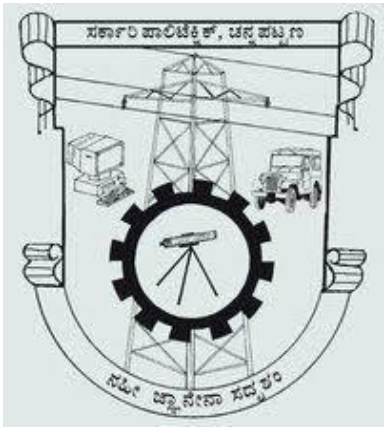
GOVERNMENT POLYTECHNIC CHANNAPATNA-562160



INTERNSHIP REPORT ON CIE-3

NAME:- YASHWANTH. K V

REG NO:- 111CS22062

PROGRAM:- COMPUTER SCIENCE & ENGG

SEM:- 6TH SEM

YEAR:- 2024-2025

TECH HEAD :- VIJAYAN .G

Submitted to -

Submitted by -

ASHWINI M.S , M Tech
Department of CSE

INTERSHIP GRADING RUBRICS (CIE-3)

NAME:

EVALUATION CRITERIAN	POOR	AVERAGE	GOOD	EXCELLENT
Intern's ability to apply the skill and technical knowledge (30 MARKS)	0-10	11-20	20-25	26-30
Intern's performance on assigned tasks and project. (10 MARKS)	0-3	4-6	7-8	9-10
Extent of Intern's ability to add value to the organization through intership (10 MARKS)	0-3	4-6	7-8	9-10
SUB TOTAL (50 MARKS)				
USE CASE - III (30 MARKS)				
TOTAL MARKS OBTAINED				

FACULTY SUPERVISOR SIGNATURE

ACKNOWLEDGEMENT

I would like to take this opportunity to thank a lot of eminent personalities, without whose constant encouragement, This endeavor of mine would not have become a reality.

At first I would like to thank the **Govt. Polytechnic Channapatna** for having this internship as part of its curriculum, which gave me a wonderful opportunity to work on my subject **Artificial Intelligence and Machine Learning** for providing me with such excellent facilities ,without which. This internship could not have acquired the shape it has now done.

My heartfelt Gratitude to our Honourable Principal **Aruna. D, Govt Polytechnic Channapatna** for is constant support and encouragement.

I am greatly indebted to **Mrs.Umadevi Y, HOD of Computer Science and Engineering** For providing me with all facilities necessary for making this internship.

I would like to thank **Mrs Ashwini .MS, Senior Scale Lecturer ,Department of Computer Science and Engineering**.For her continuous support.

I would like to thank **Vijayan.G, Managing Director and HR of GKV Global Technology ,Mandya ,Subhash Nagar,** For providing internship to me and his continuous support advice and guidance.

I am grateful to my parents ,friends and well wishers for their contribution on a personal level

Last but not least my profound thanks to the Teaching Staff and Non Teaching Staff of the Department of Computer Science and Engineering for their help and patience.

STUDENT NAME:

YASHWANTH K V

Table of Contents

INTRODUCTION

Object detection is one of the most important fields of exploration in computer vision today. It is an augmentation of image classification the objective is to identify one or more classes of objects in a picture and with the help of bounding boxes locate their presence. Consequently, object detection carries an important role in many real-world applications like image recovery and video surveillance.

The main purpose of our analysis is to elaborate the accuracy of an object detection technique SSD and the per-trained deep learning model MobileNet and additionally feature a portion of the notable elements that make this method stand out. The trial results show that the Average

OVERVIEW OF INTERNSHIP OBJECTIVES AND RESPONSIBILITIES:

This section offers a comprehensive outline of the tasks and projects assigned to Intern **Yashwanth KV** throughout their internship duration. It meticulously details the responsibilities, objectives, and anticipated deliverables, setting a clear benchmark for evaluating **Yashwanth KV's** performance in the following sections. It underscores the fundamental purpose of internships to furnish a transformative learning journey within their chosen domain. This encompasses acquiring practical wisdom, honing novel skills, and grasping the intricate dynamics of the industry.

