Computer Vision System for Tracking and Detecting Human Using Image Processing

A Mini Project Report submitted

in partial fulfilment of

the requirements for the award of the Degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

Submitted By

K.Rakesh:20WJ1A1223

Under the Esteemed Guidance of

Dr. Ravindra Changala

Associate Professor

DEPARTMENT OF INFORMATION TECHNOLOGY

GURU NANAK INSTITUTIONS TECHNICAL CAMPUS (AUTONOMOUS)

(Affiliated to JNTUH)

Ibrahimpatnam, R. R. District, Telangana - 501506.

(2020-2024)

GURU NANAK INSTITUTIONS TECHNICAL CAMPUS

Approved by AICTE - New Delhi | Affiliated to JNTU - Hyderabad | Accredited by National Assessment & Accreditation Council | Courses Accredited by National Board of Accreditation | AUTONOMOUS under Section 2(f) & 12(b) of University Grants Commission Act

Department of Information Technolgy

CERTIFICATE

This is to certify that this minor project report entitled **"COMPUTER VISION SYSTEM FOR TRACKING AND DETECTING THE HUMAN USING IMAGE PROCESSING" by K.Rakesh (20WJ1A1223)** is submitted in partial fulfillment of the requirements for the degree of **Bachelor of Technology** in **Information Technology** of the **Guru Nanak Institutions Technical Campus(Autonomous)** during the academic year 2023-2024, is a Bonafide record of work carried out under our guidance and supervision

Dr. Ravindra Changala	**Dr. M. I. Thariq Hussan**	**Dr. Rishi Sayal**
INTERNAL GUIDE	**HOD IT**	**ASSOCIATE DIRECTOR**

EXTERNAL EXAMINER

Reach Your Aim by Mastering

RAM Innovative Infotech
M : +91 9581 012 012
E : raminnovativeinfotech@gmail.com
Flat No.#309, Amrutha Ville,
Opp: Yashoda Hospital, Somajiguda,
Hyderabad-82, Telangana, India
Www.raminnovativeinfotech.webs.com

PROJECT COMPLETION CERTIFICATE

This is to certify that the following student of final year B.Tech, Department of _____- Guru Nanak Institutions Technical Campus (GNITC) has completed her training and project at GNITC successfully.

STUDENT NAME: **Roll No:**

_____ _____

The training was conducted on _____Technology for the completion of the project titled _____

_____

in _____. The project has been completed in all aspects.

RAM INNOVATIVE INFOTECH
Signature

DEPARTMENT OF INFORMATION TECHNOLOGY

VISION

To be a leading and premier department in the field of Internet of Things by providing competent and highly skilled professionals to cater the needs of the industries and society.

MISSION

M1: To establish an essential environment with state-of-the-art infrastructure and highly qualified faculty for imparting domain knowledge.

M2: To prepare the students with holistic personality by means of appropriate technical and soft skills for solving real world problems.

M3: To enrich and empower student's caliber for the positive societal contribution with emerging technologies.

M4: Extensive partnerships and collaborations with Industries for technology up gradation.

Programme Educational Objectives (PEOs)

PEO 1: To originate the professional engineering practitioners for solving the real-life industry's technological problems using exploratory and analytical skills acquired in the field of Internet of Things (IoT).

PEO 2: To connect the engineering professionals with the ability of critical thinking, analysis and design by using emerging technologies like IoT in technology development, deployment and engineering system implementation.

PEO 3: To provide the technocrats with a pleasant environment for the successful pursuing of their career in advanced education, research and development in professional manner.

Programme Outcomes (POs)

Engineering Graduates will be able to:

PO1: Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/Development of Solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct Investigations of Complex Problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The Engineer and Society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and Sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and Team Work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long Learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

Programme Specific Outcomes (PSOs)

PSO 1: To develop the culture of new emerging technologies like IoT, sensors and controllers in various applications to solve complex challenges in the real world.

PSO 2: To apply standard Internet of things-based concepts and its strategies to develop advanced smart products for skill education, business and research.

ACKNOWLEDGEMENT

I wish to express my sincere thanks to the Vice-Chairman **Sardar G. S. Kohli** of **Guru Nanak Institutions** for providing us all necessary facilities, resources and infrastructure for completing this project work.

I express a whole hearted gratitude to our Managing Director **Dr. H. S. Saini** for providing strong vision in engineering education through accreditation policies under regular training in upgrading education for the faculty members.

I express a whole hearted gratitude to our Director **Dr. K. Venkata Rao** for providing us the constructive platform to launch our ideas in the area of Information Technology and improving our academic standards.

I express a whole hearted gratitude to our Associate Director **Dr. Rishi Sayal** for providing us the conducive environment for carrying through our academic schedules and projects with ease.

I have been truly blessed to have a wonderful advisor **Dr. M. I. Thariq Hussan**, HOD Department of Information Technology and Computer Science & Engineering – Internet of Things for guiding us to explore the ramification of our work and we express our sincere gratitude towards him for leading me throughout the project work.

I specially thank our Project Coordinator**Dr.Ravindra Changala**, Assistant Professor, Department of Computer Science & Engineering – Internet of Things for his valuable suggestions and constant guidance in every stage of the project.

I specially thank our internal guide**. Dr. Ravindra Changala** Associate Professor, Department of Information Technology for his valuable suggestions and constant guidance in every stage of the project.

I express our sincere thanks to all the faculties of Computer Science & Engineering – Internet of Things department who helped me in every stage of our project by providing their valuable suggestions and support.

DECLARATION

I **K.Rakesh(20WJ1A1223)**, hereby declare that the project report entitled **COMPUTER VISION SYSTEM FOR TRACKING AND DETECTING HUMAN USING IMAGE PROCESSING** under the esteemed guidance of **Dr.Ravindra Changala**, Associate Professor, Departmentof Information Technology submitted in partial fulfilment of the requirements for the award of thedegree of Bachelor of Technology in Computer Science & Engineering – Internet of Things. Thisis a record of bonafide work carried out by us and the results embodied in this project report havenot been submitted to any other University or Institute for the award of Degree or Diploma.

