International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 12 | Dec 2020 www.irjet.net p-ISSN: 2395-0072

A Framework for Crime Prediction and Analysis in Real-Time for Smart Home

Swathi BN[1], Vijaya K[2]

[1]*Student, Dept of ECE, BMS College of Engineering, Bengaluru, Karnataka, India*
[2]*Assistant Professor, Dept of ECE, BMS College of Engineering, Bengaluru, Karnataka, India*

Abstract - *There are a wide range of smart home surveillance and control systems, which will require some sort of visual identification and classification system. Surveillance cameras for security are deployed for monitoring purpose. Images taken from these cameras for criminology are typically used to determine the people involved when an event happens in post-crime intervention. In this Paper, proposed a Framework in smart home environment for real time crime prediction analysis using Webcam. This Framework uses Artificial Intelligence and event-driven approach to send crime data to protection services and police units to identify and validate crime incidents in real time allowing a rapid action. This framework uses intelligent motion detection algorithm where the process face recognition and object detection start only hen moment is detected and face is detected. Object detection is performed by creating labels and annotations for custom data, training the data by using Darknet and by creating a YOLOV3 object detector. Face recognition is then performed by using package face-recognition where recognition is based on face distance which is Euclidean distance between 2 images. This framework at end sends an alert mail depending on weather known with unknown or unknown person with or without object encountered at smart home. The mail contains a video clips attached.*

***Key Words*: Event driven approach, Intelligent motion detection, YOLOV3, Darknet, face recognition, Euclidean distance.**

1. INTRODUCTION

A home protection system ensures that your home is secured keep valuables secure and to safeguard your family from Potential break-ins by robbers and burglars. With the growth of the social economy and the improved quality of life, Family security has become one of the most significant aspects of life [2]. Nowadays there is a drastic increase in crime, such as burglary or theft, from small homes to larger factories. As the increase in crime activity, many research communities are paying attention for smart home security. Continuous monitoring of the people's behavior is needed for protection. Cameras which are used for tracking the human behavior commonly deployed in public and smart homes to increase the safety and protection. Surveillance camera devices for surveillance purpose has been used almost everywhere: health centers, banks, universities, streets and so on. These camera footages help for monitoring perimeter for intrusion, or to track suspicious public area instincts, etc. Due to the need to resolve problems of enormous storage and drastic processing time, the videos captured from these systems have attracted researcher's interests [5]. Several methods for human detection are suggested based on video analysis. The Method for analyzing videos taken from cameras uses AI model for the purpose of human behavior prediction, human identification. Surveillance system uses Artificial intelligence, Machine learning algorithms and deep learning for motion detection, face detection and human identification [5].

Recent surveillance cameras can record the video if motion is detected but in old surveillance cameras continuously records the video regardless of motion detected [5]. This will improve system efficiency by reducing processing, searching time and storage required to save the recorded videos. The protective services and authorities often fail to respond efficiently in crime incidents, because they follow reactive approach. In reactive approach authorities depends on witness report or closed-circuit television (CCTV) footage for analyzing about the crime after it had occurred. In most of the cases when an incident was occurred, investigators visit the site of the incident, manually retrieve the footage from camera, and then try to locate the appropriate footage either by watching the full length of the video or by Processing it by using advanced algorithms [1]. An efficient crime prediction analysis system for smart home is required to enable the robust security management, thus minimizing the crime incidents and losses. In this paper a framework for real time crime analysis and prediction in smart home using webcam is implemented. This framework has three main steps they are:

- Intelligent Motion detection
- Object detection
- Face recognition

Intelligent motion detection here, the system goes in search of face detection when the motion detected, if no face detected the system stops. From this step unnecessary processing is avoided there by increasing efficiency of the system. If face is detected then next step is object detection. The objects considered here are Gun and Knife. The object detection model is trained by using Darknet and YOLO V3. If the object is found in real time, then face recognition is the next step. The face recognition model is based on python

face recognition package. Face recognition model predict as per the encodings and Euclidean distance among real time face image and the images of known person. At the end of the framework mail will be sent based on three conditions they are: if unknown person with and without object, known and unknown person with and without object. This mail will have a video clip attached. Along with location.

