International Journal of Advanced Science and Technology
Vol. 29, No. 5, (2020), pp. 12464-12478

Determination of Efficient Way of Implementing AI-based Devices for Reducing Social Crimes

Anurodh Acharya[1], Sangat Baral[2], Rajeshkannan Regunathan[3],Ramanathan L[4]
School of Computer Science and Engineering
Vellore Institute of Technology, Vellore

anurodh.acharya2018@vitstudent.ac.in[1],sangat.baral2017@vitstudent.ac.in[2],rajeshkannan.r@vit.ac.in[3], lramanathan@vit.ac.in

Abstract.
As being part of society, people have different challenges to overcome which could be as large as saving our self from vulnerable activities. Many surveys suggest that the number of crimes across the globe is increasing exponentially. The ratio between several crimes and culprit prisoned is very large which itself says about the lack of technological advancements in finding the offender. After going through series of different topics, surveys, journals, and research papers, the idea was proposed with a practical solution that can potentially reduce the risk of damage caused during the time of the incident and help us find the culprit without letting him/her escape.

Keywords: *Neural Networks, Database, OpenCV, Keras, Image Processing.*

1. Introduction

No one in the world loves anything more than himself. The bodily attachment is what makes the people more alert and conscious of dangerous situations. Even though carrying Illegal weapons such as guns is illegal across the world, it is not the case of every country. Some countries allow carrying guns as a defensive measure but are they worth it? Any random person can shoot anyone and not get caught without doing much of an effort. Let us not go too far with the guns, household items such a knife or even an acid can victimize the person that can ruin his/her life. There have been many cases in which people had been acid attacked by the biker who ran away within a fraction of seconds and not ever get caught. It will simply not be possible for the victim to chase the attacker or any other person as he/she will have gone too far ahead.

Another important aspect that is to be looked into is the rape cases which never look to being stopped. Women are scared to go alone without anyone accompanying her due to the fear of being one of the victims. Due to not getting caught, the rape cases never look to go down and there still has not been any great steps taken to reduce the same. Children are vulnerable to getting kidnapped without any information to the parents for the sake of money. Children cannot themselves call someone for help as they might not be even aware of the situations which are going around them.

All these problems have led to finding a solution that can be implemented for any of the above cases and many more which can reduce the crimes being committed across our society. This is where our technology puts its hand forward to help us. This project implements an AI-based technique that that sense, capture, detect, analyze and process the required information that can help us solve these real-time issues.

The process can be divided into the following sub-modules:

ISSN: 2005-4238 IJAST

- Voice Recognition
- Sentiment Analysis
- Face Detection
- Storing and training of pictures
- Testing the pictures with the test set
- Comparison of results

2. Related Work

The problem of weapon detection in real-time using deep learning is related to two main research areas. The first one addresses the weapon detection using classical methods whereas others focus on improving the evaluation and performance of object detection and classification using Convolution Neural Networks.

Verma, Gyanendra & Dhillon, Anamika. (2017) [1] has described automatic gun detection from a cluttered scene using Convolutional Neural Networks (CNN). The author has used Deep Convolutional Neural Network, a state-of-the-art Faster Region-based CNN model. They have evaluated gun detection over the Internet Movie Firearms Database (IMFDB) in the paper 'A Handheld Gun Detection using Faster R-CNN Deep Learning'.

Sandjai Bhulai and Shanita Biere (2018)[2] has worked on hate speech detection in the paper named "Hate Speech Detection Using Natural Language Processing Techniques". The goal of this paper is to look at how NLP applies to detect hate or some threatening speech. The author has used the classifier which assigns each tweet to one of the categories of a Twitter dataset: hate, offensive language, and neither.

Youssef Elmir et al [3] have presented a system of automatic detection of handguns in videos, suitable for both surveillance and control. This paper consists of a study of different online handgun detection methods using supervised deep learning in the paper named "Deep Learning for Automatic Detection of Handguns in Video Sequences".

Das et al (2015) [4] explained about the system which consists of two components, one for processing acoustic signal which is captured by microphone and second is to interpret the processed signal in the paper name "Voice Recognition System: Speech-to-text".

N.Niranjan et al. (2017)[5] have developed a system for face detection and recognition in real-time. The system was developed using C sharp.Net programming, Viola-Jones algorithm which used Haar Cascade Classifier, PCA (classified as either Feature-based and image-based) and EmguCV, a Computer vision Library and wrapper class of Open CV.

3. Proposed Methodology

Assumption

<u>Layer l is the convolutional layer</u>

- Filter size= $f^{[l]}$

ISSN: 2005-4238 IJAST

It is the learnable weights that can be learned via backpropagation.