Place : GNITC
Date :

K.Rakesh : 20WJ1A1223

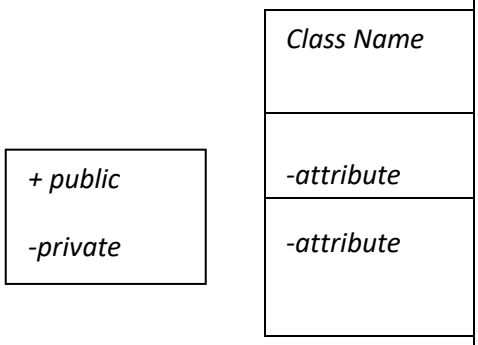
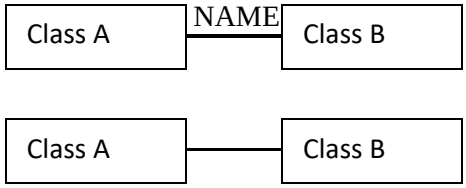
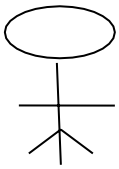
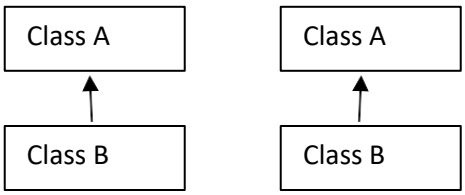
TABLE OF CONTENTS

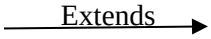
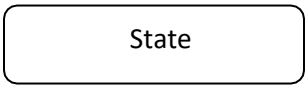
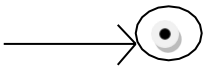
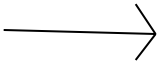
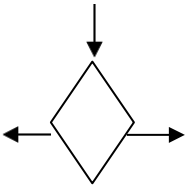
ABSTRACT

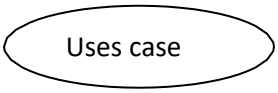
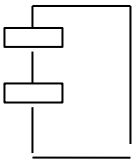
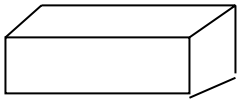
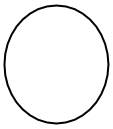
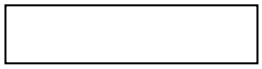
In environments where a camera is installed on a freely moving platform, e.g. a vehicle or a robot, object detection and tracking becomes much more difficult. In this paper, we present a real time system for human detection, tracking, and verification in such challenging environments. To deliver a robust performance, the system integrates several computer vision algorithms to perform its function: a hu-man detection algorithm, an object tracking algorithm, and a motion analysis algorithm. To utilize the available computing resources to the maximum possible extent, each of the system components is designed to work in a separate thread that communicates with the other threads through shared data structures. The focus of this paper is more on the implementation issues than on the algorithmic issues of the system. Object oriented design was adopted to abstract algorithmic details away from the system structure.

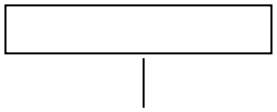
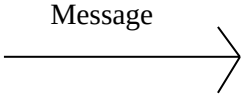
LIST OF FIGURES

LIST OF SYMBOLS

S.NO	NOTATION NAME	NOTATION	DESCRIPTION
1.	Class		Represents a collection of similar entities grouped together.
2.	Association		Associations represents static relationships between classes. Roles represents the way the two classes see each other.
3.	Actor		It aggregates several classes into a single classes.
4.	Aggregation		Interaction between the system and external environment

5.	Relation (uses)	Uses	Used for additional process communication.
6.	Relation (extends)		Extends relationship is used when one use case is similar to another use case but does a bit more.
7.	Communication		Communication between various use cases.
8.	State		State of the process.
9.	Initial State		Initial state of the object
10.	Final state		Final state of the object
11.	Control flow		Represents various control flow between the states.
12.	Decision box		Represents decision making process from a constraint

13.	Use case		Interact ion between the system and external environment.
14.	Component		Represents physical modules which are a collection of components.
15.	Node		Represents physical modules which are a collection of components.
16.	Data Process/State		A circle in DFD represents a state or process which has been triggered due to some event or action.
17.	External entity		Represents external entities such as keyboard, sensors ,etc.
18.	Transition		Represents communication that occurs between processes.

19.	Object Lifeline		Represents the vertical dimensions that the object communications.
20.	Message	Message 	Represents the message exchanged.

LIST OF ABBREVATION

S.NO	ABBREVATION	EXPANSION
1.	DB	Database
2.	JVM	Java Virtual Machine
3.	JSP	Java Server Page
4.	PWS	Personalized Web Search
5.	UPS	User Personalized Search
6.	JRE	Java Runtime Environment

CHAPTER 1
INTRODUCTION

1.1 GENERAL

The multiple object tracking method works on fixed cameras. It starts with an adaptive background modelling module which deals with changing illuminations and does not require objects to be constantly moving. The paper describes implementation of a real-time visual tracking system equipped with an active camera. The system is intended for indoor human motion tracking. Real-time tracking is achieved using simple and fast motion detection procedures based on frame differencing and camera motion compensation. Results of on-line person tracking are presented. Based on preliminary results of object detection in each image which may have missing and/or false detection, the multiple object tracking method keeps a graph structure where it maintains multiply hypotheses about the number aid the trajectories of the object in the video. the image information drives the process of extending and pruning the graph, and determines the hest hypothesis to explain the video. While the image-hued object detection makes a local decision, the tracking process confirm and validates the detection through time, therefore, it can he regards as temporal detection which makes a global decision across time. The multiple object tracking method gives feedbacks which are predictions of object locations to the object detection module. Therefore, the method integrates object detection and trucking tightly. The most possible hypothesis provides the multiple object tracking result. The experimental results are presented.

1.2 OBJECTIVE

Our multiple object tracking method is reliable to deal with occlusions, irregular object motions, changing appearances by postponing the decision of object trajectories until sufficient information is accumulated over time.

1.3.1 Existing System:

Since our focus in this paper is on the implementation issues of our system, a review of technical approaches in video surveillance will not be much relevant. For a globe overview on automated surveillance systems, the reader is referred Readers interested in broader and deeper coverage of technical details are referred to any recent sur-vey on video surveillance. For completeness of the discussion, we selected few systems that are relevant to ours in function to describe briefly here. Perhaps, a closely related system to ours is the DETER system. DETER can detect and track humans, and can analyze the trajectory of their motion for threat evaluation. The W4system detects humans and their body parts and analyzes some simple interactions between humans and objects. In, the idea of combining several cues together to enhance the robustness of tracking was utilized. Our system utilizes a similar idea with one step further. It uses different algorithms with different cues instead of one algorithm with different cues.

1.3.2 LITERATURE SURVEY:

Title: Towards Computational Baby Learning: A Weakly-Supervised Approach for Object Detection

Author: XiaodanLiang ; Si Liu ; Yunchao Wei ; Luoqi Liu.

