REAL TIME LIVE OBJECT DETECTION WITH TENSOR FLOW

B.S.MURTHY,G.UPENDRA
PG student, D.N.R. COLLEGE, P.G. COURSES (AUTONOMOUS), Bhimavaram, Andharapradesh.
Email id:guripalliupendra@gmail.com
PG Student of MCA, Dantuluri Narayana Raju College, Bhimavaram, Andharapradesh
Email id:suryanarayanamurthy.b@gmail.com

ABSTRACT

This project 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

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 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 Plaid ML 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

1.1 TENSOR FLOW AND KERAS

TensorFlow 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.

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). Th e 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.

2. LITERATURE SURVEY AND RELATED WORK:

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 number. 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.

3 EXISTING SYSTEM

Current detection systems repurpose classifiers to per- form 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 screen.

DISADVENTAGE

These complex pipelines are slow and hard to optimize because each individual component must be trained separately.

4 PROPOSED WORK AND ALGORITHM

Many academics employ the well-known machine learning technology TensorFlow with Keras for data classification. Dataflow graphs serve as the representation of computation in TensorFlow. A complex calculation can be carried out relatively easily by describing it as a graph and effectively transferring the graph's component components to a machine in the form of a cluster using Google's framework TensorFlow.

This real-time object identification system, called You Only Live Once (YOLO), is really good. The class probabilities are determined for each of the bounding boxes while object detection is taken into account as a regression problem. Over the past century, technology has made enormous strides, from the Internet of Things (IoT) to machine learning and deep learning. CNN is utilized in numerous Machine learning has become a well-known application area in numerous industries, including medical, marine science, and many more. Implementing object detection techniques using TensorFlow, Keras, and YOLO results in increased accuracy, robustness, and detection speed

5. METHODOLOGIES

MODULES

DATASET

This paper utilizes the dataset provided by revolution analytics for the detection the fraudulent credit card transaction from Kaggle. Dataset has 51149 legal transactions and 3312 fraudulent transactions. The dataset is divided as 60%, 20% and, 20% in the Train, Valid and Test set, respectively

DATA PREPROCESSING

For efficient implementation of the classification algorithm, data preprocessing is performed before feature selection. Under-sampling is performed to make the dataset balanced to avoid the biasing of the classification algorithm towards the majority class. Feature Selection is implemented on a balanced dataset.

FEATURE SELECTION

Feature selection methods are used to remove unnecessary, irrelevant, and redundant attributes from a dataset that do not contribute to the accuracy of a predictive model or which might reduce the accuracy of the model. In this paper seven feature selection techniques namely Select-K-best, Feature Importance, Extra tress classifier, Person's correlation, Mutual Information, Step forward selection and Recursive feature elimination are used.

FEATURE IMPORTANCE

Feature importance is a class of techniques for assigning scores to input features to a predictive model that indicates the relative importance of each feature at the time of making a prediction. It reduces the number of input features. In this paper, feature importance is implemented using an extra tree classifier from the decision tree. Extra Trees is similar to Random Forest, it builds multiple trees and splits nodes using random subsets of features, but unlike Random Forest, Extra Tree samples without replacement and nodes are split on random

6 RESULTS AND DISCUSSION

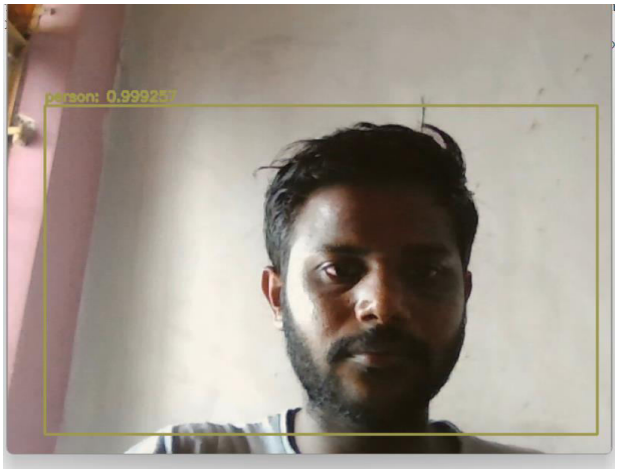

FIG1 : IMAGE DETECTION

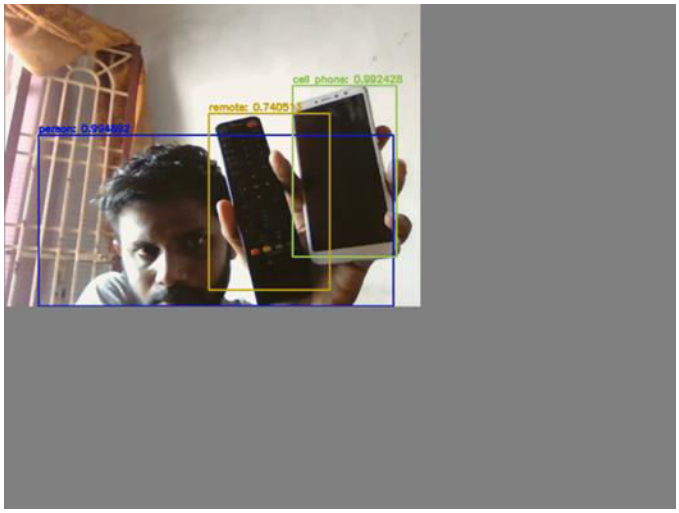

FIG 2 MULTI OBJECT DETECTION

6.CONCLUSION AND FUTURE SCOPE

CONCLUSION

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 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 These 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

FUTURE SCOPE

The computational cost and time in a neural network is higher as compared to any other network models (R-CNN, Boltzmann machines, etc.) The most crucial requirement necessary to train CNN is a GPU (graphical processing unit). If the desktop/laptop used for training does not contain a GPU, the processing required for training a model increases which affects the performance. Therefore, it is imperative that the computer we use for training must have a GPU.

The more, the model is trained the accurate it is. Hence, a huge amount of training data is required. This sometimes leads to the slow processing speed of the computer. Despite the shortcomings there is no limit to where a CNN can be used. From developing a Facial recognition software to using it in the advancement of Self Driving Cars. A voice-over interface along with the object detection model can prove to be a boon in the everyday lives of visually impaired people.

7 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