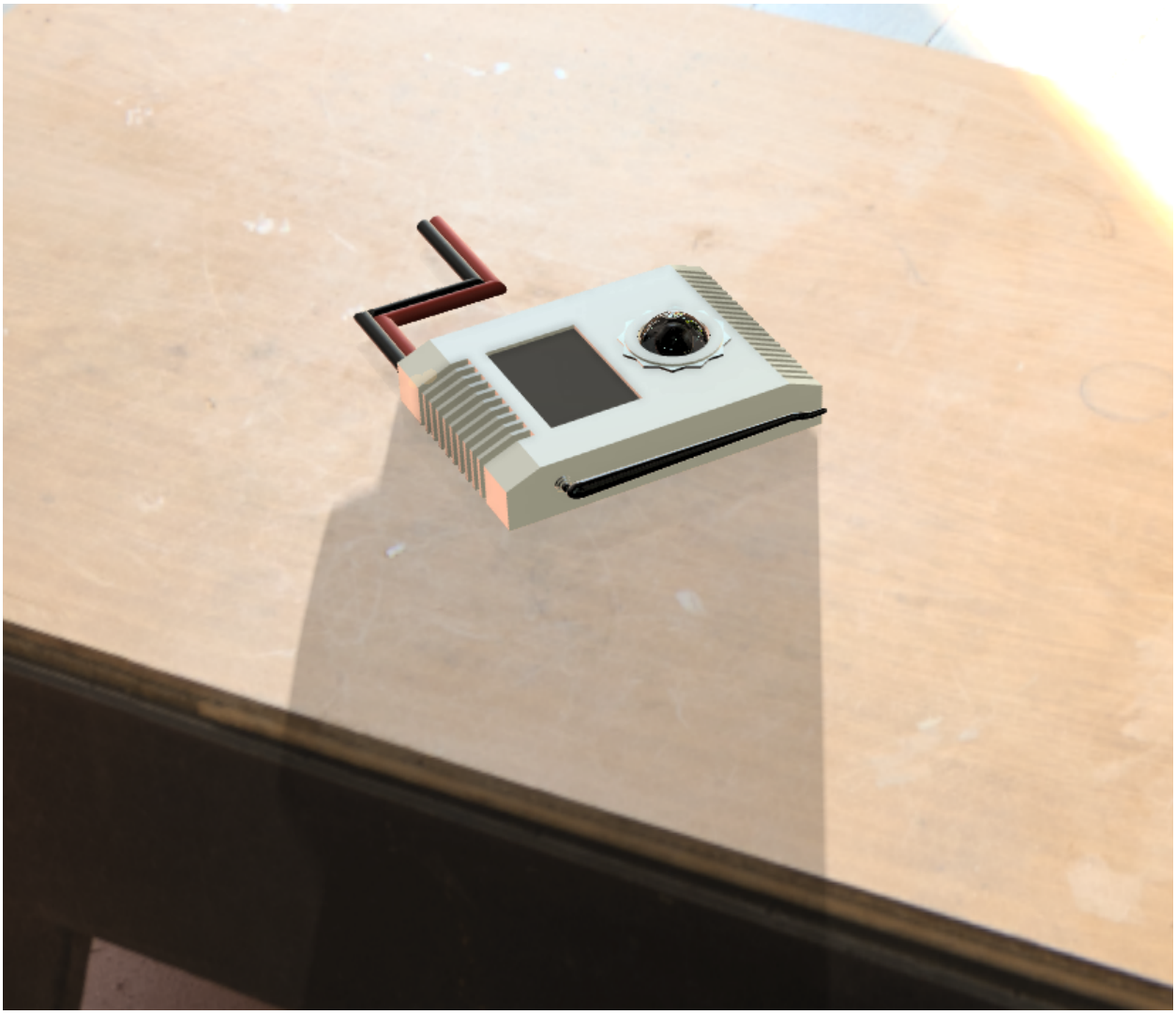
Automated Security System for Safe Homes

By: Pratye Aggarwal



(A hypothetical model of the final product I had in my mind.)