2. Related Background Study

Tanin sultana et.al [1] implemented a framework called IOT-guard which is Intelligent, resource efficient, distributed Internet of Things, Edge-fog integrated video surveillance framework for real time security in smart home environment. In real-time, the IoT-guard will detect and confirm crime events using Artificial Intelligence (AI) and an event-driven approach. Then send the crime data police units and security services for immediate actions, preserving resources such as band width, memory, energy and CPU Usage. IOT-guard framework has three-layer architecture, they are edge node processing at crime location, fog node processing and crime prevention unit. In edge node processing motion detection is performed, in fog node by using VGGNet which is pre trained CNN model object detection is performed depending on weather the object found the crime prevention unit get the alert message having detected object image with crime location so that they can take prevention actions.

Jiang Landa et.al [2] based on the background modelling algorithm in computer vision, a remote embedded intelligent security monitoring system was developed that can proactively detect intruders. In this system by using camera background images are acquired and modeled by using ViBe algorithm and then perform object detection in the monitored area. The system will automatically trigger an alarm when a moving object (including human beings) is identified and send a message or call the user to take preventive measures. To detect the intruder and to get a better understanding of the situation, users can log in to the server via the mobile application. This system was implemented on ARM development board.

S. V. Tathe et.al [3] proposed a system for human face detection and recognition in videos. This system is divided into 3 stages: motion detection, face detection and face recognition. By using background subtraction in haar face detector region of interest is reduced which in turn improves the system performance. By using Eigenface method face recognition is performed.

Zoltan Balogh et.al [4] implemented motion detection, object recognition and face recognition using python. In this paper author implemented as mentioned algorithms using Raspberry pi 3 and the data is stored in cloud. The author was able to recognize the face and object in a single frame and also able to detect multiple faces.

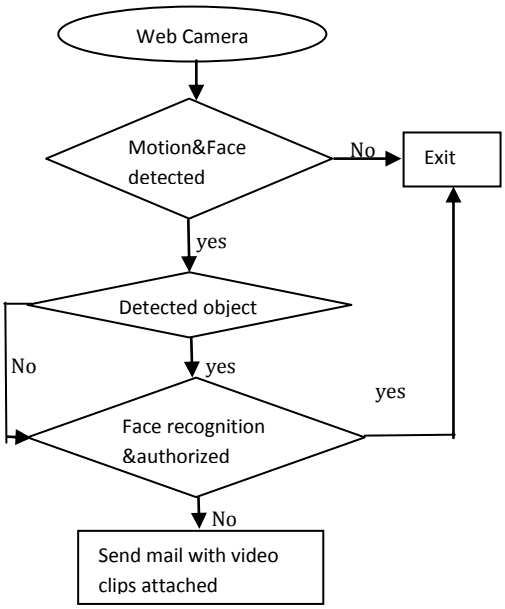
Eman Alajrami et.al [5] Designed and developed a desktop application based on AI which will start recording if person or human face is identified. This technique would increase the reliability of the system by reducing the recording processing and search time and by reducing the storage necessary to preserve recordings. This proposed framework built on Linux OS uses deep learning algorithms, OpenCV libraries.

Joseph Redmon [6] designed a YOLOV3 algorithm. Here the author has presented some design changes to YOLO for better performance in terms of speed and accuracy. Omkar Masurekar [7] implemented a real time object detection model using YOLO V3. Object detection is done by using Anchor box and by performing Non maximum suppression. Training of the object detection model is done using Google Colab for GPU support.

Veta Ghenescu [8] implemented a method for detecting object on long range thermal images by using YOLO Darknet 2.0. This method was able to detect and identify the object that are hard to detect with human eye. This model trained to detect the objects in visual spectrum by modifying standard network and retraining it on thermal data set which has low quality images.

