Video Surveillance Using Deep Learning

Joseph George1, Sreelakshmi Ramchandran2, Nezmi K A3 and Shreya Nair4

*Dept of CSE, Adi Shankara Institute of Engineering and Technology, Kalady, Kerala*

***Abstract*—A large amount of digital data from social media, research, agriculture, medical records and other IoT related applications are consumed in research and industry for big data applications. Video footage from surveillance cameras also add to the example of big data. Surveillance videos contribute to unstructured big data. In the places which are prone to many crimes and attacks, security is assured by implanting CCTV cameras. Cameras and recording equipment are relatively expensive and require human intervention to monitor the camera footage. Along with it, there is a large amount of data generated which requires vast storage. All this, demands more manpower and increased cost. In an era where we have smart gadgets, it is time to get our surveillance cameras smart as well. To do so we empower our surveillance cameras with deep learning algorithms. Our goal is to focus on automatic identification of unauthorized entries in an area by alerting the authorities. The paper discusses different deep learning methods that are used for surveillance. Our model is based on the deep learning technique in which the live video stream is taken from an IP camera. From that video footage, the faces of authorized and unauthorized one are recognized automatically which gives an accuracy of 99.93%. If any unknown person were found in a frame, a message notification along with the person’s image is sent to the registered mobile number of the officials. Hence this method reduces manpower and saves a lot of time. This implementation has a wide range of applications which includes personnel identification, bank security, company security, shop security etc. Also, it helps with instant crime detection which helps the community to identify the culprit easily. Finally, we suggest some directions as part of future work.**

***Keywords*—Big data, Video surveillance, Deep learning**

1. INTRODUCTION

The act of video surveillance involves analyzing a scene and looking for behavior that seem to be improper. Video surveillance are commonly used for remote video monitoring, facility protection, monitor operations, public safety etc. Artificial intelligence enables machines to process like humans. The machine is trained using datasets and is tested. Thus machines would be able to identify images as the humans do. Deep learning concepts have come about as a result of availability of large amount of data and high performance computer. Deep Learning helps identify and extract features so that we can easily distinguish between objects. Video surveillance is socially relevant in today’s world where the crime rates are increasing at an alarming rate. The output from surveillance camera contributes largely towards unstructured big data. Easy processing of this video is of priority.

In this project the camera analyses all people appearing in a frame and then use the face recognition to classify the image as authorized or unauthorized. If the person is found to be unauthorized, then the system notifies the officials about the intruder. Thus the authorities will be informed of any threats that may occur.

The cameras implanted in industrial, residential and commercial areas have widespread contribution towards surveillance data. Cameras implanted in public places such as malls, parks, bus stations, airports etc. are also contributors. Surveillance video analysis involves object recognition, face recognition and classification of actions. The accuracy rate of face recognition has significantly improved through deep learning technology. A key advantage of deep learning based algorithms over computer vision algorithms is that deep learning system can be continuously improved and trained with better and with more datasets. This paper provides with specific focus on solutions that are based on deep learning.

1. RELATED WORKS

The recent advances in deep learning technology, image processing and computer vision have brought to an increasing interest in surveillance of live videos. Due to the existence of huge and analytical information in videos and their easy availability, researchers have been interested in the analysis and processing of the videos. Security plays a vital role in every area, because of this crime rich society. So, video surveillance is becoming a controversial research topic in recent days. An efficient method based on deep generative network that is an unsupervised probabilistic method to model the surveillance and learn the future representation automatically was proposed by Hung Vu [1]. The advantage of this method was this system can detect abnormality at different levels that can be efficiently used for video analysis and scene understanding. An undirected generative network of a visible layer and a hidden layer is Restricted Boltzmann Machine (RBM) [2] because of the bipartite structure, RBM can learn the data distribution accurately in comparison to deep generative networks layer. Their success in training deep networks and the rise of deep learning [3] is due to their pre-training roles. Due to this reason, RBMs are used by them to examine the possibility of applying generative networks to video anomaly detection. Deep generative nets (e.g. DBM) have more than two layers hence they are known as multi layer networks. This will facilitate the ability of hierarchical feature representation and better distribution modeling. A novel deep network is aimed by this method that includes detection of abnormality in every laye0r of the network. This network is expected to produce better results and hierarchical detection results because of the proposal for the anomaly detection problem by a special generative network.

