**“Webcam Controlling and Object**

**Detection”**

**Project Report**

OF MINI PROJECT

**BACHELOR OF TECHNOLOGY**

**In**

Computer Science

SUBMITTED BY GUIDED BY

Sumiran Rai Mr. Kamal Soni



**DEPARTMENT OF COMPUTER SCIENCE**

**BBD Engineering College**

(2021-22)

**ABSTRACT**

Object detection is one of the most basic and central task in computer vision. Its task is **to find all the interested objects in the image**, and determine the category and location of the objects. Object detection is widely used and has strong practical value and research prospects.

The main purpose of object detection is **to identify and locate one or more effective targets from still image or video data**. It comprehensively includes a variety of important techniques, such as image processing, pattern recognition, artificial intelligence and machine learning.

**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Title** | **Page**  **no.** |
| 1 | INTRODUCTION | 3-8 |
| 2 | OBJECTIVES | 9 |
| 3 | PROJECT CATEGORY | 10 |
| 4 | TOOL/PLATEFORMS | 11 |
| 5 | **Object Detection: Then and Now** | 12-13 |
| 6 | METHODOLOGY | 14-19 |
| 7 | MODULES | 20 |
| 8 | LIMITIONS | 21 |
| 9 | FEASABILITY | 22 |
| 10 | CONCLUSION | 23 |
| 11 | FUTURE SCOPE | 24 |
| 12 | REFRENCES | 25 |

**INTRODUCTION**

Object detection is one of the most basic and central task in computer vision. Its task is to find all the interested objects in the image, and determine the category and location of the objects. Object detection is widely used and has strong practical value and research prospects.

The main purpose of object detection is **to identify and locate one or more effective targets from still image or video data**. It comprehensively includes a variety of important techniques, such as image processing, pattern recognition, artificial intelligence and machine learning.

**OBJECTIVES**

This project is based on the python; the main objective of this project is to identify and locate one or more effective targets from still image or video data.

The other objectives of this project are as follows-

1. **In a tracking.**
2. **Image Recognition.**
3. **Image Segmentation.**
4. **Easy operations for operator of the system.**
5. **No paper work requirement.**
6. **Helps in understanding and analyze scenes in images.**

**PROJECT CATEGORY**

This project as title **“Object Detection Webcam”** comes under the **Python language.** This application is developed with the help of Python 3.8.8 shell. This application is cross platform compatibility and can also be run on the any Python interpreter above 3.6.

**Tools/platform**

This project is developed using the tools, which are most suited for development of the application.

These tools are as follows: -

**Hardware and software requirement:**

**Hardware:-**

Processor : i3 and above

Processor speed : 1.1Hz-2.6Hz (clock rate)

Hard Disk space : 40 GB

RAM Memory : up to 50 GB

**Software:-**

Operating system :Windows 07, 08, 10 or above.

Coding language :Python

Software tool :Open CV

Text Editor :VScode,Atom,SublimeText

### ****Object Detection: Then and Now****

### The first deep neural network for object detection was Overfeat. They introduced a multi-scale sliding window approach using CNNs and showed that object detection also improved image classification. They were shortly followed by R-CNN: Regions with CNN features.

### The authors proposed a model that used selective-search for generating region proposals by merging similar pixels into regions. Each region was fed into a CNN, which produced a high dimensional feature vector. This vector was then used for the final classification and bounding box regression.

It outperformed the Overfeat network by a large margin but was also very slow, because the proposal generation using selective-search was very time-intensive, as well as the need to feed every single proposal through a CNN. A more sophisticated approach, Fast R-CNN, also generated region proposals with selective-search but fed the whole image through a CNN.

The region proposals were pooled directly on the feature map by ROI pooling. The pooled feature vectors were fed into a fully connected network for classification and regression. Similar to R-CNN, Fast R-CNN generated the region proposals with selective-search.

Object detection combines the tasks of object classification and localization. Current object detectors can be divided into two categories: Networks separating the tasks of determining the location of objects and their classification, where Faster R-CNN is one of the most famous ones, and networks which predict bounding boxes and class scores at once, with the YOLO, OPENCV and SSD networks being famous architectures.

