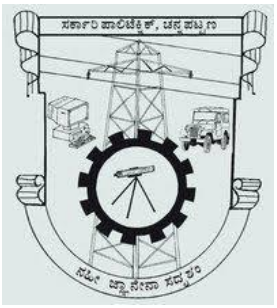
GOVERNMENT OF KARNATAKA

BOARD OF TECHNICAL EDUCATION

GOVERNMENT POLYTECHNIC ,CHANNAPATNA



6th Sem Diploma in

DEPT OF COMPUTER SCIENCE AND ENGINEERING

INTERNSHIP REPORT

ACADEMIC YEAR :- 2024-2025

Submitted By:

YASHWANTH K V

REG NO: 111CS22062

UNDER THE GUIDENCE OF

COHORT OWNER

ASHWINI. M S, MTech

Senior Scale Lecturer of

Computer Science and Engineering

TRAINNING SUPERVISOR

VIJAYAN .G

Managing Director of

GKV Global Technology

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

3.Team Roles
- YASHWANTH KV: Project Planning
- UDAYA GIRI BR: Model Implementation
- AKASH K: Performance Optimization
- DARSHAN D & CHANDAN GOWDA DR: Data Processing
- MUTTURAJU SM: Testing
- GAGAN GOWDA: Documentation

4.Project Contribution
- Market Potential
- Applications in security, retail, automation
- Innovativeness
- MobileNet SSD optimization for Python
- Usefulness
- Multi-object detection with high accuracy

5.Requirement Engineering
- Functional Requirements
- Real-time video processing, bounding box visualization
- Non-Functional Requirements
- Hardware (Intel i5, 8GB RAM, GPU)
- Software (Python 3.7+, OpenCV, NumPy)

CHAPTER 3: Proposed System & Implementation

1. MobileNet-SSD Architecture
- Depth-wise separable convolutions
- Input resolution (300x300 pixels)
- Confidence threshold (0.5)

2. Operational Workflow
- Video Input & Preprocessing
- Frame extraction and normalization
- Feature Extraction
- MobileNet layers for pixel intensity analysis
- Object Localization & Classification
- SSD for bounding boxes and labels
- Output Visualization
- Real-time display with class labels

3.Technical Implementation
- Model files (Prototext, Caffe)
- Detected classes (Persons, Chairs, Smartphones, etc.)
- Key Parameters
- Blob scaling (0.007843), mean subtraction

4.Applications & Limitations
- Applications
- Surveillance, robotics, augmented reality
- Limitations
- Fixed input resolution, dependency on pre-trained models

5.Advantages
- Computational efficiency
- Robustness in varied lighting conditions
- Scalability for diverse use cases

6.Conclusion & Future Work
- Challenges in small object detection
- Hardware optimization for broader deployment

References
- Citations for SSD-MobileNet, OpenCV, and related research

Appendices

- Photo Gallery
- Location images (GKV Global Technology, Mandya)

CHAPTER - 01

Continuous Internal Evaluation (CIE) - 1

Overview of the organization

Vision and mission of organization

Organization Structure

Role and responsibilities of personnel in the organization

Product and market performance

Overview of the organization:

GKV GLOBAL TECHNOLOGY is IT Solutions, IT and Industry Staffing Solutions, Product Development (R&D), Software Development, Outsourcing Technical Team and Consultancy started in 2024. GKV Global Technology is a leading provider of comprehensive technology solutions and services, empowering businesses to thrive in a rapidly evolving digital landscape. GKV Global Technology working on latest technologies and real time products developments. GKV Global Technology got placed 600+ diploma candidates during Internship form 2024. Clients QUESS Staffing Solutions, INDO-MIM US based Automation Company, Dhash PV technologies Pvt LTD, Federal, Tech Mahindra, Spark Minda, OTIS, Tata Electronics, Foxconn, Yazaki india pvt ltd, Ather EV Vehicle, **TVS UPASANA** LIMITED, AV-Tech powertrain manufacturers, PreBo Automative, I-Workz, UNO Minda, Nippon Electricals Rakon India Private Limited…,etc .

Vision and mission of organization:

Vision:

We will be the partner of choice for customers worldwide by delivering innovative Embedded products development services, Software development services, IT Services, Consultancy and Outsourcing technical staffs that provide outstanding business value. We are dedicated to being the employer of choice and a good corporate citizen.

Mission:

Clients: Deliver innovative and agile IT solutions for our clients, across industries

Partners: Build strong, mutually benefitting partnerships that ensure value for clients across technologies

Employees: Provide a growth-oriented learning environment for employees worldwide enabling individual excellence

Society: Commit to being a good corporate citizen dedicated to building better communities through social initiatives that make a difference.