Project overview:

The project is a Python-based multiple object detection program utilizing MobileNet SSD through OpenCV's deep neural network module. It enables real-time object detection across various input sources, supporting video files and live camera streams with configurable detection parameters. The program can identify multiple object classes including water bottles, chairs, persons, and smartphones with a confidence threshold of 0.2. Implemented with a lightweight architecture, it demonstrates efficient object detection capabilities suitable for applications like surveillance, robotics, and augmented reality. The solution offers flexible input handling, real-time processing, and potential for future enhancements in object detection technology.

A. MobileNet-SSD

Our proposed model depends on the MobileNet SSD architecture. One reason why we chose this architecture is on the because that as shown in the paper, It gives good object detection accuracy while being quicker than different architectures for example YOLO. Especially, this is valid when attempting to detect object in real time in low computing devices as in our system. MobileNet-SSD permits to lessen the detection time by addressing the model utilizing 8-bit integers rather than 32-bit floats. The input of the model was set to an image with 300 by 300 pixels and the result of the model addressed the position of the bounding box as well as the detection confidences (from 0 to 1) for each identified object. A detection confidence threshold of 0.5 was utilized to decide if the detected object was valid.

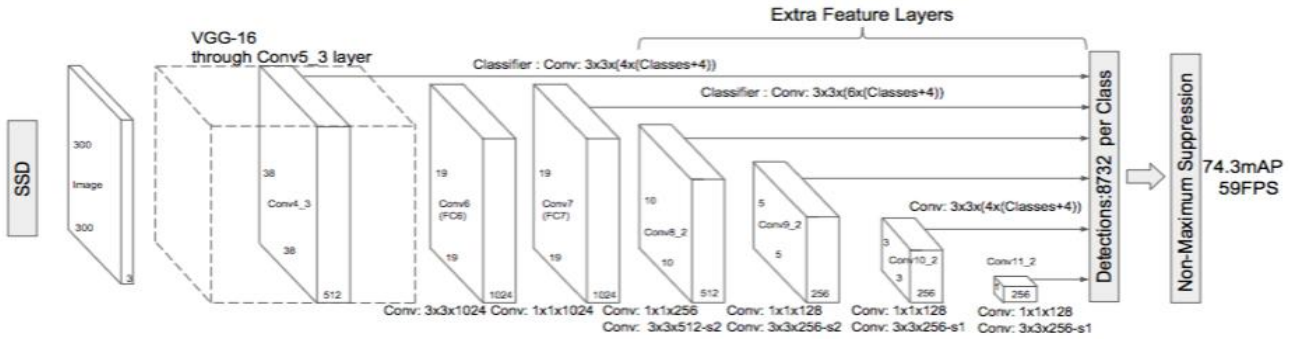


FIGURE 4.1

B. OpenCV (Open-Source Computer Vision)

Open CV is a library of programming functions basically focused on real time computer vision. OpenCV is an open-source library which is useful for computer vision applications like CCTV film analysis, video analysis and image analysis. It is an incredible tool for image processing and performing computer vision tasks. OpenCV is written by C++ and has in excess 0f 2,500 optimized algorithms. At the point when we make applications for computer vision that we do not want to make it from scratch instead we can utilize this library to begin focusing on real world problems. Open CV has a function to read video, which is cv2.VideoCapture(). We can access webcam by passing 0 as function parameter. To catch CCTV film then we can pass RTSP URL in the function parameter, which is truly valuable for video analysis.

PROPOSED SYSTEM

In the Proposed System, we are going to detect objects in real time with the help of Mobilenet-SSD model in fast and efficient way. We will create the Python script for object detection using deep neural network with OpenCV 34 Working of the system is as follow :

Input will be given through Real time video by camera, based on streamlined MobileNet Architecture which uses depth-wise separable convolutions to build light weight deep neural Networks. The input video divided into frames and pass it to MobileNet layers. Each feature value is determined as a difference between the amount of pixel intensity under the bright region and the pixel intensity under the dark area. Every one of the possible sizes and area of the image is utilized to compute these elements. An image may contain irrelevant features and few relevant characteristics that can be used to detect the object. The job of the MobileNet layers is to change over the pixels from the input image into highlights that describe the contents of the image. Then it passes to MobileNet-SSD model to determine the bounding boxes and corresponding class (label) of objects. After that the only last step is to show or display the Output.