**Methodology**

We are using many technology in our project which are given below:

**Python programing:** Python is an interpreter, object-oriented, high-level programming language with dynamic semantics.

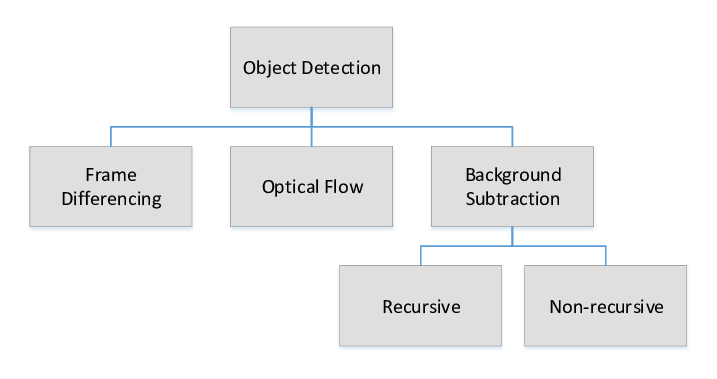
Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general purpose language, meaning it can be used to create a variety of different programs and isn't specialized for any specific problems.

* **Open cv:** OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products.

OpenCV is a huge open-source library for computer vision, machine learning, and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human.

It, can process images and videos to identify objects, faces, or even handwriting of a human. It plays a major role in real-time operation which is very important in today's systems.

**J**



**Modules**

This project includes the following module for the

development of this project. These are as follows:

* **Os-sys:** The OS module in python **provides functions for interacting with the operating system**. OS, comes under Python's standard utility modules. This module provides a portable way of using operating system dependent functionality. os.system() method execute the command (a string) in a subshell.
* The OS module in Python **provides functions for creating and removing a directory (folder)**, fetching its contents, changing and identifying the current directory, etc. You first need to import the os module to interact with the underlying operating system.
* **Open cv:** OpenCV is a huge open-source library for computer vision, machine learning, and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human.
* **Tkinter:** Tkinter is **Python's de-facto standard GUI (Graphical User Interface) package**. It is a thin object-oriented layer on top of Tcl/Tk. Tkinter is not the only GuiProgramming toolkit for Python. It is however the most commonly used one. ... Graphical User Interfaces with Tk, a chapter from the Python Documentation

**Limitations**

The field of computer vision has experienced substantial progress recently, owing largely to advances in deep learning, specifically convolutional neural nets (CNNs). Image classification, where a computer classifies or assigns labels to an image based on its content, can often see great results simply by leveraging pre-trained neural nets and fine-tuning the last few throughput layers.

Classifying and finding an unknown number of individual objects within an image, however, was considered an extremely difficult problem only a few years ago. This task, called object detection, is now feasible and has even been productized by companies like [**Google**](https://cloud.google.com/vision/docs/drag-and-drop) and [**IBM**](https://www.ibm.com/watson/services/visual-recognition/). But all of this progress wasn’t easy! Object detection presents many sizable challenges beyond what is required for image classification. After a brief introduction to the topic, let’s take a deep dive into several of the interesting obstacles these problems pose along with various emerging solutions.

# 1. Dual priorities: object classification and localization

The first major complication of object detection is its added goal: not only do we want to classify image objects but also to determine the objects’ positions, generally referred to as the object localization task. To address this issue, researchers most often use a multi-task loss function to penalize both misclassifications and localization errors.

# 2. Speed for real-time detection

Object detection algorithms need to not only accurately classify and localize important objects, they also need to be incredibly fast at prediction time to meet the real-time demands of video processing. Several key enhancements over the years have boosted the speed of these algorithms, improving test time from the 0.02 frames per second (fps) of R-CNN to the impressive 155 fps of Fast YOLO.

# 3. Multiple spatial scales and aspect ratios

For many applications of object detection, items of interest may appear in a wide range of sizes and aspect ratios. Practitioners leverage several techniques to ensure detection algorithms are able to capture objects at multiple scales and views.