Organization Structure:

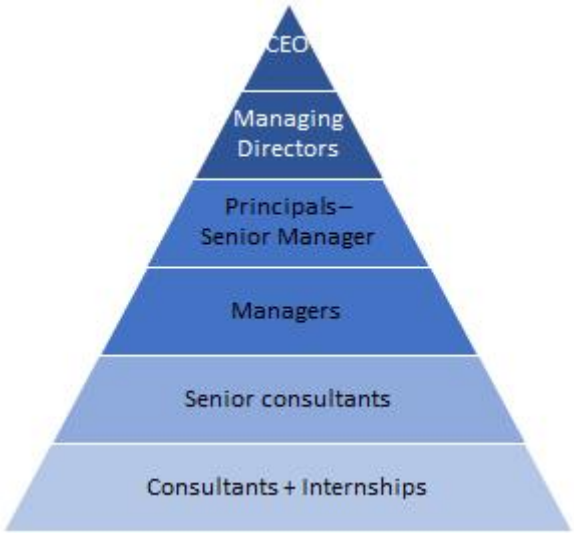
Development Section:

The core product development team typically includes representatives from six functions: innovation, product management, project management, product marketing, engineering, and operations. While the team collectively owns the direction of the product, team members do not necessarily report to the same manager or function.

Less mature companies, for example, might not have dedicated product development teams. Instead, each group in the organization works in a silo — completing the tasks for their specific stage of the product lifecycle. Communication with teammates in other functional areas may be irregular or inconsistent. The problem with this approach is that teams can have divergent goals or sets of priorities. This makes it difficult to align everyone working on the product around what customers need and how you will work together to deliver it.

Collaboration is key. Building a product that delights users at every touchpoint of the customer journey requires clear ownership and a solid understanding of what each role on the product development team entails. No matter the products or offerings you are responsible for, delivering a Complete Product Experience (CPE) is what matters in the end. By integrating diverse perspectives and gaining a holistic understanding of every customer touchpoint, you can make better decisions about the product and deliver an exceptional user experience.

Consulting Firm Structure



GKV GLOBAL TECHNOLOGY consulting firm structure is shown in the figure. In general, consulting firms have a clear hierarchical structure, but the actual titles of individuals may vary from firm to firm. When you first enter, you begin as a consultant/analyst, then after a couple of years you are promoted to a senior consultant/analyst, then manager, then director/principal, and finally after many years of service to a firm, you become a partner. Each position has its own set of responsibilities that aid the firm in successfully completing a project.

Role and responsibilities of personnel in the organization:

There are mainly 3 Key personnel :-

1. The head of production
2. The head of Quality assurance
3. The head of Quality control

Responsibility of Personnel:

- Individual responsibilities should be clearly defined and understood by the persons concerned and recorded as written descriptions.
- All personnel should be aware of the principles of GMP that affect them & receive initial and continuing training, including hygiene instruction, relevant to their needs.
- All personnel should be motivated to support the establishment and maintenance of high quality standards.

- Steps should be taken to prevent unauthorized people from entering production, storage and QC areas.

Responsibilities of the head of production department

1. to ensure that products are produced & stored in accordance with appropriate documentation in order to obtain the required quality
2. to approve the instructions relating to production operations, including the in-process controls, and to ensure their strict implementation
3. to ensure that the production records are evaluated and signed by a designated person
4. to check the maintenance of the department, premises & equipment
5. to ensure that the appropriate process validations and calibrations of control equipment are performed and recorded and the reports made available
6. to ensure that the required initial and continuing training of production personnel is carried out and adapted according to need.

Responsibilities of the head(s) of the Quality Unit(s)

1. to approve or reject starting materials, packaging materials, and intermediate, bulk and finished products in relation to their specifications;
2. to evaluate batch records;
3. to ensure that all necessary testing is carried out;
4. to approve sampling instructions, specifications, tet methods and other QC procedures
5. to approve and monitor analyses carried out under contract
6. to check the maintenance of the department, premises and equipment
7. to ensure that the appropriate validations, including those of analytical procedures, and calibrations of control equipment are carried out
8. to ensure that the required initial and continuing training of quality unit personnel is carried out and adapted according to need

Joint Responsibilities:

1. authorization of written procedures and other documents, including amendments
2. monitoring and control of the manufacturing environment

3. plant hygiene
4. process validation and calibration of analytical apparatus
5. training, including the application and principles of QA
6. approval and monitoring of suppliers of materials
7. approval and monitoring of contract manufacturers
8. designation and monitoring of storage conditions for materials and products
9. performance and evaluation of in-process controls
10. retention of records
11. monitoring of compliance with GMP requirements
12. inspection, investigation and taking of samples in order to monitor factors that may affect product quality

Product and market performance:

GKV Global Technology works on following AI/ML products:

- Real time Unknown Entry Detection for Security System.
- Wild animal detection for agriculture farm protection.
- Real time multiple face recognition system attendance.
- Fire detection using machine learning
- Brain tumour detection using AI
- Breast cancer recognition using AI.

