image prediction, custom image prediction, image prediction trainings. This library is the combination of all the state-of-art deep learning algorithms like RetinaNet, YOLOV3 and Tiny YOLOV3. Image AI uses several offline working API's that is providing object detection, video detection and object tracking API's without the need of accessing internet for these API's. One of the great use of

using this library over other technologies is that the model is pre-trained and is easily customizable. This library also contains several pre-defined functions that are purposely defined for specific tasks like for prediction of objects from images there are some other classes and for the detection of objects there are other classes. Similarly for images there are separate classes defined and for videos and other tasks there are other classes defined. Image AI is also providing support for more wider and specialized aspects of Computer Vision in special environments and fields which is not only limited to image recognition. Image AI takes input of images in the form of files only. The newer versions are also having some of the below mentioned points:

- Progress bar for detection training.
- Custom detection model metrics retrieval.
- Prediction and detection in multi-threaded code.
- Bug fixing.
- Automatic Multi-GPU utilization for detection training.

In his paper we will be further discussing about some of the pre-defined classes of the Image AI library that have showed themselves of great use in with the help of their pre-trained training models as well as the algorithms that have been designed so far which work with the help of installed dependencies.

Image Prediction: Image AI

Image AI provides with pre-defined four algorithms and models for the prediction of images which are trained on ImageNet-1000 dataset. The algorithms are summarized in the following points:

SqueezeNet algorithm has fastest prediction time and moderate accuracy.

ResNet 50 algorithm is a bit slower than SqueezeNet but more accurate.

Inception v3 has slower prediction time but higher accuracy than above algorithms.

DenseNet 121 has slowest prediction but is the most accurate algorithm of all.

Image AI does the task of prediction with the help of pre-defined function like**.'** Predictimage () /. Predict Multiple Image (). It further provides with the facility of reducing time of prediction by 20%-60% with slight changes and accurate results.

Object Detection: Image AI

Image AI provides convenient and powerful algorithms for performing object detection and further extracting the objects from the image and individually. The pre-defined and trained models for objects detection have been listed:

Retina Net is high performing and accurate but have longer detection time.

YOLOv3 is moderate performing and accurate with moderate detection time.

TinyYOLOv3 for optimized speed and moderate performance with fast detection time.

Image AI for the purpose of object detection supports only three types of inputs which are from file path to image file, numpy array of image and image file stream. On the other hand the output is given in two formats that is image file and numpy array. It also provides with the ease of hiding or showing the detected objects name from the image pr the percentage probability from being shown in the image. Image AI also provides the ways for managing the detection speeds of the object detection tasks which bring out slight changes in the result but also improves the accuracy of the object detection and provides with more accurate details.

Proposed Model

This model used several software libraries, packages and programs to implement machine learning. Python as the programming language, and TensorFlow was used for the deep learning computations, which in turn has a list of dependencies. TensorFlow offering a version of CPU usage and another of GPU, this model uses the GPU version. These versions requires extra programs from the GPU designer NVIDIA, such as CUDA Toolkit, cuDNN and their GPU drivers. So far NVIDIA is the leading GPU designer for deep learning (also crypto mining and other similar high complex tasks) since they also write programs that are compatible with their cards that enable much of this capacity. The card used for this model is a GeForce GTX 990 TI. Taking the example of self-driving cars. Considering that we have trained an object detection model which takes a few seconds (say 2 seconds per image) to detect objects in an image and we finally deployed this model in a self-driving car.

Do you think this model will be good? Will the car be able to detect objects in front of it and take action accordingly? Certainly not! The inference time here is too much. The car will take a lot of time to make decisions which might lead to serious situations like accidents as well. Hence, in such scenarios, we need a model that will give us real-time results. The model should be able to detect objects and make inferences within microseconds.

To explain the methods used to complete this it will be divided in to smaller sections. First and foremost, the previously mentioned programs and packages had to be installed, then a model had to be procured and trained. To do this training a dataset in the form of pictures was gathered and subsequently translated to the correct format. When the model was completed its inference graph could be applied to new data to detect and identify said data. Training a model from scratch can take a very long time, especially when the hardware used is far from optimal. The training for this model was done on a GTX 980 TI, so to save on time a checkpoint from a pre-trained model was used.

This model can then be further trained on your own dataset. This will result in a good model with your dataset without taking as much time as training everything yourself. The model chosen for this project is optimised for speed and not accuracy, this was done because of an accurate model will be larger with more data points which means that when new data is used for identifying objects each loop will take longer. For the real-time portion of the code to work speed is very important, since it will have to iterate several times a second.

Conclusion

On the complete execution of the code for the detection of image from the given input image using the Image AI library.

Fig 1: Image with no object identification.

We get the output depicting the objects as well as the details of those objects at the same time through a fast and more accurate way for object detection.

Fig 2: Image with marking of different objects

This model has made the detection of objects at real time much more easier and with improved accuracy which allows the faster recognition as well as the implementation of this model is much more optimized.

References

1. Felzenszwalb PF, Girshick RB, Mcallester D, Ramanan D. “Object detection with discriminatively trained part-based models,” IEEE Trans. Pattern Anal. Mach. Intell. 2010; 32(9):1627.
2. KK Sung, T Poggio, Example-based learning for view-based human face detection, IEEE Trans. Pattern Anal. Mach. Intell. 2002; 20(1):39-51.
3. Wojek C, Dollar P, Schiele B, Perona P. Pedestrian detection: An evaluation of the state of the art,” IEEE Trans. Pattern Anal. Mach. Intell. 2012; 34(4):743.
4. Kobatake H, Yoshinaga Y. Detection of spicules on mammogram based on skeleton analysis.” IEEE Trans. Med. Imag. 1996; 15(3):235-245.
5. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R *et al*. Caffe: Convolutional architecture for fast feature embedding,” in ACM MM, 2014.
6. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks,” in NIPS, 2012.