"Object Detection using Deep Learning Approach"

By—

Paromita Saha

[M.Tech, Computer Science & Engineering, 4th Semester]

Roll No.: 502120021005, **Reg. No.:** 201430411210004

Subject: Project Part-II(Dissertation-II + Defense of Project-II)

Subject Code: PGCSE491

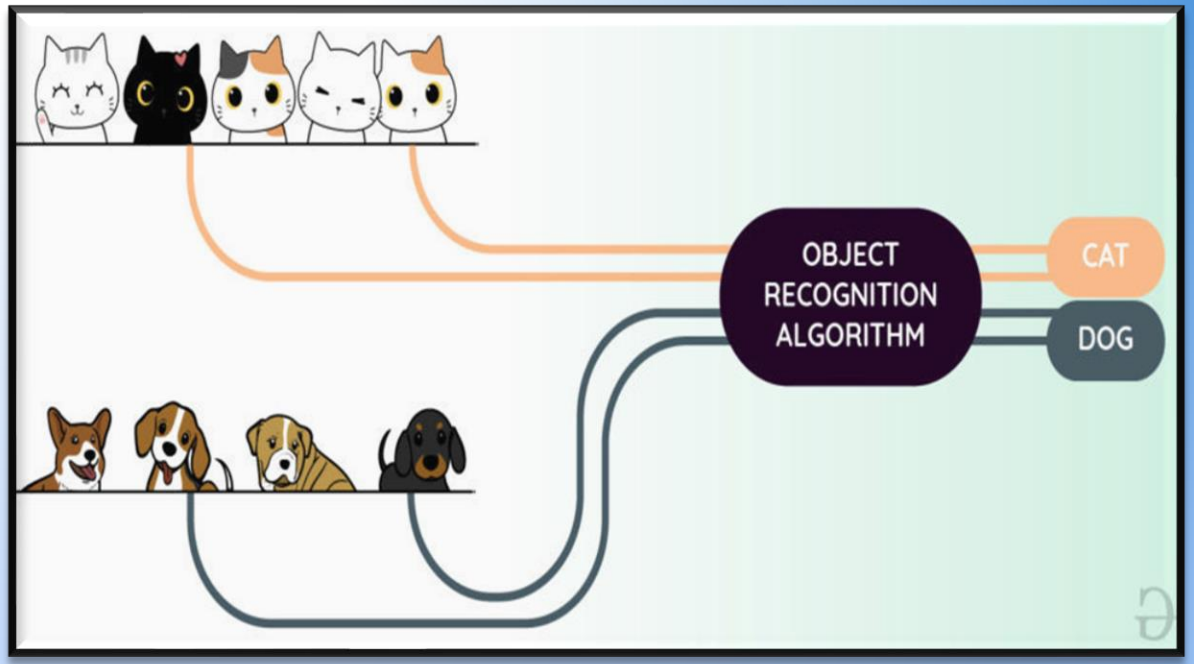
Under the guidance of

Mr. Moloy Dhar

Assistant Professor

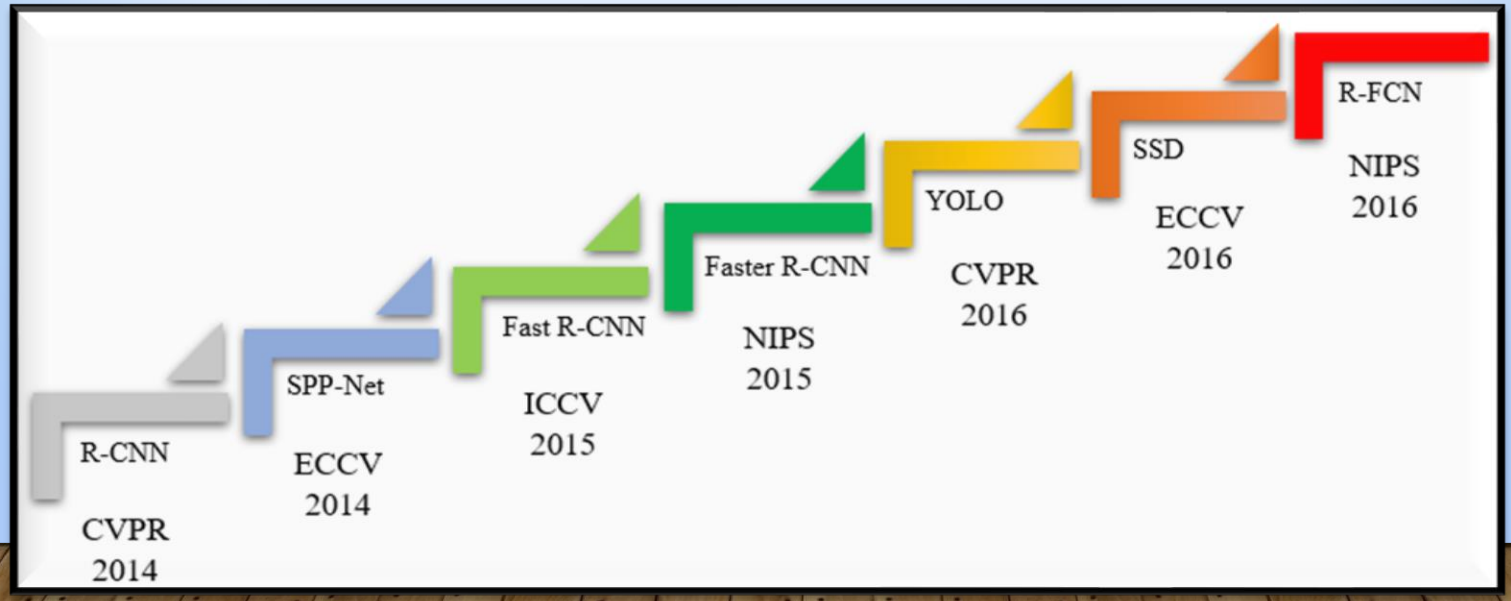
Guru Nanak Institute of Technology

Contents



Introduction

- Now a days, object detection is used globally in numerous fields such as, video surveillance, pedestrian displays, self-driving cars and appearance recognition.
- Deep Learning field, the sub-discipline which is called Object Detection, includes an image such as a photo, video, or webcam feed.
- There are many CNN models starting with classical LeNet model to AlexNet, ZFNet, GoogleNet, VGGNet, ResNet, ResNeXt, SENet, DenseNet, Xception, PNAS/ENAS.
