Software Requirement Specification Document 'VIGIL'

Hussein ElSherif, Khaled Salem , Mahmoud ElShennawy
Supervisor: Dr.Ayman Ezzat
T.A: Huda Mohammed

March 10, 2017

1 Introduction

1.1 Purpose of the document

The purpose of this document is to define the functional and non functional requirements and emphasize on our graduation project's design and functionality. Our target audience are the end users and the committee of our graduation projects for the year 2017.

1.2 Scope of this document

In this document, the defined functional and non functional requirements are derived by the development team. We are under a time constraint of 9 months and a budget constraint of about 10,000 Egyptian pounds.

1.3 Overview

Our system will be able to detect predefined abnormal behaviors related to car theft in real time using GPU.

1.4 Business Context

Vigil aims to create more reliable surveillance environments by adding real time abnormal behavior detection over GPU. We aim to decrease car theft rates and increase the rate of crime prevention through automated abnormal behavior detection to guarantee the efficiency of the surveillance system instead of relying on the traditional approach of relying on human factor.

2 General Description

2.1 Product Functions

2.1.1 Module 1 : User controls:-

1-User can define Regions of interest
2-User can define new cameras
3-User can define detection accuracy for different camera types and quality

2.1.2 Module 2 : Real time detection of the following anomalies:-

1- Suspicious loitering around car.
2- Hand swing to break windows.
3- jumping into car.
It also detects faces in case of anomalies and saves it into the system.

2.2 Similar System Information

The first similar system is NEC's Behavior detection system [1]. It can detect if a person crosses a line he shouldn't have, or loitering, or illegal parking. Similarly from our proposal, we have The Bike theft recognition according to object detection and Human activity recognition [2]. Third, we have Action recognition based on HMM approach [3]. And the last similar system we have, was that of the team running the HoG over the GPU [4].

2.3 User Characteristics

Expecting a user who is familiar with security systems,This product's end user are security guards. Those guards expected to be middle aged or young adults with little knowledge of computers.

2.4 User Problem Statement

The market currently lacks an abnormal behavior detection surveillance system that works in real-time with very good accuracy without lowering down the processing speed to have good and acceptable frame rate for a security surveillance system

2.5 User Objectives

The user objective is the detection of car theft related abnormal behaviors to prevent car theft. The objectives can be broken down into the following:
1- Real-Time detection of predefined abnormal behaviors.
2- Pop up message if an anomaly happens.

3- Save recorded video of anomaly.
4- Save the criminal's face on the system.

2.6 General Constraints

1- The camera should be mounted facing the cars.
2- The system operates in well lit environments.
3- The system will only work with NVIDIA gpus due to cuda implementation.
4- The system will function in real time.

3 Functional Requirements

Table 1: Add camera

Name	Add camera
Type	Functional requirement
Input	IP camera address
Output	Live video stream
Description	The system will load a live video from the specified camera address and use it for processing.
Priority	10/10
Preconditions	Successful login, and a successfully connected IP camera
Post-conditions	System will start monitoring abnormal behaviors in specified ROI.
Dependencies	

Table 2: Set ROI

Name	Set ROI manually
Type	Functional requirement
Input	X, Y dimensions. and also the height and width of ROI
Output	ROI around a vehicle
Description	System allow the user to create a region of interest around a vehicle for detecting abnormal behaviors. The system will take the X and Y co ordinates and the width and height of the ROI as input. It then draws the ROI and adds it to the database accordingly.
Priority	10/10
Preconditions	Loaded video stream
Post-conditions	System Detects abnormal behaviors in the defined ROI
Dependencies	Live video

Table 3: Loitering

Name	Detect Loitering
Type	Functional requirement
Input	30 FPS 480P Live camera stream
Output	Suspect who is loitering around vehicle is marked with a box, and the abnormal behavior will be saved as a video in the anomalies section in GUI.
Description	The system starts running its detection in the set Regions of Interest. The system then tracks any humans who enter the vicinity of the car, and if they loiter around the car for more than 2 minutes(for example), it would detect the human as loitering around the vehicle.
Priority	10/10
Preconditions	System was monitoring the ROI for abnormal behaviors.
Post-conditions	System will continue monitoring the ROI for abnormal behaviors.
Dependencies	Load live video and ROI settings.

