

SOFTWARE REQUIREMENTS SPECIFICATION

Human Action Recognition Under View Change

Under the guidance of
Dr.Shiloah Elizebeth

Group name:msmgp

PREPARED BY:

N. MURUGAPPAN

MA.P .MAANICCKA SENTIL

G. MOHAMED SHAMSUDEEN

TABLE OF CONTENTS

1. INTRODUCTION

1.1 PURPOSE.

1.2 PROJECT SCOPE.

1.3 DEFINITIONS.

2. DESCRIPTION

2.1 PRODUCT PRESPECTIVE.

2.2 PRODUCT FUNCTIONS.

2.3 USER CHARACTERISTICS.

2.4 USER CONSTRIANTS.

3. EXTERNAL INTERFACES REQUIRED

3.1 USER INTERFACES.

3.2 HARDWARE INTERFACES.

3.3 SOFTWARE INTERFACES.

4. SPECIFIC REQUIREMENTS

4.1 LOGON CAPABILITIES.

4.2 USABILITY.

4.3 RELIABILITY

4.3.1 AVAILABILITY.

4.3.2 MEAN TIME BETWEEN FAILURES.

4.3.3 MEAN TIME TO REPAIR.

4.3.4 ACCESS RELIABILITY.

5. SYSTEM FEATURES

5.1 SYSTEM TRAINING PHASE

5.2 SYSTEM TESTING PHASE

5.3 SYSTEM RESPONSE TO INVALID ACTION

5.4 QUERY THE SYSTEM

6. DESIGN CONSTRIANTS

6.1 SOFTWARE LANGUAGES USED

6.2 DEVELOPMENT TOOLS

6.3 IEEE REFERENCE DETAILS

7. DIAGRAMS

7.1 FLOW DIAGRAM OF VARIOUS ACTIONS PERFORMED

7.2 USE CASE DIAGRAM

7.3 ACTIVITY DIAGRAM

7.4 CLASS DIAGRAM

7.5 SEQUENCE DIAGRAM

7.6 COLLABORATION DIAGRAM

1. INTRODUCTION

1.1 PURPOSE:

In the emerging days of increasing needs discovering new technologies has become an important criterion. VISUAL recognition and understanding of human actions have attracted much attention over the past three decades and remain an active research area of computer vision. A good solution to the problem holds a yet unexplored potential for many applications, such as the search for and the structuring of large video archives, video surveillance, human-computer interaction, gesture recognition and video editing.

Recent work has demonstrated the difficulty of the problem associated with the large variation of human action data due to the individual variations of people in expression, posture, motion, and clothing; perspective effects and camera motions; illumination variations and distracting effects of scenes surroundings. Also, actions frequently involve and depend on manipulated objects, which add another layer of variability.

1.2 PROJECT SCOPE:

The following found to be the essential scope of the software from the stake holders view

- Reduces the Human Requirements.
- Increase the accuracy and precision in monitoring.
- Provides more security.
- The system can be easily modified to suit current requirements.

The features that are described in this document are used in the future phases of the software development cycle. The features described here meet the needs of all the users. The success criterion for the system is based in the level up to which the features described in this document are implemented in the system.

1.3 DEFINITIONS :

USER : A human being whose gestures are being monitored under Camera.

ADMIN: A person who is empowered to monitor the system and control it and who is responsible for addition of various trusted/valid action into the system and deletion of various invalid action from System. When a system monitors some invalid situation immediately Admin is alerted.

2. DESCRIPTION:

2.1 Product perspective:

The product aims at reducing Human labor, decreasing the cost of surveillance and increases tracking accuracy

2.2 Product Functions:

- The software requires the users to register their basic gestures.
- The registration for new users can be updated as per the needs.
- The users are monitored all the time. This guarantees safety 24 x 7.
- If any unauthorized person enters, an alert is generated.
- All the footages are automatically recorded for future references with one month backup facility for security reasons .

2.3 User Characteristics:

The user's gestures are taken as input and they are recorded in the system. Multiple view angles are used to record the gestures. Whenever a user enters, the camera records his/her gestures and compares it with the prerecorded data in the system. If the gesture match is found, then the person is assumed to be authorized to enter. In case of mismatch, a siren is sounded which alerts the surroundings which implies an unauthorized entry.

2.4 User constraints:

- a) The user can be of any gender.
- b) The users gestures are recorded in top and front view.
- c) In case of any change in gestures of pre existing users, it can be updated in the system.
- d) The systems used must ensure high level security and are accessible to only the administrators.
- e) The user should provide a casual gesture in order to record it into the system.