- With the approach of computing infrastructure, Deep Neural Networks have achieved massive success in the domain of Computer Vision to handle such massive amounts of data.



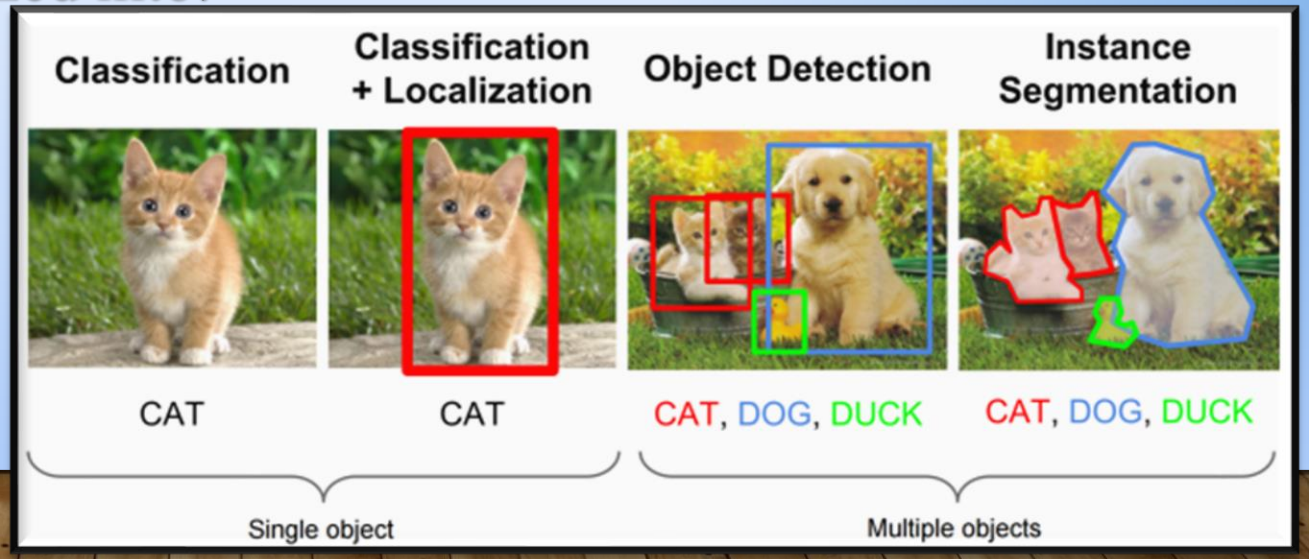
Object Detector Datasets

Objective

The goal of this project is to detect the objects from video sequences or high resolution images from the SSD MobileNet_v3 dataset. Motivated by this fact, this object detection software aims to correctly classify and detect the input data into its underlying category. MobileNet was developed to reduce the number of parameters and complexity of an algorithm to a number of resources demanded running by the embedded devices.

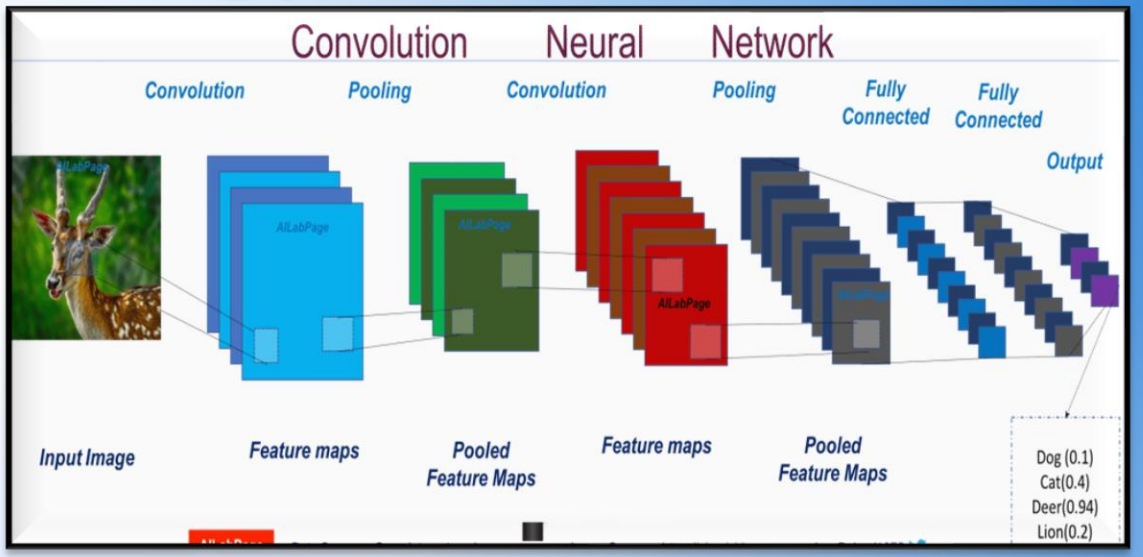
Depending on the dataset, it is categorized into:

i. Animals,
ii. Furniture's,
iii. Person,
iv. Fruits,
v. Vegetables,
vi. Vehicles, etc.