Table 4: Detect hand swing

Name	Detect hand swing
Type	Functional requirement
Input	30 FPS 480P Live camera stream
Output	Suspect who swung his arm around vehicle is marked with a box, and the abnormal behavior will be saved as a video in the anomalies section in GUI.
Description	The system starts running its detection in the set Regions of Interest. The system then tracks any humans who enter the vicinity of the car, if the human starts swinging his hand in order to break the glasses of a car, it would detect him as doing a hand swing.
Priority	10/10
Preconditions	System was monitoring the ROI for abnormal behaviors.
Post-conditions	System will continue monitoring the ROI for abnormal behaviors.
Dependencies	Load live video and ROI settings.

Table 5: Detect jumps

Name	Detect Jumps into vehicle
Type	Functional requirement
Input	30 FPS 480P Live camera stream
Output	Suspect who jumps inside the vehicle is marked with a box, and the abnormal behavior will be saved as a video in the anomalies section in GUI.
Description	The system starts running its detection in the set Regions of Interest. The system then tracks any humans who enter the vicinity of the car, if the suspect starts jumping into a car, it would detect him as jumping into the car and mark him.
Priority	10/10
Preconditions	System was monitoring the ROI for abnormal behaviors.
Post-conditions	System will continue monitoring the ROI for abnormal behaviors.
Dependencies	Load live video and ROI settings.

Table 6: Detect Faces

Name	Detect Faces
Type	Functional requirement
Input	30 FPS 480P Live camera stream
Output	Snapshot of the suspect's face is displayed in the GUI and is also saved on the system.
Description	The system will detect the face of the suspect if they perform an abnormal behavior. If a face is ever detected in our region of interest in parallel with an abnormal behavior, that face is saved as reference for the crime on the internal storage of the system.
Priority	8/10
Preconditions	An anomaly has been detected by the system.
Post-conditions	System will continue monitoring ROI for abnormal behaviors.
Dependencies	Load live video and ROI settings.

Table 7: Save video of detected anomalies

Name	Save videos of detected anomalies
Type	Functional requirement
Input	Abnormal behavior detected in a live feed or video.
Output	Confirmation message
Description	System will a video of the detected anomaly. If an anomaly is detected, the system saves the frames that the abnormality occurs in onto the internal storage of the system.
Priority	10/10
Preconditions	Pre-detected anomaly
Post-conditions	System continues monitoring abnormal behaviors.
Dependencies	Detection of an abnormal behavior

Table 8: Feedback alert

Name	Provide feedback alert
Type	Functional requirement
Input	Abnormal behavior detected in a live feed or video.
Output	Pop-up message
Description	System will show a pop-up message whenever an abnormal behavior is detected. When an anomaly detected it a red line will be displayed on the cam view detecting this anomaly with the anomaly type flashing in the tool bar of the systems UI
Priority	10/10
Preconditions	Pre-detected anomaly
Post-conditions	System will continue monitoring abnormal behaviors.
Dependencies	Detection of the Behaviors.

Table 9: Remove/modify ROI

Name	Remove/modify ROI
Type	Functional requirement
Input	X, Y dimensions. and also the height and width of ROI
Output	Modified ROI around a vehicle
Description	System allow the user to modify or remove a predefined ROI around a vehicle in a certain camera.
Priority	7/10
Preconditions	Live video stream, and a predefined ROI
Post-conditions	Detection now follows the updated ROI
Dependencies	Setting the ROI.

Table 10: Alter parameters

Name	Alter detection parameters
Type	Functional requirement
Input	New values for basic detection parameters
Output	Detection now follows the new parameters according to user's preference
Description	The user will have the ability to alter basic parameters of any added camera for detection to adapt the system manually
Priority	5/10
Preconditions	System runs detection on predefined parameters.
Post-conditions	System now follows the altered parameters in detection.
Dependencies	Adding camera.

Table 11: Reset parameters

Name	Reset detection parameters
Type	Functional requirement
Input	None.
Output	Detection now follows default parameters
Description	The user will have the ability to reset detection parameters to their default values
Priority	4/10
Preconditions	System runs detection on predefined parameters.
Post-conditions	Detection now follows recommended default parameters
Dependencies	Adding camera.

Table 12: View detected anomalies

Name	View detected anomalies
Type	Functional requirement
Input	Saved video of a detected anomaly
Output	Play video of saved anomaly
Description	The user will select and play a video of a saved anomaly. When anomaly detected the frames in which the anomaly happened in it will be saved internally on the system to be opened later for further deductions by the user.
Priority	10/10
Preconditions	System was monitoring abnormal behaviors.
Post-conditions	System continues monitoring abnormal behaviors in specified ROI.
Dependencies	Saving anomalies.