3. EXTERNAL INTERFACES REQUIREMENTS:

3.1 User Interfaces:

As the proposed project will be closely monitored only by the admin of the company and will not be open to all and the front end could be designed with Java programming Language compatible Net Beans or any other IDE.

3.2 Hardware Interfaces:

- Processor : Dual Core and above processor compatible
- RAM : 2 GB
- Hard Disk : 80 GB(minimum free space for storing good resolution video as a backup)

3.2 Software Interfaces:

- Domain: Digital Image Processing, Transactions on Pattern Analysis and Machine intelligence
- Operating system: Windows XP, Windows Vista, Windows 7.
- System: MySQL 4.0
- Platform : J2SE 1.5 Java 2 Platform, Standard Edition(J2SE).
- Special Tool : Java Media Framework
- Protocol : TCP

4. SPECIFIC REQUIREMENTS:

4.1 Logon capability:

The software shall provide the logon capability only to the administrator who has the power to record other authorized users gestures. Other persons can view their footage but cannot modify it without the administrator's permission.

4.2 Usability

The software is user friendly and self-explanatory with minimum knowledge about using computer is sufficient for effective usage of the proposed system. Java provides many user friendly and interactive features for the project to be effectively used. Java language enabled with Multi threaded interface which means handling multiple tasks simultaneously. Java supports multi threading programs. This means that we need not wait for the application to finish one task before beginning another.

4.3 Reliability:

The system has to be very reliable due to the importance of data and the damages incorrect or incomplete data can do. Reliability will be enhanced by taking a backup of the video recorded and log history once in a month. Java is a robust language. It provides many safeguards to ensure reliable code. It has strict compiler time and runtime checking for data types. It is designed as a garbage-collected language relieving the programmers virtually all memory management problems.

4.3.1 Availability:

The software and the server is available 100% for the user and is used 24 hrs a day and 365 days a year. The server shall be operational 24 hours a day and 7 days a week.

4.3.2 Mean time between failures (MTBF):

The software is developed such that it fails once in 3-5 years. The server performance depends on the number of surveillance camera and the video input quality given to the system

4.3.3 Mean time to repair (MTTR):

The software/server will be recovered with a back up of 1 or 2 hours of time with minimum loss in the system working.

4.3.4 Access Reliability:

The system shall provide 100% access reliability and as java is a robust language

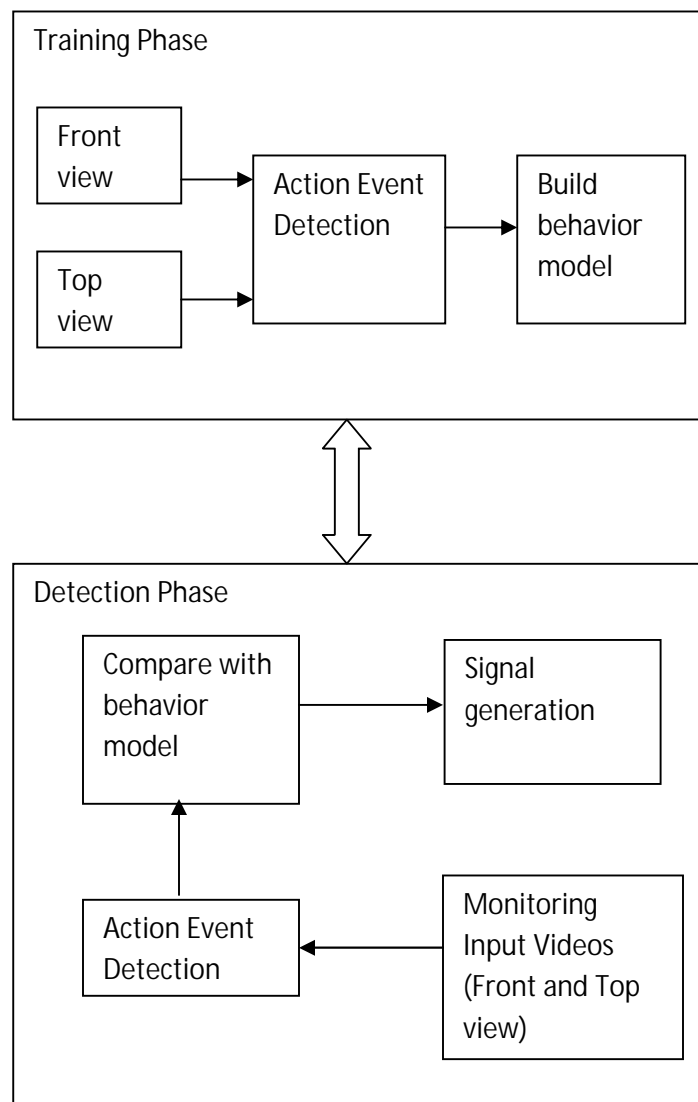
4.3.5 Maintenance:

The proposed software will definitely undergo change once it is delivered to the customer. There can be many reasons for this change to occur. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations.

