**Chapter 1**

**Introduction**

Background subtraction is a major preprocessing steps in many vision-based applications. For example, consider the cases like visitor counter where a static camera takes the number of visitors entering or leaving the room, or a traffic camera extracting information about the vehicles etc. In all these cases, first you need to extract the person or vehicles alone. Technically, you need to extract the moving foreground from static background.

Background subtraction is a way of eliminating the background from image. To achieve this, we extract the moving foreground from the static background. Background subtraction is a major pre-processing step in many vision-based applications. For example, consider the cases like visitor counter where a static camera takes the number of visitors entering or leaving the room, or a traffic camera extracting information about the vehicles etc. In all these cases, first you need to extract the person or vehicles alone. Technically, you need to extract the moving foreground from static background.

If you have an image of background alone, like image of the room without visitors, image of the road without vehicles etc., it is an easy job. Just subtract the new image from the background. You get the foreground objects alone. But in most of the cases, you may not have such an image, so we need to extract the background from whatever images we have. It became more complicated when there is shadow of the vehicles. Since shadow is also moving, simple subtraction will mark that also as foreground. It complicates things.

Background Subtraction has several use cases in everyday life, it is being used for object segmentation, security enhancement, pedestrian tracking, counting the number of visitors, number of vehicles in traffic etc. It is able to learn and identify the foreground mask.

Background subtraction (BS) is a common and widely used technique for generating a foreground mask (namely, a binary image containing the pixels belonging to moving objects in the scene) by using static cameras.

As the name suggests, BS calculates the foreground mask performing a subtraction between the current frame and a background model, containing the static part of the scene or, more in general, everything that can be considered as background given the characteristics of the observed scene



Fig 1.1: layering of image

**Chapter 2**

**2.1 Company Profile**

TAKE IT SMART (OPC) PVT.LTD is an Indian based engineering and Software Company headquartered in Bangalore, Karnataka, India. It is both product and service-oriented software company. All offices employ an experienced team of professionals, with an outstanding track record of handling complex web & Apps development projects.

**2.2** **COMPANY STRATERG**

* **Purpose:** To be a leader in the software Industry by providing enhanced services, relationship and profitability.
* **Vision:** To provide quality services that exceeds the expectations of our esteemed customers.

· **Mission:** To build long term relationships with our customers and clients and provide exceptional customer services by pursuing business through innovation and advanced technology.

**2.3 COMPANY SERVICES**

TAKE IT SMART (OPC) PVT.LTD have its own services such as,

* Embedded Applications development
* Web design and development
* IT Service
* Android app Development
* Web Bases Software Solutions
* Web Based Ads Mobile Based Services: Mobile Web Apps

a. Android Apps

b. Windows Apps

c. IOS Apps

d. Cross Plate forms Apps

* Hybrid apps Get trained for industry requirements while you pursuing degree The Different verticals that we operate in are: ¬ Internship & Software Training

**2.4 DOMAINS**

TAKE IT SMART (OPC) PVT.LTD have working with several domains like-

* IT
* Digital marketing

**2.5 Business Address:** **Take It Smart (OPC) Pvt.Ltd**

14,SGN Arcade, 1st Floor, 2nd stage, 1st Main Rd,

RPC Layout, Hoshalli Extension, Stage 1,

Vijayanagar, Bengaluru, Karnataka 560040

**Mobile**: +91-8050104212

**Email**: [info@takeitsmart.in](mailto:info@takeitsmart.in)

**Website**: www.takeitsmart.in

**Chapter 3**

**Work Carried Out at the Company**

**Background Subtraction on Real time Video using OpenCV**

Background subtraction is a way of eliminating the background from image. To achieve this we extract the moving foreground from the static background. Background subtraction is a major pre-processing step in many vision-based applications. For example, consider the cases like visitor counter where a static camera takes the number of visitors entering or leaving the room, or a traffic camera extracting information about the vehicles etc. In all these cases, first you need to extract the person or vehicles alone. Technically, you need to extract the moving foreground from static background.

**3.1 Proposed System:**

* Background modelling consists of two main steps:
  1. Background Initialization;
  2. Background Update.
* In the first step, an initial model of the background is computed, while in the second step that model is updated in order to adapt to possible changes in the scene.