Table 13: View detected faces

Name	View detected faces
Type	Functional requirement
Input	Saved snapshot of a suspect
Output	Open a snapshot of a suspect
Description	The user will select and open a snapshot of a saved suspect that made an abnormal behavior
Priority	5/10
Preconditions	System was monitoring abnormal behaviors.
Post-conditions	System continues monitoring abnormal behaviors in specified ROI.
Dependencies	Saving faces.

4 Interface Requirements

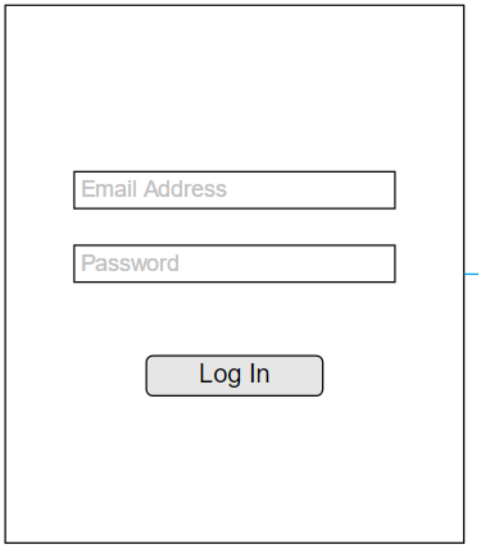
Vigil will display the results of post processing on screens in security management room.