- Padding= $p^{[l]}$

It is the number of pixels that has been added into the image at the time of it being processed by a kernel

- Stride=$s^{[l]}$

It is the no. of pixels that has been shifted in the input matrix

Formulas Used

Input: $n_h^{[l-1]} \times n_w^{[l-1]} \times n_c^{[l-1]}$ (1)

Each Filter: $f^{[l]} \times f^{[l]} \times n_c^{[l-1]}$ (2)

Weights: $f^{[l]} \times f^{[l]} \times n_c^{[l-1]} \times n_c^{[l]}$ (3)

Output: $n_h^{[l]} \times n_w^{[l]} \times n_c^{[l]}$ (4)

Bias: $1 \times 1 \times 1 \times n_c^{[l]}$ (5)

Inputs to CNN are image pixel values. CNN extracts the features in the input image and learns some weight over the epochs.

W**eight** represents the strength of the connection between units. A weight brings down the importance of the input value

The **output** layer in an artificial neural network is the last layer of neurons that produces given outputs for the program.

Bias is an additional parameter in the Neural Network which is used to adjust the output along with the weighted sum of the inputs to the neuron.

3.1 Google Voice Recognition for speech to text conversion

Recognizing and processing voice is an important part of knowing that is happening at the crime scene. For every person having a device with him/her, he/she can provide the voice commands to the device that enables its working mechanism. But different cases are to be taken care of.

What if the attacker firstly closes your mouth and does not allow you to speak? The voice commands that have to be provided by the person will not be processed and the camera will never get activated and thus the further processes will never happen. Considering this situation, the concept of Natural Language Processing is implemented for analyzing the text detected from the offender. The system tries to find out if an offender is intended in hurting the person by finding the compound value using the Natural Language Processing libraries that will help to find if he means to hurt or not.

Google Voice Recognition will provide a better platform to convert the voice into text without much complexity in the program that can be easily decoded.

3.2 Sentiment Analysis using Natural Language Processing

ISSN: 2005-4238 IJAST

Analyzing the speaker's intention will be the main focus of our project while working with Natural Language Processing. The concept of deep learning will help in achieving the requirement as a system moves onto the sequence models. One way to proceed forward would be to use a Simple Neural Network consisting of few hidden layers and an output layer. The problem lies in the fact that these models will not be able to process or memorize the intention of the text. Say for example 'Cats are beautiful. They meow when they are hungry'. For a human, it can be easily recognized that the word 'They' in the sentence refers to the Cat but that is not the case in case of computers. The system will have to make them learn these things which will simply not be possible in the case of Simple Neural Networks.

Then as an alternative came LSTM(Long Short Term Memory) which resolves the above issue and is currently one of the widely used models for the sequence processing along with GANs. Since the system is trying to detect the intention of the other person, the system should be fed with a dataset to train the model. Our dataset consisted of thousands of samples with positive and negative labels for every sentence to be fed into the Neural Network. The model is trained through these data. After successfully fitting the model, after 5 epochs the accuracy of about 87% is obtained which is acceptable. If the number of neurons is increased in the hidden layer, there always exists a chance of overfitting of data where the model may successfully predict the models with the data and not be able to generalize with the test data. To overcome this problem validation set is used to choose the model that performs well in most of the testing data.

The system cannot be tested with our data which in this case is the words spoken by the person. Our system does not recognize the texts so firstly text is converted into numbers which can be achieved with the help of Tokenizer to convert the texts into equivalent numbers. Since the RNN models require a set of fixed length inputs, the length of the text is limited and incase the length of the sentence is below the threshold value, the sentences are sliced to match the length. After performing the required text preprocessing using the Keras library the system now proceed further in predicting our model. Once the data is fed for prediction, the value obtained will provide the sentiment of the sentence according to which the intention is a person can be decided.

```
Detected code word is help me someone
Recorded audio during incident is I will Kill you!
I will Kill you!
{'neg': 0.714, 'neu': 0.286, 'pos': 0.0, 'compound': -0.7177}
```

Fig 3.1 Detection of Threat

The person carrying the device can provide the predefined code words as in this case it is 'help me, someone'. If the device detects any of the work in 'help me someone' or on the other hand if it detects any offensive words from the attacker as in Fig 3.1, the camera automatically opens and starts capturing the images. Our model detected the words 'I will kill you' on the negative side with a probability of 0.7 in the offender's sentence.

Fig 3.2 Face Detection

3.3 Face Detection

Detecting the faces of the offender is one of the most important parts of the device. Finding the faces in the image with high accuracy is not easy as there may exist a chance of it failing any time which could make a huge impact on the outcome**.**

Computer Vision is one of the fields of Artificial Intelligence that helps to capture, analyze and store images as per the requirements. But, why not other techniques? Well, comparing the performance of other techniques such as Neural Networks for detecting the faces, OpenCV library which is primarily used in Computer Vision came on top in terms of its performance and provides a better result as compared to other techniques.