# 4. Limited data

The limited amount of annotated data currently available for object detection proves to be another substantial hurdle. Object detection datasets typically contain ground truth examples for about a dozen to a hundred classes of objects, while image classification datasets can include upwards of 100,000 classes. Furthermore, crowdsourcing often produces image classification tags for free (for example, by parsing the text of user-provided photo captions). Gathering ground truth labels along with accurate bounding boxes for object detection, however, remains incredibly tedious work.

# 5. Class imbalance

Class imbalance proves to be an issue for most classification problems, and object detection is no exception. Consider a typical photograph. More likely than not, the photograph contains a few main objects and the remainder of the image is filled with background. Recall that selective search in R-CNN produces 2,000 candidate RoIs per image–just imagine how many of these regions do not contain objects and are considered negatives!

**Feasibility**

Object Detection has come to the forefront as one of the most important applications of deep learning which is characterized by its learning ability of features and depiction of features compared to the traditional object detection methods. Region proposal algorithms form the basis of which Object Detection networks hypothesize object regions. Algorithms that have cut down on the running time of these detection networks are SPPnet and Fast R-CNN. In this paper we propose a method for object detection through webcam. The method used is a combination of Region Proposal Network (RPN) and Fast R-CNN, where high quality region proposals are generated by training the RPN end to end, which are in turn is used by FastRCNN for detection. These two modules combine to generate an object detection system called Faster R-CNN. Deep models can automatically act as a classifier and detection device, they do not require hand engineered technologies. Therefore, one of the greatest prospects of deep learning technology is in Object Detection. One of the major focus in computer vision technologies is object detection, which has been applied in the pedestrian detection driverless car, video surveillance robotics and counting techniques. The Faster R-CNN models: InceptionV2 and ResNet50 have been used for object detection using webcam, i.e. real time object detection.

**Future scope**

The future of object detection technology is in the process of proving itself, and much like the original Industrial Revolution, it has the **potential to free people from menial jobs that can be done more efficiently and effectively by machines**.

Object detection is a key ability required by most computer and robot vision systems. The latest research on this area has been making great progress in many directions. In the current manuscript, we give an overview of past research on object detection, outline the current main research directions, and discuss open problems and possible future directions.

**Conclusion**

**Object detection is key ability required by most computer and robot vision systems.**

**The latest research on this area has been making great progress in many directions.**

**In the current manuscript, we gave an overview of past research on object detection, outline the current main research directions, and discuss open problems and possible future directions.**

**References**

[1] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, [Focal loss for dense object detection](https://arxiv.org/abs/1708.02002) (2017), arXiv preprint arXiv:1708.02002.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik, [Rich feature hierarchies for accurate object detection and semantic segmentation](https://arxiv.org/pdf/1311.2524.pdf) (2014), IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[3] R. Girshick, [Fast R-CNN](https://arxiv.org/pdf/1504.08083.pdf) (2015), IEEE International Conference on Computer Vision (ICCV).

[4] S. Ren, K. He, R. Girshick, and J. Sun. [Faster r-cnn: Towards real-time object detection with region proposal networks.](https://arxiv.org/pdf/1506.01497.pdf) (2015) Neural Information Processing Systems (NIPS).

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, [You only look once: Unified, real-time object detection](https://arxiv.org/abs/1506.02640) (2015), arXiv preprint arXiv:1506.02640.

[6] J. Redmon and A. Farhadi, [Yolo9000: Better, faster, stronger](https://arxiv.org/abs/1612.08242) (2017), Computer Vision and Pattern Recognition (CVPR).

[7] J. Redmon and A. Farhadi, [Yolov3: An incremental improvement](https://arxiv.org/pdf/1804.02767.pdf) (2018), arXiv.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. E. Reed, [SSD: single shot multibox detector](https://arxiv.org/pdf/1512.02325.pdf) (2015), CoRR.

[9] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, [Feature pyramid networks for object detection](https://arxiv.org/abs/1612.03144) (2017), Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[10] GitHub user Chuangi/darknet/coco.names.

[11] Udemy instructor Ardit Sulce .