A SURVEY ON OBJECT DETECTION TECHNIQUES USING TENSORFLOW, KERAS AND YOLO

Mahshiya VM, Manju VM, Dr G.Kiruthiga

[1] *Student, Department of Computer Science, IES College of Engineering, Kerala, India*
[2] *Assistant Professor, Department of Computer Science, IES College of Engineering, Kerala, India*
[3] *Associate Professor, Department of Computer Science, IES College of Engineering, Kerala, India*

ABSTRACT

This paper aims to discuss about some project works that implements CNN with the help of TensorFlow, Keras and YOLO object detection and improvements in object detection. TensorFlow with Keras is a very famous machine learning system and used by various researchers in data classification areas. In TensorFlow computation is represented in the form of dataflow graphs. With the help of TensorFlow, Google's library a complex computation can be performed very easily by representing them in the form of graph and efficiently mapping the graph parts to machine in the form of cluster. You Only Live Once (YOLO) is an effective real time object detection system. It considers object detection as a regression problem and finds the class probabilities for each of the bounding boxes.

Technology has advanced tremendously over the past century, everything starting from the Internet of Things (IoT) to machine learning and deep learning. CNN is used in various fields like medical, marine science and many other applications and has become a prominent domain of machine learning. Implementation of TensorFlow, Keras and YOLO object detection methods gives more accuracy, robustness and faster detection

Keyword : - *Machine Learning, TensorFlow, Keras, YOLO , CNN, OpenCV*

1. INTRODUCTION

Humans glance at an image and instantly know what objects are in the image, where they are, and how they interact. The human visual system is fast and accurate, allowing us to perform complex tasks like driving with little conscious thought. Fast, accurate algorithms for object detection would allow computers to drive cars without specialized sensors, enable assistive devices to convey real-time scene information to human users, and unlock the potential for general purpose, responsive robotic systems. Current detection systems repurpose classifiers to perform detection. To detect an object, these systems take a classifier for that object and evaluate it at various locations and

scales in a test image. Systems like deformable parts models (DPM) use a sliding window approach where the classifier is run at evenly spaced locations over the entire image

More recent approaches like R-CNN use region proposal methods to first generate potential bounding boxes in an image and then run a classifier on these proposed boxes. After classification, post-processing is used to refine the bounding boxes, eliminate duplicate detections, and rescore the boxes based on other objects in the scene . These complex pipelines are slow and hard to optimize AQ because each individual component must be trained separately. We reframe object detection as a single regression problem, straight from image pixels to bounding box coordinates and class probabilities. Using our system, you only look once (YOLO) at an image to predict what objects are present and where they are.

TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks TensorFlow provides a collection of workflows to develop and train models using Python or JavaScript, and to easily deploy in the cloud, on-prem, in the browser, or on-device no matter what language you use. The tf. data API enables you to build complex input pipelines from simple, reusable pieces.

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library. Up until version 2.3, Keras supported multiple backends, including TensorFlow, Microsoft Cognitive Toolkit, Theano, and PlaidMLKeras is used for creating deep models which can be productized on smartphones. Keras is also used for distributed training of deep learning models. Keras is used by companies such as Netflix, Yelp, Uber, etc

2. TENSORFLOW AND KERAS

nsorFlow is an open-source platform that is used for Machine Learning, created by the Google Brain team. It is explicitly used for complex numerical computation,that packs together a bunch of machine learning and deep learning models and algorithms. It can be used for a variety of applications such as classifying handwritten digits, object detection, image recognition, natural language processing (Natraj, 2019) by training and running deep neural networks. Application Development for Mask Detection and Social Distancing Violation Detection using Convolutional Neural Networks Keras which acts as an interface for TensorFlow is an open-source library that provides an efficient way of implementing neural networks. It consists of useful functions such as activation functions, and optimizers.