Year: 2015

Description:

Intuitive observations show that a baby may inherently possess the capability of recognizing a new visual concept (e.g., chair, dog) by learning from only very few positive instances taught by parent(s) or others, and this recognition capability can be gradually further improved by exploring and/or interacting with the real instances in the physical world. Inspired by these observations, we propose a computational model for weakly-supervised object detection, based on prior knowledge modelling, exemplar learning and learning with video contexts. The prior knowledge is modeled with a pre-trained Convolutional Neural Network (CNN). When very few instances of a new concept are given, an initial concept detector is built by exemplar learning over the deep features the pre-trained CNN. The well-designed tracking solution is then used to discover more diverse instances from the massive online weakly labeled videos. Once a positive instance is detected/identified with high score in each video, more instances possibly from different view-angles and/or different distances are tracked and accumulated. Then the concept detector can be fine-tuned based on these new instances. This process can be repeated again and again till we obtain a very mature concept detector. Extensive experiments on Pascal VOC-07/10/12 object detection datasets [9] well demonstrate the effectiveness of our framework. It can beat the state-of-the-art full-training based performances by learning from very few samples for each object category, along with about 20,000 weakly labeled videos.

Title: Track and Transfer: Watching Videos to Simulate Strong Human Supervision for Weakly-Supervised Object Detection

Author: Krishna Kumar Singh, Fanyi Xiao.

Year: 2016

Description:

The status quo approach to training object detectors requires expensive bounding box annotations. Our framework takes a markedly different direction: we transfer tracked object boxes from weakly-labeled videos to weakly-labeled images to automatically generate pseudo ground-truth boxes, which replace manually annotated bounding boxes. We first mine discriminative regions in the weakly-labeled image collection that frequently/rarely appear in the positive/negative images. We then match those regions to videos and retrieve the corresponding tracked object boxes. Finally, we design a though transform algorithm to vote for the best box to serve as the pseudo GT for each image, and use them to train an object detector. Together, these lead to state-of-the-art weakly-supervised detection results on the PASCAL 2007 and 2010 datasets.

Title: Watch and Learn: Semi-Supervised Learning of Object Detectors from Videos

Author: Ishan Misra, Abhinav Shrivastava, Martial Hebert

Year: 2015

Description:

We present a semi-supervised approach that localizes multiple unknown object instances in long videos. We start with a handful of labeled boxes and iteratively learn and label hundreds of thousands of object instances. We propose criteria for reliable object detection and tracking for constraining the semi-supervised learning process and minimizing semantic drift. Our approach does not assume exhaustive labeling of each object instance in any single frame, or any explicit annotation of negative data. Working in such a generic setting allow us to tackle multiple object instances in video, many of which are static. In contrast, existing approaches either do not consider multiple object instances per video, or rely heavily on the motion of the objects present. The experiments demonstrate the effectiveness of our approach by evaluating the automatically labeled data on a variety of metrics like quality, coverage (recall), diversity, and relevance to training an object detector.

Title: Weakly supervised localization of novel objects using appearance transfer

Author: Mrigank Rochan ; Yang Wang.

Year: 2015

Description:

We consider the problem of localizing unseen objects in weakly labeled image collections. Given a set of images annotated at the image level, our goal is to localize the object in each image. The novelty of our proposed work is that, in addition to building object appearance model from the weakly labeled data, we also make use of existing detectors of some other object classes (which we call “familiar objects”). We propose a method for transferring the appearance models of the familiar objects to the unseen object. Our experimental results on both image and video datasets demonstrate the effectiveness of our approach.

Title: Large Scale Semi-Supervised Object Detection Using Visual and Semantic Knowledge Transfer

Author: YuxingTang ; Josiah Wang ; Boyang Gao

Year: 2016

Description:

Deep CNN-based object detection systems have achieved remarkable success on several large-scale object detection benchmarks. However, training such detectors requires a large number of labeled bounding boxes, which are more difficult to obtain than image-level annotations. Previous work addresses this issue by transforming image-level classifiers into object detectors. This is done by modeling the differences between the two on categories with both image level and bounding box annotations, and transferring this information to convert classifiers to detectors for categories without bounding box annotations. We improve this previous work by incorporating knowledge about object similarities from visual and semantic domains during the transfer process. The intuition behind our proposed method is that visually and semantically similar categories should exhibit more common transferable properties than dissimilar categories, e.g. a better detector would result by transforming the differences between a dog classifier and a dog detector onto the cat class, than would by transforming from the violin class. Experimental results on the challenging ILSVRC2013 detection dataset demonstrate that each of our proposed object similarity based knowledge transfer methods outperforms the baseline methods. We found strong evidence that visual similarity and semantic relatedness are complementary for the task, and when combined notably improve detection, achieving state-of-the-art detection performance in a semi-supervised setting.

1.4 Proposed System

It introduced a fast human detection and classification method using HOG features and Support Vector Machine (SVM) classifier. The proposed method comprised of three primary steps. Detection of moving regions from the video sequence, extraction of HOG features of moving regions, and utilizing SVM to classify the moving regions. Similarly, Qixiang presented an Error-Correcting Output Code (ECOC) based on manifold clustering strategy for human detection. The proposed technique worked efficiently with multi-view and multi-posture problems. Dewei introduced an online expectation-maximization (EM) algorithm in order to estimate foreground and background. Later, the human samples are cropped from the estimated foreground for HOG feature extraction.

CHAPTER 2

PROJECT DESCRIPTION

2.1 GENERAL:

In this project, two seamlessly integrated solutions, self-paced learning and multi-modal learning, are used to achieve high precision and recall during training sample generation.

2.2 METHODOLOGIES

MODULE:

1. INPUT VIDEO
2. PRE-PROCESSING
4. SEGMENTATION
5. CLASSIFICATION

➢ INPUT VIDEO:

Firstly we take an video for detecting the objects whose presented on the video at the anytime. The multiple object tracking method works on fixed cameras. It starts with an adaptive background modelling module which deals with changing illuminations and does not require objects to be constantly moving.

PRE-PROCESSING:

Pre-processing is a common name for operations with images at the lowest level of abstraction both input and output are intensity images. These iconic images are of the same kind as the original data captured by the sensor, with an intensity image usually represented by a matrix of image function values (brightness) The aim of pre-processing is an improvement of the image data that suppresses unwilling distortions or enhances some image features important for further processing,

although geometric transformations of images (e.g. rotation, scaling, and translation) are also classified among pre-processing methods. Since here similar techniques are used.

- **SEGMENTATION:**

We can divide or partition the image into various parts called segments. It's not a great idea to process the entire image at the same time as there will be regions in the image which do not contain any information. By dividing the image into segments, we can make use of the important segments for processing the image. That, in a nutshell, is how image segmentation works.

