Computers, Materials & Continua
DOI: 10.32604/cmc.2022.027571
Article

Tech Science Press

Automatic Detection of Weapons in Surveillance Cameras Using Efficient-Net

Erssa Arif[1,*], Syed Khuram Shahzad[2], Muhammad Waseem Iqbal[3], Muhammad Arfan Jaffar[4], Abdullah S. Alshahrani[5] and Ahmed Alghamdi[6]

[1]Department of Computer Science, Superior University, Lahore, 54000, Pakistan
[2]Department of Informatics & Systems, University of Management & Technology, Lahore, 54000, Pakistan
[3]Department of Software Engineering, Superior University, Lahore, 54000, Pakistan
[4]Faculty of Computer Science & Information Technology, Superior University, Lahore, 54000, Pakistan
[5]Department of Computer Science & Artificial Intelligence, College of Computer Science & Engineering, University of Jeddah, Jeddah, 21493, Saudi Arabia
[6]Department of Software Engineering, College of Computer Science and Engineering, University of Jeddah, Jeddah, 21493, Saudi Arabia
*Corresponding Author: Erssa Arif. Email: erssaarif1@gmail.com
Received: 20 January 2022; Accepted: 08 March 2022

Abstract: The conventional Close circuit television (CCTV) cameras-based surveillance and control systems require human resource supervision. Almost all the criminal activities take place using weapons mostly a handheld gun, revolver, pistol, swords etc. Therefore, automatic weapons detection is a vital requirement now a day. The current research is concerned about the real-time detection of weapons for the surveillance cameras with an implementation of weapon detection using Efficient–Net. Real time datasets, from local surveillance department's test sessions are used for model training and testing. Datasets consist of local environment images and videos from different type and resolution cameras that minimize the idealism. This research also contributes in the making of Efficient-Net that is experimented and results in a positive dimension. The results are also been represented in graphs and in calculations for the representation of results during training and results after training are also shown to represent our research contribution. Efficient-Net algorithm gives better results than existing algorithms. By using Efficient-Net algorithms the accuracy achieved 98.12% when epochs increase as compared to other algorithms.

Keywords: Detection algorithms; machine learning; machine vision; video surveillance

1 Introduction

Today's, a huge number of criminal activities are taking place using handheld arms e.g., guns, pistols, revolver and semi machine guns of shot guns also in some cases [1]. These activities can be reduced by monitoring and identifying at early stage. The way to minimize the violence is by the early detection of suspicious activity so that the law enforcements can take necessary action.

Current control and surveillance still need human supervision and interference. Close circuit television (CCTV) surveillance cameras are broadly in use for monitoring and the other security purposes [2]. Currently, deep learning approaches are increasingly adopted because of the capability of giving data-driven solutions to such problems [3]. Extraordinary results of image classification using deep neural networks have exceeded the human performance [4]. Fig. 1 illustrates an abstraction of common process flow of such detection applications.

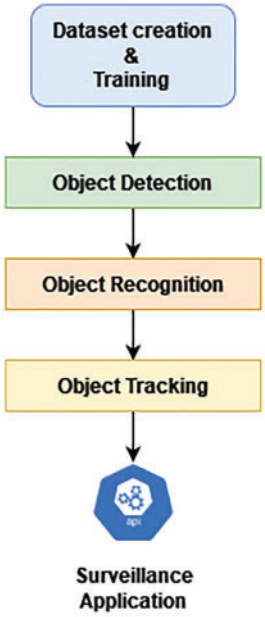

Figure 1: Detection and tracking application process flow

1.1 Data Detection and Extraction

The first step is data extraction and detection model training using deep learning-based algorithms. There are varieties of different algorithms used for object detection. Histogram of oriented gradients (HOG) is used as a feature descriptor for the detection of the objects. It is comprised of gradient occurrences and their orientations in a localized part of image like Region of interest (ROI) and detection window etc. The basic advantage of HOG is that it is easy to implement and understand [5].

Single shot detector (SSD) is a single deep network to detect objects from an image. This algorithm is easy to integrate into multiple systems that require the detection of objects [6]. The Spatial pyramid pooling) (SPP-net) has the ability to generate a representation of fixed length irrespective of the image sixe or scale. Pyramid pooling shown robust results for object deformation and the performance of all Convolutional based neural networks can be improved by using SPP-net [7].

1.2 Efficient-Net Architecture

Efficient-Net formulates the basic backbone of Efficient-Net by studying the scaling method of ConvNet systems. It is quite tedious for researchers to add more layers to ConvNet, make them wider and deeper with high resolutions. To tackle all these problems, Efficient-Net model is used by using a few numbers of Floating point operation per second (FLOPS) to create a baseline ConvNet, which

is called Efficient-Net B0. After that, Efficient-Net B1 is built by scaling up Efficient-Net B0. This scaling function is then applied to Efficient-Net B7. This Efficient-Net architecture is further used to form Efficient-Net, which used fused features with different resolutions for object detection [8].