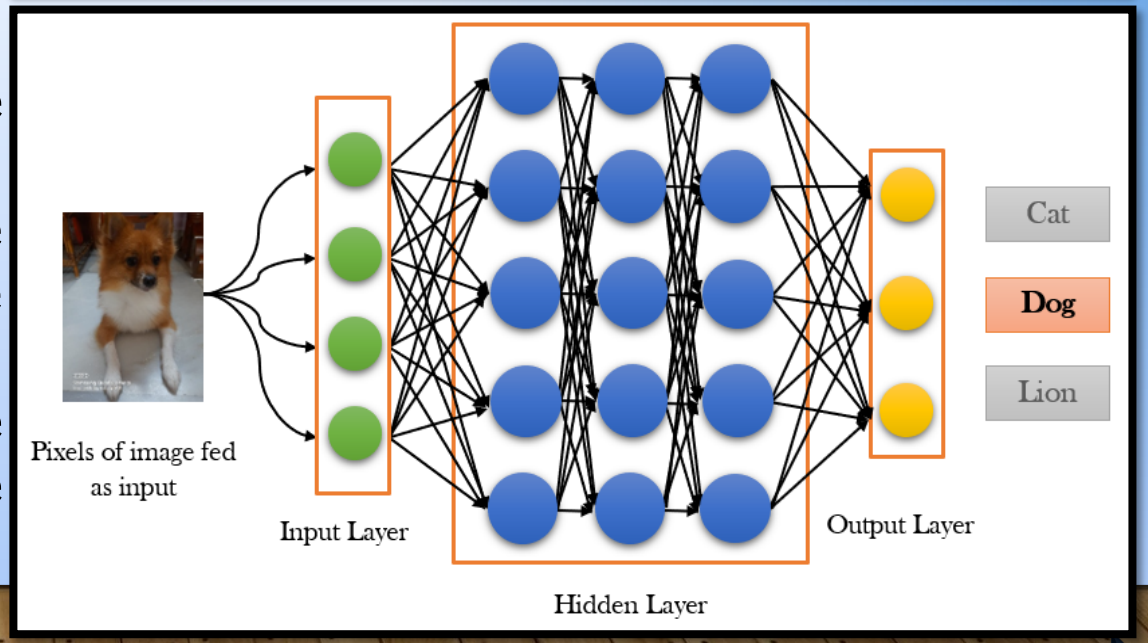


Methodology

1. **CNN model a Convolutional Neural Network:**
   - A class of deep neural networks, most commonly applied to analyze visual imagery.
   - It uses a special technique called Convolution. Convolution is the dot product of two matrices.
   - The whole CNN process has four layers: convolutional layers, pooling layers, fully connected layers and normalization layers.



2. **Deep Neural Network (DNNs):**
   - An Artificial Neural Network (ANN) with multiple layers between the input and output layers.
   - The components of neural networks functions are similar to human brains and can be trained like any other ML algorithm.
   - **Example:** a DNN can be trained to recognize the category from the given image and calculate the probability of the certain image.



CONTINUE--

CONTINUE--

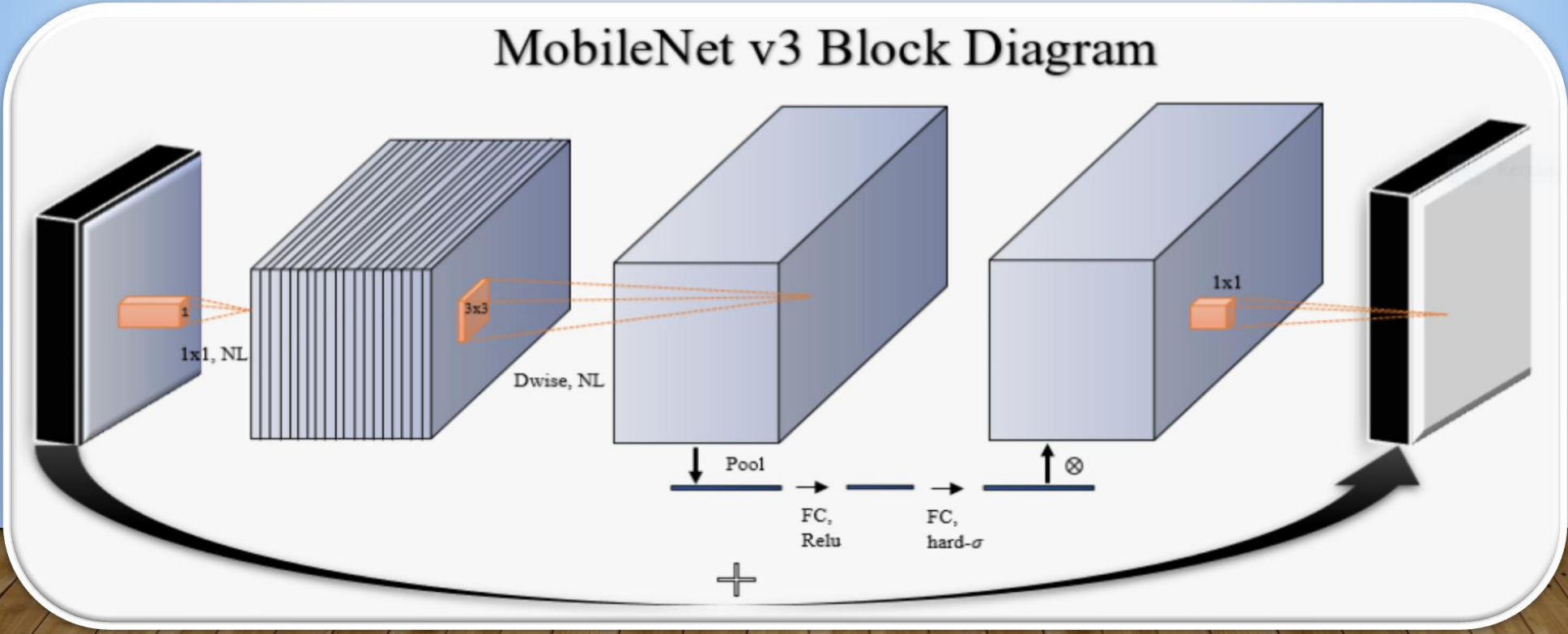
3. **Opensource Computer Vision (OpenCV):**
   - General-purpose programming language started by Guido van Rossum.
   - Library of Python bindings designed to solve computer vision problems.
   - Enables the programmer to express ideas in fewer lines of code without reducing readability.

