OBJECT IDENTIFICATION USING WEBCAM AND OPENCV PYTHON

B. SURYA NARAYANA MURTHY, K KRANTHI KIRAN KUMAR
Assistant Professor in DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS, BHIMAVARAM-534202
Email id: - suryanarayanamurthy.b@gmail.com
PG student, D.N.R. COLLEGE, P.G. COURSES (AUTONOMOUS), BHIMAVARAM-534202.
Email id: - kannajikranthikumar@gmail.com

ABSTRACT

OpenCV object tracking is a widely used technique in the field. A variety of object tracking-specific features are already incorporated into OpenCV. Media and Flow and MIL are some of the object trackers in OpenCV. Tracks the path taken by an object in a movie using an Object Tracking System. We're detecting objects in movies and webcam images with Python and the OPENCV module in this project. "Browse system videos" and "Start webcam video tracking" are two modules included in this application. Working with methods such as frame differencing, color-space transformation, background separation, optical flow, and a classifier based on the Haar cascade, the project entails implementing numerous object recognition and tracking techniques in video. These approaches include: 1. Besides these methods, a widely used and highly effective edge detection method is also used. Python is used for all of the implementations. The results are extensive, and they are thoroughly evaluated. OpenCV and python are two of the most often used terms.

1 INTRODUCTION

Tracking an object when there is a lot of variation is extremely difficult. Background movement, partial and complete occlusions of complex-shaped objects, and varying degrees of illumination One of the most crucial applications for industries to ease the user, save time and achieve parallelism is object detection and localization in digital images. A more efficient and precise method of object detection is still needed in order to attain the desired result.

The primary goal of computer vision research and study is to design a system that reduces human effort by showing the fundamental block diagram of detection and tracking by using a computer. Detection and tracking are implemented in a python environment using SSD and Mobile Nets-based techniques. Object detection is the process of identifying an object's specific area of interest in a certain type of image. Frame differencing, optical flow, and background subtraction are a few of the ways that can be used. With the use of a camera, this is a way to track down and locate an object in motion. By extracting the properties of images and videos for security applications, detection and tracking methods are explained .

Object tracking, using video sensing technique, is one of the major areas of research due to its increased commercial applications such as surveillance systems, Mobile Robots, Medical therapy, security systems and driver assistance systems. Object tracking, by definition, is to track an object (or multiple objects) over a sequence of images. Tracking is usually performed on higher-level applications that require the location and object in every frame.

The most popular application in this area is vision-based surveillance, to help understand the movement patterns of people with suspicious actions. Traffic scene analysis is also a well-known application, to get the tracking information for keeping the vehicles in lane and preventing the accidents. Thus, object detection and tracking under dynamic conditions is still a challenge for real-time performance which requires the computational complexity to be minimum. Various methods for object detection have been proposed; such as feature-based, template-based object detection and background subtraction [1]. But selection of the best technique for a specific application is relative and dependent upon the hardware resources and scope of the application. Feature-based detection searches for corresponding features in successive frames, including Harris corner, edges, SIFT, contours or colour pixels.

Background subtraction is a popular method which uses static background and calculating the difference between the hypotheses background and the current image. This approach is fast and good for fixed background but it cannot deal with

the dynamic environment, with different illumination and motions of small objectsThe goal of tracking is to establish a correspondence between the detected target objects of images over frames.

Tracking using mean shift kernel is also introduced. This method performs well when there is occlusion, which can be solved using templates [2]. Camshift (Continously Adaptive Meanshift) can track a single object fast and robust using color features, but it is ineffective for occlusion. There is also research on appearance-based object detection [3]. It uses whole 2-D images to perform tracking for navigation in faster time. However, this kind of approach requires several templates and does not work when the target object, color or perspective view is changed.