There are techniques based on Deep Neural Network (DNN) [4] to model normal behavior. A DNN is created that learns to predict future frames from past frames using a normal dataset. The results from predictions are then compared with testing video for similarity, and the resulting error is used to detect anomalies. Here the anomalies are detected using the general features that are automatically extracted from the video input. Precisely for each frame of an input sequence, a stack of Convolution Neural Network (CNN) is used. After that based on the input sequence, a Convolution Long- Short Term Memory (ConvLSTM) stack is used to predict the future motion sequence. These steps jointly capture the spatiotemporal patterns contained in the input videos. For the detection of an anomaly, after getting the output video sequence from an input video sequence, an error is computed between the two, which is the threshold to check if the input video sequence is anomalous or not. This approach gives a regularity score for frames and does not attempt to localize the anomalies within the frame.

A study for anomaly detection in video using deep learning method has been done in [5]. This paper comprises of two components. One is the extraction and learning of feature and the other is identifying anomalies. Both training and testing phases are done, where, in the testing face normal frames are used and in the test face frames with anomalies are used. In the paper four different types of features are used. The first feature is appearance which includes object detection in an individual frame. In the second feature, density is considered which helps identify density of objects in each frame. For the third feature motion is identified by comparing consequent patch frames. And in the last feature the scene is identified from the patch frames in order to recreate a scene from the learned model. The combination of these features is also used for the detection and creation of scores.

There is another method using Mask R- CNN [6] that suggests intelligent monitoring of indoor surveillance video. Here in this method they are trying to understand the semantic meaning of the video data by using instance segmentation and target detection. Also, it helps to retain the important information of the original video. This method reduces the memory and processing power of monitoring video. Also the alert mechanism for the abnormal event greatly reduces the effort that taken by the human personnel.

1. EXISTING METHOD
2. *Traditional cameras and manual surveillance*

Surveillance systems may be used as smart watchdogs, which are used to observe a restricted area, identify trespassers, unauthorized entries and therefore identify authorization. They are often connected to a recording device (Surveillance camera) and IP network watched by a security guard or law enforcement officer. Manual operation is needed for this traditional cameras and a person want to monitor the video footage continuously. In normal cases what we are doing is, if any abnormal activity took place we will check for the video footage and find the culprit. It is the manual way of video surveillance. This method requires human personnel to monitor the camera footage. Also real-time alert is not possible in this method. We have to check the video after a particular unauthorized event has occurred and we want to find the unauthorized one. Also, the video data generated is in huge amount, hence monitoring the video footage is becoming a difficult task and time-consuming. By using this method we didn’t get a real-time status of the security places also the lack of appropriate alert systems.

1. *Video surveillance using the Raspberry PI architecture*

In this method, Raspberry PI architecture is used as a surveillance system. Raspberry PI B+ model can be used for this purpose; it will have the features of the basic computer. The hardware component included in the project was Raspberry Pi processor, stepper motor, webcam, PIR sensors and GSM module. The video footage is captured using the webcam and PIR sensors that are interfaced using a Raspberry Pi. The footage can be locally and remotely visualized. A 360° Surveillance can be achieved using a stepper motor. Video is processed in the Raspberry Pi and data is send to the officials. PIR sensor is used for motion detection. If the PIR sensor detects any motion the raspberry pi will send notification using the GSM module. LCD and alarm are used to give the surveillance message locally. Data can send anywhere at any time because of the wireless medium. This method can be used in hospitals, city buses or WIFI enabled train. But it does not detect intruder person by identifying authorized or unauthorized personal. It just detects the motion using PIR sensors and sending notification. It requires hardware components and its installation. Also, require power consumption.