It starts by training the model with thousands of images consisting of both positive and negative sets. Images in the positive sets consist of images of different people's faces which are used to train our model on. On the other side, negative sets will have a similar number of pictures to be trained but consist of pictures that are not having any face of the person. On having the model trained, the obtained result is used to find the face in the real-time video capturing every frame and locating the face of the person. More the number of pictures used in training more will be the precision and chances of detecting faces in the images increase. All these detected images will be stored in the temporary storage which will later come handy while analyzing the picture.

3.4 Understanding Various Convolutional Neural Network Models

Neural Networks have always been highly regarded in terms of their performance and the type of results they can produce. The main question here remains how many layers of Neural Networks should be used to implement training and testing our images?

There are different ways of choosing the number of layers that are required in our Neural Network Architecture. If less than 10 layers of ConvNet are used then manually adding layers would still have been possible. However, the most efficient way is to modify the existing in-built models which have been pre-trained for various other tasks. Some of the most popular ones include the VGG16 model, MobileNet model, and ResNets. ResNets also is known as Residual Network provides skip connections that allow us to take the activation from one of the layers to even much deeper into the neural network. Thus this will eventually help us build ResNets which enables us to train even deeper networks. These are made of what is called a Residual Block. So, rather than following the usual traditional path, the information can follow a shortcut to reach much deeper into the Neural Network. Residual blocks of networks are taken and stacked them together to form a deep network. In theory, having a deeper neural network should help us in reducing the training error. But in reality, it turns out that if the very deep neural network is being used, the optimization algorithm will have a much harder time to train the model and thus the error goes even words if the neural network tends to become very deep. MobileNet with comparatively smaller in size and VGG16 model with a higher rate of accuracy would be a better choice considering the circumstances that we are working on. Thus we have carried forward with both the Neural Network Models statistically comparing both the models and choosing the best one.

ISSN: 2005-4238 IJAST

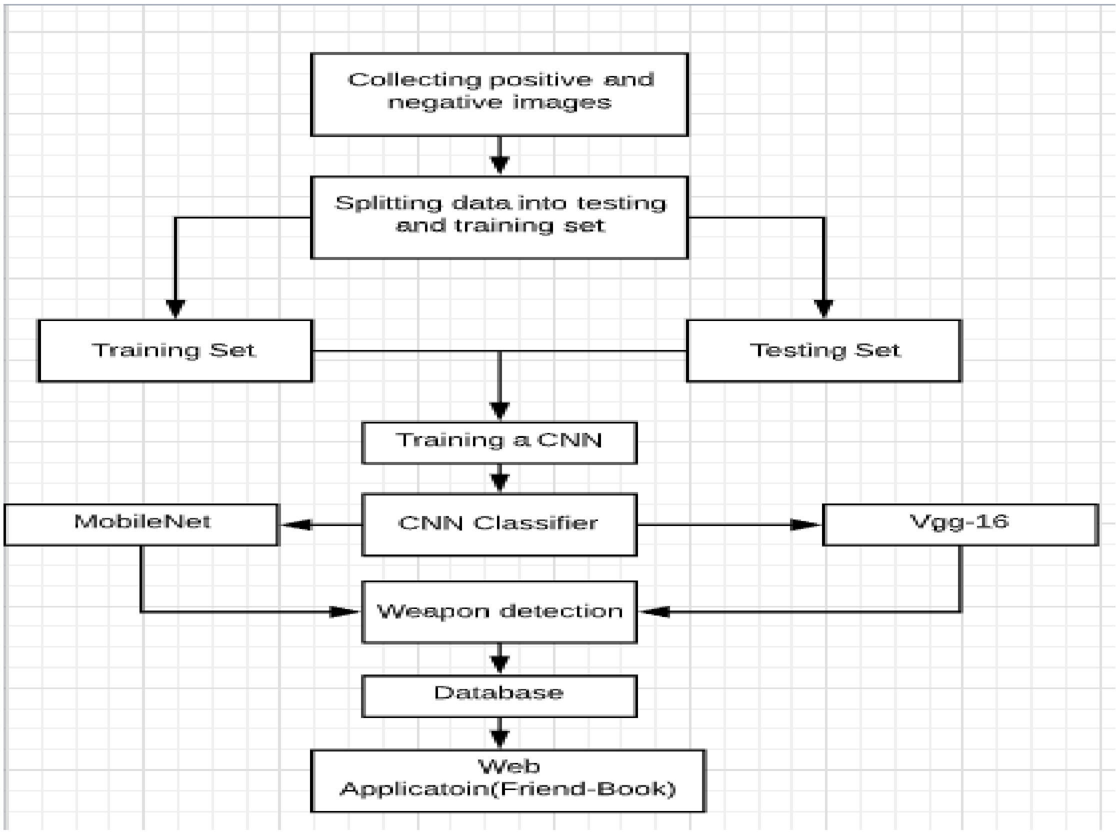


Fig 3.3 Block Diagram of Work Flow

Mechanism of Detection

The images of weapons are collected as a sample into the folder which is further split into subfolders namely training and testing data. The CNN Classifier is trained with the help of training data images. In this case, we perform the same operation for VGGNet as well as MobileNet. The real-time image which is extracted via camera is then tested with the trained classifier for both the Neural Network classifiers i.e. VGGNet and MobileNet. If the weapon is detected with an accuracy of a provided threshold value, the information is transferred into the database which is further connected to the social networking sites that provide the information to the connected users.