4. **Single Shot Detector (SSD):**
   - Detecting numerous objects in an image which requires only one single shot.
   - Utilizes the anchor boxes with various aspect ratios and scales.
   - Faster than those methods which are based on two-shot RPN.

CONTINUE--

5. MobileNet v3:

- MobileNet is a type of convolutional neural network designed for mobile and embedded vision applications.
- MobileNetV3 is the third version of the architecture powering the image analysis capabilities of many popular mobile applications.
- MobileNetV3 is the use of AutoML to find the best possible neural network architecture.
- In object detection tasks, MobileNetV3 operated with 25% less latency and the same accuracy of previous versions.



CONTINUE--

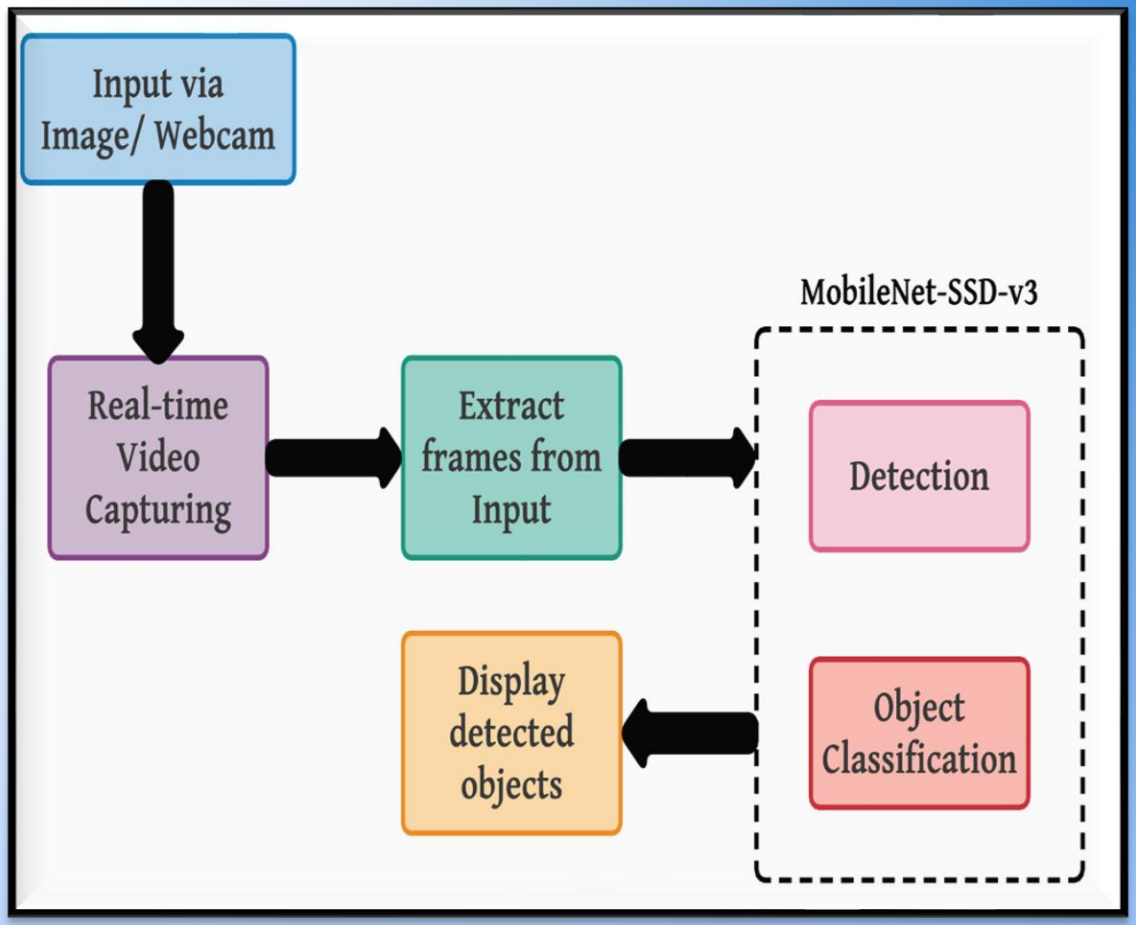
6. Non-Maximum Suppression (NMS):

- NMS (Non-Maximum Suppression) is a computer vision approach utilised in a variety of tasks.
- It's a set of algorithms for selecting one entity (like bounding boxes) out of several other overlapping ones.
- Windowing is used by most object detectors to create hundreds of thousands of windows (anchors) of varying sizes and shapes, containing only one object, and each class is assigned a probability/score by a classifier.
- It reduces observed bounding boxes to the ones having best probability(confidence) score.
- This method is used to "suppress" the less likely bounding boxes and only maintain the best ones.