2. LITERATURE SURVEY AND RELATED WORK:

Single Shot Multi Box Detector: A single deep neural network is used to recognize objects in photos. The output space of bounding boxes is discretized into a series of default boxes with varied aspect ratios and scaling depending on the feature map location. We call this approach SSD. At prediction time, the network generates scores for the existence of each item category in each default box and creates updates to the box in order to better match the object's shape. To accommodate objects of varying sizes, the network uses numerous feature maps with varied resolutions to forecast their location.

Methods that require a proposal generation stage and subsequent resampling of pixels or features are more complicated than our SSD model, which does not require any proposal generation or resampling. A detecting component is not required; therefore, SSD is a snap to learn and integrate into systems. SSD's performance on the PASCAL VOC, MS COCO, and ILSVRC datasets is comparable to those of approaches that use an additional object proposal phase and is substantially faster, while providing a unified framework for both training and inference. SSD's accuracy is superior to that of other single-stage approaches, even when the input image size is less.

With 300x300 input, SSD is capable of attaining 72.1% MAP on the VOC2007 test, while with 500x500 input, SSD is capable of attaining 75.1% MAP, exceeding the Faster RCNN model. Convolutional Neural Networks for Mobile Vision: 2.2 MobileNets. For mobile and embedded vision applications, we offer a new class of models termed MobileNets.

In order to construct lightweight deep neural networks, MobileNets rely on a simplified architecture that makes extensive use of depth wise separable convolutions. It is our goal to introduce two basic global hyper-parameters that efficiently balance off latency versus precision. Using these hyper-parameters, the model builder can pick the appropriate model size based on the constraints of the task. Extensive studies on resource and accuracy trade-offs are presented, and we demonstrate high performance on ImageNet classification when compared to other popular models.

3 EXISTING SYSTEM

Real-time object detection and tracking is a vast, vibrant yet inconclusive and complex area of computer vision. Due, or multi-camera multi-objects tracking make this task strenuously difficult. Though, many to its increased utilization in surveillance, tracking system used in security and many others applications have propelled researchers to continuously devise more efficient and competitive algorithms. However, problems emerge in implementing object detection and tracking in real-time; such as tracking under dynamic environment, expensive computation to fit the real-time performance methods and techniques have been developed.

DISADVANTAGES OF EXISTING SYSTEM

- Problems emerge in implementing object detection and tracking in real-time and tracking under dynamic environment.
- Expensive computation to fit the real-time performance methods and techniques have been developed.

4 PROPOSED WORK AND ALGORITHM

We're detecting objects in movies and webcam images with Python and the OPENCV module in this project. "Browse system videos" and "Start webcam video tracking" are two modules included in this application. Allows the user to upload a video from his system and the application will begin to play that video; if the application detects any objects, those objects will be marked with bounding boxes. If the user wishes to stop tracking the video while it is playing, he or she can press the "q" key on their keyboard to do so. In order to use this module, the program must first connect to the built-in system webcam and

begin streaming video; if the application detects an object during this process, the application will surround that object with bounding boxes. To stop the webcam streaming, simply hit 'q'.

5. METHODOLOGIES

MODULES

DATASET

This paper utilizes the dataset provided by revolution analytics for the detection of the fraudulent credit card transaction from Kaggle. Dataset has 51149 legal transactions and 3312 fraudulent transactions.The dataset is divided as 60%, 20% and, 20% in the Train, Valid and Test set, respectively

DATA PREPROCESSING

For efficient implementation of the classification algorithm, data preprocessing is performed before feature selection. Under-sampling is performed to make the dataset balanced to avoid the biasing of the classification algorithm towards the majority class. Feature Selection is implemented on a balanced dataset.