- **CLASSIFICATION:**

As a result of the model executed during object detection. In this detection we can detect multiple objects. It is used to filter weak calculations with the Confidence value. Unwanted values were filtered out in this way.

2.3 TECHNIQUE USED OR ALGORITHM USED

Our multiple object tracking method is reliable to deal with occlusions, irregular object motions, changing appearances by postponing the decision of object trajectories until sufficient information is accumulated over time. It makes a global decision. The most possible hypothesis generates the multiple objects tracking result. The trajectories provide information of object identifications, motion histories, timing and object interactions. The information can be applied to detect abnormal behaviors in video surveillance and collect traffic data in traffic control systems.

CHAPTER 3

REQUIREMENTS ENGINEERING

3.1 GENERAL

We provide a brief discussion on the relationship between FEOD and other types of supervisions, excluding the methods using strong labels. First, strictly speaking, FEOD is a semi-supervised task. But to the best of our knowledge, most works on semi-supervised object detection (SSOD) assume around50% of all the labeled bounding boxes.

3.2 HARDWARE REQUIREMENTS

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It should what the system do and not how it should be implemented.

- PROCESSOR : DUAL CORE 2 DUOS.
- RAM : 4GB DD RAM
- HARD DISK : 250 GB

3.3 SOFTWARE REQUIREMENTS

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the teams and tracking the team's progress throughout the development activity

SOFTWARE REQUIREMENTS

- Operating System : Windows 7/8/10
- Platform : Spyder3
- Programming Language : Python
- Front End : Spyder3

3.4 FUNCTIONAL REQUIREMENTS

A functional requirement defines a function of a software-system or its component. A function is described as a set of inputs, the behavior, Firstly, the system is the first that achieves the standard notion of semantic security for data confidentiality in attribute-based deduplication systems by resorting to the hybrid cloud architecture.

3.5 NON-FUNCTIONAL REQUIREMENTS

EFFICIENCY

Our multi-modal event tracking and evolution framework is suitable for multimedia documents from various social media platforms, which can not only effectively capture their multi-modal topics, but also obtain the evolutionary trends of social events and generate effective event summary details over time. Our proposed mm ETM model can exploit the multi-modal property of social event, which can effectively model social media documents including long text with related images and learn the correlations between textual and visual modalities to separate the visual-representative topics and non-visual-representative topics.

CHAPTER 4

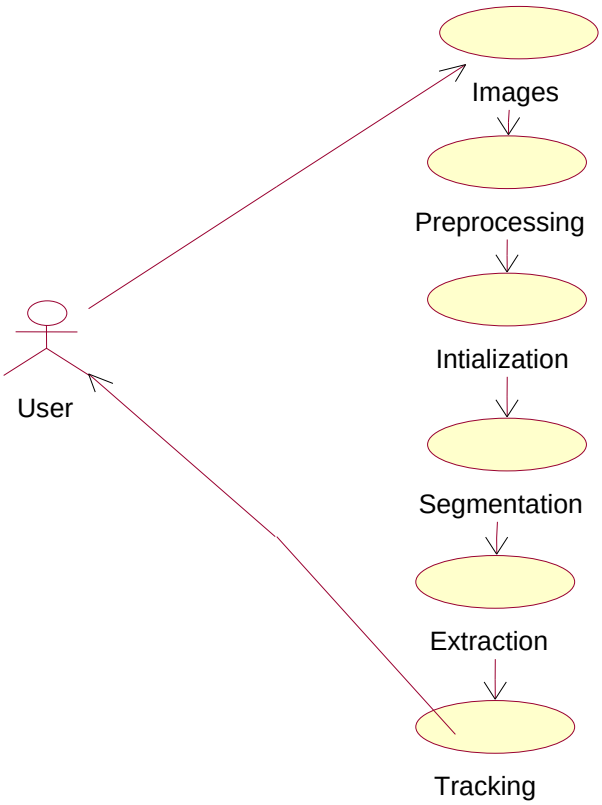
DESIGN ENGINEERING

4.1 GENERAL

Design Engineering deals with the various UML [Unified Modelling language] diagrams for the implementation of project. Design is a meaningful engineering representation of a thing that is to be built. Software design is a process through which the requirements are translated into representation of the software. Design is the place where quality is rendered in software engineering. Design is the means to accurately translate customer requirements into finished product.

UML Diagrams

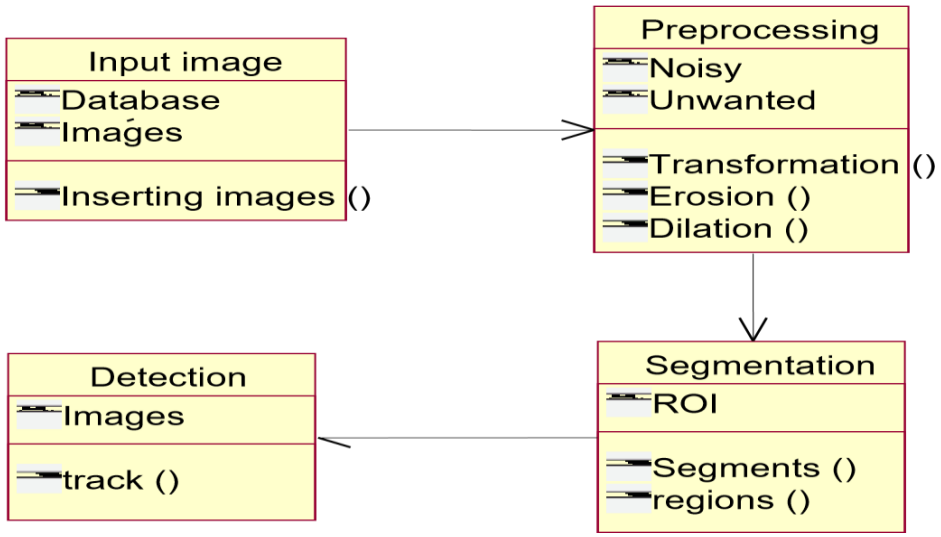
Use case diagram



EXPLANATION:

The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted. The above diagram consists of user as actor. Each will play a certain role to achieve the concept.