PROPOSED SYSTEM ARCHHITECTURE DIAGRAM:

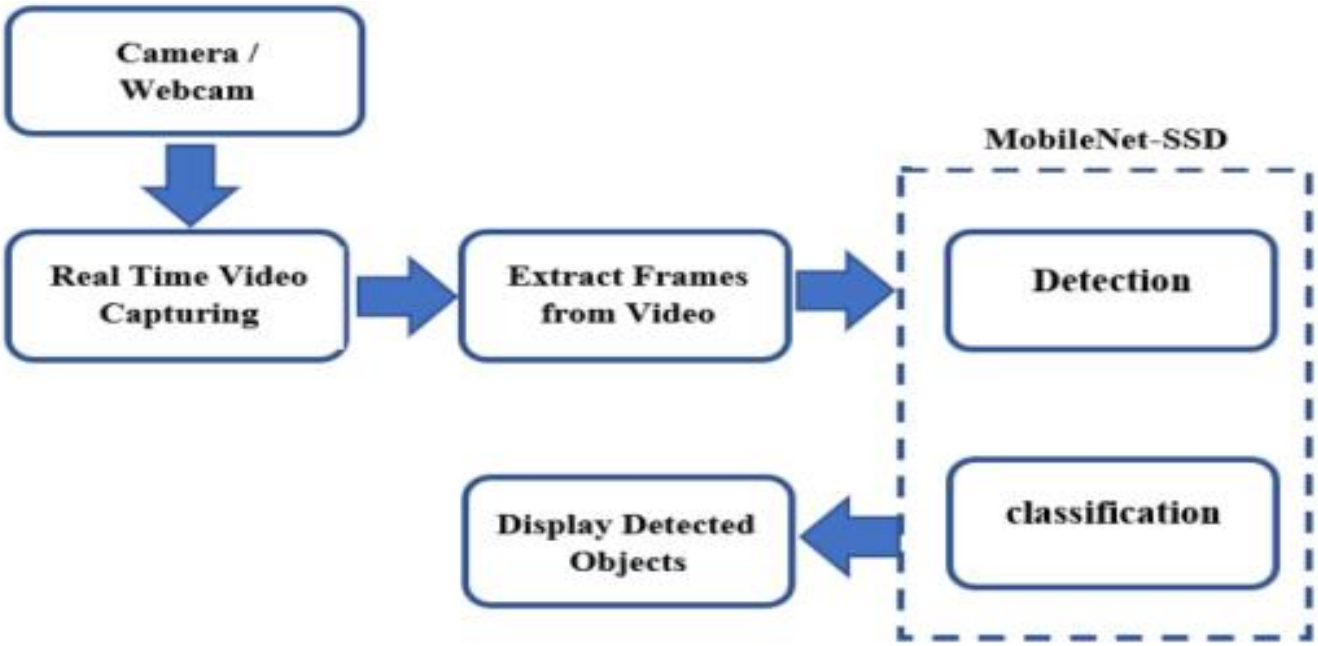


FIGURE 4.2

Operational Workflow:

1. Video Input and Preprocessing:

The system initiates by capturing realtime video input from a camera. This continuous stream of video frames serves as the primary data source for object detection.

2] Feature Extraction with MobileNet:

Each video frame undergoes preprocessing and segmentation into individual frames. These frames are fed into the MobileNet layer, which executes depth-separable convolutions to extract features. This process involves analyzing pixel intensity variations across different scales and regions within the image, distinguishing relevant objects from irrelevant elements.

3] Object Localization and Classification:

The MobileNet SSD model utilizes the extracted features to localize objects within each frame. By employing SSD, the model predicts bounding boxes that tightly enclose identified objects and assigns corresponding class labels (e.g., person, car, dog) based on learned object categories. This step is crucial for accurately identifying and categorizing objects present in the video stream.