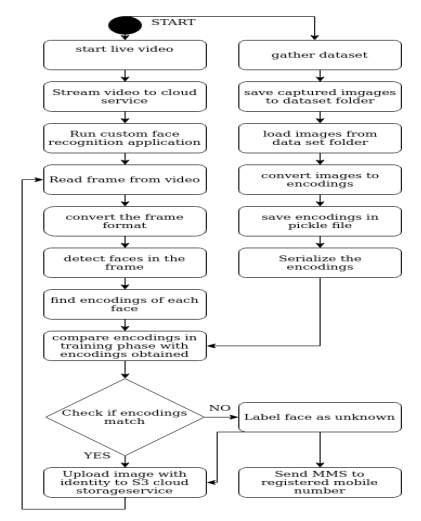
1. PROPOSED METHOD

In the proposed system, live video stream is captured from an IoT camera, which is streamed and analyzed through cloud services. Custom made face recognition application is deployed through the modern techniques used in cloud. During the training phase, the faces of authorized people is converted into encodings using DLIB’s face recognition tool which gives an accuracy of 99.38% on the standard LFW face recognition benchmark. The model used for training is DLIB’s face recognition RESNET v1. In the live streaming, the face recognition application analyzes one frame at a time. The detected faces in each stream are converted to encodings and compared with encodings obtained in the training phase. After comparison, we get a list of identities which comprises people that appeared in a frame. If the encoding of a person captured in a frame did not match the encodings obtained in the training phase he/she is labeled as “unknown”. Images of all people appearing in the frame are uploaded to Amazon S3 service along with the name and timestamp and if an unknown person appears in a frame a message notification along with the person’s image is sent to a registered mobile number using Twilio notification API. Figure 1 demonstrates the flowchart for the proposed system.

1. *Data Gathering*

The faces of authorized people are gathered live from a webcam. We need to detect faces in each frame from the video captured from the webcam. Face detection is used in numerous applications. Viola-Jones face detection algorithm [7] is the most popular for face detection. Facial features are detected and others are ignored. Single or multiple faces can be detected using this algorithm. OpenCV library is utilized for face detection as they are the best open source library available to perform image processing.

In Haar cascade based detection algorithm a face candidate passes to the next phase when any feature is discovered. A face candidate is a rectangular part of the original image known as a sub-window. This



**Figure 1: Architecture of proposed system**

Sub window size is usually 24x24 pixels. To get the faces, window is often rotated. The Haar cascade algorithm scans the entire image within this window and denotes all relevant part as a face candidate [8] [9]. There are four stages for Viola-Jones face detection algorithm. They are Creating Integral Images, Haar feature selection, Cascading Classifiers and Adaboost Training.

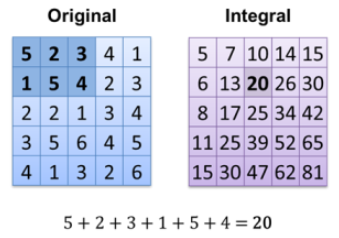
Haar features are to be collected as the first step. Features used are similar to Haar basis functions which have been used by Papageorgiou et al. [10]. Adjacent rectangular regions at a specific location in a detection window is considered by a Haar feature and sums up the pixel intensities in each region and calculates the difference between these sums.

The process is made faster by using integral images. Integral images are those images in which the pixel value at any (x, y) location is the sum of all pixel values present before the current pixel. The idea is to change input images into a summed-area table, where the value at any point (x, y) in that table is the summation of all the pixels left and to the above of (x, y), inclusive. This is shown in equation 1.

Where I(x, y) is the value of the integral image pixel in the position (x,y), while i(x,y) is the original image’s corresponding intensity. It is a recursive formula; hence, if we take the input image and start from one of its corner, we will have the same output in the integral image. This is demonstrated in figure 2 and 3:



**Figure 2: Haar like Feature Example**



**Figure 3: Integral Image numeric calculation**

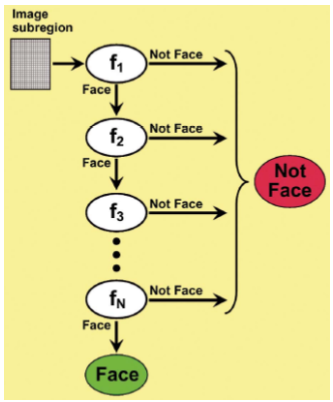
The number of possible rectangular features is 180,000 for a 24x24 detection zone. The best featured is selected by using Adaboost which selects the best features and trains the classifiers that use them. The better performance feature is selected based on the weighted error it generates. The weighted error is a function of the weights belonging to the training instances. The weight of a correctly classified case is decreased and the weight of a misclassified case is kept fixed.

Cascade classifier phase is the final step for the Viola-Jones face detection algorithm [7]. Cascade stage is utilized to eliminate face candidates speedily. A cascade classifier comprises numerous phases of filters, to determine if a given sub-window is definitely not a face or a face. If a face candidate fails any one of the phases the candidates exit the cascade. The cascade classifier will directly reject the area if it fails to pass the threshold of a phase. If a face candidate passes all phases, the face candidate will be classified as a face. This is demonstrated in figure 4.