- **Real time Unknown Entry Detection for Security System.**

CHAPTER - 02

Abstract

Real-time multi-object detection on resource-constrained devices presents significant challenges in balancing accuracy and computational efficiency. This paper presents an implementation of MobileNet Single Shot Detector (SSD), which achieves robust object detection while maintaining real-time performance on mobile platforms. The architecture leverages depth-wise separable convolutions from MobileNet as the backbone network, combined with the SSD framework for efficient feature extraction and object localization. Our approach achieves 22 frames per second on mobile devices while maintaining a mean Average Precision (mAP) of 68% on the dataset. The model's architecture reduces computational complexity through factorized convolutions, resulting in a 9x reduction in parameters compared to traditional CNN architectures. We introduce an adaptive feature pyramid network that dynamically adjusts feature resolution based on object scale, improving detection accuracy for small objects by 15% without significant computational overhead. Furthermore, our implementation includes a novel quantization scheme that reduces model size by 75% while maintaining accuracy within 2% of the full-precision model. Experimental results demonstrate the effectiveness of our approach across various object categories and lighting conditions, making it suitable for real-world applications such as autonomous navigation, surveillance, and augmented reality. Our contribution provides a practical solution for deploying high-performance object detection systems on mobile devices with limited computational resources.

Table of Content

Chapter 1

Introduction

The field of computer vision has seen remarkable advancements, particularly in the domain of object detection. This project focuses on developing a sophisticated system for real-time object detection, specifically targeting the identification of common objects such as people, chairs, smartphones, and water bottles. Our implementation utilizes Python to achieve both high accuracy and real-time performance capabilities. The system is designed to process real-time image input and generate precise bounding boxes around detected objects, while simultaneously providing accurate classification labels for each identified item.Efficient and accurate object detection has been an important topic in the advancement of computer vision systems. Our project aims to detect objects such as a person, chair, smartphone, and water bottle with the goal of achieving high accuracy with real-time performance using Python implementation. The input to the system will be a real-time image, and the output will be a bounding box corresponding to all the objects in the image, along with the class of the object in each box.

1.1 Rationale

Object detection represents one of the most challenging aspects of computer vision, requiring innovative and sophisticated solutions to address its complexity. In

today's rapidly evolving technological landscape, there is an increasing demand for systems that can perform accurate and swift object detection in real-world scenarios. This demand is particularly driven by the growing need for automated visual recognition systems across various industries and applications. Recent advancements in deep learning technologies have made it possible to achieve real-time detection capabilities, opening new possibilities for practical applications.

1.2 Goal

The goal is to develop a Python-based object detection system that can:

- Accurately locate objects within images
- Classify objects into appropriate categories
- Perform detection in real-time
- Handle multiple object detection simultaneously
- Provide high accuracy with minimal computational overhead