4.1 User Interfaces

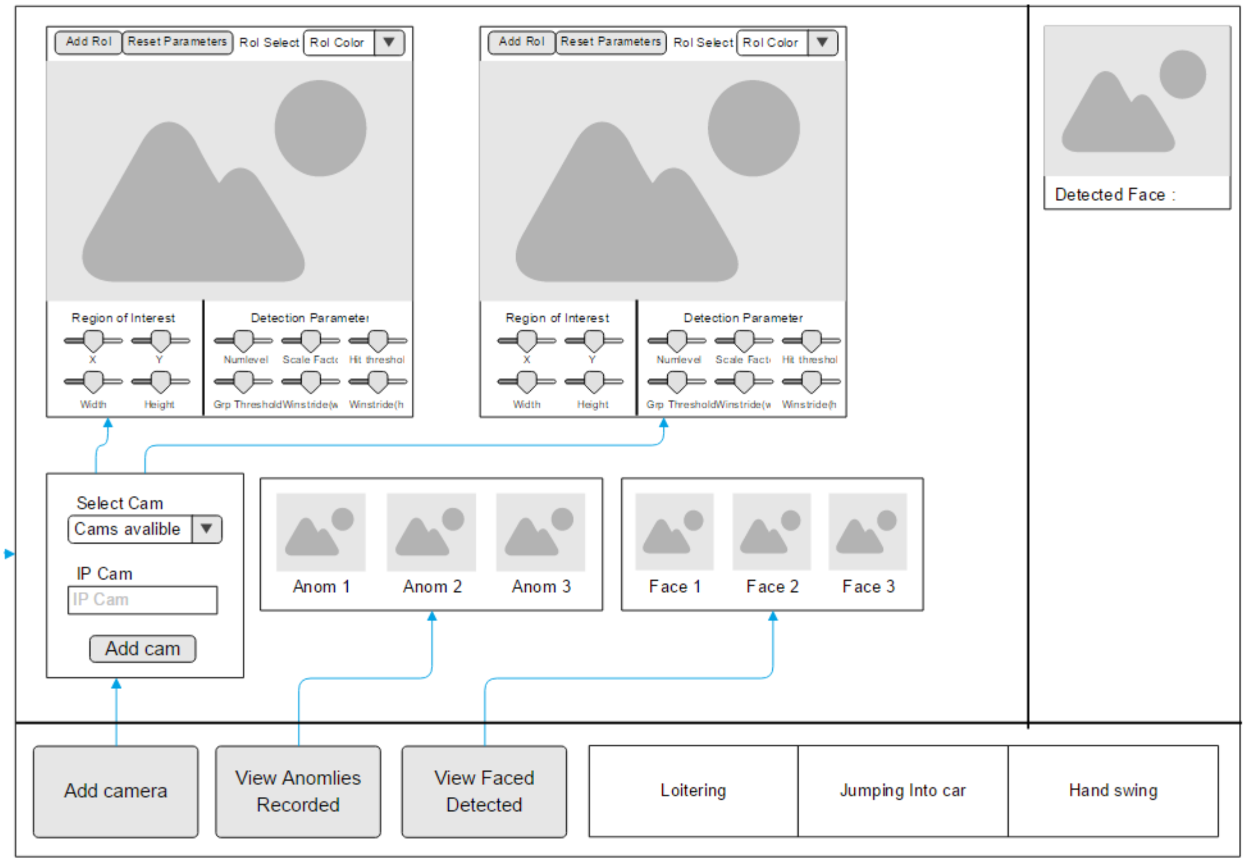
4.1.1 GUI

Vigil's user interface will feature the stream output window along with the detected face window. Bundled with track bars to manage all parameter related changes in a user friendly environment.

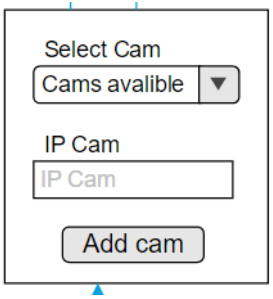
1.Login Screen



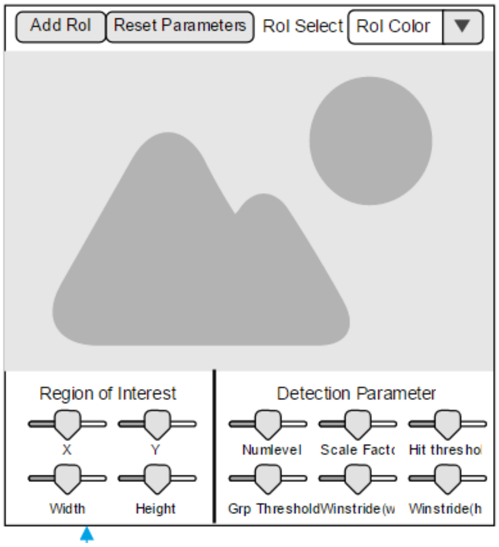
2.Main program UI



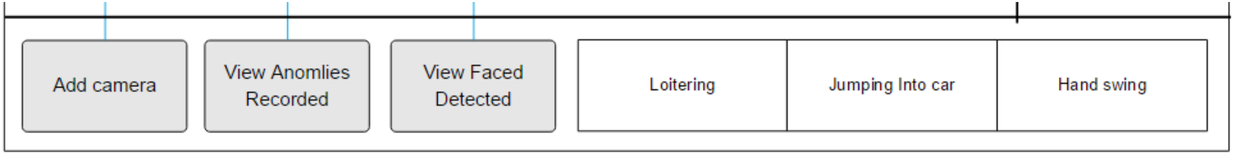
3.Add Camera pop-up



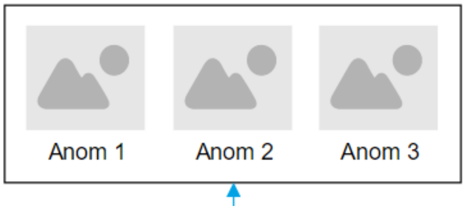
4.Camera Frame



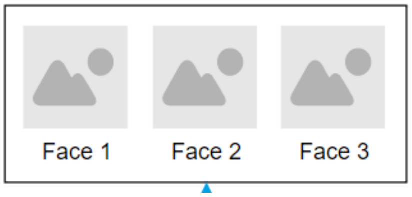
5.UI Controls



6.View Anomalies



7.View Faces



4.2 Hardware Interfaces

The system uses one IP camera, a system to process on with an Nvidia GPU, and a screen to display the output on.

5 Performance Requirements

The system must be capable of at least 20 frames per second due to the real-time detection requirement.
Required graphics memory is currently set at 2 gb DDR5 for ONE camera.
The hard disk storage requirement depends on the company using the system.

6 Design Constraints

The camera must be mounted facing the cars in a well lit environment.

6.1 Hardware Limitations

The camera's maximum frame rate and resolution. We are currently operating with an IP camera of maximum resolution of 480p and a frame rate of 10, however, we plan on investing in a 720p camera with a rate of 24 frames per second.

7 Other non-functional attributes

Table 14: NFR1

Non-Functional requirement 1	
Name	Processing over GPU
Type	Speed requirement
Date	6/2/2017
Description	The system must run an average of around 15 frames per second to be considered running around real time.
Priority	10/10.