2.1 How Does TensorFlow Work?

With the help of TensorFlow, developers can create dataflow graphs which are structures that show how data passes through the graph, or a series of nodes. Think of each node as a mathematical operation and each edge representing a multidimensional data array or a tensor. This can be easily implemented in python where these nodes and tensors act as objects. However, the mathematical operations are performed in C++ binaries which shows an optimal performance. Python takes care of directing the traffic and combines them to work together as a unit. TensorFlow can be run on multiple platforms such as in a cloud, a local machine, CPUs or GPUs, iOS, and Android devices. It can also be run on Google's custom TensorFlow Processing Unit (TPUs). The trained models can be run on any system for predicting results. TensorFlow 2.0 which was released in October 2019 made many significant changes from user feedback. It works more efficiently and is more convenient with simple Keras API for training models and better performance. With the help of TensorFlow Lite,it is possible to train models on a wide variety of devices.

3. YOLO OBJECT DETECTION

You Only Look Once (YOLO) is an effective real time object detection system. It considers object detection as a regression problem and finds the class probabilities for each of the bounding boxes. In one evaluation the neural

network predicts bounding boxes and class probabilities from the image, hence the name YOLO. The base model detects images at an astonishing speed of 45 frames per second whereas a smaller version called Fast YOLO detects at 155 frames per second. It performs better than other detection methods such as Deformable Part Models (DPM) and Region-based convolutional neural networks (R-CNN). There are two types of algorithms that work towards object detection-Algorithms based on classification, and Algorithms based on regression. YOLO falls under the latter category.

3.1 How Does YOLO Work?

A single neural network is used to combine separate parts of object detection. It takes in features from an image to predict each bounding box. The detection is modeled as a regression problem where an image is divided into SxS grids. In Fig. 7, the authors set the value of S as 7. This can be changed in the YOLO configuration file. Each grid predicts bounding boxes (B), their confidence scores, and their class probabilities (C). These predictions are encoded as an S x S x (B*5 + C) tensor.

Each bounding box has 5 predictions: (x, y, w, h) and confidence. Each grid cell predicts conditional class probabilities (C) as P (Classi | Object).

The architecture for this Convolutional Neural Network was inspired from GoogLeNet model that is used for image classification. Just like GoogLeNet their model was implemented using 24 convolutional layers that helps to extract features from the image followed by 2 fully connected layers to predict the output probabilities and coordinates. They also came up with a faster version on YOLO named Fast YOLO that uses a CNN with less layers (9 instead of 24) and less filters for those layers. Other than that, the training and testing parameters were the same between YOLO and Fast YOLO.

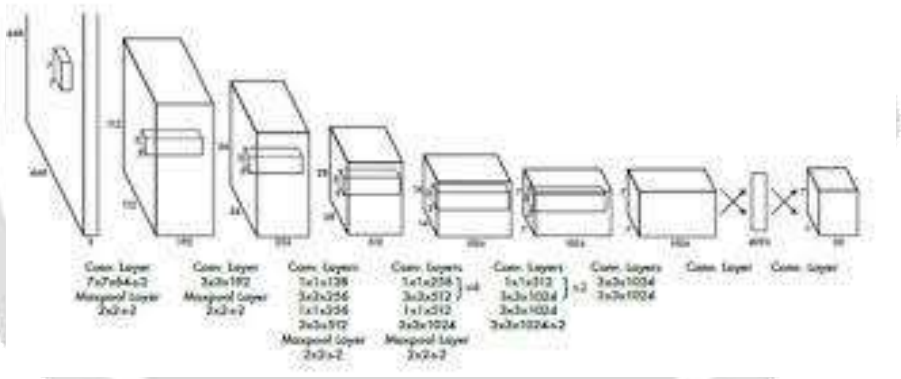

Fig-1: The YOLO architecture

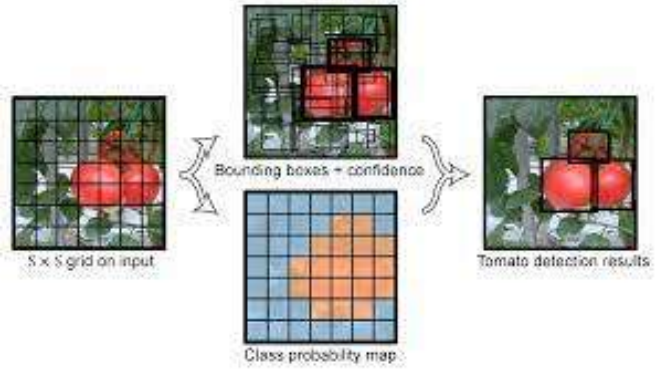

Fig-2: The YOLO Model

4. LITERATURE REVIEW

Akanksha Soni et al. (Soni, 2020) developed a model that detects whether a person is wearing a helmet in real time thereby, detecting any violations. This project was also implemented with the help of TensorFlow, Keras and OpenCV. Their proposed model showed major improvements when compared to some previous models that gave wrong predictions whenever a rider wears clothes over their face. They achieved an overall accuracy of 98% when tested.