All the information is now visible to the concerned people via database information transfer which thus allows an effective and faster transfer of information as compared to any other method.

ISSN: 2005-4238 IJAST

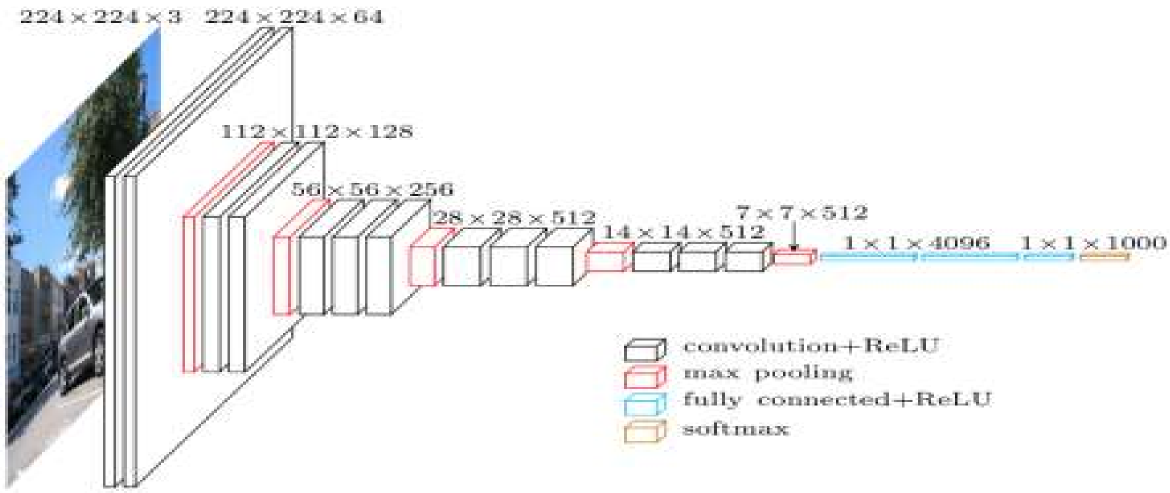


Fig 3.4 VGG-16 Net

Source: https://neurohive.io/en/popular-networks/vgg16/

3.5 Forward Propagation for classifying pictures

The pictures will be divided as the training and testing after the capturing of images takes place. Every image will forward propagate along with the provided labels for the images. After every epoch, a result will be generated by the ConvNet comparing the provided label as well as the obtained output. That result will be an estimation of how well our Neural Network has been in predicting the objects in the images. In a single epoch, the chances of obtaining maximum accuracy are minimal as there is not much thing that this Network has offered. Now to reduce the boundary between the images of the weapon and the obtained output the Network had predicted, Back-Propagation Method is used to minimize the loss.

3.6 Back-Propagation for minimizing a loss function

After every epoch cycle, backpropagation is performed to reduce the gap between the actual label and the predicted output. This can be approached by changing the weights of the Neural Net in each layer. After many continuous cycles of prediction of weapon, the weight of the network is varied to the point where minimum loss is obtained. Thus this will mean that our model is sufficient enough to predict a weapon given a set of images.

3.7 Training Pictures

Keras deep learning library provides us with an easier way to implement Artificial Neural Networks under fewer lines of codes with TensorFlow as our backend. The reason to choose Keras rather than other libraries lays in that particular fact. Both of our images are placed onto the drive before starting any operation. The folders were structured in the form of the train set, test set, and valid set what was to be fetched for training, testing as well as validating.

ISSN: 2005-4238 IJAST

The images are fetched sequentially from the drive to the Neural Network for training every picture with some provided weights. Due to a larger number of levels as well as the parameters, the model took some amount of time for varying the weights and training the ANN. Epochs are set to 100 so that the accuracy obtained would be high. Images along with the labels (cat: 0, dog: 1) were provided to the network which forward propagated to produce the result. The obtained result was compared to the labels which were provided before and after a series of backpropagation, it improved its accuracy in predicting the cat or a dog. The weights of the networks constantly changed to match the label to the produced output by the Neural Network and thus after about 100 epochs, the system went of obtaining the accuracy of 0.91 which is about 91 percent. This was the initial process that takes place before the beginning of any operations.