Wei Cui [9] designed a garbage detection model using YOLOv3-darknet model which is based on adaptive clustering anchor box. Using a camera in a sanitation vehicle, pictures of waste on the roadside can be Captured. This model was able to produce good results and it was able to accurately classify three types of garbage in complex scene in less time.

3. Methodology



3.1 Motion Detection

Motion detection is the mechanism by which a change in the location of an object relative to its background or a change in the background relative to an object is detected. The key

applications of motion detection are the detection of unauthorized entry and the detection of a moving object that allows a camera to record subsequent events. A simple motion detection algorithm compares the current image to a reference image and simply counts the number of different pixels. Due to factors such as changing lighting, camera flicker, and CCD dark currents, images will naturally differ, pre-processing is useful to minimise the number of false positive output. For detecting the moving objects in video, background subtraction model is used. Motion detection algorithm has following procedure: A video capture object is created to capture live stream video using OpenCV. Obtain numpy array for the frame1 and frame3 then find the difference between those frames and calculate Pythagorean distance between first frame and third frame using equation: Consider A has first frame matrix (NumPy array), B has a third frame matrix (NumPy array), C has a difference matrix, P has an obtained Pythagorean distance in matrix form (NumPy array). Considering A, B is 3*3 matrix:

C=A-B

$$P = \frac{\sqrt{(C[:: 0] ** 2 + C[:: 1] ** 2 + C[:: 2] ** 2)}}{\sqrt{(255^2 + 255^2 + 255^2)}}$$

Then normalize and convert the data type of image from float32 to uint8. Next step is to calculate the Gaussian blur, threshold and mean Standard Deviation which are built in OpenCV functions for calculating the standard deviation. The threshold standard deviation is to be properly selected which is usually selected between 15-20. If the obtained standard deviation is greater than the predefined threshold standard deviation then the system concludes that motion is detected else it concludes as no motion detected. This process continues for whole frame in live stream video until a face is detected if no face detected for 10 steps then system exits from further processing.

3.2 Face Detection

If motion is detected then next step is to detect face in live stream. Face detection is performed by using Haar feature based cascade classifier which is an effective detector of objects. It is an approach based on machine learning. Lot of Positive and negative images are used to train the cascade function, then it is used for comparing with other images for object detection. There are huge individual XML files with lot of features, each xml files have a specific use case feature. Here for face detection, haarcascade_frontalface_default.xml is used which has features for detecting the front face. This xml has values which is obtained when training with lot of positive and negative images for detecting the front face. Face detection model is designed using OpenCV which is most familiar way to detect the face.

3.3 Object detection and Recognition

Object Recognition is one of the computer vision technique for identifying instances of objects in images or videos. The primary objective of object detection is to replicate the human intelligence of detecting object in a video or image to computers. The use cases of object detection are infinite some of them are monitoring objects, video surveillance, pedestrian identification, identification of anomalies, people counting, self-driving cars or face detection and so on. Object detection model is designed based on YOLOV3 and Darknet for custom data. Custom data here considered are Knife and Gun. Following are procedures involved in designing the object detection model are:

1. Set up YOLO V3 on windows. Install all dependencies they are Visual Studio 2019, CUDA>=10.0, cuDNN>=7.0, CMake>=3.12, OpenCV>=2.4. Ensure to add OpenCV, CUDA, cuDNN directory in environmental variables. Then clone the darknet directory from https://github.com/AlexeyAB/darknet. Set up the config file with the CUDA version installed cuDNN directory in environmental variables. Then clone the darknet directory from https://github.com/AlexeyAB/darknet. Set up the config file with the CUDA version installed and then build the solution using visual studio 2019 which will generate darknet.exe file. Copy cuDNN64_7.dll, OpenCV ffmpeg420_64.dll, OpenCV_world420.dll file to darknet bin folder.
2. Using Image annotation tool Labelimg which is a powerful tool used for image annotation and labelling. Using this tool labelImg and annotation and labelling is done and save the file generated for each custom image in txt file which contains annotation and labelled values for each image. Define class file which has names of the objects that should be detected.
3. Prepare config file for custom data by modifying the yolo config file in darknet. Create object name folder for training and object data folder which has train data path, validation data path, classes, names of custom object file and path for storing the trained data. Download pre trained CNN weights for YOLO.
4. Train using darknet for custom data using pre trained weights.
5. Using an object detector model coded using yolo v3 and OpenCV for detect the objects in real time which is able to detect the knife and gun.

