



OpenCV (Computer Vision) Library:

- *VideoCapture*: This class will be used for capturing and showing real time videos
- *Mat*: This class will be used for creating matrix so the video frame can be saved in matrix
- *CascadeClassifier*: this class will be used for face detection

- *Rect*: This class will be used for drawing a rectangular around the face detected by CascadeClassifier class

CNN (Machine Learning) Library and Eigen Library:

- *Eigen* library is library for saving data in format of matrix, vector and doing linear algebra
- *Convolution*: The output from Cascade Library, would first input to the Convolution as a subclass of Layer.
- *Relu*: The output from Convolution would input to the Relu as a subclass of activation function
- *Pooling*: The output from Relu would input to the pooling as a subclass of layer.
- *FullyConnected*: The output from Pooling would input to the Fullyconnected as a subclass of layer:
- *Softmax*: The output from Fullyconnected would input to the Softmax as a subclass of activation.
- *Output*: The output from Softmax would input to this class for applying loss function.

Database:

- *Admin*: This class contains all the required information about the administrator. To use the security camera, the administrator must register their information to the Admin class.
- *TrainedEntity*: it holds information of any other person who is deemed an authorized person. This class will hold names, ID and other information that will help recognize the person's face. Also, the TrainedEntity class will obtain a person's face matrix from the Output class in the Machine Learning Convolutional Neural Network section.

NotificationModule:

- The Admin class interacts with the NotificationModule class by sending the administrator's information to the NotificationModule when an alert/email needs to be sent.
- The TrainedEntity class will interact with NotificationModule class and Output class. The TrainedEntity class interacts with NotificationModule by sending the information of a person who just entered the home.

User Interface:

- *MainWindows*: includes the push button which captures and shows the live video frame. This live video frame is obtained from opencv library. This class has two

subclass: Pushbutton (for capturing video frame) and label (to set text for push button).

- *QApplication*: will execute the application.