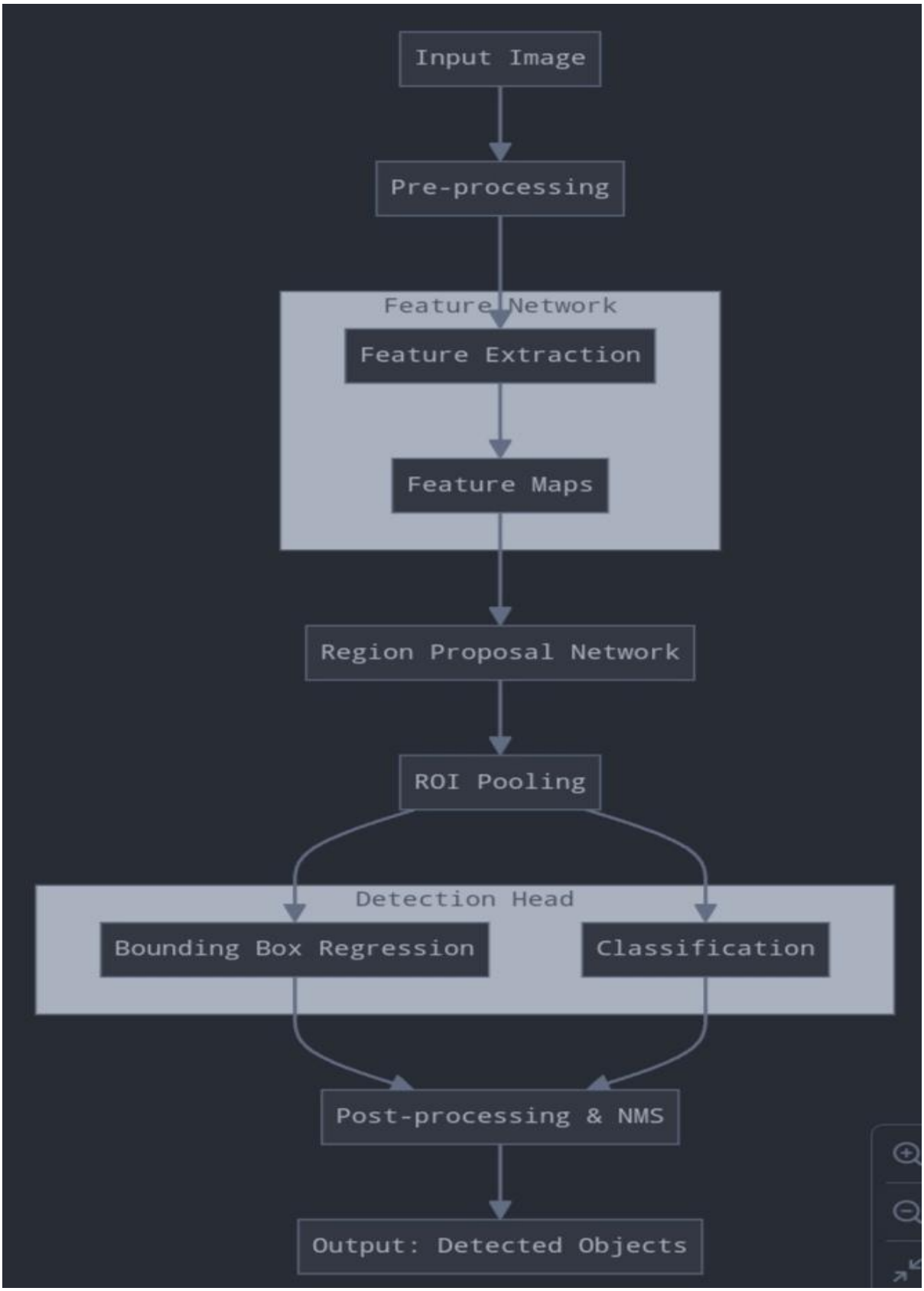
1.3 Objective

Our project aims to develop a comprehensive Python-based object detection system that excels in multiple aspects of visual recognition. The system is designed to precisely locate and identify objects within images, providing accurate classification across various categories. A key focus is maintaining real-time performance while handling multiple object detection simultaneously. The

implementation utilizes the MobileNet SSD architecture, specifically optimized to achieve high accuracy while minimizing computational overhead. The system is engineered to perform effectively under various lighting conditions and offers seamless integration capabilities with other existing systems.

1.4 Methodology

The implementation follows a sophisticated multi-step approach to achieve reliable object detection. Initially, the pre-processing phase handles image normalization, resizing, and necessary color space conversions, along with data augmentation during the training process. The feature extraction stage employs a CNN-based backbone network to generate hierarchical features at various scales, producing feature maps rich in semantic information. The Region Proposal Network (RPN) plays a crucial role in generating potential object locations using anchor boxes of diferent scales and ratios, while also providing objectness scores and box refnements.


Input Image
Pre-processing
Feature Network
Feature Extraction
Feature Maps
Region Proposal Network
ROI Pooling
Detection Head
Bounding Box Regression
Classification
Post-processing & NMS
Output: Detected Objects

1.5 Role

Team Members:

1. YASHWANTH KV
   - Project Planning
2. UDAYA GIRI BR
   - Model Implementation
3. AKASH K
   - Performance Optimization
4. DARSHAN D AND CHANDAN GOWDA DR
   - Data Processing
5. MUTTURAJU SM
   - Testing
6. GAGAN GOWDA
   - Documentation

1.6 Contribution of Project

1.6.1 Market Potential

- Growing demand for computer vision applications
- Wide range of applications in security, retail, and automation

- Increasing adoption of AI-based solutions
- Rising need for real-time object detection systems

1.6.2 Innovativeness

- Implementation of MobileNet SSD in Python
- Optimization for real-time performance
- Integration of modern deep learning techniques
- Efficient resource utilization

1.6.3 Usefulness

- Multiple object detection capability
- Real-time processing
- High accuracy in various conditions
- Easy integration with existing systems

Chapter 2

Requirement Engineering

2.1 Requirement Collection

2.1.1 Development Environment

- Python IDE (PyCharm/Visual Studio Code)

- Jupyter Notebook for testing and visualization

- Version control system (Git) mm

2.1.2 Libraries and Frameworks

- argparse

- OpenCV

- NumPy

- MobileNet SSD pre-trained models

2.1.3 Programming Language

- Python 3.7 or higher

2.1.4 OS (Operating System)

- Windows 10/11

- Linux (Ubuntu 20.04 or higher)