Class Diagram

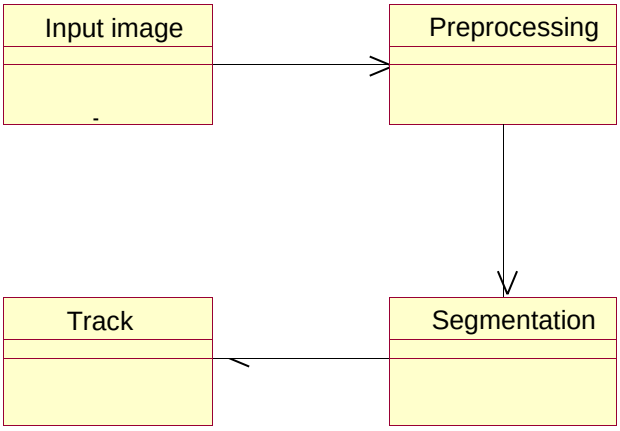


()

EXPLANATION

In this class diagram represents how the classes with attributes and methods are linked together to perform the verification with security. From the above diagram shown the various classes involved in our project.

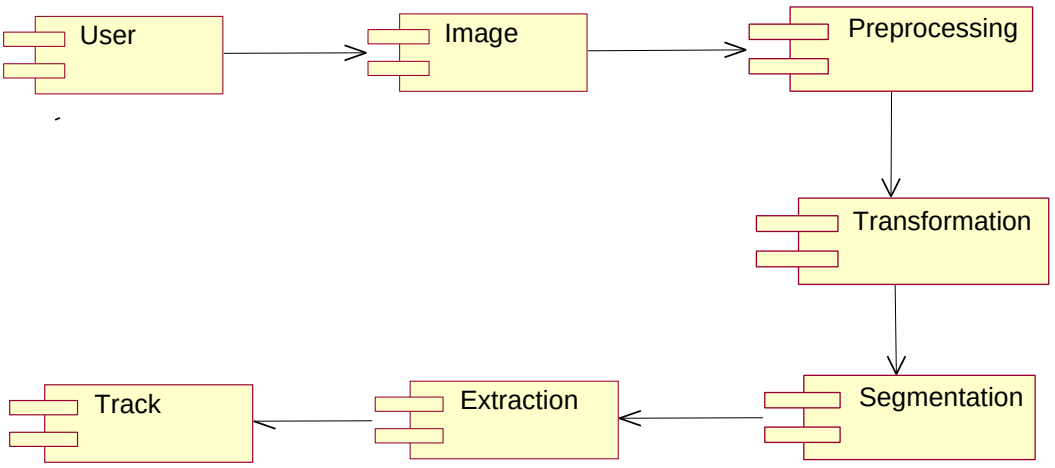
Object Diagram



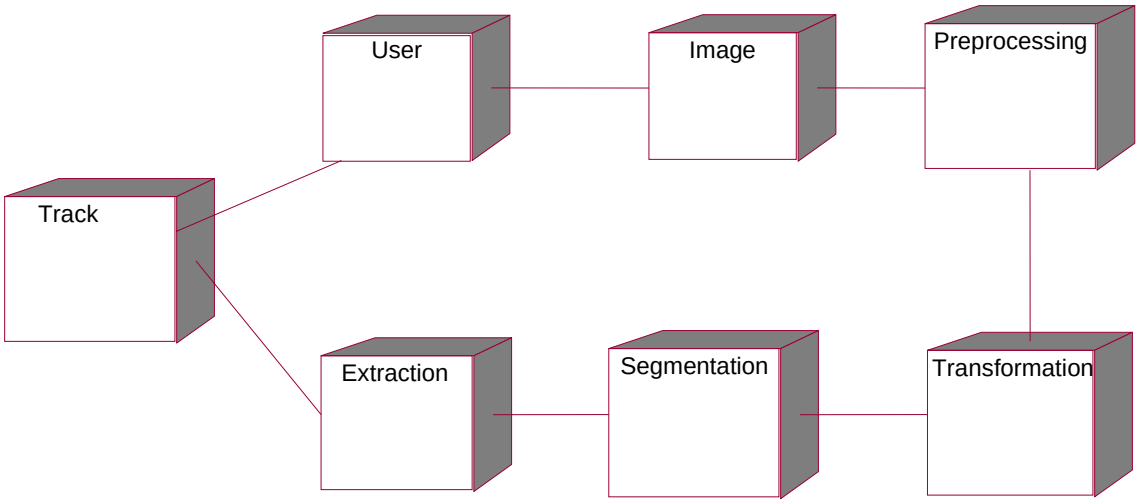
EXPLANATION:

In the above digram tells about the flow of objects between the classes. It is a diagram that shows a complete or partial view of the structure of a modeled system. In this object diagram represents how the classes with attributes and methods are linked together to perform the verification with security.

Component Diagram



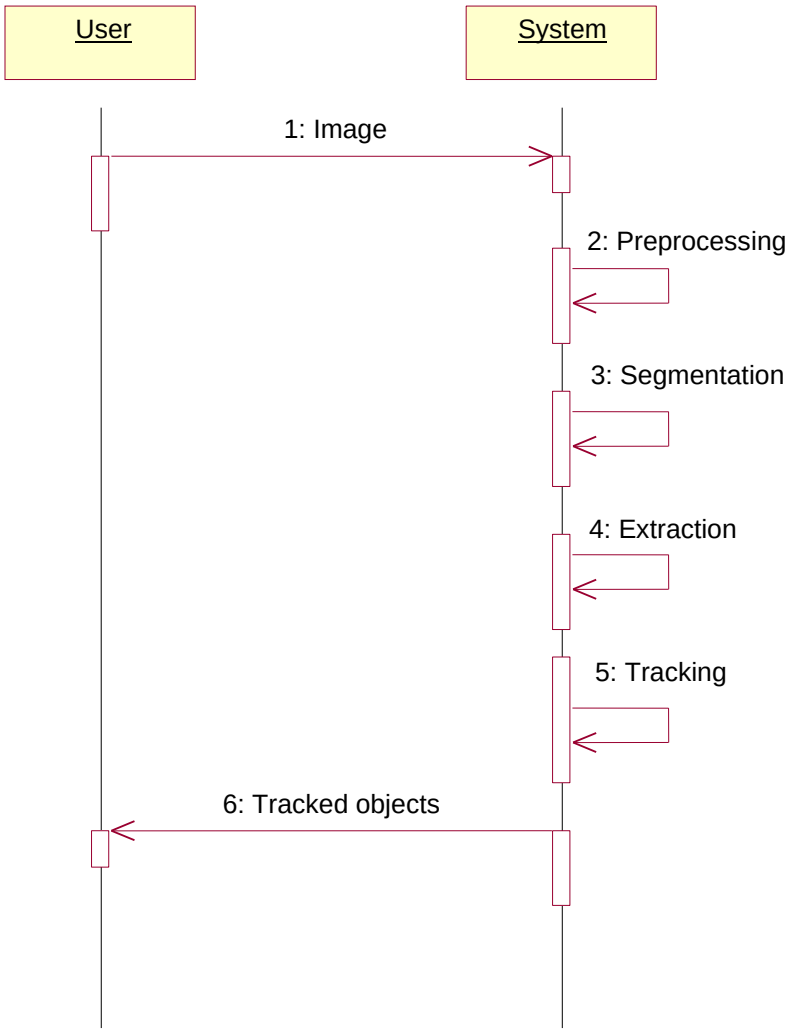
Deployment Diagram



EXPLANATION:

In the above digram tells about the flow of objects between the classes. It is a diagram that shows a complete or partial view of the structure of a modeled system. In this object diagram represents how the classes with attributes and methods are linked together to perform the verification with security.