3.4 Face Recognition

The next step after object detection is face recognition. Face recognition model is built using a Face recognition function defined by adam geitgey. Face recognition package is downloaded using pip command. Then built a model to recognize the face in real time. Humans are capable to identify the person easily and quickly but computers cannot. In order to make computers to do that following procedures are involved they are: find face in image, analyse facial

features, compare against known face and then prediction. The first step is finding the face which involves, convert the RGB image to gray image then divide the image into 16*16 pixel each. For each pixel calculate the gradients point in each major direction replace that square in the image with the arrow directions that were the strongest. Using HOG Face is detected for given image. Face landmark estimation algorithm is used for locating the 68 face landmarks on given image, condition is that eyes and nose should be visible in image. A pre trained convolution neural Network "OpenFace" is used to generate 128 measurements for each face. By calculating the Euclidean distance between the image encodings, comparing the distance face prediction is performed.

4. Obtained Results

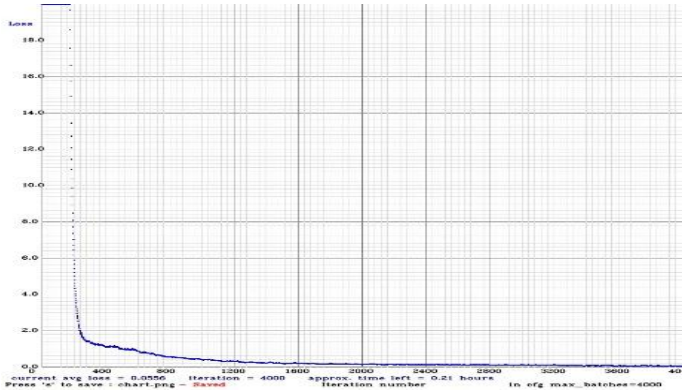


Figure 1: Graph obtained while training for object detection

In Figure 1 is the graph obtained while training the object detection model using yolov3 and darknet. This graph implies that loss occurred during the training drastically decreased with number of iterations.

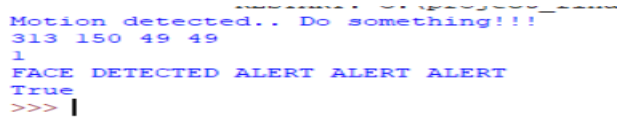


Figure 2: Result obtained when motion is detected.

In figure 2 when motion is detected and face detected, in python shell we will get a statement mentioned above

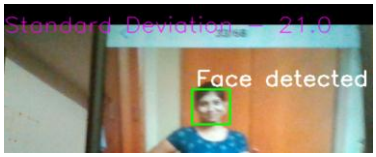


Figure 3: Face detected in real time

In figure 3 the face detection model was accurately detecting the face when the motion is detected.

Figure 4: Object detected in real time

Figure 5: Object detected in real time

Figure 6: Knife Object detected in real time

In figure 4 and figure 5 object detection model was able to detect the object Gun in real time with efficiency 0.4231. In figure 6 Object knife was accurately detected using object detection model.

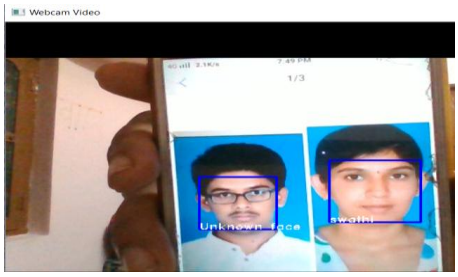


Figure 7: Face recognition output

In figure 7 the Face recognition model was able to predict the known person and unknown person effectively.