Now as pictures are taken from the camera, the pictures of the offender are stored into the folder where there pre-exist negative examples and positive examples being the pictures recently taken. Now the images are tested which consists of the pictures of the offender and tries to detect if there exists any potential weapon threat. Keras Deep Learning library begins testing the new set of test images with Adam optimizer and loss being calculated as tropical-cross entropy. Since the numbers of images are very big, steps per epoch could be set as close to 5-10 which will not affect the result much and reduce the time it takes to compute the result.

The obtained result was staggering as the system correctly predicted the presence of any weapon in the picture with about 97 percent of accuracy. This thus confirmed the situation not being ideal and further steps had to be quickly taken into place. Now the system cannot assure that the offender will have his face uncovered. Well, then there lies a possible use of OpenCV Machine Learning library DLIB.

Fig 3.5 Training Images

3.8 Testing Pictures

After the successful training of pictures, the system now has to test our pictures with the trained set. Keras library will perform the task as data is fed into the Neural Network. On specifying the loss as well as other parameters our model is successfully compiled and trained. The result is visualized with the help of the confusion matrix as well as the accuracy score to compare the performance of Neural Net. If the system obtains a satisfying score of above 50%, the presence of the weapon can be assured.

ISSN: 2005-4238 IJAST

Fig 3.6 Testing Images

Number of Epochs	50
Batch Size	10
Steps per epoch	4
Total Number of training data	1000
Total Number of testing data	350
Learning rate	0.001
Optimizer	Adam
Loss	Categorical Cross Entropy

Table 1: Experimental parameters

3.9 Transferring the information into the database

The project used by MySQL to send the data from our running program into the database. The database names, entities, and tuples will be pre-defined before the information is sent to the server. The system will obtain all the database information such as dB name, dB password to execute the connection into the database. Once the command is run to connect into the database server, the elements can be inserted into the database with the help of SQL queries. Once the information is updated into the database system will continue with the other part of the database i.e. execution fetch.

PHP is one of the most popular languages used for backend development. In our case, it will provide us a platform to retrieve the stored data from the database. Virtually many people will be connected to an online platform known as Friend-Book. It will allow the friends in contact to message each other, share posts with other features influenced by Facebook. The account creation will take place at the beginning of the entire process. Once the account has been created, that information will be stored in the database from which the system will now look to retrieve information. Once the weapon is detected in the camera frame, the image of a gun is extracted.

4. Experimental Conclusions

4.1 Result

The model is trained with the pictures as shown in **Fig 3.5**. It consists of different weapon types and orientation so that the model can be trained effectively to predict and detect all different types of weapons. During the process of training, the same images have been oriented in several different ways with different angles to not fail to predict the correct images.

ISSN: 2005-4238 IJAST

A sufficient number of image of weapon is taken and tested against the trained model whereas the complete frame is stored separately. The weapon detected is tested against the model to find accuracy. If the testing result meets the desired accuracy then the picture of the weapon and the complete frame is stored in the database.

The concept of ROI (Region of Interest) is used to save the image of a gun. Similarly, the whole camera frame is also being captured. The reason for extracting the weapon from the frame is that it will be easier to test the object against the trained model. As well as, the accuracy will be high and precise.

Fig 4.1 Gun detection

In **fig4.1**, it can be seen that the gun has been recognized by the device which is shown by a blue square box. To make the faces of the offender highlight, a yellow squared box has been put around that distinct the faces and guns to any other object or body part.

When a particular person's family members along with contact numbers are registered into the database, the system will query into the database. Once the system receives the information that his/her close person is in trouble and which is verified by detecting weapons in the picture. The messages will be queried and will be visible into the person's account as sent by an AI-based model and thus alerts the near and close ones that they require help which will help to take immediate actions and not allow the culprit escape away.