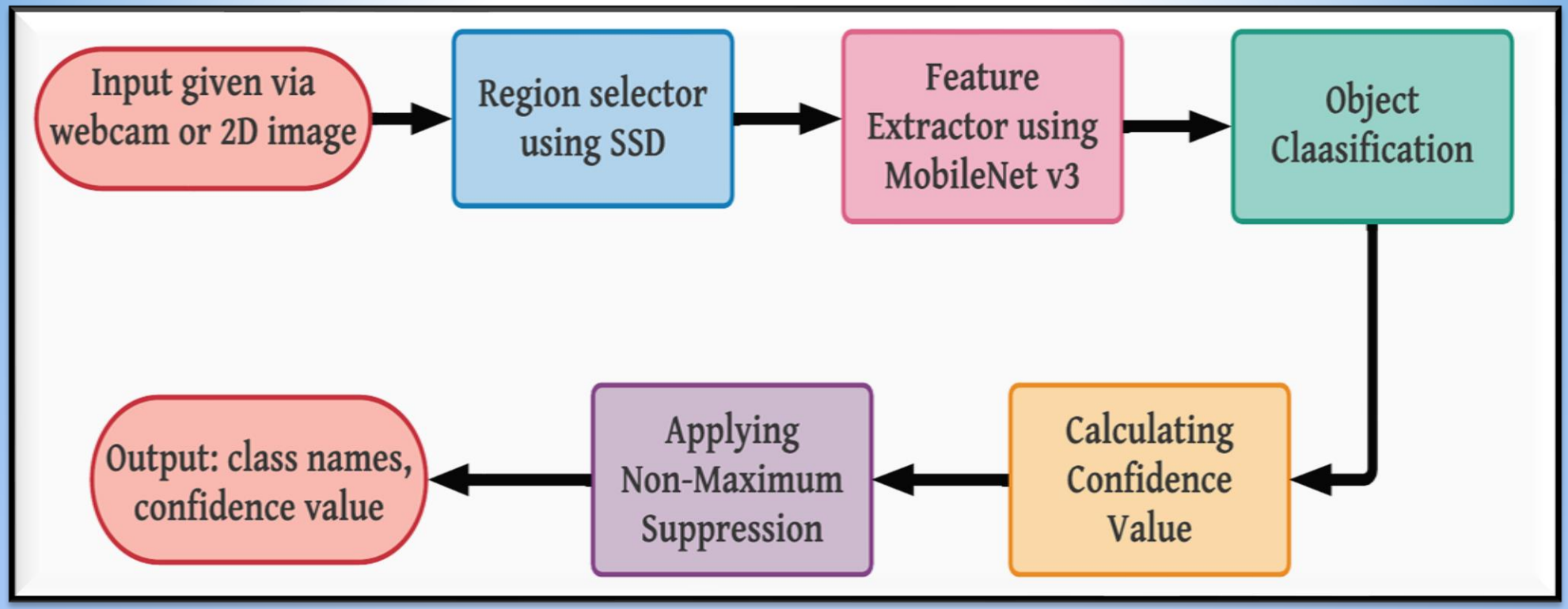
Proposed System

We will detect objects through webcam and static images using Mobilenet-SSD architecture written in Python code and using OpenCV. The system operates as follows:

- Input given via real-time webcam and static images.
- MobileNet architecture optimizes the images and builds light-weight deep neural networks.
- MobileNet layers receives the input images, and separates into frames.
- Each feature value is computed by reducing the pixel intensity of brighter region from darker region.
- The MobileNet layers converts the pixel in the input image into highlights that describes the image's contents.
- The bounding boxes and related class labels are then determined.
- The confidence score is displayed with the object classification class labels.