5. SYSTEM FEATURES

5.1 SYSTEM TRAINING PHASE:

This module would initially train the system with new input Behavior of various Authenticated person and store the modeling into System as a new system doesn't the list of valid and invalid activities. After this step the new activity has been recorded into the System and next time when the system detects the same new activity by the user there will be no mismatch with the DB entry and it will be ascertained as valid person. Main objective of this phase is to recognize and differentiate between the authenticated valid person and the unauthenticated invalid person.



5.2 SYSTEM TESTING PHASE:

Once the initial training phase is completed the system is ready to be deployed for checking the human action recognition under the view change and when it has detected when a person enters a room, video of him is captured and stored (both side view and the top view) then it is given to the training module here the video is checked if it is a normal behavior splited image is taken, whenever the action is recognized blob images are saved, and the frame counts are taken . In case, the anomaly is detected the red color will be displayed. The abnormal behavior is achieved by tracking the videos and blob frames and checking each frame values.

5.3 SYSTEM RESPONSE TO INVALID ACTION:

When the system encounters an abnormal behavior by tracking the video and the blob frames and checking each frame values with that of system entry, an alert is sent to the Admin and he determines whether the particular person whose action has been not present in the system has to be updated into trusted set of actions or not.

5.4 QUERY THE SYSTEM FOR LIST OF AUTHENTICATED ACTION:

Admin wants the system to retrieve various trusted activities stored in the System during the course of its training phase. Only Admin has the privilege to check the actions and if necessary edit, add or delete the data i.e. trusted set of activities in the system.

6. DESIGN CONSTRAINTS:

6.1 Software Language Used

The languages that shall be used for coding the Human Recognition Under View Change is JAVA and a special tool called JAVA MEDIA FRAMEWORK . The system used is MYSQL 4.0 and preferable operating system is Windows XP or higher versions.

6.2 Development Tools

Will make use of the available Java Development Tool kits. Also will make use of the online references available for developing programs using Java and Java Media Framework.

6.3 IEEE Reference Details:

This paper appeared in: Pattern Analysis and Machine Intelligence, IEEE Transactions on

Title: View-Independent Action Recognition from Temporal Self-Similarities

Authors:

- Imran N. Junejo, Member, IEEE,
- Emilie Dexter,
- Ivan Laptev,
- Patrick P´erez