4] Real-Time Output Visualization:

Object detection and classification, the system proceeds to visualize the results in real-time. Detected objects are displayed along with their respective bounding boxes overload on the image and video feed. Class labels associated with each object provide additional context, facilitating immediate understanding and action based on the detected objects.

Technical Implementation

The technical implementation of the program focuses on providing maximum flexibility and configurability.It supports optional video file inputs and allows users to set custom confidence thresholds.

The detection process relies on two critical model files: a prototext file for network configuration and a caffemodel file containing pre-trained weights.

The program is configured to detect multiple object classes including:

- Water Bottles
- Chairs
- Persons
- SmartPhones
- Background Elements

Key Components and Parameters

The program features a dedicated Object Detector class with the following characteristics:

- Configurable confidence threshold (default: 0.2)
- Predefined class name mapping
- Efficient network loading mechanism
- Comprehensive frame processing method

Detection parameters are carefully calibrated:

- Input frame size: 300x300 pixels

- Blob scaling factor: 0.007843
- Normalization technique: Mean subtraction

Potential Applications

The object detection program has diverse potential applications, including:

- Surveillance systems
- Autonomous navigation
- Robotics
- Interactive mobile applications
- Security monitoring
- Augmented reality environments

Limitations

Current technical constraints include:

- Fixed input resolution of 300x300 pixels
- Limited object class detection
- Dependency on pre-trained model
- Hardware-dependent performance variations

System Advantages:

- **Efficiency and Computational Speed:**

The streamlined architecture of MobileNet SSD ensures high computational efficiency without compromising on detection accuracy. Depth-separable convolutions reduce the computational load compared to traditional convolutional neural networks, making it well-suited for deployment on resource constrained devices such as mobile platforms or embedded systems.

- **Robustness to Varied Environmental Conditions:**

The system's capability to operate effectively with low-quality video input and varying environmental conditions underscores its robustness. By leveraging efficient feature extraction and object localization techniques, MobileNet SSD maintains reliable performance across different lighting conditions, camera perspectives, and environmental settings. This object detection algorithm achieves good results with any fps low quality camera and can detect objects in real time with decent accuracy. In our experiment we gave different images as input and the model has identified then with a good accuracy. And then we used the webcam to detect objects in the real-time which also produced the desired results.

- **Scalability and Application Versatility:**

Designed with scalability in mind, the system can be adapted for various applications including surveillance, traffic monitoring, retail analytics, and more. lts ability to process real-time video streams and provide actionable insights enhances its utility in dynamic and time-sensitive scenarios.

5. Conclusion and Future Work

According to our research, the most important problems of one-stage methods in object detection is the difficulty to detect small objects as well as inaccurate localization. As these issues are combined with hardware limitations, they have become an important challenge. We attempted to tackle hardware limitations while striving to keep the process in the real-time range.

REFERENCES

[1] Yundong Zhang, Haomin Peng haomin and Pan Hu, "Towards Realtime Detection and Camera Triggering," CS341.
[2] Ibai Gorordo Fernandez and Chikamune Wada, "Shoe Detection Using SSD-MobileNet Architecture,"2020 IEEE 2nd Global Conference on Life Sciences and Technologies (LifeTech 2020).
[3] Yu-Chen Chiu, Chi-Yi Tsai, Mind-Da Ruan, Guan-Yu Shen and TsuTian Lee, "Mobilenet-SSDv2: An Improved Object Detection Model for Embedded Systems," ©2020 IEEE.
[4] Andres Heredia and Gabriel Barros-Gavilanes," Video processing inside embedded devices using SSD-Mobilenet to count mobility actors," 978-1-7281-1614-3/19 ©2019 IEEE.
[5] G. Bradski and, A. Kaehler, "Learning OpenCV", OReilly Publications, 2008.
[6] Animesh Srivastava1, Anuj Dalvi2, Cyrus Britto3, Harshit Rai4, Kavita Shelke5," Explicit Content Detection using Faster R-CNN and SSD MobileNet v2," e-ISSN: 2395-0056 © 2020, IRJET.
[7] R. Huang, J. Pedoeem, and C. Chen, "YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers,"