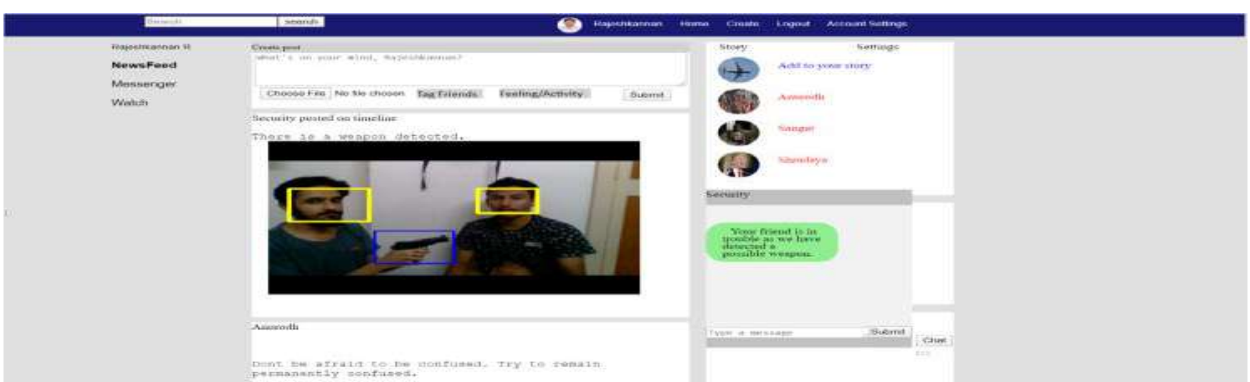


Fig4.2 User Interface for sending the emergency message to concerned people

ISSN: 2005-4238 IJAST

The use of the database can be visualized in **fig 4.2** as the analyzed frame that is received from the device is transferred into the database once the weapon detection has been detected as per the level of accuracy obtained by the neural network. MySQL has been used to query the commands that further are handled by PHP at the backend. The message, as well as the picture, can be uploaded by the concerned security and specifically the alert information can be personally sent to the parents/guardians.

4.2 Comparison between VGGNet and MobileNet

VGG16 model is preferred since after the trial implementation with all the models, the MobileNet and ResNets could not match the accuracy provided by the VGG16 and thus VGG16 model was a better choice. However, there lies a tradeoff of the complexities of each of the models. Looking at the weights of both of the models, VGG16 has the size over 500MB whereas MobileNet does not exceed 20MB this makes MobileNet more preferable for low-end devices where the situation is more time-critical than speed-critical. Let us go through both the models in detail and find the best one for us in this case.

VGG16 model with its massive size has a total number of 138,357,544 parameters which in itself after few epochs tend to provide a better result. But the system cannot directly use this model as a system needs to fine-tune this as per the requirement. Since the system wants to classify whether it is a weapon or not, the system shall have only two final Dense Layer but VGG16 has 1000 dense layers that can be redefined using the fine-tuning technique. On the other side, MobileNet initially consisted of 4,253,864 parameters with 4,231,976 trainable parameters which are significantly lesser as compared to the VGG model. The result was not surprising as the VGG16 model provided us with a higher accuracy rate but not with a bigger margin. Since our task was more of accuracy oriented, VGG16 has been chosen over MobileNet.

4.2.1 Confusion Matrix

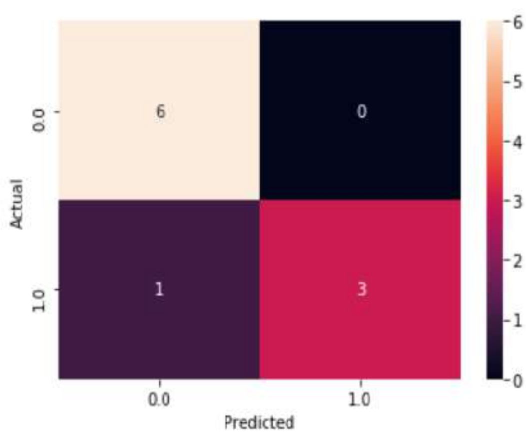


Fig 4.3 Confusion matrix of VGGNet

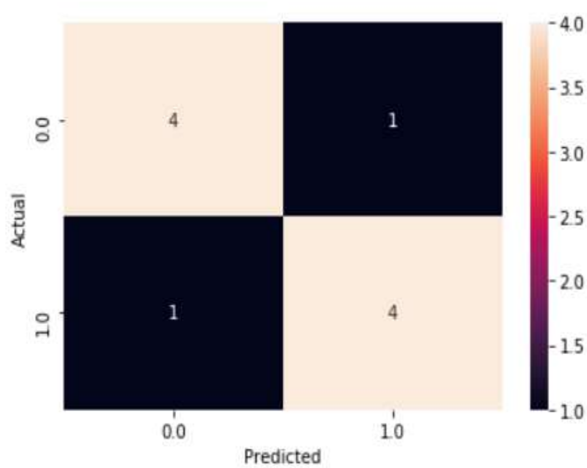


Fig 4.4 Confusion matrix of MobileNet

The confusion matrix of VGGNet for the gun prediction on the frames shows that the accuracy of 90% can be obtained in most of the cases. It brings us to the conclusion that it can accurately predict the presence of a gun in 9 out of 10 possible causes.

The confusion matrix of Mobile Net shows that the system has accurately predicted the detection of gun 8 out of 10 times with an accuracy of 80% and 2 of the predictions coming on the wrong side.