**3.2 Methodology:**

OpenCV is a handy Python library that allows us to do a number of critical operations. One of the functions of this library includes the creation of Background Subtraction. We will cover as much possible in this article but the main things which we will go through are as follows:

* Create Background Subtraction using Real Time Camera
* Define the MOG function
* Adjust the position of the Camera to suit your content
* Press Letter ‘b’ in keyboard to capture Background.

**3.3 System Architecture**

Library OpenCV, NumPy

Input Real Time Camera

Output

python

Background Subtraction Algorithm

Fig 3.1: System Architecture

**3.5 Hardware Requirements:**

* Computer – Processor, high speed is preferred. 32/64 - bit
* RAM 4GB
* Hard disk – Free space of 5GB
* Laptop built-in Camera

**3.5 Software’s Used**

1. Python 3.7
2. Open CV tool
3. OS – Windows 11, 32/64 – bit.
4. Idle Software

**3.6 Libraries Requirement:**

pip install pandas

pip install matplotlib

pip install OpenCV-python

**3.7 Python Code:**

import cv2

import numpy as np

#cap = cv2.VideoCapture("highway.mp4")

cap = cv2.VideoCapture(0)

subtractor = cv2.createBackgroundSubtractorMOG2(history=100, varThreshold=10, detectShadows=True)

while True:

\_, frame = cap.read()

mask = subtractor.apply(frame)

cv2.imshow("Frame", frame)

cv2.imshow("mask", mask)

key = cv2.waitKey(30)

if key == 27:

break

cap.release()

cv2.destroyAllWindows()

print("Project End")

**3.8 Machine learning** (**ML**)

Machine learning is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to do so.

Machine learning algorithms are used in a wide variety of applications, such as email filtering and computer vision, where it is difficult or infeasible to develop conventional algorithms to perform the needed tasks. Machine learning is closely related to computational statistics, which focuses on making predictions using computers. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. In its application across business problems, machine learning is also referred to as predictive analytics. Machine learning approaches are traditionally divided into three broad categories, depending on the nature of the "signal" or "feedback" available to the learning system:

* **Supervised learning:** The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
* **Unsupervised learning:** No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).
* **Reinforcement learning:** A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.

Other approaches have been developed which don't fit neatly into this three-fold categorization, and sometimes more than one is used by the same machine learning system. Single CPU systems, GPUs as well as mobile devices and large scale distributed systems of hundreds of machines.

**3.9 Applications**

* Background Subtraction method can be used in Vehicle counting applications in highway.
* Intruder Detection can be implemented using Background Subtraction Algorithm.

**3.10 Advantages**

* It saves time.
* Efficient and reliable process to monitor changes in privacy conditions