FEATURE SELECTION

Feature selection methods are used to remove unnecessary, irrelevant, and redundant attributes from a dataset that do not contribute to the accuracy of a predictive model or which might reduce the accuracy of the model. In this paper seven feature selection techniques namely Select-K-best, Feature Importance, Extra tress classifier, Person's correlation, Mutual Information, Step forward selection and Recursive feature elimination are used

FEATURE IMPORTANCE

Feature importance is a class of techniques for assigning scores to input features to a predictive model that indicates the relative importance of each feature at the time of making a prediction. It reduces the number of input features. In this paper, feature importance is implemented using an extra tree classifier from the decision tree. Extra Trees is similar to Random Forest, it builds multiple trees and splits nodes using random subsets of features, but unlike Random Forest, Extra Tree samples without replacement and nodes are split on random

6 RESULTS AND DISCUSSION

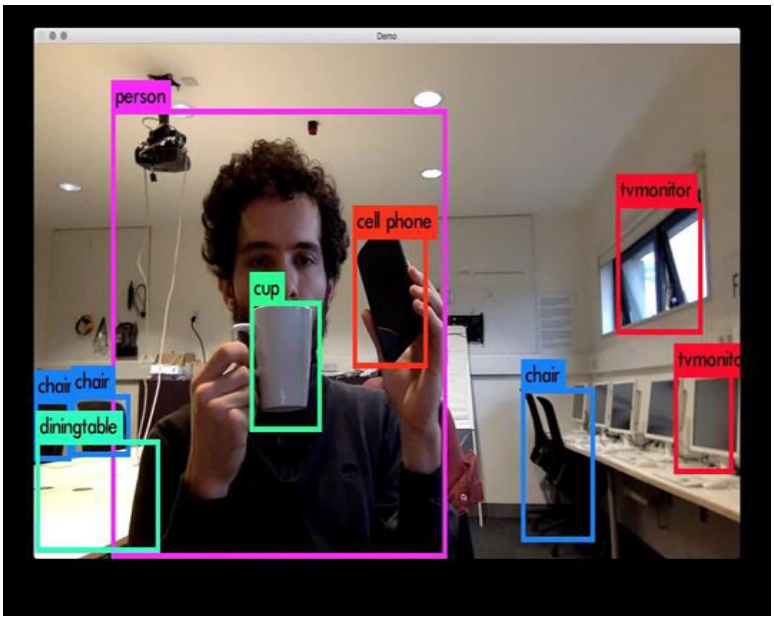

FIG1 : CAPTURE IMAGE

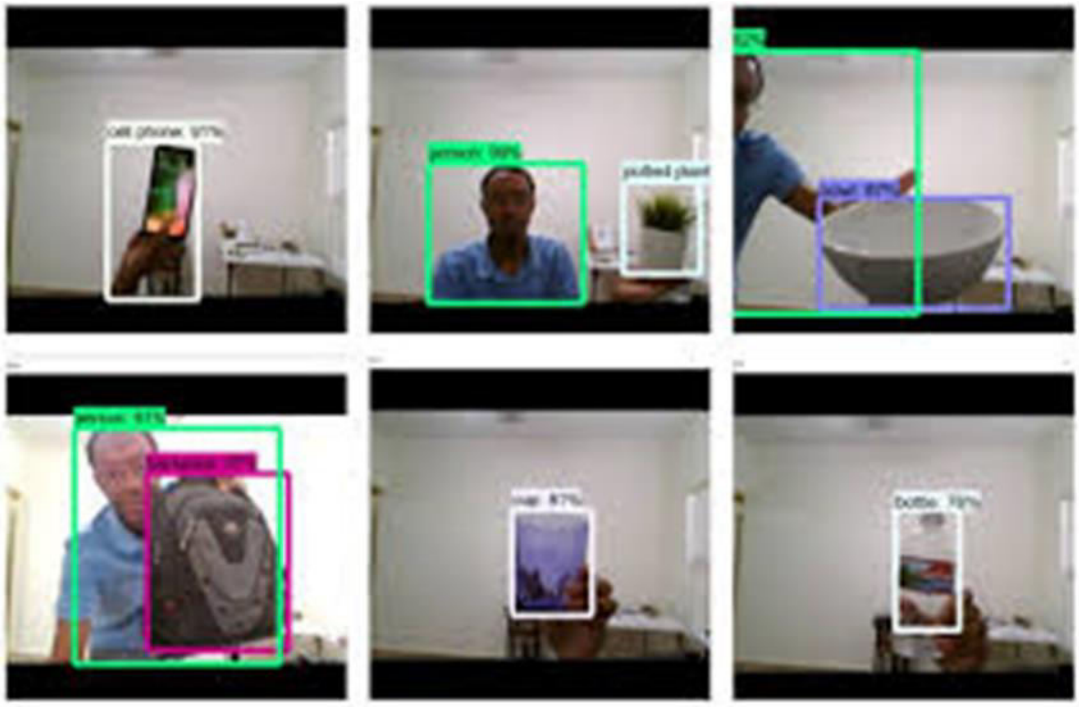

FIG 2 DETECTING OBJECT

6.CONCLUSION AND FUTURE SCOPE

CONCLUSION

The current state-of-the-art object detection technology has been replaced by a new system that is both accurate and efficient. OpenCV is used for object detection in this project, which employs Python and the OPENCV package. In this literature review, we have analyzed five methods for real time object detection and tracking, in terms of accuracy, 152 computational time and memory consumption. The success of a video tracking algorithm depends upon the quality of its response to a high frame- rate input video. The higher the frame-rate is, more difficult it becomes for an algorithm to produce accurate results.

Based on the simulation results, we recommend the 'Kanade-Lucas' algorithm for the hardware (real-time) implementation in mobile robots. The 'Kanade Lucas' algorithm is the fastest, consumes the least memory and the most accurate with no implementation complexities. It gives high accuracy in the scenarios where the distortion is very high. This algorithm is one of the standards in motion detection. Because of its iterative nature, it provides a good support for the video sequences having high frame rates. For example, an algorithm that tracks object by identifying a specific feature may work very efficiently in one scenario but in case of video containing different objects with similar features, it may fail completely.