4.2.2 Precision Score

ISSN: 2005-4238 IJAST

	precision	recall	f1-score	support
0.0	1.00	0.86	0.92	7
1.0	0.75	1.00	0.86	3
accuracy			0.90	10
macro avg	0.88	0.93	0.89	10
weighted avg	0.93	0.90	0.90	10

Fig 4.5 Precision Score of VGGNet

Fig 4.5 shows the precision score for the VGG16 Model. It can be seen that our classifier has performed considerably well in separating the positive sample which is not negative and vice versa. Accuracy stands on the higher side with a 0.9 probability score.

	precision	recall	f1-score	support
0.0	0.80	0.80	0.80	5
1.0	0.80	0.80	0.80	5
accuracy			0.80	10
macro avg	0.80	0.80	0.80	10
weighted avg	0.80	0.80	0.80	10

Fig 4.6 Precision Score for MobileNet

Fig 4.6 shows the precision score for MobileNet Model. It can be seen that the model is 80% certain the separating the positive samples that are not negative and vice versa. Accuracy also remains to be 80%. The precision is the ratio $\frac{Tp}{Tp+Fp}$ where Tp is the number of true positives and Fp is the number of false positives. This is the ability of the model not to classify as positive a sample which is negative.

4.2.3 Performance Measure

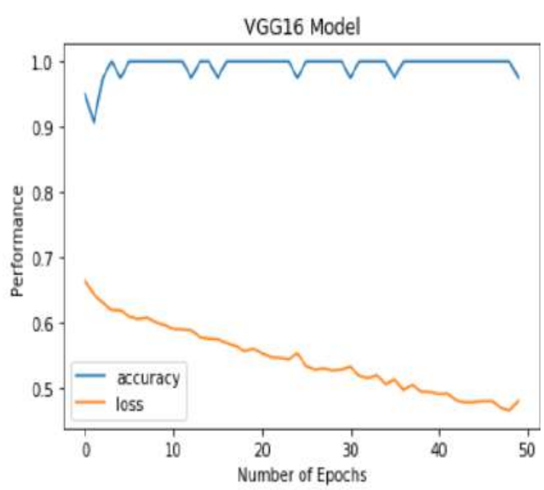


Fig 4.7 Performance Graph for VGGNet

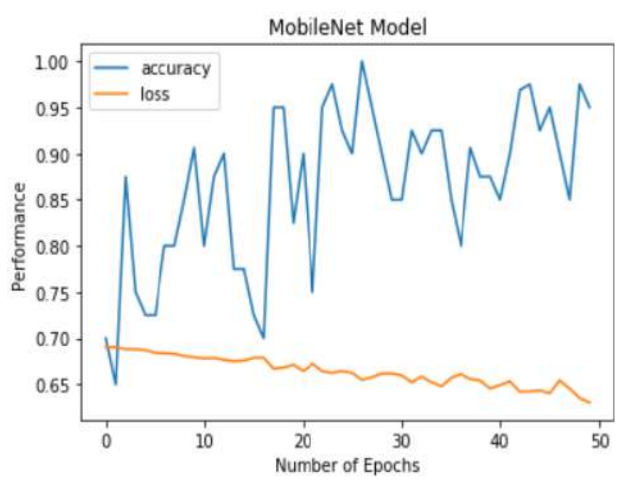


Fig 4.8 Performance Graph for MobileNet

ISSN: 2005-4238 IJAST

Fig 4.7 shows the performance vs. epochs graph for the VGG16 model. It can be seen that the difference between the correct and wrong prediction tend to decrease as our model differentiates between whether or not a weapon is present and thus loss decreases and the accuracy of the model increases.

Fig 4.8 shows the performance vs. epochs graph for MobileNet Model. Even though the loss of our model decreases with a similar rate as the vgg16 model, the accuracy revolves around 0.7-0.9 which does not reliably differentiate between the objects captured as a frame.

5. Conclusion and Future Work

This paper demonstrates how to make an effective model for personal safety equipment to reduce the number of crimes around the globe. The offender will be caught with immediate effect and also the person using this will be more comfortable outdoors. The paper has analyzed various aspects of Artificial Intelligence and Machine Learning algorithms and the better one is chosen as per our analysis within the dataset which mostly is the pictures. Since the system works with an RGB color image, the complexity of the program tends to rise comparatively high as working with real-time data consists of more resources.

6. References

1. Verma, Gyanendra & Dhillon, Anamika. (2017). A Handheld Gun Detection using Faster R-CNN Deep Learning.

2. Das, Prerana & Acharjee, Kakali & Das, Pranab & Prasad, Vijay. (2015). VOICE RECOGNITION SYSTEM: SPEECH-TO-TEXT. Journal of Applied and Fundamental Sciences. 1. 2395-5562.

3. Niranjani, & B.Tharmila, & Sukirtha, C. & Kamalraj, K. & Thanujan, S. & P.Janarthanan, & Thiruchchelvan, Nagarathnam & Kanagasabai, Thiruthanigesan. (2017). The Real-Time Face Detection and Recognition System. International Journal of Advance Research in Computer Science and Management. 51.

4. Nadhir Ben Halima and Osama Hosam. 2016. Bag of Words Based Surveillance System Using Support Vector Machines. 10 (04 2016), 331–346.