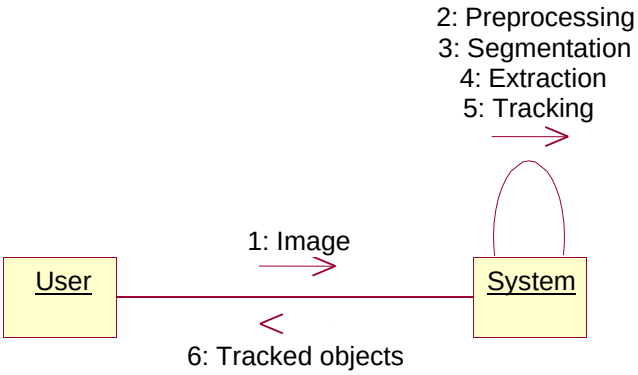
Sequence Diagram



EXPLANATION:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

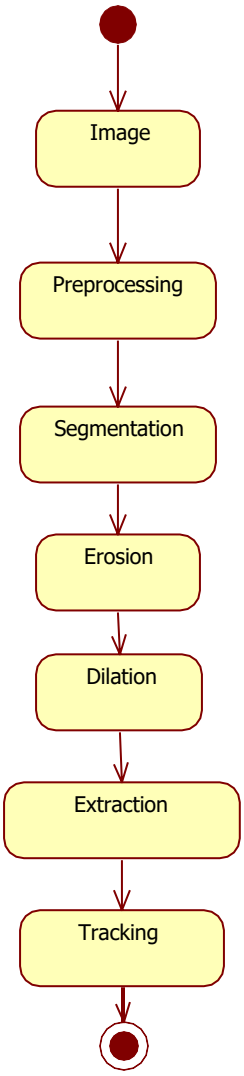
Collaboration Diagram



EXPLANATION:

In the above Collaborationdigram tells about the flow of objects between the classes. It is a diagram that shows a complete or partial view of the structure of a modeled system. In this object diagram represents how the classes with attributes and methods are linked together to perform the verification with security.

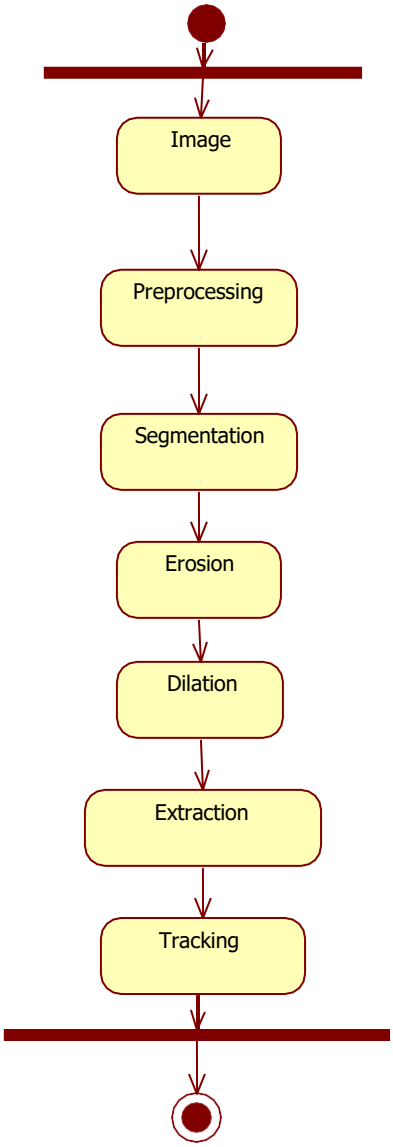
State Diagram



EXPLANATION:

State diagram are a loosely defined diagram to show workflows of stepwise activities and actions, with support for choice, iteration and concurrency. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. Many forms of state diagrams exist, which differ slightly and have different semantics.

Activity Diagram

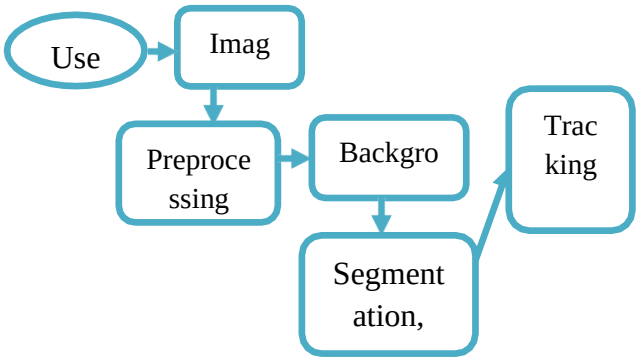


EXPLANATION:

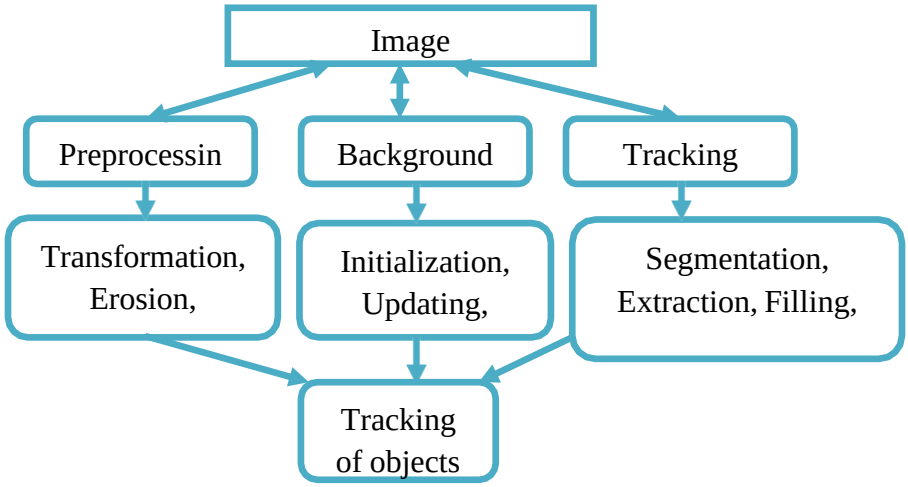
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

