International Journal of
Image Processing and Pattern Recognition

ISSN: 2456-6985
Volume 7, Issue 2, 2021
DOI (Journal): 10.37628/IJIPPR

http://computers.journalspub.info/index.php?journal=JIPPR&page=index

Review IJIPPR

Recognition and Detection of Content in Video Using OpenCV

Akshit Rawat*

Abstract

The emergence and continued reliance on the Internet and related technologies has resulted in massive amounts of data that can be analysed. Humans, on the other hand, do not have the cognitive abilities to comprehend such vast amounts of data. Machine learning (ML) is a mechanism that enables humans to process large amounts of data, gain insights into the data's behaviour, and make more informed decisions based on the analysis's results. ML has a wide range of applications, such as efficient and accurate object detection, and has been a hot topic in the advancement of computer vision systems. Since the introduction of deep learning techniques, the accuracy of object detection has increased dramatically. The project intends to incorporate cutting-edge object detection techniques with the goal of achieving high accuracy with real-time performance. The reliance on other computer vision techniques to assist the deep learning-based approach is a major challenge in many object detection systems, resulting in slow and suboptimal performance. The resulting system is fast and accurate, making it useful for applications that require object detection.

Keywords: Recognition, Detection, OpenCV, E-R, DFD

INTRODUCTION

Detection

Face detection has gotten a lot of attention in recent years due to its applications in computer and human interaction. Image processing includes face detection as a subset. Image processing is primarily used to compress, improve, or extract valuable information from images. Facial recognition technology can identify single or multiple faces in a photograph, removing distracting background noise. A face recognition algorithm must basically divide images into two groups based on whether or not they contain a face. The goal of the face detection algorithm is to thoroughly examine the image, identify the presence of faces in the image, and remove the background from the image. Face detection errors are divided into two categories: false negatives and false positives. When a face is recognised, a false positive occurs. When the algorithm rejects the existence of anything in the image, this is referred to as a false negative. The detection rate is the ratio of the number of faces correctly detected by the system to the number of faces identified by humans. The face detection algorithm's detection rate should be as high as possible [1–3].

***Author for Correspondence**
Akshit Rawat
E-mail: akshit@openlogicsys.com

Student, Department of Computer Science & Engineering, Dr. M.C. Saxena College of Engineering & Technology, Lucknow, Uttar Pradesh, India

Received Date: December 22, 2021
Accepted Date: December 28, 2021
Published Date: December 31, 2021

Citation: Akshit Rawat. Recognition and Detection of Content in Video Using OpenCV. International Journal of Image Processing and Pattern Recognition. 2021; 7(2): 42–47p.

Recognition

Facial recognition is the most advanced and quick biometric technology in the world. It makes use of the most visible human body component, the face, in a non-intrusive way. According to global data, most people are unaware of the face recognition process that is taking place on them, making it one of the least invasive procedures with the least amount of delay. The facial recognition algorithm examines the numerous features of a face in the

input image. This biometric has been widely, and perhaps exaggeratedly, lauded as a great method for identifying potential threats such as terrorists, scam artists, and so on, but it has yet to gain widespread acceptance in high-level use. Biometric face recognition technology is expected to overtake fingerprint biometrics as the most commonly used method of identification and authentication [4], [5].

OpenCV Library

It is a massive open-source library for image processing, machine learning, and computer vision. OpenCV works with a wide range of programming languages, including Python, C++, and Java. It will analyse photos and videos to identify artefacts, faces, and even human handwriting. When combined with many other libraries, such as Numpy, a high-performance library for turning machines, OpenCV achieves good performance; that is, all services that can be performed in Numpy can also be integrated with OpenCV. It is written in C++ and has a C++ interface as its main interface, but it also has an older Language training that is less robust but still detailed. The C++ GUI displays the most recent technologies and algorithms. There are Python, Java, and MATLAB/OCTAVE bindings available Wrappers in a variety of programming languages have been developed to promote wider acceptance. In version 3.4, JavaScript plugins for a variant of OpenCV functions are published as OpenCV.js, which can be used on web platforms. The OpenCV project, which was officially launched in 1999, began as Intel's research programme to support CPU-intensive applications [6], [7]. OpenCV is a popular platform for developing face detection and recognition algorithms.