At the end of the four stages, we get the faces that have been detected. Each frame is resized to just get the faces in that framed and stored in the dataset folder.

1. *Training Phase*

In the training phase, we need to convert the images in the dataset to encodings. To achieve this deep metric learning technique is used. Deep metric learning is used in order to generate a real valued feature vector rather than generating a single label. Here we use dibs face recognition tool. It gives a 128-d feature vector i.e., 128 real valued number to quantify a face.

**

**Figure 4: Phases of Cascade**

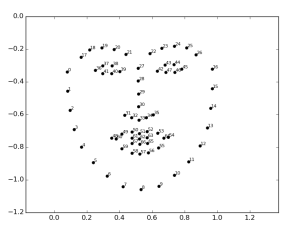
Dlib face recognition tool helps us to map faces to 128D vector space. Different images of the same person will be more similar and images of different people will be far apart. To check the similarity of different images of the same person it makes use of the Euclidean distance. If the Euclidean distance is small enough then the images belong to the same person else images belong to different people.

We use a distance threshold is kept 0.6. This helps the dlib model obtain an accuracy of 99.38% on the standard LFW (Labeled Faces in the Wild) face recognition benchmark. This accuracy implies that, when given a pair of face images, the tool can properly establish if the pair belongs to the same person or different person 99.38% of the time.

This model is a ResNet network which is a pre-trained model with 29 Convolution layers derived from a 34 layer network described in Deep Residual learning for image classification [11]. The network has been trained from scratch on a dataset of about 3 million faces derived from the scrub dataset, VGG dataset and other random images from the internet. The network training was done with randomly initialized weights and then used a structured metric loss. For each image from the data set the images are converted using the face recognition library which is wrapped around the dlib face recognition tool. These encodings are then stored in a pickle file. Using a pickle file helps to serialize the object before it is written into the file. This is done in order to easily reconstruct the python script.

1. *Face Recognition*

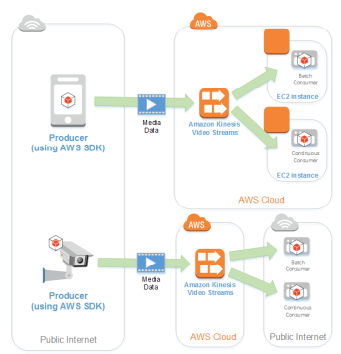
In the face recognition section, we stream the video from IoT camera to cloud. Amazon web services have been used as the cloud service. The stream is sent to AWS Kinesis Video stream. Amazon Kinesis Video Streams is a service provided by AWS which helps to stream live video from IoT devices to make custom applications for real-time video processing. Real- time video can be viewed in the cloud. Figure 5 demonstrates how Kinesis Video streams work. Figure 5 demonstrates the interaction between the following components:



**Figure 5: 128-D feature vector**

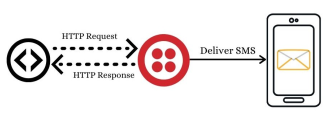
1. Producer - Producer is the Kinesis video stream source. Devices such as CCTV cameras, phone cameras, web- cam, raspberry pi camera or any video-generating device can act as a producer. A producer can also send data in other formats such as images, audio, RADAR data.
2. Kinesis video stream - Kinesis video streams are resources which help with transfer storage and consumption of generated data. Usually there is one producer publishing data into the stream as shown in figure 6.
3. Consumer - The Consumer uses data from the Video Stream in the form of fragments or frames to analyze it. Kinesis video streams applications are called consumers. Consumer applications can be created to run on Amazon EC2 instances.

To stream the live video, GStreamer and Kinesis producer library are installed into the IoT camera. The stream is made active by running the GStreamer once data is streamed, the consumer application can be built in AWS EC2 engine. In the EC2 engine, each frame is assessed one at a time. All faces in a frame are found using the face recognition library. All the faces in the frame are converted to encodings.