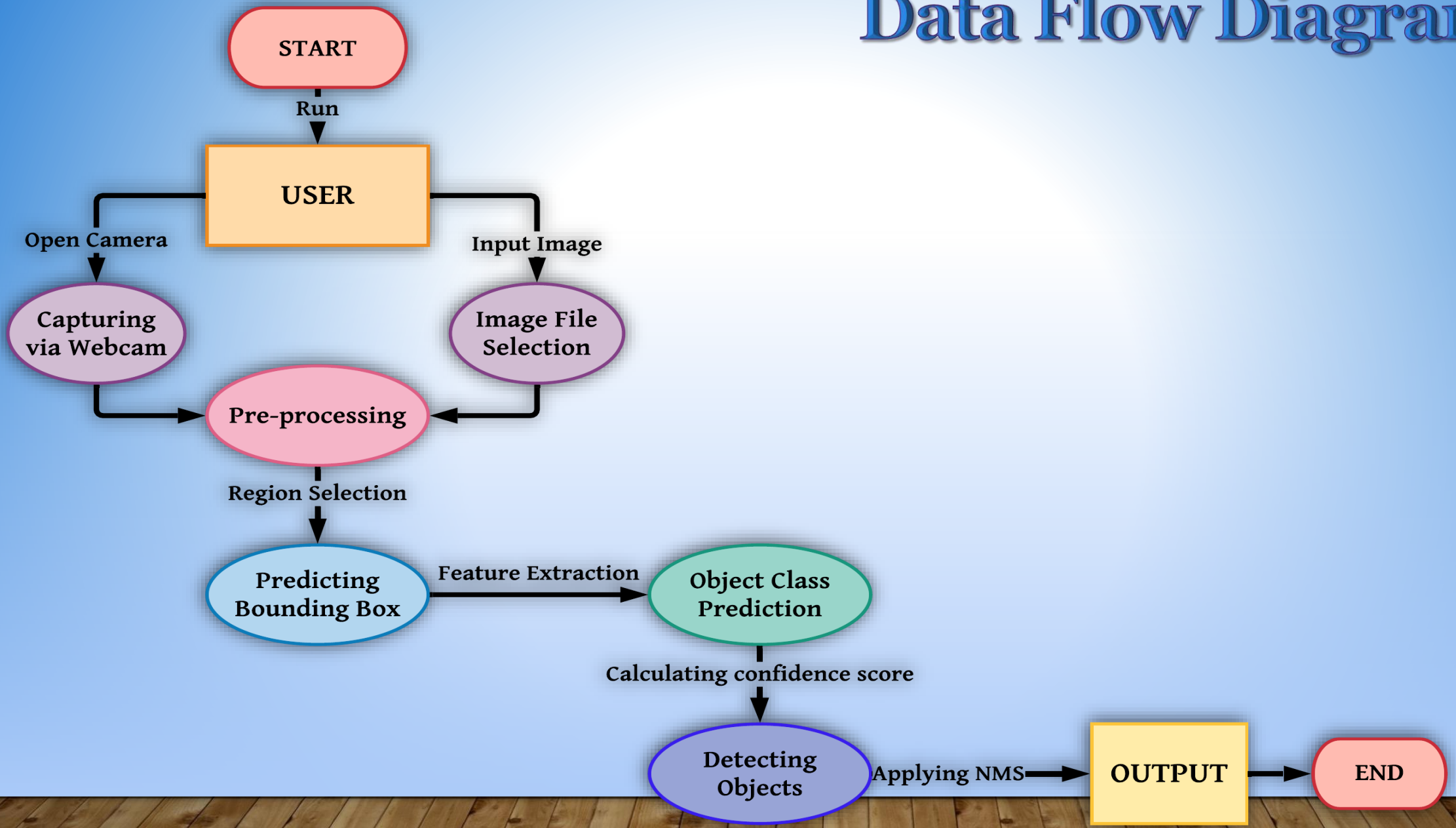


Flow Chart

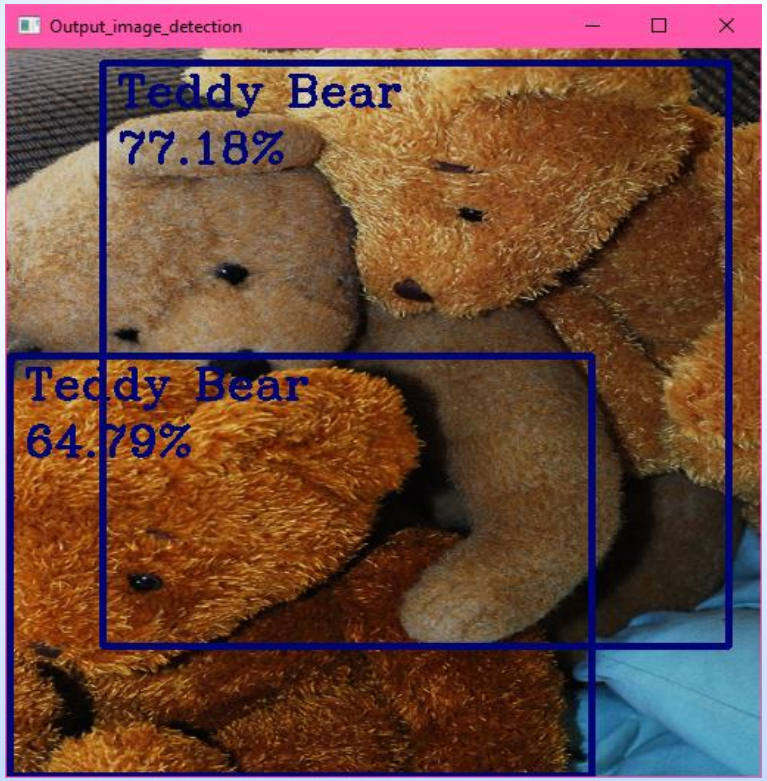


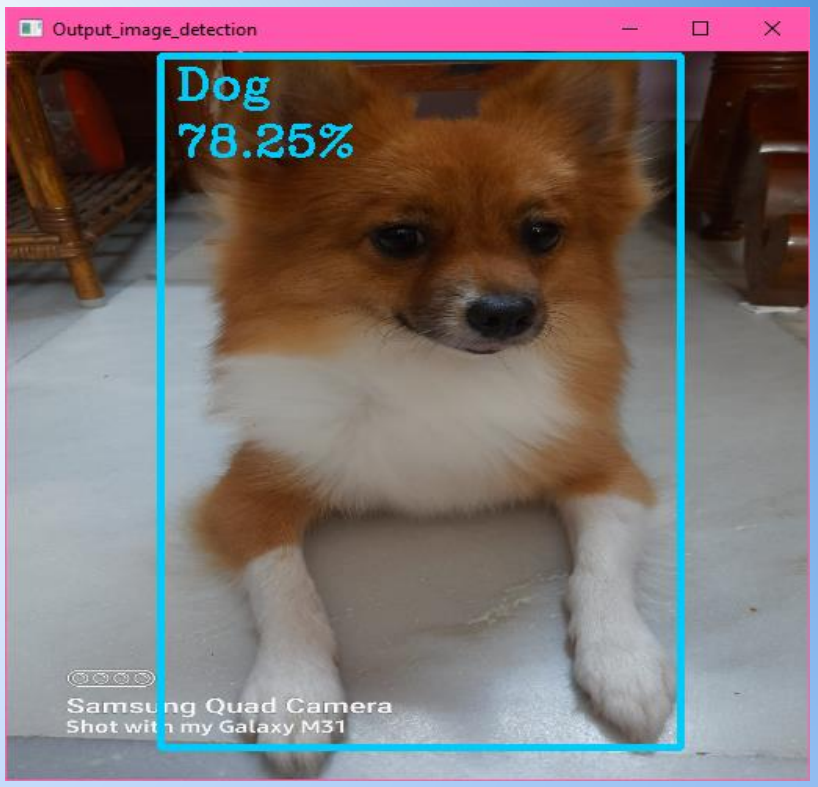
CONTINUE--

Data Flow Diagram



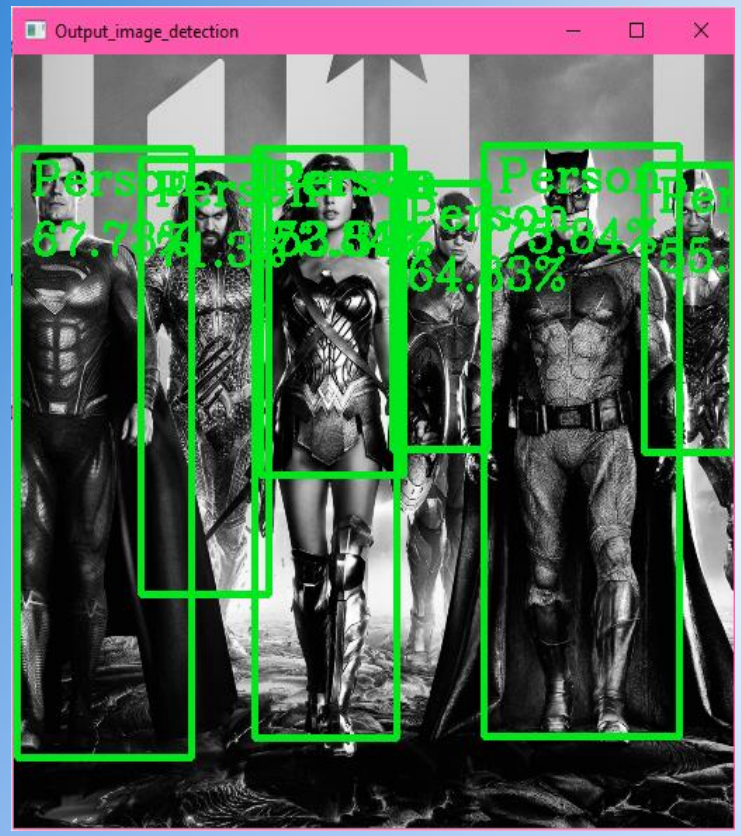
Results





Outputs taken through Colour Static Images

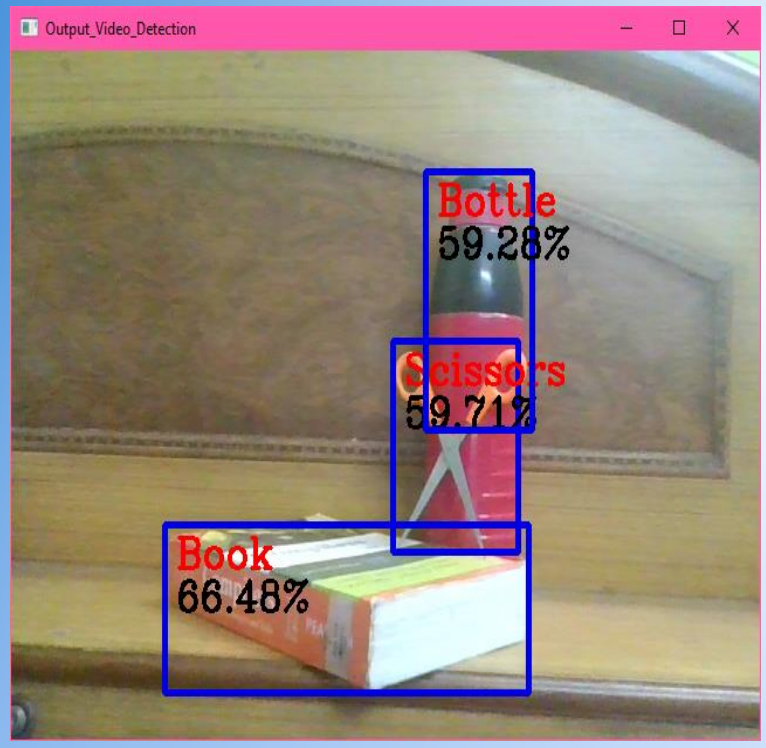
CONTINUE--

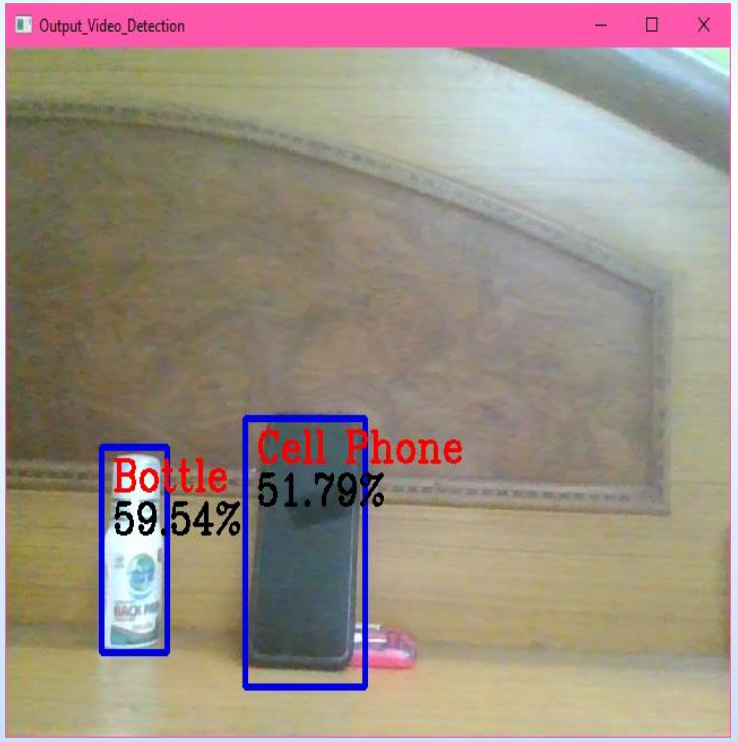


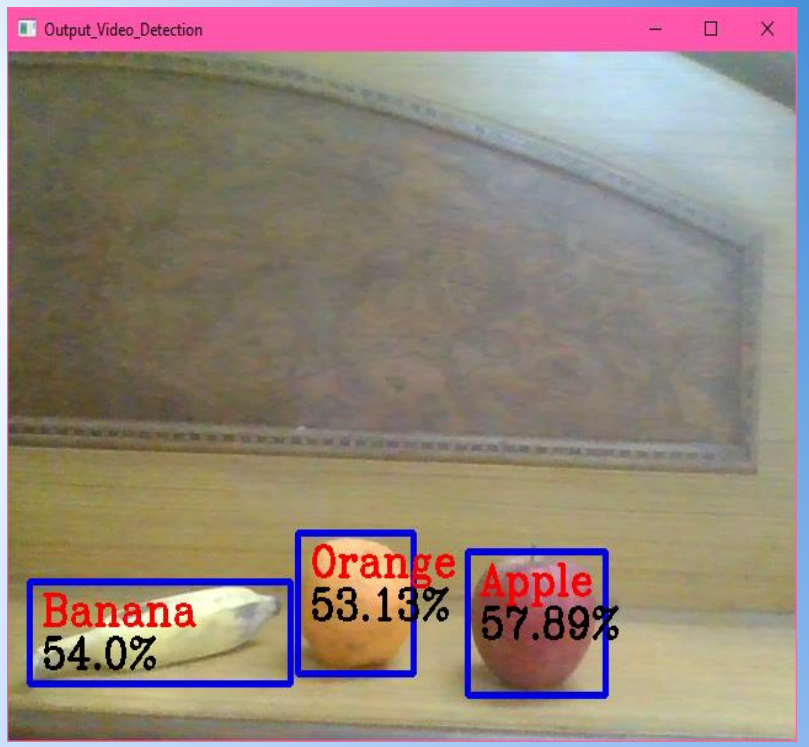
Outputs taken through Black & White Static Images

CONTINUE--

Continue --







Outputs taken through real-time webcam

Conclusion

- Deep learning techniques have recently been introduced in the detection of objects and humans using real-time webcam as well as high resolution images.
- This project is accomplished by combing two things: Object detection with deep learning and for quick and efficient, threaded video streams with OpenCV.
- A high accuracy object detection procedure is achieved by using SSD detector and MobileNet to improve the feature map and detection accuracy of the back-end detection network.
- Experiments shows Mobilenet-SSDv3 detector also improves detection performance significantly.