Existing System

- The existing system performs poorly when detecting multiple moving objects. The proposed system is intended for real-time tracking of multiple moving objects.
- Before a decade, many problems in computer vision had reached saturation in terms of accuracy.
- However, as deep learning techniques became more popular, the accuracy of these problems improved tremendously.
- One of the major issues was image classification, which is defined as predicting the image's class.

Need for New System

- Works effectively with mathematical expressions involving multidimensional arrays.
- Strong backing for deep neural networks and machine learning concepts.
- GPU/CPU computing, in which the same code can run on both architectures.
- High computational scalability across machines and large data sets.
- The TensorFlow Object Detection API (as shown in Figure 1) is a TensorFlow-based open source framework that makes it simple to build, train, and deploy object detection models.

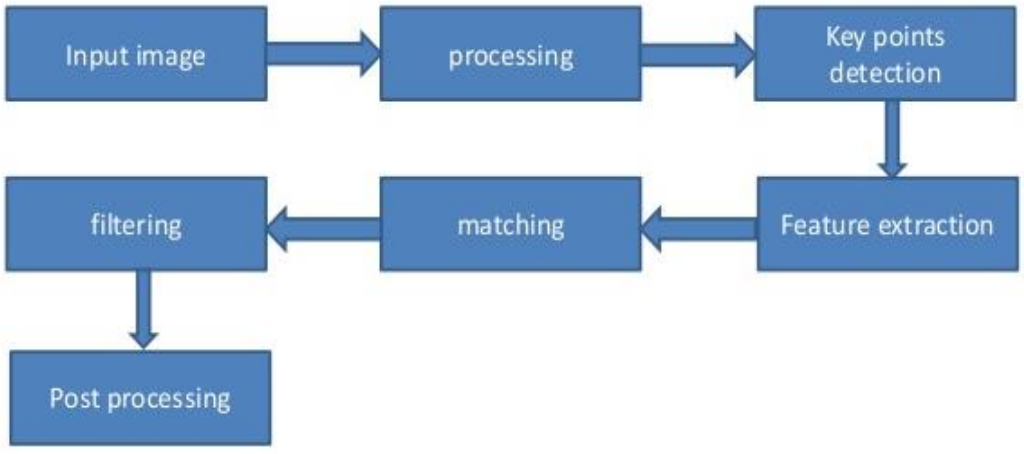

Figure 1. Block diagram tensorflow object detection API.

Applications

Face detection is a well-known application of object detection that is found in almost all mobile cameras. In autonomous driving, a more generalised (multi-class) application can be used to detect a variety of objects (shown in Figure 2). It also plays an important role in surveillance systems. These systems can be combined with other tasks, such as pose estimation, in which the first stage of the pipeline detects the object and the second stage estimates the pose in the detected region. It can be used to track objects and thus has applications in robotics and medicine. As a result, this problem has a wide range of applications as shown in Table 1 [8].

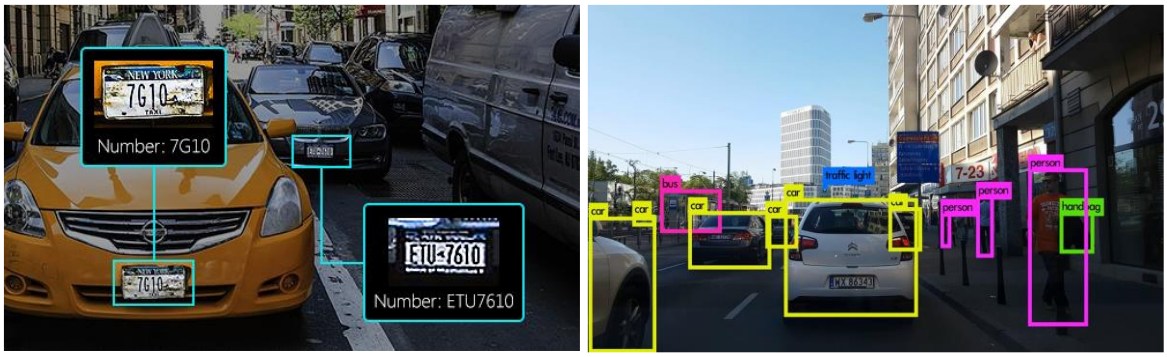

Figure 2. Applications of object detections, (a) Surveillance, (b) Autonomous vehicles

Table 1. (a & b) Object detector data.