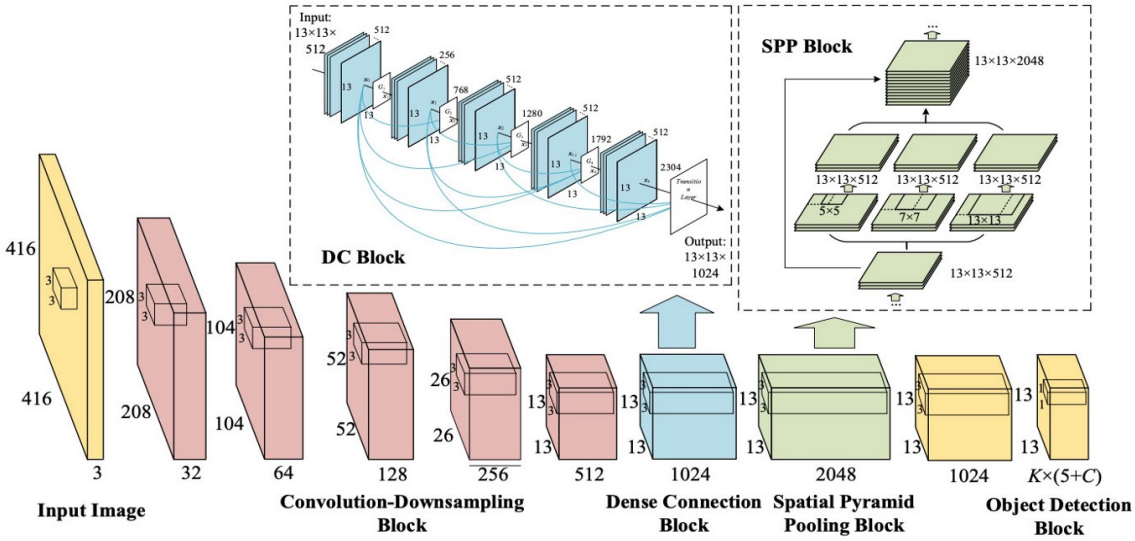
(made with fusion 360)

Working of the model:

1. It uses computer vision as a source of detection of the residents, household workers.
2. Through Computer Vision, our model detects Weapons, Criminal personalities through the police database.
3. The security system allows the entrance of residents through facial recognition. It also allows household helpers through a confirmation of the residences
4. When it detects something suspicious, it goes through a three-stage breakdown algorithm. Which have stages as follows:
    a. Weapon detection: weapon detected. And the face of the person with the weapon saved. It immediately notifies the residents. And secures the entrance with a double-locking system.
    b. The attempt of breach or breach results in closing of windows if the house is not breached. If residents are inside the house, then stage C is initiated.
    c. Breach detected. The model sends an SOS along with the data of the weapon, and the face of the person holding the weapon is sent to the police helpline “100”

Machine Learning Model:

For training our Machine Learning model, we used the network structure of the YOLOv4 model. The YOLOv4 architecture provides excellent results with extensive and diverse datasets. We trained for four classes: Knife, Pistol, Obama(being the residential identity), and Person (i.e., Unidentified people). 1000 Images for each class were used. The Images were labeled according to their classes. We then divided the dataset into a train and test set, with 1.8k images in the test and 2.2k images in the train. To create more training examples, we used a couple of image augmentation methods on the train set, producing over 6.8k images. Annotation and Augmentation of the images were done through an online tool, Roboflow(www.roboflow.com).



The data was exported and fed into the model for training purposes. The model was trained for over 2200 iterations and achieved an accuracy of 94.85%. The model loss shows that there is more room for improvement. The model can be trained for more iterations along with larger data sets yielding better and more consistent results.

Detection Process:

First, we take input from the camera to detect the different classes. The frames from the camera stream are taken, and then after a bit of preprocessing, like frame resizing, it is given as input to our trained model. The model outputs the detected classes and coordinates for the bounding boxes. These are first filtered according to their prediction accuracy and then used to perform different tasks like deciding whether or not to allow access and call the police.