**3.11 Output:**

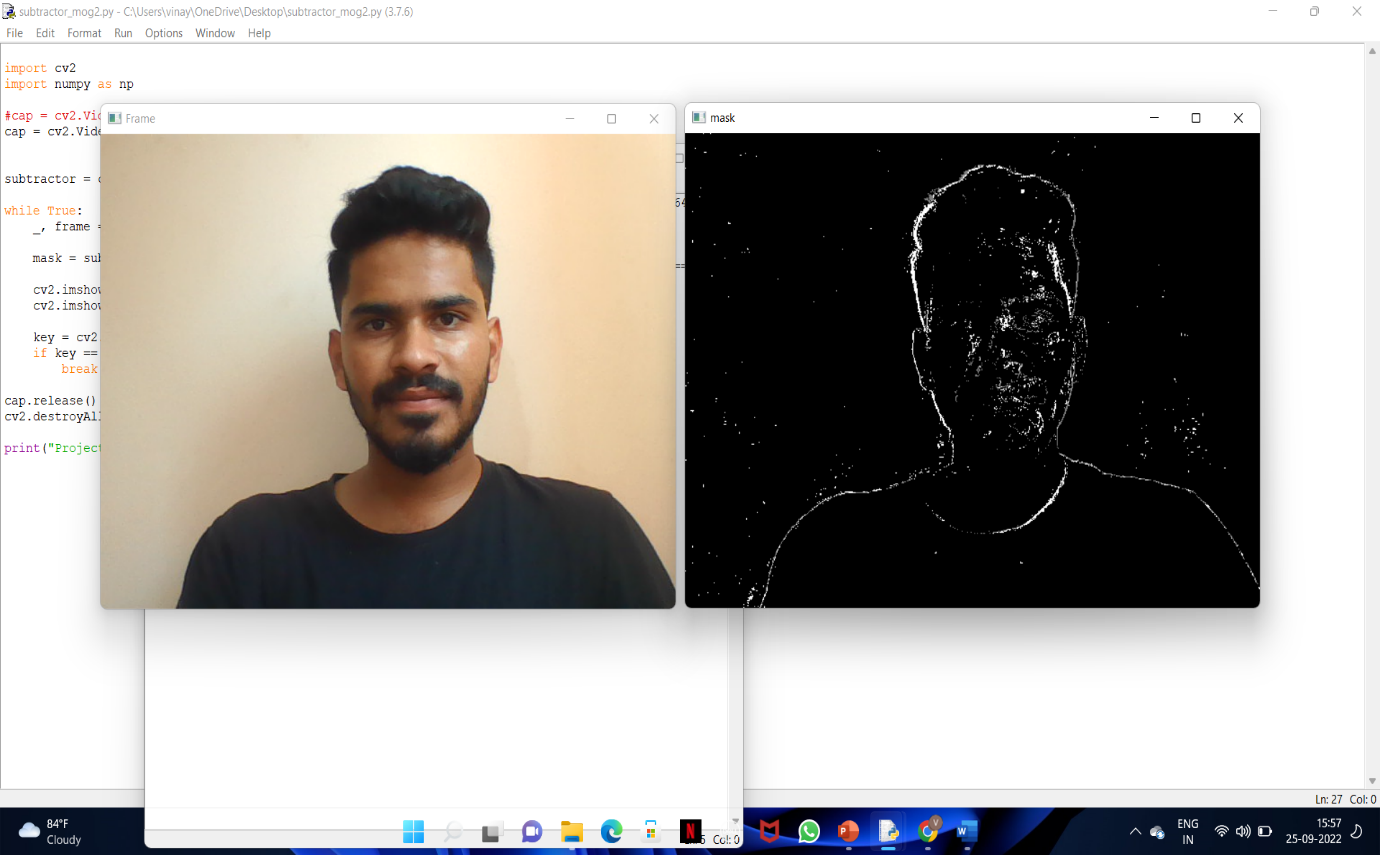
****

Fig 3.2: Output

**Chapter 4**

**CONCLUSION**

A system for Background Subtraction on Real time Video using OpenCV can be developed. Background subtraction is a way of eliminating the background from image. To achieve this, we extract the moving foreground from the static background. Background subtraction is a major pre-processing step in many vision-based applications

In future, this project can be taken to the product level. To make this project as user friendly and durable, we need to make it compact and cost effective. Going further, most of the units can be embedded along with the controller on a single board with change in technology, thereby reducing the size of the system.

**References**

[1] Henry et al., "Anomaly Detection in Videos Recorded by Drones in a Surveillance Context", International Carnahan Conference on Security Technology (ICCST), Madrid, 2017.

[2] Mohammad Farhadi et al., “AAD: Adaptive Anomaly Detection through traffic Surveillance videos”, arXiv:1808.10044, 2018.

[3] Abhiraj Biswas et.al.,“Classification of Objects in Video Records using Neural Network Framework,” International conference on Smart Systems and Inventive Technology (ICSSIT-2018).

[4] S. Loncaric, ‘‘A survey of shape analysis techniques,’’ Pattern Recognit., vol. 31, no. 8, pp. 983–1001, 1998.

[5] Abhiraj Biswas et.al.,“Classification of Objects in Video Records using Neural Network Framework,” International conference on Smart Systems and Inventive Technology (ICSSIT-2018).

**Guide’s Remarks/Comments**

**Signature of the Guide**

**Panel Remarks/Comments**

**Name and Signature of the Panel Head/Panel Member**