(a) User details.

Table Name	user_detector
Description	This table provide the information about the detected image
Primary Key	Reg_id
Foreign Key	-

(b) User data and description.

S.N.	Field Name	Data type(Size)	Constrains	Description
1	reg_id	Int	Primary Key	It stores image register id
2	picture_id	Int	Not Null	It stores picture id
3	Image_path	Varchar(10)	Not Null	It stores path of the image
4	result_id	Int	Not Null	It stores output result

TECHNICAL CONCEPT

Hardware Requirements

The specifications of a hardware device are known as hardware requirements. Most hardware only has requirements for the operating system or compatibility. These specifications include a minimum processor speed, memory, and disc space as shown in Table 2.

Table 2. Shows hardware and its requirements.

S.N.	Hardware	Requirement
1.	Processor	i3 or higher, AMD Fx or higher
2.	RAM	4GB or higher
3.	HDD	1 TB
4.	Monitor	15 inch
5.	Keyboard	108 Key Normal Keyboards
6.	Mouse	3 Button Normal Mouse
7.	Others	Webcam, High speed Internet

Software Requirements

The software requirements are a description of the target system's features and functionalities as described in Table 3.

Table 3. Different software requirements.

S.N.	Software	Minimum System Requirement
1.	Interpreter	Python 3.6 or higher
2.	Database	SQLite
3.	IDE	VS code,Spyder or Pycharm
4.	Libraries	Numpy,Scipy,Pandas,jupyter,tensorflow,etc.,
5.	Operating System	Windows 10 or Linux or Mac os
6.	Documentation Tool	MS Word, MS Power Point

Feasibility Study

A feasibility study evaluates the viability of a proposed project. A feasibility study seeks to objectively and rationally uncover the strengths and weaknesses of an existing business or proposed venture, as well as the opportunities and threats that exist in the environment, the resources needed to carry out the project, and, ultimately, the prospects for success.

Technical Feasibility

This evaluation is based on an outline design of system requirements and is used to determine whether the company has the technical expertise to complete the project.

The technical feasibility assessment focuses on gaining an understanding of the organization's current technical resources and their applicability to the anticipated needs of the proposed system. It is an assessment of the hardware and software and how well it meets the requirements of the proposed system.

Economic Feasibility

The goal of the economic feasibility assessment is to determine the positive economic benefits that the proposed system will provide to the organisation. It includes the quantification and identification of all expected benefits. This evaluation usually includes a cost/benefit analysis.

Legal Feasibility

Determines whether the proposed system is in conflict with legal requirements, such as whether a data processing system must comply with local data protection regulations, and whether the proposed venture is acceptable in accordance with digital data laws [9].

MODULE DESCRIPTION

Features

Camera Management: A system that allows you to control camera or webcam-related code.

AI Training System: This section contains neural network training for the preparation of various models.

Model Deployment System: Enables the loading of Trained Models for Object Detection.

Management system: Enables us to write coding logic for dynamic table creation [10, 4].

Use case Diagram

A use case diagram is a graphical representation of how a user might interact with a system. A use case diagram depicts the various use cases and user types that exist in the system and is frequently accompanied by other types of diagrams shown in Figure 3 and Table 4.

Data Flow Diagram

A data-flow diagram depicts the flow of data through a process or system. The DFD also contains information about each entity's and the process's outputs and inputs. There is no control flow in a data-flow diagram; there are no decision rules or loops as described in Figure 4.