S Chen et al. (Chen, 2020) implemented a model with the help of TensorFlow to identify ID card numbers. With the help of OpenCV the image of an ID card is preprocessed and the number on the ID card is recognized and given as output with the help of a trained CNN model. When tested it was observed that training speed is fast and the accuracy is high.

Emily Caveness et al. (Caveness, 2020) developed TensorFlow Data Validation (TFDV) which offers a scalable solution for data analysis and validation for machine learning. It is deployed in production which is integrated with TensorFlow Extended (TFX), which is an end-to-end ML platform. Their system has gained a lot of traction ever since they open sourced their project. Other open-source data validation systems such as Apache Spark were also heavily inspired from their project. Apache Spark packs with built-in modules for streaming and has a fast, easy to use system for big data processing.(Nair, 2018)

Yonghui Lu et al. (Lu, 2020) proposed an efficient YOLO Architecture, YOLO-compact for a real time single category detection. As we know in most practical applications, the number of categories in object detection is always single and the authors aimed to make detections faster and more efficient for these scenarios. By performing a series of experiments, the authors were able to come up with an efficient and compact network with the help of YOLOv3. It was observed that YOLO-compact is only of 9MB size, about 26 times smaller than YOLOv3, 6.7 times smaller than tiny-yolov2 and 3.7 times smaller than tiny-yolov3. The average precision of YOLO-compact is 86.85% which is significantly higher than other YOLO models.

M. B. Ullah (Ullah, 2020), proposed a CPU-based YOLO object detection model that is intended to run on non-GPU computers. In the proposed method, the author optimized YOLO with OpenCV in a way that real time object detection can be much faster on CPU based computers. Their network architecture comprises 2 Convolutional layers each followed by pooling layers and 3 fully connected layers. Their model detects objects from videos in 10.12 to 16.29 FPS with 80-99% confidence in CPU-based computers.

J.Redmon et al.(Redmon,2016),introduced YOLO, a unified model for object detection. This model is simple to construct and can be trained directly on full images. Unlike classifier-based approaches, YOLO is trained on a loss function that directly corresponds to detection performance and the entire model is trained jointly. Fast YOLO is the fastest general-purpose object detector in the literature and YOLO pushes the state-of-the-art in real-time object detection. YOLO also generalizes well to new domains making it ideal for applications that rely on fast, robust object detection.

Daniele Gratterola et al.(Daniele,2021), presented Spektral, an open-source Python library for building graph neural networks with TensorFlow and the Keras application programming interface. Spektral implements a large set of methods for deep learning on graphs, including message-passing and pooling operators, as well as utilities for processing graphs and loading popular benchmark datasets. The purpose of this library is to provide the essential building blocks for creating graph neural networks, focusing on the guiding principles of user-friendliness and quick prototyping on which Keras is based. Spektral is, therefore, suitable for absolute beginners and expert deep learning practitioners alike. This work presented an overview of Spektral's features and report the performance of the methods implemented by the library in scenarios of node classification, graph classification, and graph regression. In

the future, we will keep Spektral up to date with the ever growing field of GNN research, and we will focus on improving the performance of its core components.

Neeraj Chauhan et al.(Neeraj,2018),discussed about the TensorFlow a Python framework for testing data classification accuracy of ANN (Artificial Neural Network). TensorFlow with Keras is a very famous machine learning system and used by various researchers in data classification areas. In TensorFlow computation is represented in the form of dataflow graphs. With the help of TensorFlow, Google's library a complex computation can be performed very easily by representing them in the form of graph and efficiently mapping the graph parts to machine in the form of cluster.