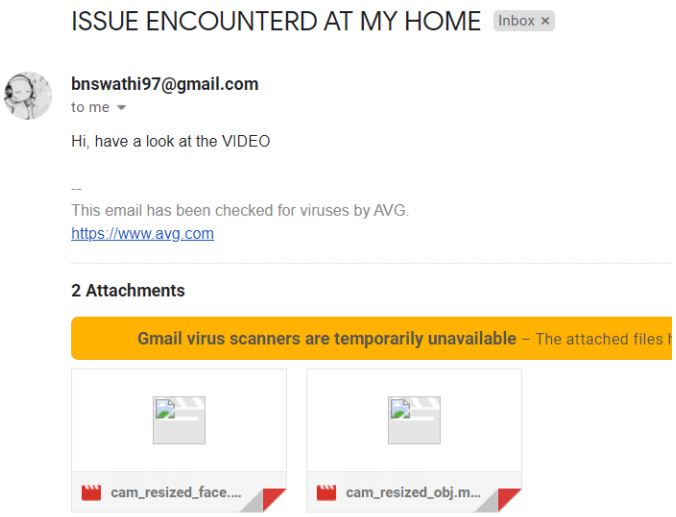


Figure 8: Mail output

Figure 8 is the mail output got when both objects detected and unknown with known person encountered.

5. CONCLUSIONS

In this paper an intelligent crime prediction framework is designed. This framework includes 3 main steps they are intelligent motion detection, Object detection and face recognition. Intelligent motion detection is the one which effectively detects the motion and also face detection is added to make the system to proceed further only if person is encountered when motion is detected if no person present then this framework ends its process. Proceeding with next step is Object detection and face recognition. This model can be used to analyse and predict the actual crime event at smart home. This framework can be used in various scenarios not only in smart home scenarios. Object detection model here can predict only 2 objects which can be scaled further to detect multiple objects.

REFERENCES

[1] Tanin sultana, Khan A. Wahid "IOT-Guard: Event driven fog-based video surveillance system for real time security management" IEEE Access, pp-134881-134894 17 September 2019.

[2] Jiang Landa, Chu Jun, Miao Jun "Implementation of a remote real-time surveillance security system for intruder detection" 2017 9th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA) 26 January 2017.

[3] S. V. Tathe, A. S. Narote, S. P. Narote "Human Face Detection and Recognition in Videos" 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI) 03 November 2016.

[4] Zoltán Balogh, Martin Magdin, György Molnár "Motion Detection and Face Recognition using Raspberry Pi, as a Part of, the Internet of Things" June 2019 Acta Polytechnica Hungarica 16(3): 2019-167 DOI: 10.12700/APH.16.3.2019.3.9.

[5] Eman Alajrami, Hani Tabash, Yassir Singer, M.-T. El Astal "On using AI-based human identification in improving surveillance system efficiency" 2019 International Conference on Promising Electronic Technologies (ICPET) 09 December 2019.

[6] Dr. Jayakumar Kaliappan, Jain Shreyansh, Shanmuga Sundari. P, Mohan Sai Singamsetti "Surveillance Camera using Face Recognition for automatic Attendance feeder and Energy conservation in classroom" 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN) 14 November 2019

[7] https://pjreddie.com/media/files/papers/YOLOv3.pdf

[8] Omkar Masurekar, Omkar Jadhav, Prateek Kulkarni, Shubham Patil "Real Time Object Detection Using YOLOv3" International Research Journal of Engineering and Technology Volume: 07 Issue: 03 | Mar 2020

[9] Wei Cui, Wei Zhang, Juli Green, Xu Zhang, Xiang Yao "YOLOv3-darknet with Adaptive Clustering Anchor Box for Garbage Detection in Intelligent Sanitation" 2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE) 18 May 2020

[10] https://en.wikipedia.org/wiki/Motion_detection#:~:text=Motion%20detection%20devices%2C%20such%20as,disturbance%20in%20the%20infrared%20spectrum.&text=A%20simple%20algorithm%20for%20motion,the%20number%20of%20different%20pixels.