Table 15: NFR2

Non-Functional requirement 2	
Name	Ease of system maintainability
Type	Maintainability requirement
Date	6/2/2017
Description	Ensure the systems ease of maintainability through the implementation of MVC design patterns, observer, and strategy design patterns.
Priority	10/10

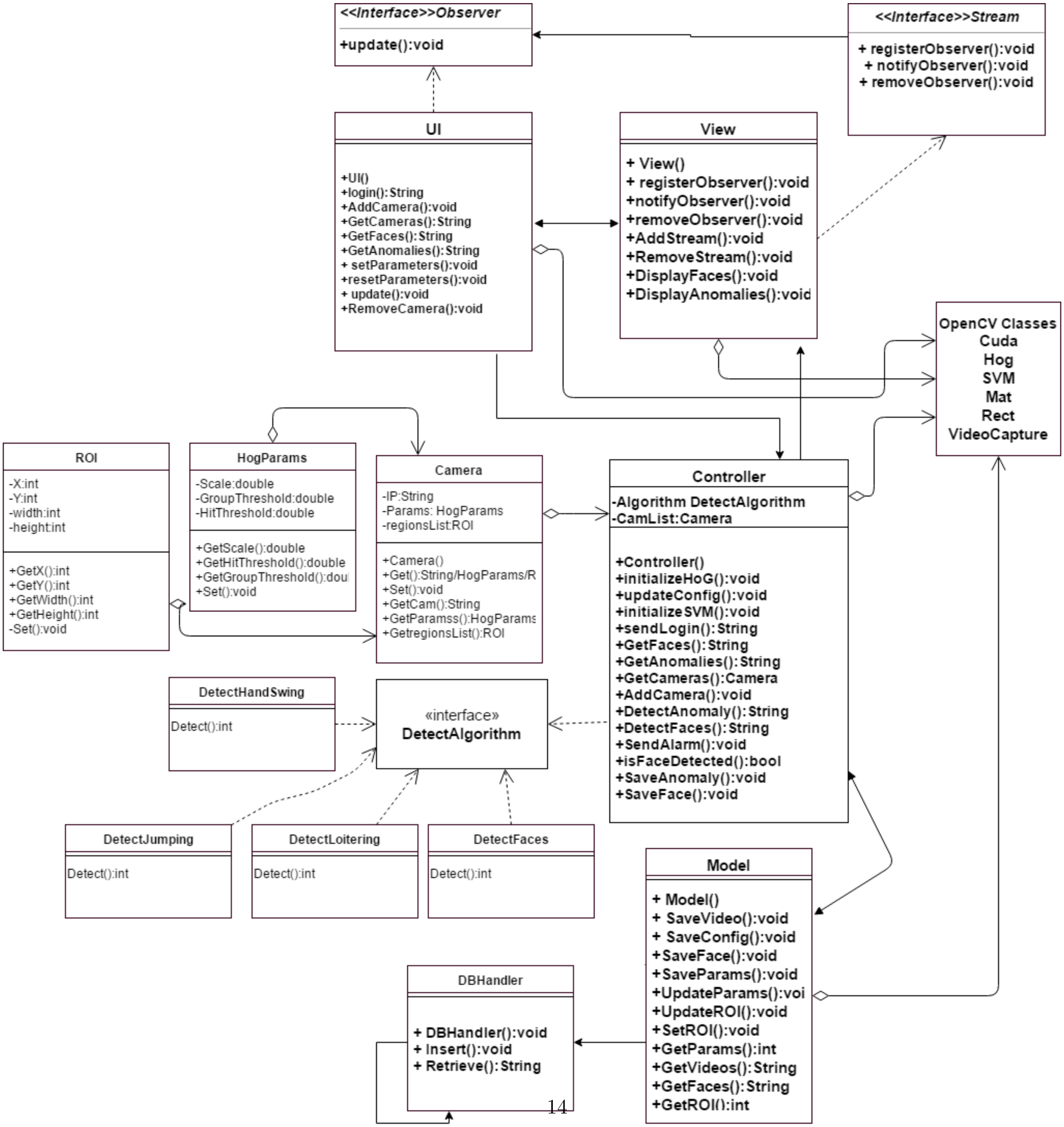
Table 16: NFR3

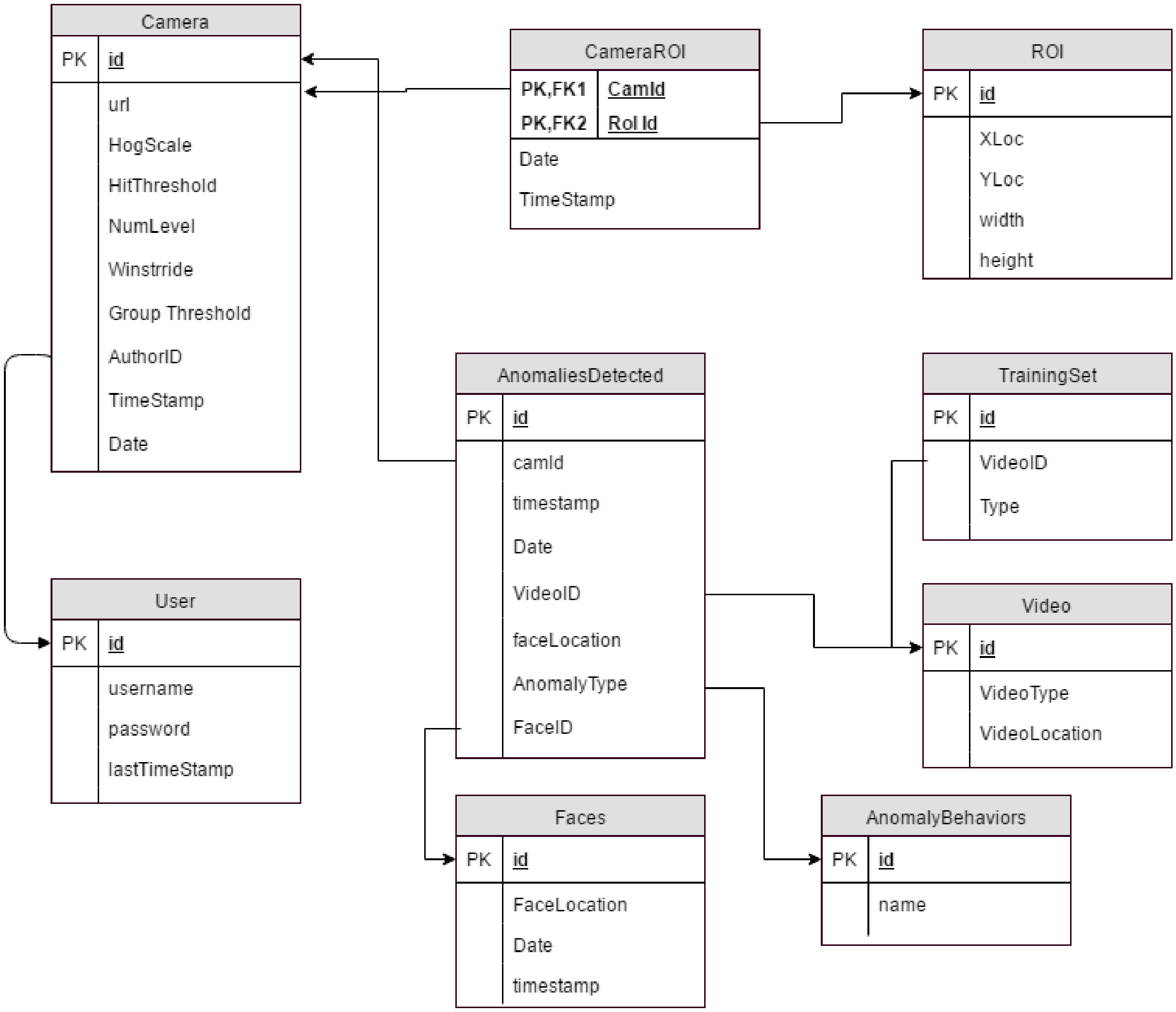
Non-Functional requirement 3	
Name	Stable consistent performance
Type	Reliability requirement
Date	6/2/2017
Description	The system should be reliable with a consistent performance rate averaging minimum 15 frames per second.
Priority	10/10

Table 17: NFR4

Non-Functional requirement 4	
Name	Ease of setup on multiple platforms.
Type	Portability requirement
Date	6/2/2017
Description	The system should be easy to deploy on multiple platforms, therefore, we will package the system with all the needed files and dependencies.
Priority	10/10

8 Preliminary Object-Oriented Domain Analysis




Camera
PK
id
url
HogScale
HitThreshold
NumLevel
Winstrride
Group Threshold
AuthorID
TimeStamp
Date
CameraROI
PK,FK1
CamId
PK,FK2
RoI Id
Date
TimeStamp
ROI
PK
id
XLoc
YLoc
width
height
User
PK
id
username
password
lastTimeStamp
AnomaliesDetected
PK
id
camId
timestamp
Date
VideoID
faceLocation
AnomalyType
FaceID
TrainingSet
PK
id
VideoID
Type
Video
PK
id
VideoType
VideoLocation
Faces
PK
id
FaceLocation
Date
timestamp
AnomalyBehaviors
PK
id
name