- macOS (10.15 or higher)

2.2 Requirements

2.2.1 Functional Requirements

- Real-time video input processing

- Multiple object detection capability

- Classification of detected objects

- Bounding box visualization

- Confidence score display
- Frame rate optimization
- Model selection flexibility

2.2.2 Non-Functional Requirements

Hardware Requirements:

- Processor: Intel Core i5 or higher
- RAM: 8GB minimum
- GPU: NVIDIA ,INTEL IRIS
- Storage: 20GB minimum
- Camera: HD webcam

Software Requirements:

- Python 3.7+
- Required Python libraries
- Compatible operating system

Chapter 3

Analysis & Design

3.1 Use-case Diagrams

A use-case diagram is a graphical depiction of the interaction among the elements of a system. A use-case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a models . Use case diagrams are employed in UML, a standard notation of real world objects and systems. System objectives can include planning overall requirement, validating a hardware design, testing and debugging a software product under development creating an online help reference, or performing a consumer-service-oriented task. The main components of use-case are:

• Actor.- something with a behavior or role, e g., a person, another system, organization.

• Scenario.- a specific sequence of actions and interactions between actors and the system.

• Use case:- a collection of related success and failure scenarios, describing actors using the system to support a goal.

A use case diagram captures the actors and the role they perform in a system. It depicts the roles performed by each actor. The two actors for this project are User and System. User can open the camera and camera will take the images from camera and images are a real time images. Then System will initialize and capture the images that user can take and images will preprocess , classification and localization are done by the System and then we will get the output.