Issue Date: Jan. 2011

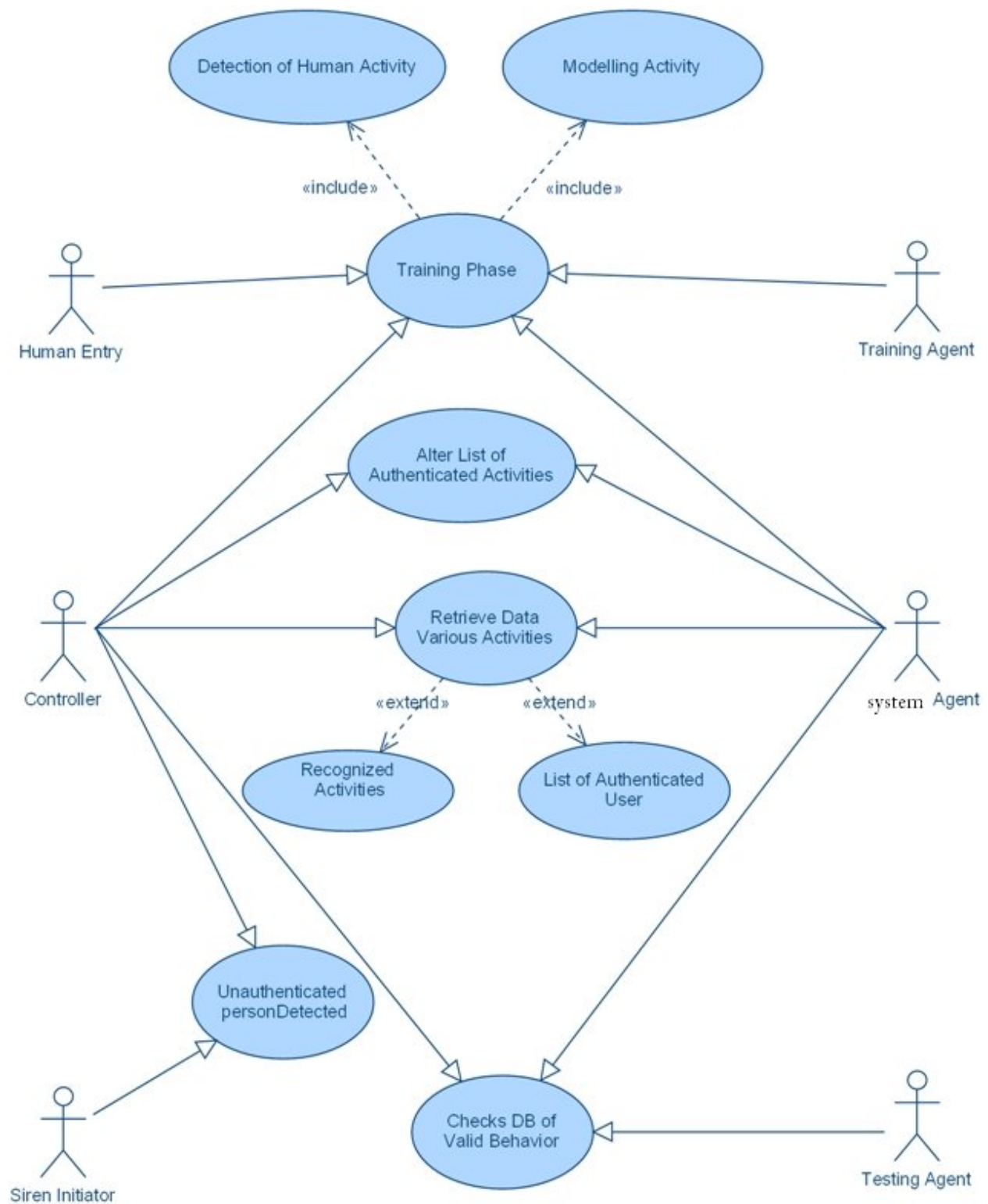
Volume: 33 Issue:1

On page(s): 172 - 185

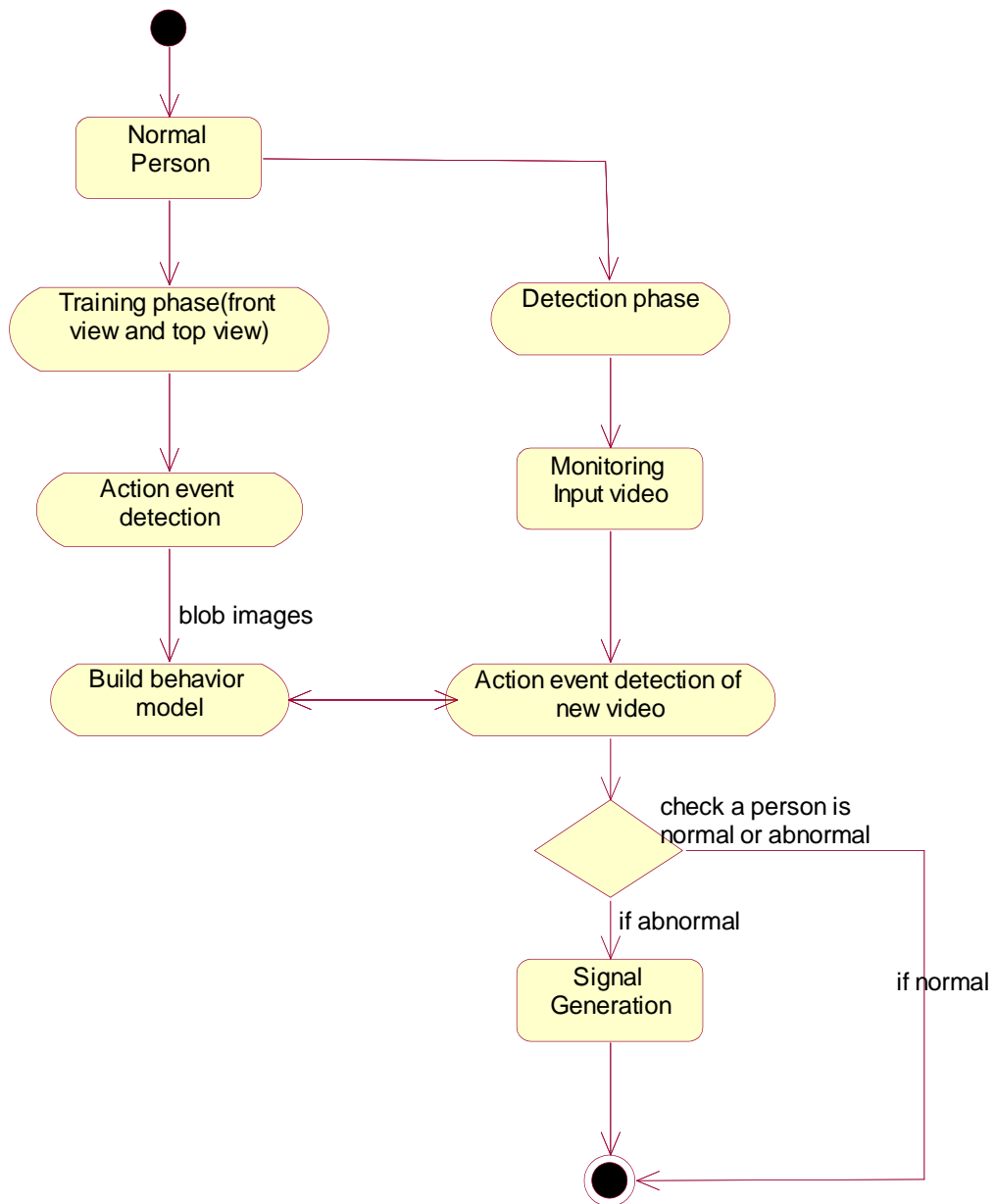
ISSN: 0162-8828

INSPEC Accession Number: 11661266

7.2 USE CASE DIGRAM

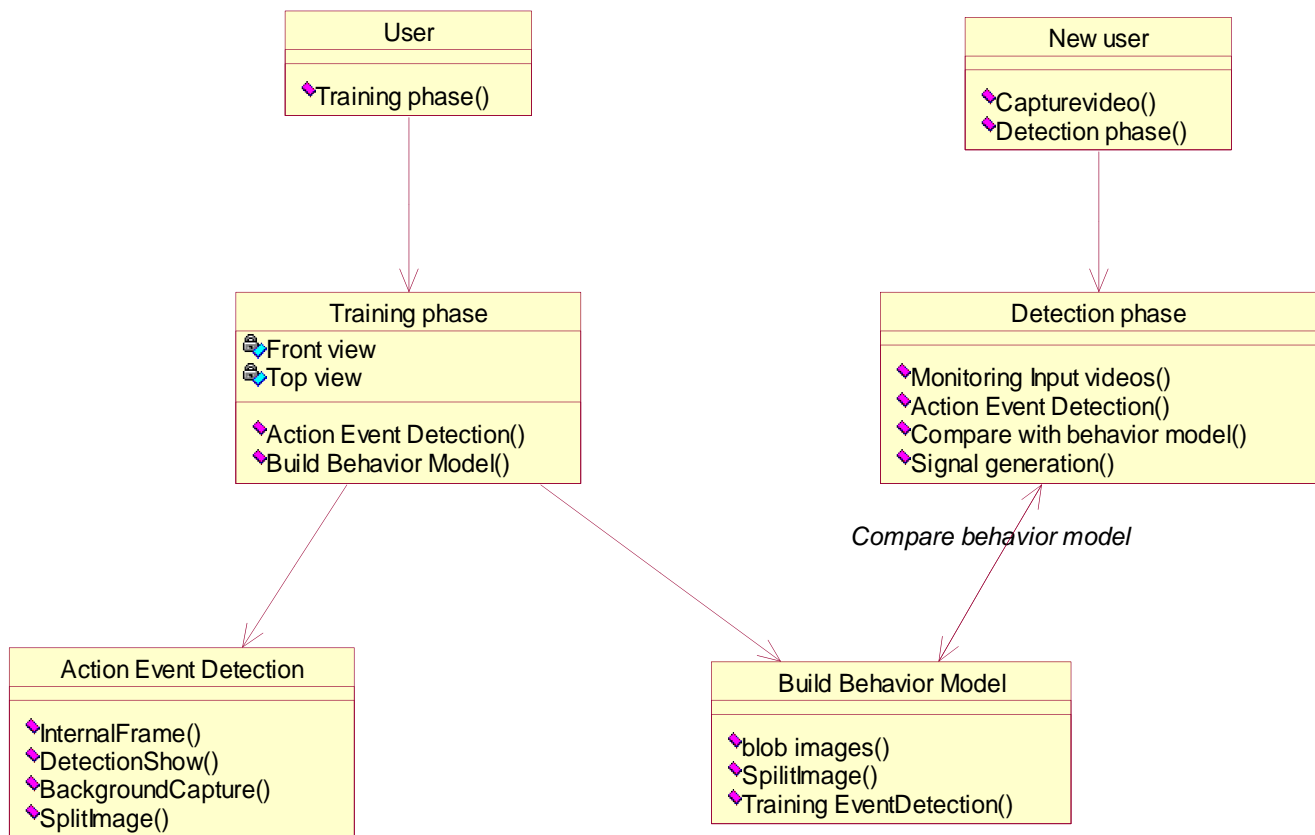


7.3 Activity Diagram



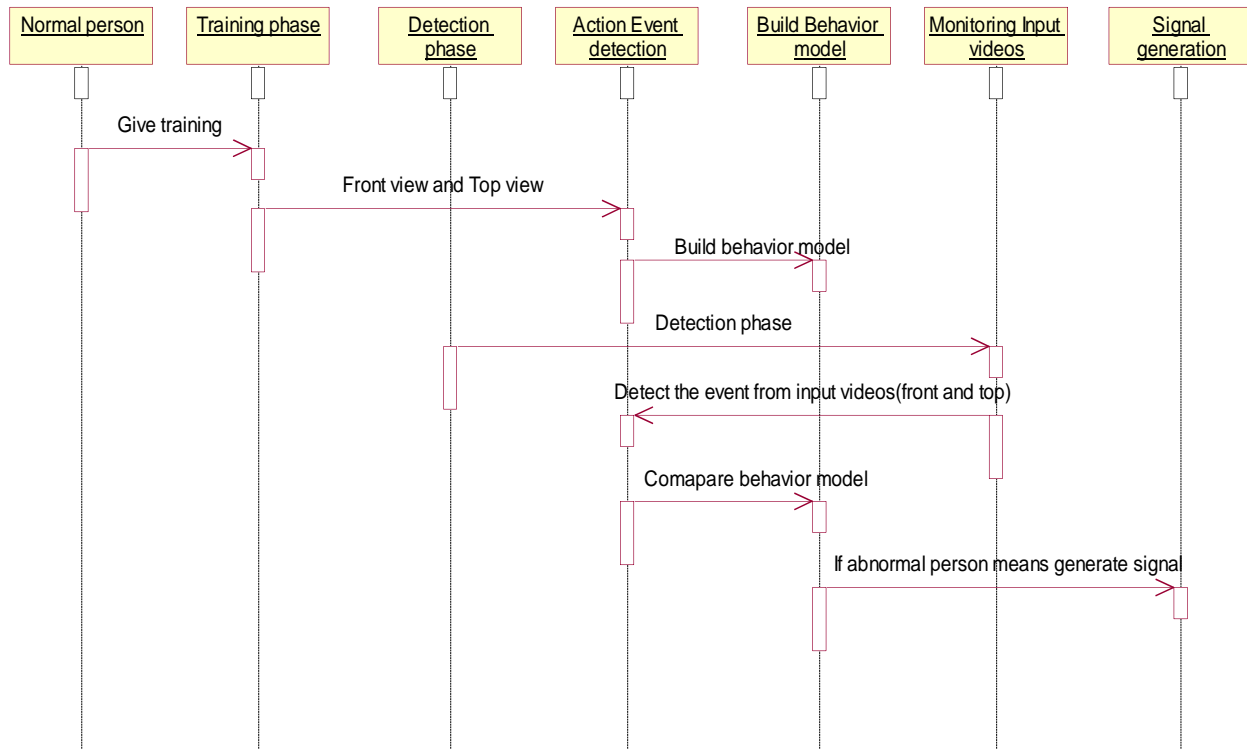
7.4 CLASS DIAGRAM

Class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system classes, their attributes, and the relationship between the classes



7.5 SEQUENCE DIAGRAM:

Sequence diagram are an easy and intuitive way of describing the behavior of a system by viewing the interaction between the system and its environment. A sequence diagram shows an interaction arranged in time sequence



7.6 COLLABORATION DIAGRAM:

Achieve Collaboration diagram represents a collaboration, which is a set of object related in a particular context and interaction which is a set of messages exchanged among the object within the collaboration to a desired outcome