Data Flow Diagram

Level 0



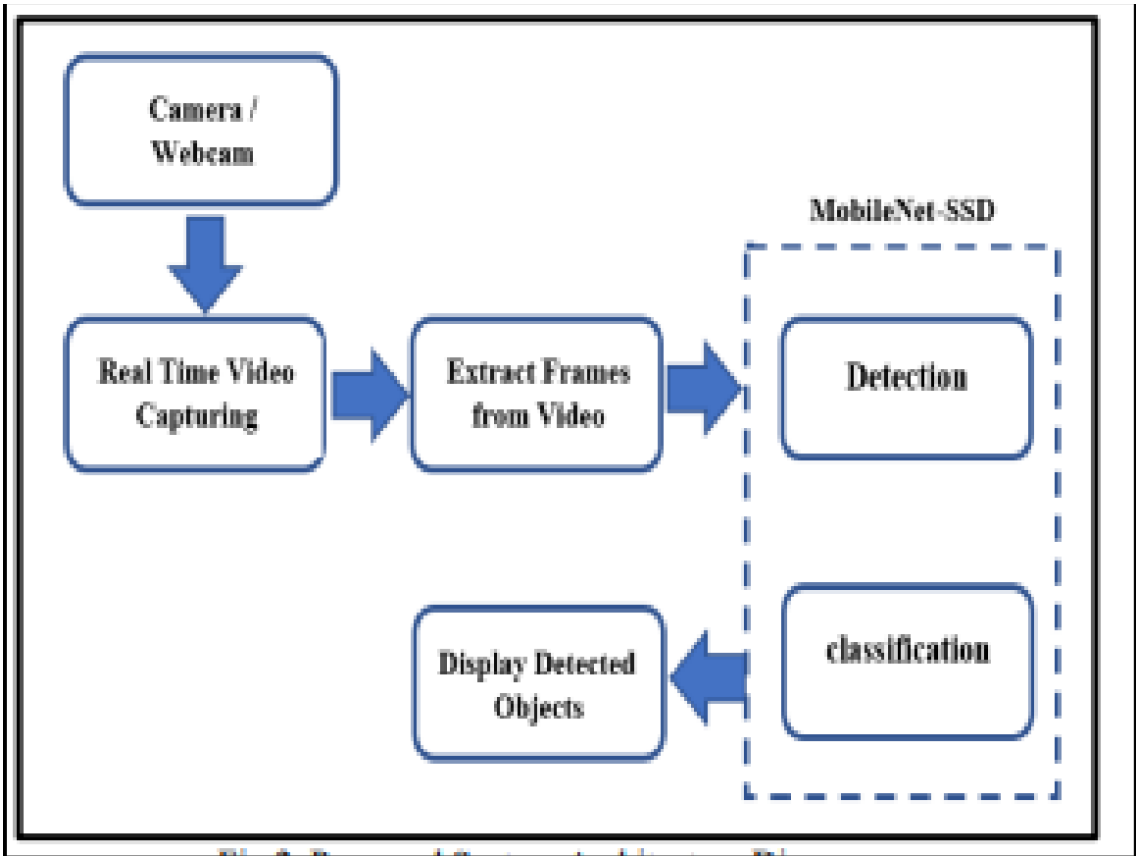
Level1



EXPLANATION:

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

System Architecture



CHAPTER 5

DEVELOPMENT TOOLS

Python

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Small Talk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Importance of Python

- **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

- **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

Features of Python

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
- **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** − Python provides interfaces to all major commercial databases.
- **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** − Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below −

- It supports functional and structured programming methods as well as OOP.

- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- IT supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Libraries used in python:

- NumPy - mainly useful for its N-dimensional array objects.
- pandas - Python data analysis library, including structures such as data frames.
- matplotlib - 2D plotting library producing publication quality figures.
- scikit-learn - the machine learning algorithms used for data analysis and data mining tasks.

Figure : NumPy, Pandas, Matplotlib, Scikit-learn

CHAPTER 6

IMPLEMENTATION

6.1 GENERAL

Coding:

 # USAGE

 # python human_activity_reco_deque.py --model resnet-34_kinetics.onnx --classes action_recognition_kinetics.txt --input example_activities.mp4

 # python human_activity_reco_deque.py --model resnet-34_kinetics.onnx --classes action_recognition_kinetics.txt

 # import the necessary packages

from **collections** ***import*** **deque**

import **numpy** ***as*** **np**

import **imutils**

import **cv2**

 # load the contents of the class labels file, then define the sample

 # duration (i.e., # of frames for classification) and sample size

 # (i.e., the spatial dimensions of the frame)

```
CLASSES = open('action_recognition_kinetics.txt').read().strip().split("\n")

SAMPLE_DURATION = 16

SAMPLE_SIZE = 112



# initialize the frames queue used to store a rolling sample duration

# of frames -- this queue will automatically pop out old frames and

# accept new ones

frames = deque(maxlen=SAMPLE_DURATION)



# load the human activity recognition model

print("[INFO] loading human activity recognition model...")

net = cv2.dnn.readNet('resnet-34_kinetics.onnx')



# grab a pointer to the input video stream

print("[INFO] accessing video stream...")

vs = cv2.VideoCapture("examples_activities.mp4")



# loop over frames from the video stre

while True:
```

```
# read a frame from the video stream
(grabbed, frame) = vs.read()

# if the frame was not grabbed then we've reached the end of
# the video stream so break from the loop
if not grabbed:
  print("[INFO] no frame read from stream - exiting")
  break

# resize the frame (to ensure faster processing) and add the
# frame to our queue
frame = imutils.resize(frame, width=400)
frames.append(frame)

# if our queue is not filled to the sample size, continue back to
# the top of the loop and continue polling/processing frames
#if len(frames) < SAMPLE_DURATION:
  #continue
```

```
# now that our frames array is filled we can construct our blob
blob = cv2.dnn.blobFromImages(frames, 1.0,
  (SAMPLE_SIZE, SAMPLE_SIZE), (114.7748, 107.7354, 99.4750),
  swapRB=True, crop=True)
blob = np.transpose(blob, (1, 0, 2, 3))
blob = np.expand_dims(blob, axis=0)

# pass the blob through the network to obtain our human activity
# recognition predictions
net.setInput(blob)
outputs = net.forward()
label = CLASSES[np.argmax(outputs)]

# draw the predicted activity on the frame
cv2.rectangle(frame, (0, 0), (300, 40), (0, 0, 0), -1)
cv2.putText(frame, label, (10, 25), cv2.FONT_HERSHEY_SIMPLEX,
  0.8, (255, 255, 255), 2)

# display the frame to our screen
```

```
  cv2.imshow("Activity Recognition", frame)

  key = cv2.waitKey(1) & 0xFF



  # if the `q` key was pressed, break from the loop

  if key == ord("q"):

    break

cv2.destroyAll-Windows()
```

CHAPTER 7

SNAPSHOTS

General:

This project is implements like application using python and the Server process is maintained using the SOCKET & SERVERSOCKET and the Design part is played by Cascading Style Sheet.