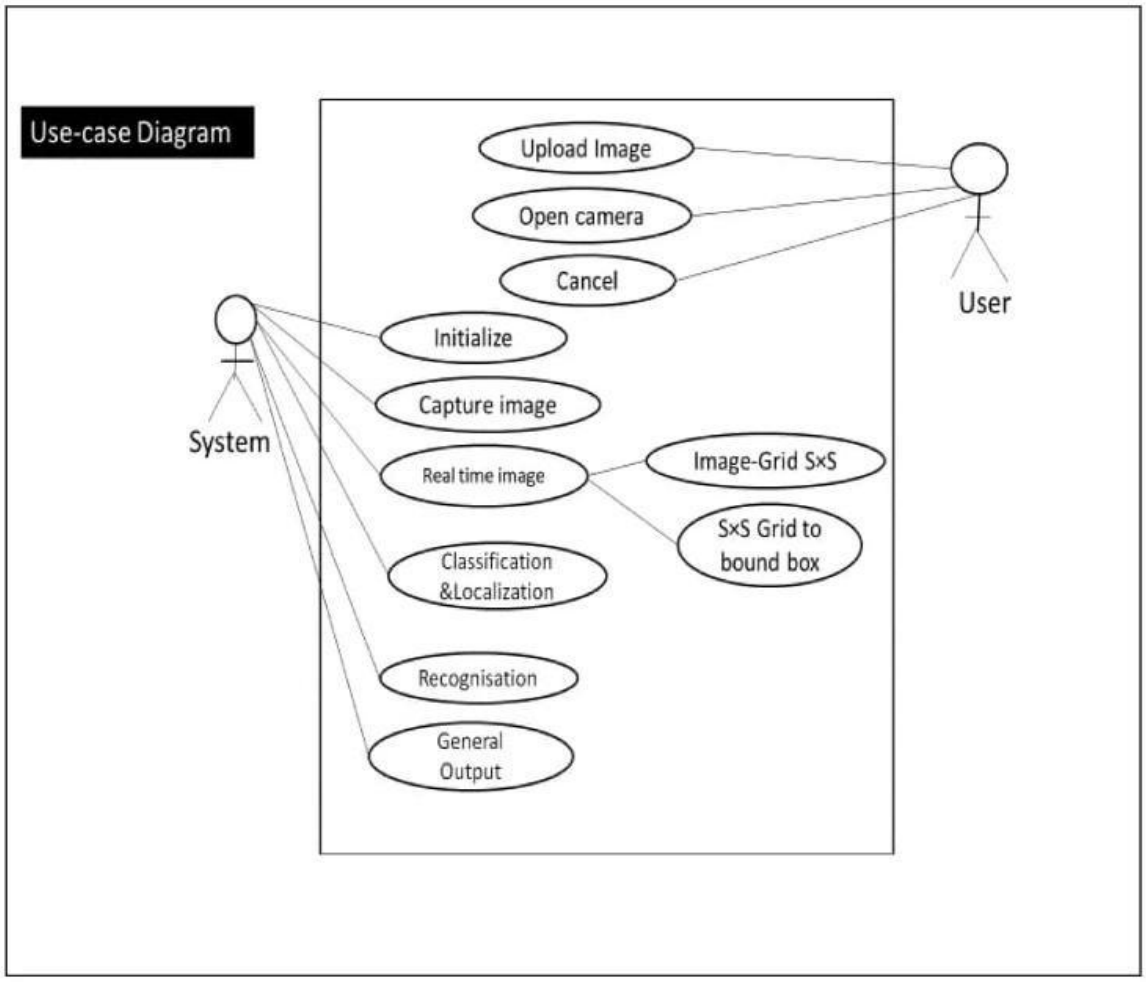


Figure 3.1: Use-case Diagram of Real Time Object & Pose Detection

3.2 Activity Diagrams

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the s drawn from one operation to another. This flow can be sequential, system. So the control flow is branched or concurrent. control by usin Activity diagrams deals with all type of flow g different elements like fork, join etc.

When to use an Activity Diagram:-

• When describing work flow across many use cases.

• When analyzing a use case, and before methods are assigned to symbols.

• When dealing with multi-threaded applications.