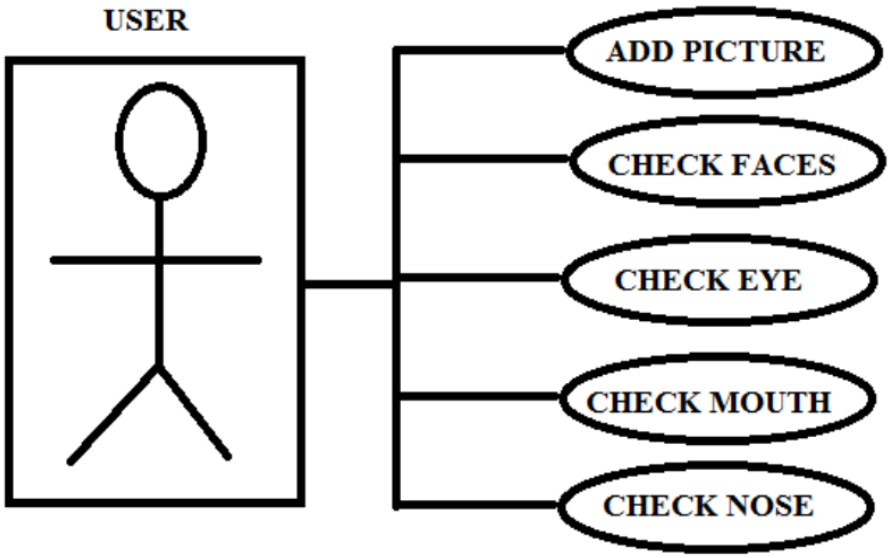

Figure 3. Use case diagram of object detection.

Table 4. User detail detection.
(a) User data.

Table Name	user_data
Description	This table provide the information about the user
Primary Key	user_id
Foreign Key	-

(b) User description.

S.N.	Field Name	Data type(Size)	Constrains	Description
1	user_id	Int	Primary Key	It stores User id
2	face_id	Int	Not Null	It stores Face id
3	name	Varchar(10)	Not Null	It stores User name
4	check_info	Varchar(10)	Not Null	It stores User Information
5	picture_id	Int	Not Null	It stores User picture id

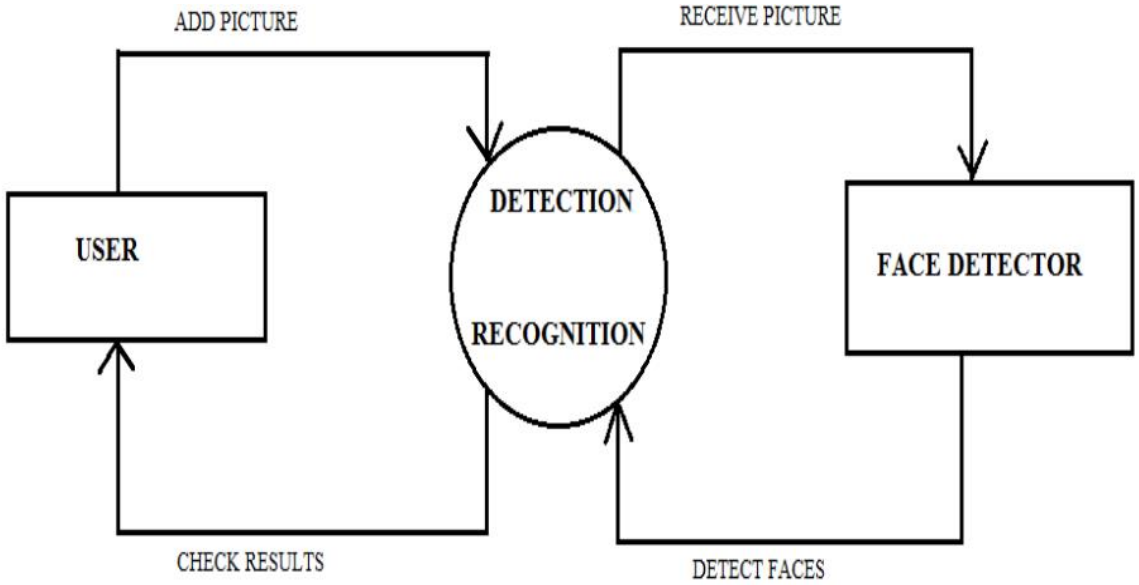

Figure 4. Data flow diagram.