All the above-mentioned techniques have already been used by researchers for the detection of concealed weapons. These algorithms also help in detecting a weapon under a loose clothing, its shape, size, type of weapon and total number of weapons. Each technique has certain benefits for various scenarios in terms of speed, efficiency and accuracy. By using Efficient-Net algorithm, the accuracy increase as compared to other algorithms. Like CNN, RCNN etc [9].

The purpose of this work is to dive into existing systems and conduct brief research and proposes a solution that overcomes the drawback on some new and state of the art techniques or methodologies, on real-time gathered data in deep learning. Hence, the aim of this work is to represent automated hand-held weapon detection and alarm system by using Convolutional neural network (CNN) deep learning algorithm.

Detecting a gun is a challenge because of its many subtleties. Using CNNs for automatic detection of guns in video arise several challenges:

- Weapon may be handled with tow or one hand in several ways so the large part of the weapon is occluded.
- Designing a new data set is time taking.
- Automatic Weapon detection triggering alarm in real-time.
- Accurate location of weapon in the scene for triggering alarm.

Some other problems can also occur while detecting a weapon and are necessity of real-time processing such as deformation of gun and noise in an image [4,6,8].

2 Literature Review

In this field, most of the previous research depends upon either context or pose and does not allow our system to deal with both factors. Using this approach, they built a model based on You only looks once (YOLO) neural network, which is able to classify both kind of (low level and high level) threats with an efficiency of 84%.

Multiple simulated experiment to classify and evaluate disease found in five cassava leaf data set and our framework is capable of producing relatively accurate classification results despite small difference between the test set and image [10].

To achieve a high detection rate, the authors of this research increased the total number of images by taking different possible directions of pistols [8]. The functionality of the architectures of Convolutional neural networking (CNN) by giving the complete description of CNN models, which started from LeNet model and further involved ZFNet, VGGNet, AlexNet, SENet, ResNet, GoogleNet, ResNeXt, Xception, PNAS/ENAS and DenseNet [9].

The experiment results show the improved model is more efficient and finally achieved the optimal accuracy of 99.69%. Compared with other human behavior recognition methods has stronger environment and lower human invasivene [10].

Deep convolutional network (DCN), a revolutionary model: F-R CNN model, through transfer learning and evaluated the weapon detection on IMFDB which is a standard weapon database. In the paper CNN implementation using MatConvNet, MATLAB toolbox for the implementation of Convolutional neural networks for the applications of computer vision without the usage of Graphical

processing unit (GPU) [10]. Each CCTV image was able to capture the image by taking care of indoor and outdoor conditions with different resolutions to represent various scales of gun. To train data, M2Det network was used and then this trained network was authenticated by taking images from the dataset of University of central florida (UCF) crime videos. The experimental results of this research indicated that by using the proposed model, the average accuracy of weapon detection can be increased up to 18% when we compared it with the previous approaches [11].

To improve the results of object detection and its classification the domain of terrorism and military, a Multi spectral fusion system (MSFMT) is presented in this paper. This system mainly depends on the combination of dempster-shafer statistical method and deep learning techniques. In this research, MSFMT system is used to help in improving the results of classifications by creating an algorithm for fusion between multiple spectrums [12].

Single shot detection (SSD) and Region convolutional neural network (RCNN) for self-created and pre labeled image dataset for the detection of weapon. The experimental results showed that both were efficient algorithms but their real time application gave results based on a compromise between accuracy and speed. Faster RCNN method was better in terms of accuracy as it gave the accuracy of 84.6% and the accuracy of SSD method was 73.8% [13].

CCTV depends on human supervision that may cause human prone errors such as a person can miss some crime events while monitoring multiple screens at the same time. To tackle this situation, a crime intension detection system to detect crimes happening in real time images, videos and after detection this system sends warnings to human supervisor by using Short message service (SMS) sending module. Fast RCNN and RCNN methods were also used to mark the objects in the CCTV images like knife, gun, pistol and person [14].

Deep learning-based algorithms were used for object classification and detection. By using techniques of sensor fusion, a framework consisting of multi-sensor data was not only designed but also embedded by extracting the features of image modules using Raspberry pi and intel movidious stick. This framework helped in reducing the sub problems like resolution, noise by the implementation of a modified R-CNN algorithm [15]. Not only the techniques of recognizing the features of a moving object are explained in this research work but also its explained that how to classify the concealed objects present in the video frames. This paper also reviews some research gaps like it is difficult to identify the concealed objects in loose cloths and the shape and size of weapon varies, etc [16].

