OBJECT DETECTION USING TENSOR FLOW OPENCV AND FLASK

Hemalatha T*1, K Venkata Chalapathi*2

*1,2Assistant Professor, Department of CSE, SV U College of Engineering, Tirupati, AP, India.
DOI : https://www.doi.org/10.56726/IRJMETS33262

ABSTRACT

Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings or cars) in digital images and videos.

This detects the objects of a class in digital images and videos. Object detection has applications in many areas of computer vision, image retrieval and video surveillance. With the help of python popular framework Flask we can detect an object. The applications of real time object detection include tracking objects, video surveillance, pedestrian detection, people counting, self-driving cars, face detection, ball tracking in sports.

Keywords: Object Detection, Image Processing, Digital Images, Python popular framework FLASK.

I. INTRODUCTION

Object Detection is the process of finding and recognizing real-world object instances such as car, bike, TV, flowers, and humans out of an images or videos. An object detection technique lets you understand the details of an image or a video as it allows for the recognition, localization, and detection of multiple objects within an image. It is usually utilized in applications like image retrieval, security, surveillance, and advanced driver assistance systems (ADAS).

Object detection from a video in video surveillance applications is the major task these days. Object detection technique is used to identify required objects in video sequences and to cluster pixels of these objects. The detection of an object in video sequence plays a major role in several applications specifically as video surveillance applications. Object detection in a video stream can be done by processes like pre-processing, segmentation, Humans can easily detect and identify objects present in an image.

The human visual system is fast and accurate and can perform complex tasks like identifying multiple objects with little conscious thought. With the availability of large amounts of data, faster GPUs, and better algorithms, we can now easily train computers to detect and classify multiple objects within an image with high accuracy.
II. LITERATURE SURVEY

The problem of image captioning is a complex and widely interested research topic since the evolution of deep learning. There are many proposed solutions for this problem which are replacing the previous solutions every single day.

In [1] Karpathy proposed a system which uses multimodal neural networks to generate novel descriptions of the image by providing suitable descriptions for the image.

In [2], Deng proposed a model which uses a database called ImageNet which is build using the core called WordNet. This model uses ImageNet to generate sentence descriptions from the image.

In [3] proposed an attention based model, which generates captions of the images based on the region of interest. It generates the captions based on the region the image is surrounded.

In [4], Yang proposed a multimodal recurrent neural network based model, which generates the descriptions of the image by detecting the objects and converting them to sentences, which is almost similar to human visual system.

In [5], Aneja proposed a convolutional neural network based model to generate descriptions from the image after thorough training given to the model.

In [6], Pan proposed a multiple neural network model, which is experimented with large sets of datasets to generate the accurate sentence descriptions from the image.

In [7], Vinyals proposed a model that uses Natural Language Processing and Computer Vision to detect the objects in the image and generate captions based on word processing and keyword retrieval techniques.

III. METHODOLOGY

DATASET

In performing the task of image captioning, we have used flickr8k dataset from flickr.com website. It consists of a total of 8092 images. Those 8092 images were splitted into 6000 training images and 1000 each for development and testing purpose. It consists of daily life images with features covering many objects. The images were of high clarity with good resolution and were easily recognizable for the model to get trained.
an open source dataset, which is available freely on the internet.

PHASES OF MODEL:

Creating a pre-trained model using Transfer Learning:
Transfer Learning is used to create and use pre-trained models in solving complex machine learning problems. It is the way of preserving knowledge we gained by solving a problem and use it later to solve another complex problem. In this model, we are using a pre-trained model with Transfer learning to make our model learn things from the existing knowledge.

Object Detection:
Objects in the images were detected in this phase. It uses Convolutional Neural Network (CNN) to extract the features from

Probabilistic NLP model:
The objects detected in the image were sent through this NLP(Natural Language Processing) based probabilistic model, which removes the unnecessary features from the image. It processes only the features which are relevant and meaningful with context of the image and ignore the odd ones which are irrelevant. It also removes the stop words which are repeated and have the same meaning.the image. We will use pre-trained model like Inception V4 or VGG 16 which is a Convolutional Neural Network for the task of object detection. Each and every object in the image is detected in this phase.

Figure 3: Object Detection

Deployment to Web Server:
The final phase of the project is to deploy the caption generation model in the form of web application. We are using Flask Rest API, which is a web framework of Python to deploy the working model. Flask is one of the popular Python Web development framework to develop and deploy models into web applications.
We also use html, css and bootstrap to design the interface of the web application.

LIBRARIES, TENSOR FLOW

- Tensor flow is an opensource software library for high performance numerical computation.
- To construct, train and deploy ObjectDetection Models TensorFlow is used
- TensorFlow framework allows you to create highly flexible CNN architectures for computer vision tasks.
- It provides a collection of Detection Models pre-trained on the COCO dataset and the Open Images dataset.

OPEN CV

- OpenCV is a cross-platform library using which we can develop real-time computer vision applications.
IV. IMPLEMENTATION AND RESULTS

Python is used to implement this model. It uses Scipy environment. Scipy is a machine learning library of Python. It uses Keras environment. Keras is a deep learning library written in Python. This model uses Tensorflow as a backend, which is popular Deep Learning framework of Python. Convolutional Neural Network and Long Short Term Memory are two neural networks used to implement this model. Transfer Learning is used to create and use pre-trained models to deal with complex object detection tasks. Flickr8k dataset is trained with the model, which generates sentence based captions using CNN to detect the objects and LSTM to generate the captions from the input image. The interface is used to deploy our model with the help of Flask Rest API of python.

![Figure 4: Uploading input into the model.](image)

Object detection has many advantages in almost every complex area of Artificial Intelligence. The main use case of our model is to help visually impaired to understand the environment and made them easy to act according to the environment. As, this is a complex task to do, with the help of pre-trained models and powerful deep learning frameworks like Tensorflow and Keras, we made it possible. This is completely a Deep Learning project, which makes use of multiple Neural Networks like Convolutional Neural Network and Long Short Term Memory to detect objects and captioning the images. To deploy our model as a web application, we have used Flask, which is a powerful Python's web framework. We are going to extend our work in the next higher level by enhancing our model to generate captions even for the live video frame. Our present model generates captions only for the image, which itself a complex task and captioning live video frames is much complex to create. This is completely GPU based and captioning live video frames cannot be possible with the general...
CPUs. Video captioning is a popular research area in which it is going to change the lifestyle of the people with use cases being widely usable in almost every domain. It automates the major tasks like video surveillance and other security tasks.

VI. REFERENCES