8.1 Class descriptions

8.1.1 Class name

Concrete: View
Concrete: UI
Concrete: Controller
Concrete: Model
Concrete: DBHandler
Concrete: Camera
Concrete: HogParams
Concrete: ROI
Concrete: DetectHandSwing
Concrete: DetectJumping
Concrete: DetectLoitering
Concrete: DetectFaces

8.1.2 Purpose:

View : Handles output views
UI : Handles the components needed in the view
Controller : Handles all the processing in the system
Model : It handles saving the data of the system
DBHandler : It handles all operations within the database
Camera : It handle camera control and receives hog controlling parameters
HogParams : it takes the HOG parameters to modify detection accuracy
ROI : it takes region of interest parameters to set the detecting region
DetectHandSwing : Detects hand swing anomaly
DetectJumping : Detects jumping anomaly
DetectsLoitering : Detects Loitering Anomaly
DetectFaces : Detects faces

8.1.3 Collaborations:

DBHandler , Dcryptor and Encryptor are all associative to the model class , and the model class is associative with the controller , and the controller is associative with the UI class , UI class is using an observer interface, the video class is using stream interface and it is associated with controller

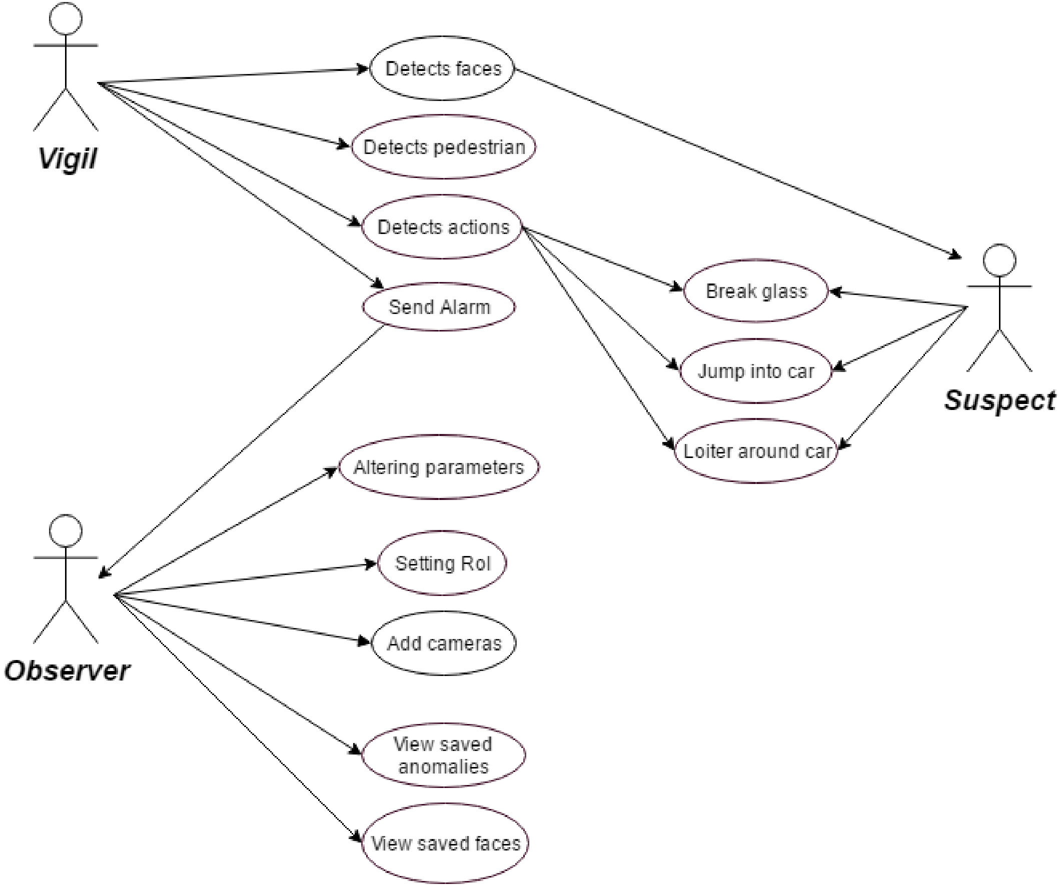
8.1.4 Operations

InitializeHOG : Starts HOG to start Detection
updateConfig : used to update the configurations of the system
InitializeSVM : use to start initalizing SVM
SaveVideo : Used to save video
SaveConfig : Used to save configurations
SaveFace : Use to save face detected