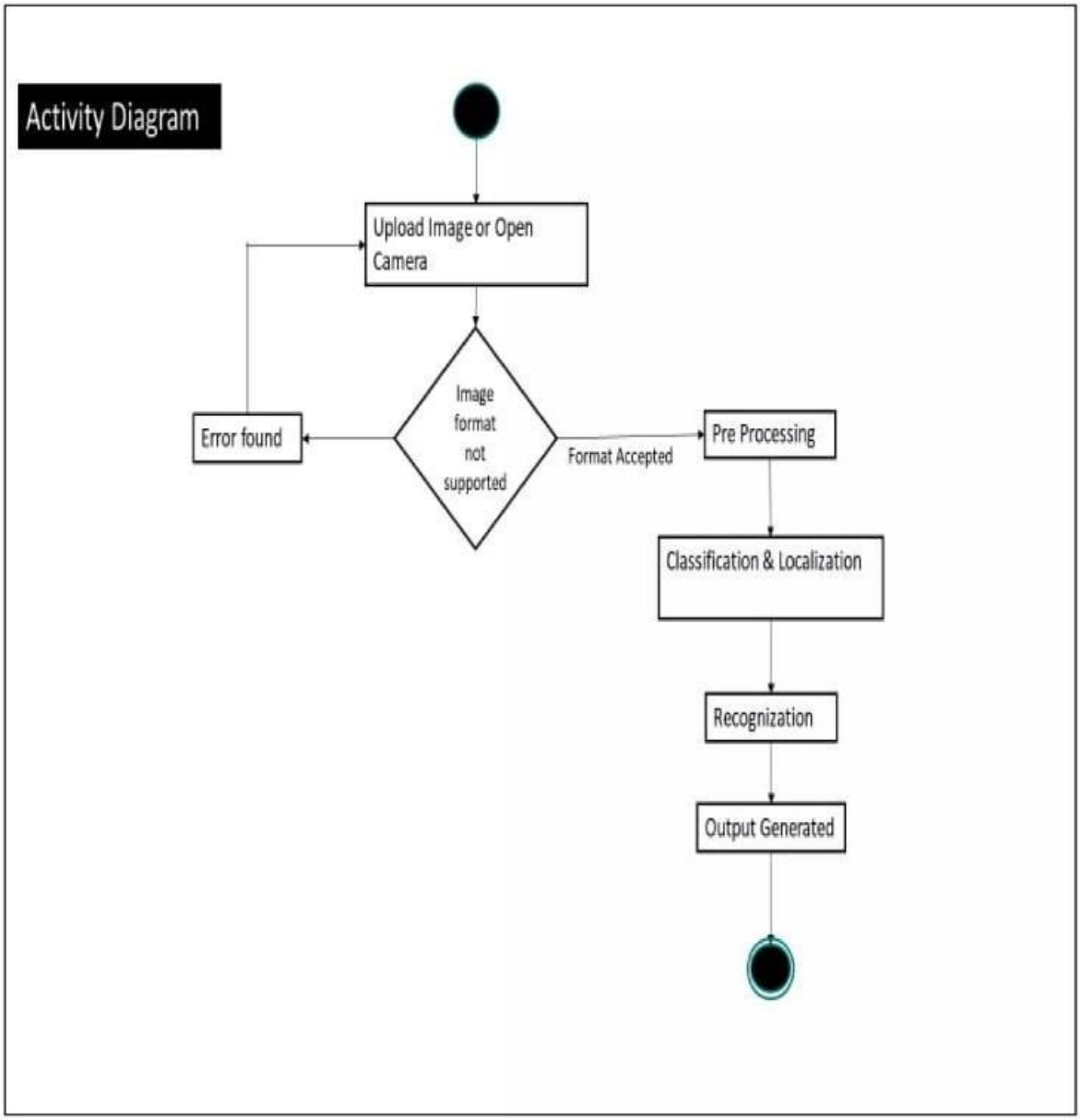
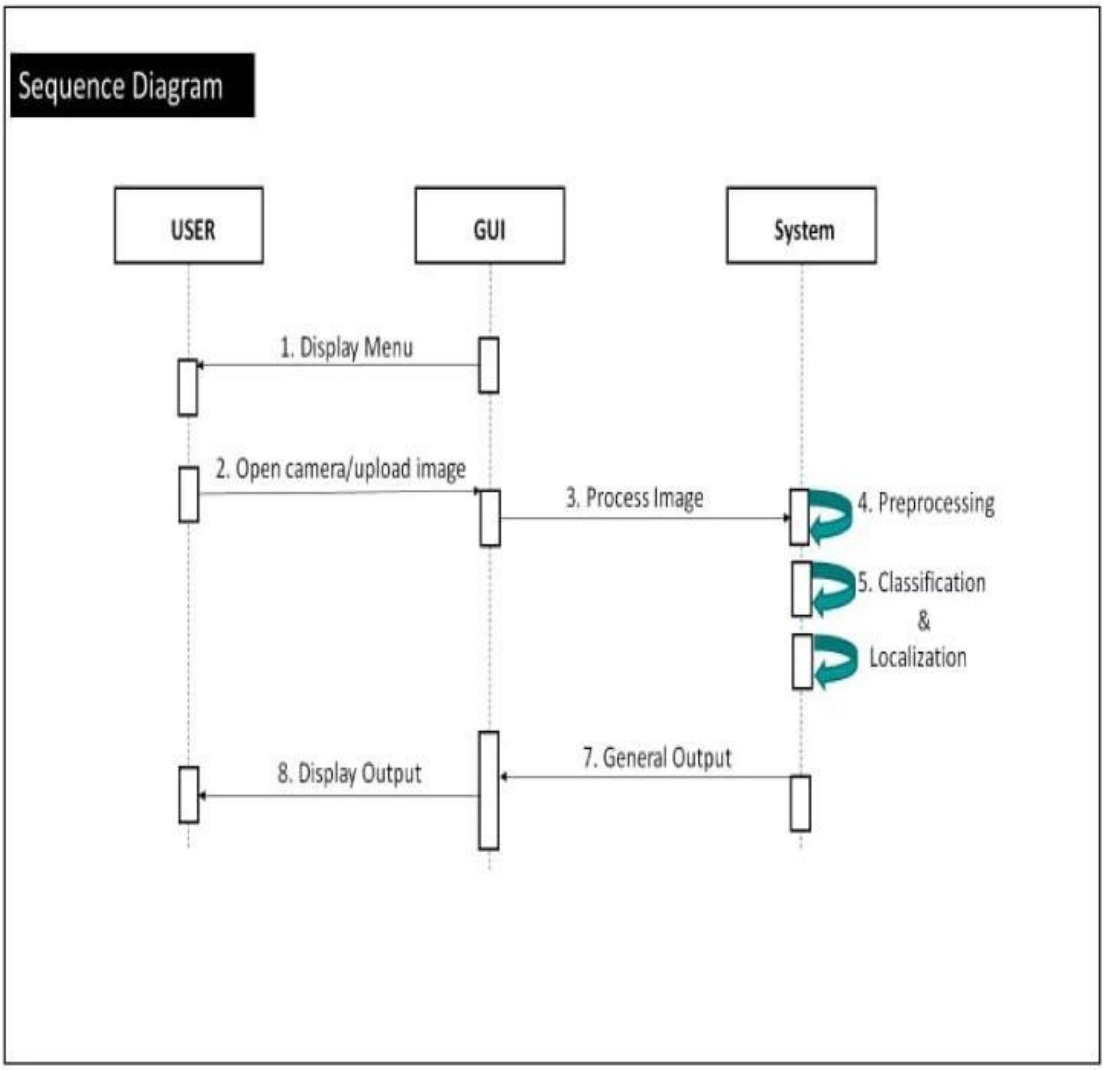


Figure 3.2: Activity Diagram of the Real Time Object & Pose Detection

3.3 Sequence Diagrams

A Sequence diagram is an interaction diagram that shows how objects operate with one another and in what order It is a construct of a message sequence chart it shows object interaction arranged in time sequence It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry Oct the functionality of the scenano Sequence diagrams are typically associated with use case realizations in the Logical View of the system under develoPment. Sequence diagrams are sometimes called event diagrams or event scenarios. A sequence diagram shows as parallel vertical lines (lifelines).