Sung.M.et al. (sung 2017). proposed convolutional neural network based techniques based on You Only Look Once algorithm. Actual fish video images were used to evaluate the reliability and accuracy of the proposed method. As a result, the network recorded 93% classification accuracy, 0.634 intersection over union between predicted bounding box and ground truth, and 16.7 frames per second of fish detection. It also outperforms another fish detector using sliding window algorithm and classifier trained with histogram of oriented gradient features and support vector machine.

Daniel Kold et al. described how to adapt the state-of-the art deep learning-based detector of You Only Look Once (YOLO) for the purpose of detecting barcodes in a fast and reliable way. The detector is capable of detecting both 1D and QR barcodes. The detector achieves state-of-the-art results on the benchmark dataset of Muenster BarcodeDB with a detection rate of 0.991. The developed system can also find the rotation of both the 1D and QR barcodes, which gives the opportunity of rotating the detection accordingly which is shown to benefit the decoding process in a positive way. Both the detection and the rotation piction shows real-time performance.

5. CONCLUSIONS

TensorFlow provides a collection of workflows to develop and train models using Python or JavaScript, and to easily deploy in the cloud, on-prem, in the browser, or on-device no matter what language you use. The tf. data API enables you to build complex input pipelines from simple, reusable pieces

Keras is a powerful and easy-to-use free open source Python library for developing and evaluating deep learning models. It is part of the TensorFlow library and allows you to define and train neural network models in just a few lines of code.Keras is used for creating deep models which can be productized on smartphones. Keras is also used for distributed training of deep learning models. Keras is used by companies such as Netflix, Yelp, Uber, etc

YOLO is an algorithm that uses neural networks to provide real-time object detection. This algorithm is popular because of its speed and accuracy. It has been used in various applications to detect traffic signals, people, parking meters, and animals.The biggest advantage of using YOLO is its superb speed – it's incredibly fast and can process 45 frames per second. YOLO also understands generalized object representation. This is one of the best algorithms for object detection and has shown a comparatively similar performance to the R-CNN algorithms

This methodologies can be used to improve real time object detection. So that we can make lot of changes in computer vision approaches of any field and can make the world run fast.

6. REFERENCES

[1]Soni, A., & Singh, A. P., 2020. Automatic Motorcyclist Helmet Rule Violation Detection using Tensorflow & Keras in OpenCV. In 2020 IEEE International Students Conference on Electrical, Electronics and Computer Science (SCEECS).

[2]Chen, S., Wei, Y., Xu, Z., Sun, P., & Wen, C., 2020. Design and Implementation of Second-generation ID Card Number Identification Model based on TensorFlow. In IEEE International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA).

[3]Caveness, E., C., P. S., Peng, Z., Polyzotis, N., Roy, S., & Zinkevich, M., 2020. TensorFlow Data Validation: Data Analysis and Validation in Continuous ML Pipelines. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data.

[4]Lu, Y., Zhang, L., & Xie, W., 2020. YOLO-compact: An Efficient YOLO Network for Single Category Realtime Object Detection. In 2020 Chinese Control and Decision Conference (CCDC).

[5]Ullah, M. B., 2020. CPU Based YOLO: A Real Time Object Detection Algorithm. In 2020 IEEE Region 10 Symposium

[6]Redmon, J., Divvala, S., Girshick, R., & Farhadi, A., 2016. You Only Look Once: Unified, Real-Time Object Detection. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

[7]Grattarola G., & Alippi C., 2021. Graph Neural Networks in TensorFlow and Keras with Spektral[Application Notes]. IEEE Computational Intelligence Magazine

[8]Chauhan N., Bhatt A.K., Dwivedi R.K., & Belwal R., 2018. Accuracy Testing of Data Classification using TensorFlow a Python Framework in ANN Designing. In IEEE International Conference on System Modeling & Advancement in Research Trends (SMART)

[9]Sung, M., Yu, S., & Girdhar, Y., 2017. Vision based realtime fish detection using convolutional neural network. In OCEANS 2017 - Aberdeen. Hansen, D. K., Nasrollahi, K., Rasmusen, C. B., & Moeslund, T. B. (2017).

[10]Real-Time Barcode Detection and Classification using Deep Learning. In Proceedings of the 9th International Joint Conference on Computational Intelligence