SNAPSHOTS

EXPLANATION:

In this image, a person is captured in the midst of a dynamic leg stretch, showcasing flexibility and focus as the leg extends gracefully, forming a compelling visual representation of physical wellness.

EXPLANATION:

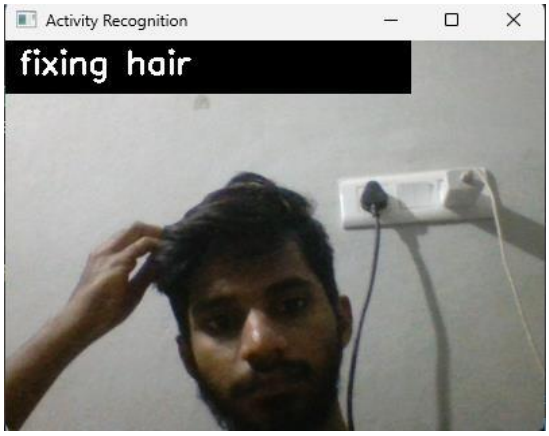
The yoga image encapsulates a serene moment, with a person engaged in a yoga pose that exudes tranquility. The harmonious alignment of the body reflects the balance sought in yoga practice, creating a visually calming and health-focused snapshot.

EXPLANATION:

A captivating instance is frozen in time as fingers snap together, conveying a sense of rhythm and precision. The image captures the swift, deliberate movement, adding a touch of energy and vibrancy to the visual narrative.

EXPLANATION:

The image depicts a powerful moment of physical exertion as the subject engages in a push-up, showcasing strength and determination. The dynamic posture conveys a commitment to fitness, making it a compelling representation of active and disciplined living.



EXPLANATION:

A candid snapshot reveals a person in the act of fixing their hair, adding a personal and relatable touch to the collection. The image conveys a moment of self-care and attention to appearance, enriching the overall narrative with a blend of everyday activities and health-conscious behaviors.

CHAPTER 8
SOFTWARE TESTING

8.1 GENERAL

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

8.2 DEVELOPING METHODOLOGIES

The test process is initiated by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process verifies that the application meets the requirements specified in the system requirements document and is bug free. The following are the considerations used to develop the framework from developing the testing methodologies.

8.3.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program input produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

8.3.2 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.
Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised.
Systems/Procedures : interfacing systems or procedures must be invoked.

8.3.3 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

8.3.4 Performance Test

The Performance test ensures that the output be produced within the time limits, and the time taken by the system for compiling, giving response to the users and request being send to the system for to retrieve the results.

8.3.5 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.
The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

8.3.6 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Acceptance testing for Data Synchronization:

- The Acknowledgements will be received by the Sender Node after the Packets are received by the Destination Node
- The Route add operation is done only when there is a Route request in need
- The Status of Nodes information is done automatically in the Cache Updating process

8.2.7 Build the test plan

Any project can be divided into units that can be further performed for detailed processing. Then a testing strategy for each of this unit is carried out. Unit testing helps to identity the possible bugs in the individual component, so the component that has bugs can be identified and can be rectified from errors.

CHAPTER 9

FUTURE ENHANCEMENT

Image based likelihood is then computed to give a probability to each hypothesis. This computation is based on the object detection probability, appearance similarity, trajectory smoothness and image foreground coverage and compactness. The probabilities are calculated based on a sequence of images, therefore, they are temporally global representations of hypotheses likelihood. The hypotheses are ranked by their probabilities and the unlikely hypotheses are pruned from the graph in the hypotheses-management step.

CONCLUSION & REFERENCES

CONCLUSION

The tracking module provides feedbacks to the object detection module to improve the local detection performance. According to the trajectories in the top hypothesis, the multiple object tracking module predicts the most likely locations to detect objects. This interaction tightly integrates the object detection and tracking, and makes both of them more reliable.

REFERENCES

[1] J.P. MacCormick and A. Blake, “A probabilistic exclusion principle for tracking multiple objects,” in ICCV99, 1999, pp. 572–578.

[2] H. Tao, H.S. Sawhney, and R. Kumar, “A sampling algorithm for tracking multiple objects,” in Vision Algorithms 99, 1999.

[3] M. Isard and J.P. MacCormick, “Bramble: A bayesian multiple-blob tracker,” in ICCV01, 2001, pp. II: 34–41.

[4] C. Hue, J.P. Le Cadre, and P. Perez, “Tracking multiple objects with particle filtering,” IEEE Trans. on Aerospace and Electronic Systems, vol. 38, no. 3, pp. 791–812, July 2002.

[5] D.B. Reid, “An algorithm for tracking multiple targets,” AC, vol. 24, no. 6, pp. 843–854, December 1979.

[6] T.E. Fortmann, Y. Bar-Shalom, and M. Scheffe, “Sonar tracking of multiple targets using joint probabilistic data association,” IEEE Journal Oceanic Eng., vol. OE-8, pp. 173–184, July 1983.

[7] R.L. Streit and T.E. Luginbuhl, “Maximum likelihood method for probabilistic multi-hypothesis tracking,” in Proceedings of SPIE International Symposium, Signal and Data Processing of Small Targets, 1994.

[8] H. Gauvrit and J.P. Le Cadre, “A formulation of multi target tracking as an incomplete data problem,” IEEE Trans. on Aerospace and Electronic Systems, vol. 33, no. 4, pp. 1242–1257, Oct 1997.

[9] S. Avidan, “Support vector tracking,” in CVPR01, 2001, pp. I:184–191.

[10] I. Haritaoglu, D. Harwood, and L.S. Davis, “W4s: A real-time system for detecting and tracking people in 2 1/2-d,” in ECCV98, 1998.