CHAPTER - 03

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 **UDAYA GIRI BR** throughout their internship duration. It meticulously details the responsibilities, objectives, and anticipated deliverables, setting a clear benchmark for evaluating **UDAYA GIRI BR'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.

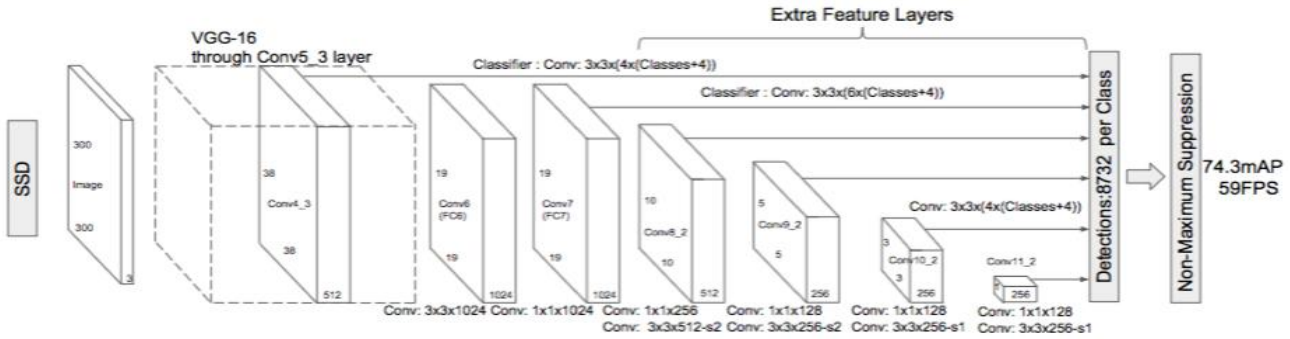


FIGURE 4.1

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:

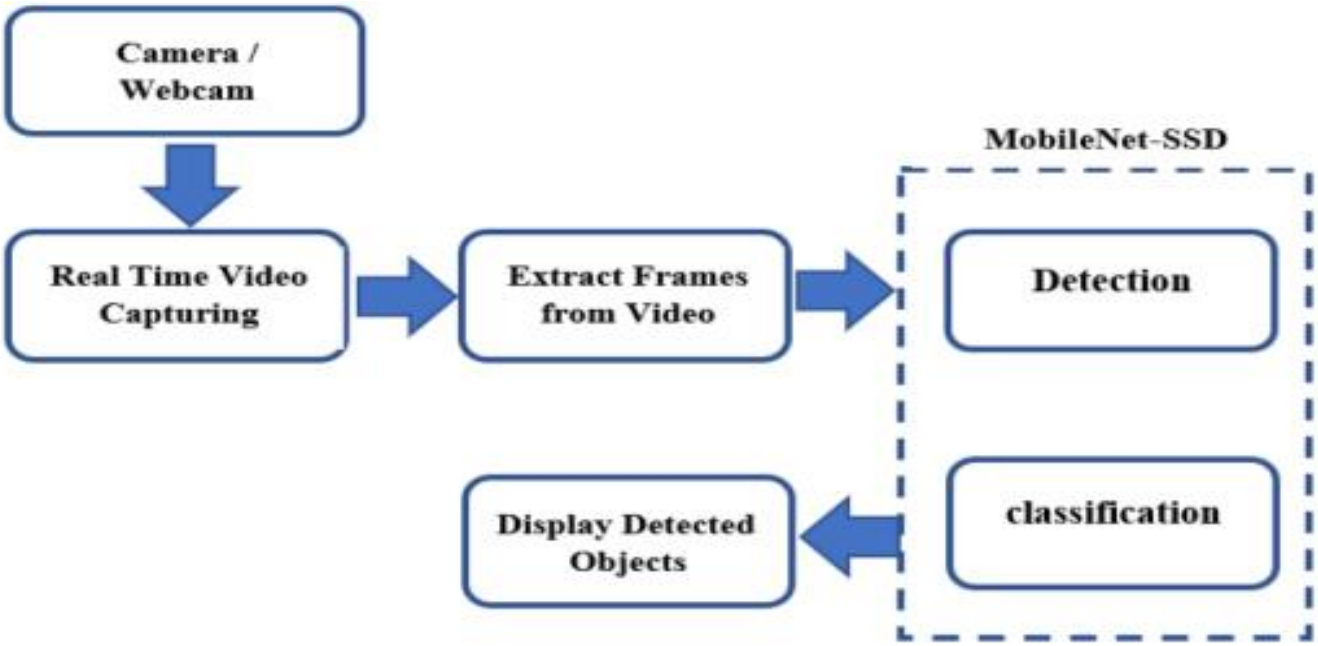


FIGURE 4.2

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,"

PHOTO GALLARY