E-R Diagram

Entity Relationship Diagram (ER Diagram), also known as ERD, is a diagram that depicts the relationship of entity sets stored in a database. ER Diagrams use a variety of symbols, including rectangles to represent entities, ovals to define attributes, and diamond shapes to represent relationships. E-R Diagram is shown in Figure 5.

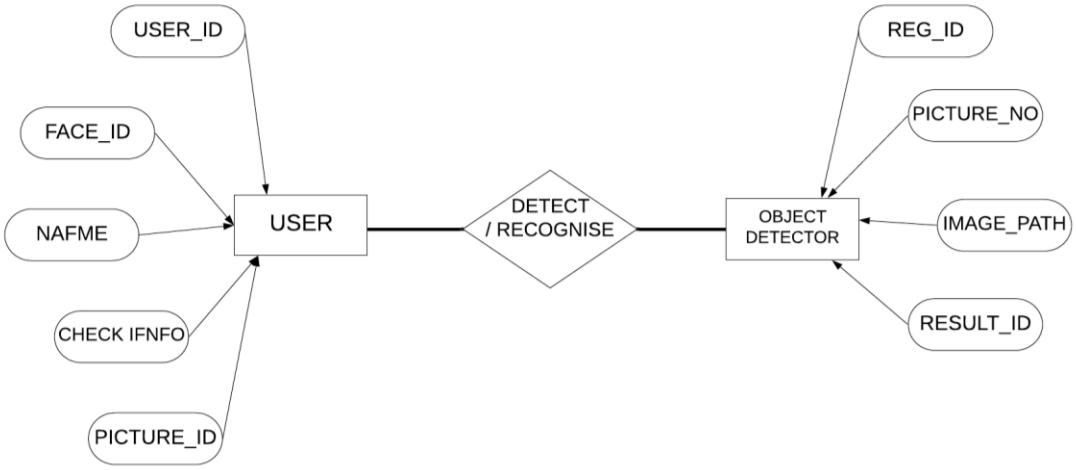

Figure 5. E-R diagram.

CONCLUSION

This project makes use of cutting-edge computer vision and deep learning techniques. This can be used in real-time applications that require object detection as part of their pipeline pre-processing. Training the system on a video sequence for use in tracking applications would be a significant scope. The addition of a temporally consistent network would enable smooth detection, which would be preferable to per-frame detection.

REFERENCES

1. Joshi P, Escrivá DM, Godoy V. OpenCV By Example. Packt Publishing Ltd; 2016.
2. Khan M, Chakraborty S, Astya R, Khepra S. Face Detection and Recognition Using OpenCV. In2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) 2019:116-119. IEEE.
3. Culjak I, Abram D, Pribanic T, Dzapo H, Cifrek M. A brief introduction to OpenCV. In2012 proceedings of the 35th international convention MIPRO 2012:1725-1730. IEEE.
4. Gornale SS, Babaleshwar AK, Yannawar PL. Analysis and detection of content based video retrieval. Int. J. Image, Graph. Signal Process. 2019;11(3):43.
5. Harini V, Prahelika V, Sneka I, Adlene Ebenezer P. Hand gesture recognition using OpenCv and Python. InInternational Conference On Computational Vision and Bio Inspired Computing 2018:1711-1719. Springer, Cham.
6. Howse J, Joshi P, Beyeler M. Opencv: computer vision projects with python. Packt Publishing Ltd; 2016 Oct 24.
7. Gollapudi S. Object Detection and Recognition. In Learn Computer Vision Using OpenCV 2019:97-117. Apress, Berkeley, CA.
8. Brostow GJ, Fauqueur J, Cipolla R. Semantic object classes in video: A high-definition ground truth database. Pattern Recognition Letters. 2009 Jan 15;30(2):88-97.
9. Brostow GJ, Fauqueur J, Cipolla R. Semantic object classes in video: A high-definition ground truth database. Pattern Recognition Letters. 2009;30(2):88-97.
10. Yu L, Sun W, Wang H, Wang Q, Liu C. The design of single moving object detection and recognition system based on OpenCV. In2018 IEEE International Conference on Mechatronics and Automation (ICMA) 2018:1163-1168. IEEE.