CNN framework is used to classify handguns in the CCTV video frames by only using edge features. Moreover, IAGMM and ViBe algorithms are used to evaluate the experimental models. These algorithms are important to give more inner detail and they have high capacity of resisting sudden changes and noise, which could happen while operating in an outdoor environment. They simulate the results by taking 1869 positive and 4000 negative images to train the CNN model [17]. Object detection structures based on deep learning are reviewed to solve many sub problems like low resolution, clutter and blocking by making several modifications in the R-CNN method. This research also provides the experimental analyses to make a comparison between different methods like R-CNN, YOLO and CNN. This review gives a basic architecture for object detection, pedestrian detection and facial detection. Finally, they proposed to work on multimodal information fusion, multitask joint optimization, contextual modeling, spatial correlations and scale adaption [14,16].

By carrying out a detailed study of the previous state-of-the-are detection models, we were able to enhance the computational efficiency which led to use EfficientDet model. EfficientDet as a model needs less Floating point operations per second (FLOPS) than YOLO, also its the needs much less parameters compared to algorithms like RCNN, Mask R-CNN, etc. which reduces the complexity

and training time of the model. Most of the prior object detection algorithm use top to bottom or either bottom to top approach, while using regular efficient connections; EfficientDet allows the flow of information in both directions using BiFPN [18].

3 Methodology

The research methodology comprised of two basic steps first the research planning and the second is research execution. The research plan comprised of two tasks including research problem identification and consequent research experiments directions while the second task is data acquisition for the model training and testing. The research execution involves a step-by-step algorithm application at the collected data sets for training and testing for the model generated using Efficient-Net.

3.1 Research Plan

3.1.1 Problem Statement

Main aim for using multi-scale feature fusion is to aggregating features in many resolutions. It gives us a list of different multi scale features as in which is representation of feature at particular level but, objective of using multi scale feature fusion is to locate the transformation that could efficiently combine the different features to generate new features list.

3.1.2 Data Collection

We collect data from session training data in our research like public data and use data with the permission of the authority. The recorded live stream video from the CCTV camera is first preprocessed into frames to clean data and avoid any noise. Later these frames are labelled by annotation to train the model. We apply algorithm on labelled images extracted from videos.

3.2 Research Execution

The step-by-step research execution flow is illustrated in the Fig. 2.

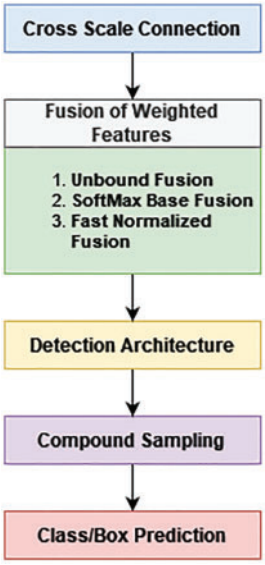

Figure 2: Research execution flow

3.2.1 Cross Scale Connection (CSC)

Conventional top-down Feature pyramid network (FPN) is fundamentally constrained by the one-way flow of information. To tackle this situation, Path of network (PANet) architecture contains an additional network, which follows the bottom-up path. Moreover, for finding more suitable feature network topology, neural architecture is used by Neural Architecture Search (NAS_FPN). But its drawback is that it requires much GPU consumption during search and beside this it is difficult to modify/interpret the found network.

After analyzing efficiency and performance of above- mentioned networks, it is observed that PANet able to achieve maximum accuracy as compare to others. Nevertheless, PANet uses a greater number of parameters and flop in it so required more Computation power. For achievement of more accuracy of the model, we proposed a methodology in which our main aim is to reducing the parameters and to achieve the better accuracy with a lesser number of modes. At first, after inspection of architecture shown in Fig. 3, we excluded those layers which have only single input [9].