FUTURE SCOPE:

It's possible to make the visual tracking method provided here more robust by removing some of the following limitations: When using Single Visual tracking, the template's size remains constant. When the size of the monitored object shrinks with time, the background takes on more importance. The object may not be able to be tracked in this situation. There can be no tracking or new object consideration for an object that is completely obscured. Multi-view tracking with many cameras may be used in the future to automate the system and overcome the constraints listed above. In comparison to single view tracking, multi view tracking has an evident advantage due to its ability to cover a much larger area with a wide range of viewing angles for the tracked objects.

7 REFERENCES

1. Wei Liu and Alexander C. Berg, SSD: Single Shot MultiBox Detector, Google Inc., Dec 2016.

2. Andrew G. Howard, and Hartwig Adam, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, Google Inc., 17 Apr 2017.

3. Justin Lai, Sydney Maples, Ammunition Detection: Developing a Real-Time Gun Detection Classifier, Stanford University, Feb 2017

4. Shreyamsh Kamate, UAV: Application of Object Detection and Tracking Techniques for Unmanned Aerial Vehicles, Texas A&M University, 2015.

5. Adrian Rosebrock, Object detection with deep learning and OpenCV, pyimagesearch.

6. Mohana and H. V. R. Aradhya, "Elegant and efficient algorithms for real time object detection, counting and classification for video surveillance applications from single fixed camera," 2016 International Conference on Circuits, Controls, Communications and Computing (I4C), Bangalore, 2016, pp. 1-7.

7. Akshay Mangawati, Mohana, Mohammed Leesan, H. V. Ravish Aradhya, Object Tracking Algorithms for video surveillance applications International conference on communication and signal processing (ICCSP), India, 2018, pp. 0676-0680.

8. Apoorva Raghunandan, Mohana, Pakala Raghav and H. V. Ravish Aradhya, Object Detection Algorithms for video surveillance applications International conference on communication and signal processing (ICCSP), India, 2018, pp. 0570-0575.