**Figure 6: Amazon kinesis architecture**

These encodings are compared with the encodings that have been obtained in the training phase. If a match is found then the person is labeled as the person with which the encoding is matched else the person is labeled unknown. When an unknown person is identified as a message notification along with the image and timestamp is immediately sent to a registered mobile number through Twilio notification API as shown in figure 7. The Twilio notification API provides 99.95% uptime SLA achieved with automated failover and zero maintenance windows. This platform can be used for SMS, video, chat, MMS and two-factor authentication. For using Twilio we purchase a mobile number from which the notification will be sent and register another mobile number which will receive the notification. To send messages we install the Twilio library for python and use the pre-defined functions.



**Figure 7:** Twilio notifier

The images of all persons appearing in a frame will be uploaded to AWS Simple Storage Service or S3, every 15 to 20 seconds which assures scalability, data availability, security, and performance. It provides easy to use management features to store data efficiently. To upload to S3 we use the boto library which is an AWS SDK for Python. Boto provides built-in functions to upload to S3. In S3 we create a bucket with the desired name and region to store the data. Every time the upload function is called, the image is uploaded to S3.

1. RESULT

In “Comparison of Haar-like, HOG and LBP approach for face detection in video sequences” [12], a comparison has been done on using Haar with two different classifiers; HOG and LBP. The HOG classifier had an accuracy of 92.68% whereas LBP classifier achieved the worst rate which was 32% inferior to the HOG classifier. Therefore, the best accuracy available with Haar is 92.38%. In the proposed system, we have used dlib face recognition tool which gives higher accuracy. The comparison is given in Table 1. We successfully created the encodings from the dataset, installed Gstreamer and producer libraries for the IoT camera. The video from the camera was streamed to Kinesis. The face recognition application in the EC2 engine consumed the video stream from Kinesis. Images were uploaded to S3 bucket along with name and timestamp. Message of the intruder was received in the registered mobile number.

|  |  |
| --- | --- |
| **Method** | **Accuracy** |
| Haar | 92.68% |
| Dlib | 99.38% |

1. FUTURE WORK

The system would include recognizing the weapons held by the person who was being viewed by the camera. The system would also include identifying the person in mask by recognizing the eyebrow feature.

1. ACKNOWLEDGEMENT

The completion of this project could not have been possible without the lab facilities, support and encouragement provided by our institute, Adi Shankara institute of engineering and technology.

1. REFERENCES

[1] Hung Vu, ”Deep Abnormality Detection in Video Data” IJCAI-17

[2] Y. Freund and D. Haussler, ”Unsupervised learning of distributions on binary vectors using two layer networks”. Technical report, CA, USA, 1994.

[3] R. Salakhutdinov and G. Hinton,” Deep Boltzmann machines”. In AISTATS, volume 5, pages 448-455, 2009.

[4] Prakhar Singh and Vinod Pankajakshan , ”A Deep Learning Based Technique for Anomaly Detection in Surveillance Videos”, in 2018 Twenty Fourth National Conference on Communications (NCC)

[5] Ali Khaleghi, Mohammad Shahram Moin ”Improved Anomaly Detec- tion in Surveillance Videos Based on A Deep Learning Method ” in Artificial Intelligence and Robotics 2018.

[6] Yun-Xia Liu, Yang Yang, Aijun Shi, Peng Jigang and Liu Haowei, ”Intelligent monitoring of indoor surveillance video based on deep learning” on International Conference on Advanced Communications Technology(ICACT)

[7] P. Viola and M. J. Jones,“Robust real-time face detection”, International Journal of Computer Vision, 57 (2004), pp.

[8] P. Viola, M. Jones, ”Rapid object detection using a boosted cascade of simple features,” in: Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognition (CVPR), vol. 1 (2001), pp. 511-518.

[9] R. Lienhart and J. Maydt, ”An extended set of Haar-like features for rapid object detection” In: Proc. of the International Conference on Image Processing (ICIP), pages 900-903, 2002.

[10] C.Papageorgiou,M.Oren,and T.Poggio ”A general frame-work for object- detection”. International Conference on Computer Vision, 1998.

[11] Kaiming He, Xiangyu Zhang,Shaoqing Ren, Jian Sun, ”Deep Residual Learning for Image Recognition”,The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778

[12] Amal Adouani, Wiem Mimoun Ben Henia, Zied Lachiri, ”Comparison of Haar-like, HOG and LBP approaches for face detection in video se- quences”,2019 16th International Multi-Conference on Systems, Signals & Devices (SSD’19)