- The end result is a deep learning- based object detector using camera or webcam that can process around 6-30 FPS.

CONTINUE--

Future Work

In this project entitled "**Object Detection using Deep Learning Approach**", the followings are future working to be done:

- Surveillance systems, face identification, fault detection, character recognition, and other applications can all benefit from the object recognition system.
- Night vision mode could potentially be included as an option in tracking devices and CCTV cameras.
- We can also utilize real-time object detection to create self-driving automobiles that can identify items approaching from the front.
- We can teach new era children's to identify new things digitally.

References

[1] D. Wu, N. Sharma and M. Blumenstein, "**Recent advances in video-based human action recognition using deep learning: A review**," *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 2865-2872, doi: 10.1109/IJCNN.2017.7966210. Accessed at: https://ieeexplore.ieee.org/document/7966210

[2] S. Hu, X. Jia and Y. Fu, "**Research on Abnormal Behavior Detection of Online Examination Based on Image Information**," *2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 2018, pp. 88-91, doi: 10.1109/IHMSC.2018.10127. https://ieeexplore.ieee.org/document/8530188

[3] Dhillon, Anamika, and Gyanendra K. Verma. "**Convolutional neural network: a review of models, methodologies and applications to object detection**." Progress in Artificial Intelligence 9.2 (2020): 85-112, https://doi.org/10.1007/s13748-019-00203-0

[4] Srivastava, A., Dalvi, A., Britto, C., Rai, H., & Shelke, K. (2020). "**Explicit Content Detection using Faster R-CNN and SSD MobileNet v2**". International Research Journal of Engineering and Technology (IRJET), 7, 5572-5577.

[5] Xiao, Y., Tian, Z., Yu, J., Zhang, Y., Liu, S., Du, S., & Lan, X. (2020). "**A review of object detection based on deep learning**". Multimedia Tools and Applications, 79(33), 23729-23791, https://doi.org/10.1007/s11042-020-08976-6

[6] Younis, Ayesha, et al. "**Real-time object detection using pre-trained deep learning models MobileNet-SSD**." Proceedings of 2020 the 6th International Conference on Computing and Data Engineering. 2020, https://doi.org/10.1145/3379247.3379264

Thank You