Encrypt : Use to encrypt the files saved
Insert : to start inser DB query
Retrieve : to Retrieve from DB using select query
Decrypt : to decrypt retrived files

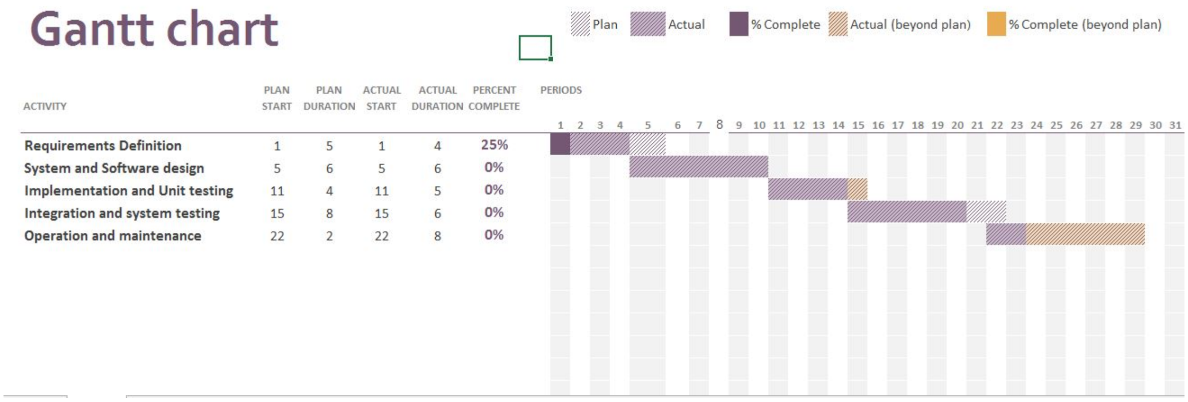
9 Operational Scenarios

Vigil is installed in a parking lot of an organization. Vigil will detect cars and draw a ROI around them. It then will start running detection in the defined ROI. Afterwards, if a person enters this ROI and starts doing one of the anomalies that Vigil was trained to detect, it will display an alarm on the screen to notify the security officer monitoring the screens. Vigil then also saves footage of the anomaly and the face of the culprit. The security officer sends an alarm for the guards to check the situation and resolve it.



10 Preliminary Schedule Adjusted

Our schedule has not changed from the initial proposed schedule.



Gantt chart

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Requirements Definition	1	5	1	4	25%
System and Software design	5	6	5	6	0%
Implementation and Unit testing	11	4	11	5	0%
Integration and system testing	15	8	15	6	0%
Operation and maintenance	22	2	22	8	0%

11 Preliminary Budget Adjusted

Same as our initial proposed budget amounted to about: 10,341.1 EGP

12 Appendices

12.1 Definitions, Acronyms, Abbreviations

GPU - Graphical processing unit.
ROI - Region of interest.
FPS - Frames per Seond.

12.2 Collected material

CCTV car theft videos from YouTube with unknown sources. (22 videos)
Arena 2014 Data set. (22 videos)
HumanEva Dataset. (7 training sequences for HoG)
A recorded data set will be produced by the team as well.

13 References

1. Human Behavior Monitoring Solution, Human Behavior Monitoring Solution: Video Security Solution — NEC Informatec Systems, Ltd. [Online]. Available: http://www.nec-nis.co.jp/en/product/videosecurity/hbms/. [Accessed: 12-Feb-2017].

2. D. Mai and K. Hoang, Motorbike theft detection based on object detection and human activity recognition,2013 International Conference on Control, Automation and Information Sciences (ICCAIS), 2013.

3. B. Chakraborty, O. Rudovic, and J. Gonzalez, View-invariant human-body detection with extension to human action recognition using component-wise HMM of body parts,2008 8th IEEE International Conference on Automatic Face Gesture Recognition, 2008.

4. M. Hirabayashi, S. Kato, M. Edahiro, K. Takeda, T. Kawano, and S. Mita, GPU implementations of object detection using HOG features and deformable models, 2013 IEEE 1st International Conference on Cyber-Physical Systems, Networks, and Applications (CPSNA), 2013.