**Security Access System Flow**

# System Flow Description

1. System Setup

* CAM1 - Camera 1 for Intrusion Detection à pointing at an immediate proximity area
* CAM2 - Camera 2 for Face Recognition à pointing forward

1. User Setup

* people authorized to work on the Smart City Demo asset
* Train 3 people, along with some negative samples

1. Intrusion Detection
   1. Idle State

CAM1 – idle

CAM2 – idle

* 1. Presence Detected
* CAM1 – when someone gets very close to the asset, trigger CAM2 Face Recognition
  1. Start Face Recognition
* CAM2 – triggered and run Python program
* Check if the detected person is one of the 3 authorized people, or unauthorized
* Capture the face of the unauthorized person
* Output the result
* Name of the authorized person OR Face of the unauthorized person
* Alert Mechanism
  1. Alert
* Send SMS to asset owner (94389636)

State the following message

* Detection Alert
* Unauthorized person: <date>, <time>
* Send email (francis.han@oracle.com)

State the following message

* Detection Alert
* Unauthorized person: <date>, <time>
* Attach picture of unauthorized person
* Append message into log file

# System Flow Charts

## Overall System Flow



## Intrusion Detection Flow

Purpose:

Check whether there is any suspicious object for intrusion and report detection.

Proposed API:

# Parameters:

# frame: Open CV captured frame image or video file frame image.

# Return:

List of detection result and mask image; Detection result is true if intrusion is detected, and False if not detected.

def perform\_intrusion\_detection(frame)

## Face Recognition Flow

Purpose:

Perform face recognition and retrieve list of recognized and unknown face name and images.

Proposed API:

# Parameters:

# frame: Open CV captured frame image or video file frame image.

# Return:

list of face name, face location (x, y, w, h) and face image;

Unrecognized face name is “unknown”.

E.g.

[ [“unkown”, [10,10,200,200], unknown\_face\_image],

[“people1\_name”, [15,15,200,200], people1\_face\_image],

[“people2\_name”, [25,25,200,200], people2\_face\_image] ]

def perform\_face\_recognition(frame)

## Alert Message Flow

Call AlertClient API:

* 1. Send SMS message

send\_sms(sms\_message, target\_phone=94389636)

* 1. Send email

send\_email\_with\_images(email\_content, subject\_message, alert\_type='Notice',

image\_name\_prefix='image\_', image\_list=[])