5. Yu-Ming Liang, Sheng-Wen Shih, and Arthur Chun-Chieh Shih. 2013. Human action segmentation and classification based on the Isomap algorithm. Multimedia Tools and Applications 62, 3 (01 Feb 2013), 561–580. https://doi.org/10.1007/ s11042-011-0858-2

6. Rohit Kumar Tiwari and Gyanendra K. Verma. 2015. A Computer Vision-based Framework for Visual Gun Detection Using Harris Interest Point Detector. Procedia Computer Science 54, Supplement C (2015), 703 – 712. https: //doi.org/10.1016/j.procs.2015.06.083

7. R. K. Tiwari and G. K. Verma. 2015. A computer vision-based framework for visual gun detection using SURF. In 2015 International Conference on Electrical, Electronics, Signals, Communication, and Optimization (EESCO). 1–5. https://doi. org/10.1109/EESCO.2015.7253863

8. Tammina, Srikanth. (2019). Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images. International Journal of Scientific and Research Publications (IJSRP). 9. p9420. 10.29322/IJSRP.9.10.2019.p9420.

9. Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going Deeper with Convolutions." 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

ISSN: 2005-4238 IJAST

10. Muhammad, Usman & Wang, Weiqiang & Pervez, Shahbaz & Ali, Sajid. (2018). Pre-trained VGGNet Architecture for Remote-Sensing Image Scene Classification. 1622-1627. 10.1109/ICPR.2018.8545591.

11. Shi, Cheng, and Chi-Man Pun. ”Superpixel-based 3D deep neural networks for hyperspectral image classification.” Pattern Recognition 74 (2018): 600-616

12. Simonyan, Karen, and Andrew Zisserman. ”Very deep convolutional networks for large-scale image recognition.”arXiv preprint arXiv:1409.1556 (2014)

13. Jain, Aditya & Kulkarni, Gandhar & Shah, Vraj. (2018). Natural Language Processing. International Journal of Computer Sciences and Engineering. 6. 161-167. 10.26438/ijcse/v6i1.161167.

14. Brian Milch, Alexander Franz “Searching the Web by Voice” Taipei, Taiwan — August 24 - September 01, 2002.

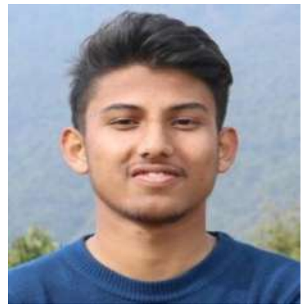
ISSN: 2005-4238 IJAST

International Journal of Advanced Science and Technology
Vol. 29, No. 5, (2020), pp. 12464-12478

Authors

Anurodh Acharya

Anurodh Acharya is pursuing his B.E. Computer Science & Engineering Degree at VIT University, Vellore, India. He is currently the second year student. His area of interest is Artificial Intelligence and Cyber security. He has been working on project which is based on predictions of cyber-attack and provides solution to cyber threat and security problem using AI. He have already completed his project on hybrid Job scheduling algorithm based on round robin, priority and SJF and "Friendbook" as a social networking medium..



Sangat Baral

Sangat Baral is currently pursuing his B.Tech Degree in Computer Science and Engineering at VIT University, Vellore, India. He is currently the third year student. His area of interest is Computer Networking and Machine Learning. His project includes Voice Navigated Hospital Appointment System, Robo-Stick for Visually Impaired Person using Deep Learning and Nepali-English Translation using natural language processing.

Rajeshkannan Regunathan

Rajeshkannan Regunathan has received M.Tech in Computer Science & Engineering from SASTRA University, Tanjore, and a Ph.D. degree in Computer Science and Engineering from Vellore Institute of Technology, Vellore, India. He is currently working as an Associate Professor in VIT, Vellore, India. His area of interest is Cloud Computing, Artificial Intelligence, Natural Language Processing, and Data Science. He is having 13+ years of teaching experience. He has published more papers in International Journals and Conferences. He is a member of CSI, IEEE (WIE).

Ramanathan L

Ramanathan L has received his B.E. in Computer Science & Engineering from Bharathidasan University, Tiruchirappalli, India, and M.E in Computer Science from Sathyabama University, Chennai, India and a Ph.D. degree in Computer Science and Engineering from VIT University, Vellore, India. He is currently working as an Associate Professor in VIT, Vellore, India. His area of interest is Data Mining, Database Systems, Software Engg, Cloud Computing, and Virtualization. He is having 14+ years of teaching experiences. He has published more papers in International Journals and Conferences. He is an Editorial board member/reviewer of International/ National Journals and Conferences. His ongoing research is on Prediction, Classification, and Clustering for Educational systems. He is a member of IACSIT, CSI, ACM, IACSIT, IEEE (WIE), ACEEE

ISSN: 2005-4238 IJAST

ISSN: 2005-4238 IJAST