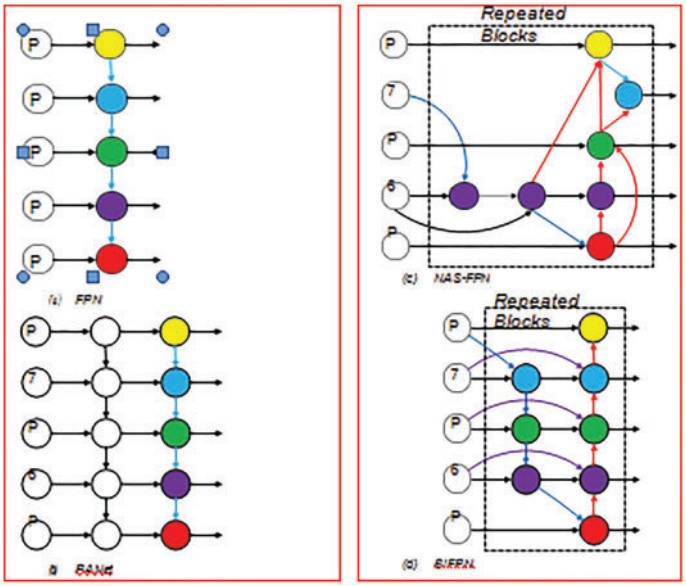

Figure 3: Directional path of networks

Intuition behind this is as simple that if a layer only has one input edge it means its contribution is minimal in network for fusing different features. By doing this, we are able to achieve simplified bidirectional architecture. In the second step, when they (input and output node) are at the same point, we add one extra edge from the original node to the destination node to fuse additional features with less cost. At last, for enabling a larger number of high-level feature fusions, we use every bidirectional path as a single layer for extracting features and utilize it multiple times. In this way, we differentiate our network from PANet that contains only one bidirectional path, we differentiate our network from PANet that contains only one bidirectional path [19].

After analyzing efficiency and performance of above- mentioned networks, it is observed that PANet able to achieve maximum accuracy as compare to others. Nevertheless, PANet uses a greater number of parameters and flop in it so required more Computation power. For achievement of more accuracy of the model, we proposed a methodology in which our main aim is to reducing the parameters and to achieve the better accuracy with a lesser number of modes. At first, after inspection

of architecture we excluded those layers which have only single input. Intuition behind this is as simple: that if a layer only has one input edge it means its contribution is minimal in network for fusing different features [13].

By doing this, we are able to achieve simplified bidirectional architecture. In the second step, when they (input and output node) are at the same point, we add one extra edge from the original node to the destination node to fuse additional features with less cost. At last, for enabling a larger number of high-level feature fusions, we use every bidirectional path as a single layer for extracting features and utilize it multiple times. In this way, we differentiate our network from PANet that contains only one bidirectional path, we differentiate our network from PANet that contains only one bidirectional path [19].

3.2.2 Fusion of Weighted Features

A common method of fusing feature with different resolutions is to first resize them to the same resolution and then summarize them. The traditional method uses the approach of considering all features with equal contribution. However, we notice that because specific input characteristics are at various resolutions, they typically contribute unequally to the output function. We suggest introducing an additional weight for each input to resolve this problem, and making the network know the value of each input feature. According to this, we used three different weighted fusion techniques [20].

3.2.3 Unbounded Fusion: $O = \Sigma I\ WI.\ II$

Here WI indicating a weight to learn and which can be treated as a multidimensional tensor i.e., per pixel. This learnable weight can also be treated as a scaler and as a vector according to need. It's also observed that it is possible to achieve comparable accuracy with less computation in comparison with other different approaches. Nevertheless, because the scalar weight is unbounded, this may theoretically contribute to uncertainty in preparation. Consequently, we turn to weight normalization to restrict the significance spectrum of increasing weight [19].

3.2.4 Softmax-Based Fusion

An interesting concept is to apply softmax to each weight, to normalize all weights to be likelihood with a significance set of 0 To 1, which reflects the value of each data. But, it leads us to extra latency cost on GPU. To tackle this problem, a fast fusion approach is proposed by us [20].

3.2.5 Fast Normalized Fusion

In this to track the feature whose weight is > 0 we used Relu to avoid any numerical instability. Likewise, the value of increasing weighted weight still dropped between 0 and 1, but it is much more effective because there is no softmax process here. This make sure that the fast fusion method produces same learning accuracy as softmax but its computation on GPUs is 30 percent less than as compare to softmax. We named this network as a Bidirectional FPN. Our final model, combine the fast-normalized fusion and bidirectional cross scale network.

$$P_6^{td} = Conv\left(\frac{w_1.\ P_6^{in} + w_2\ .\ Resize\left(P_7^{in}\right)}{w_1 + w_2 + \varepsilon}\right) \tag{1}$$

$$P_6^{out} = Conv\left(\frac{w_1'.\ P_6^{in} + w_2'.\ P_6^{td} + w_3'.\ Resize\left(P_6^{out}\right)}{w_1 + w_2 + w_3 + \varepsilon}\right) \tag{2}$$

Ptd is showing the intermediate features of level 6. Moreover, the output features of level 6 are indicated by and these are on bottom to up route shown in in Eqs. (1) and (2). Construction of other features is carried in the same manner. We also use the depth wise separable technique in order to increase the efficiency for feature fusion. After each convolution we are adding batch for activation and normalization [21].

3.2.6 Detection Architecture

On the basis of our bidirectional FPN, we proposed a new method of detection of object with lesser computation and parameters. This section includes the discussion of our network's architecture and how we proposed a new method of compound scaling for our model.

Fig. 4 shows the architectural diagram of our proposed model. In which it is clearly visible that our model is utilizing one stage detector paradigm. As the back bone of our network we used efficient nets which are pre-trained ImageNets.

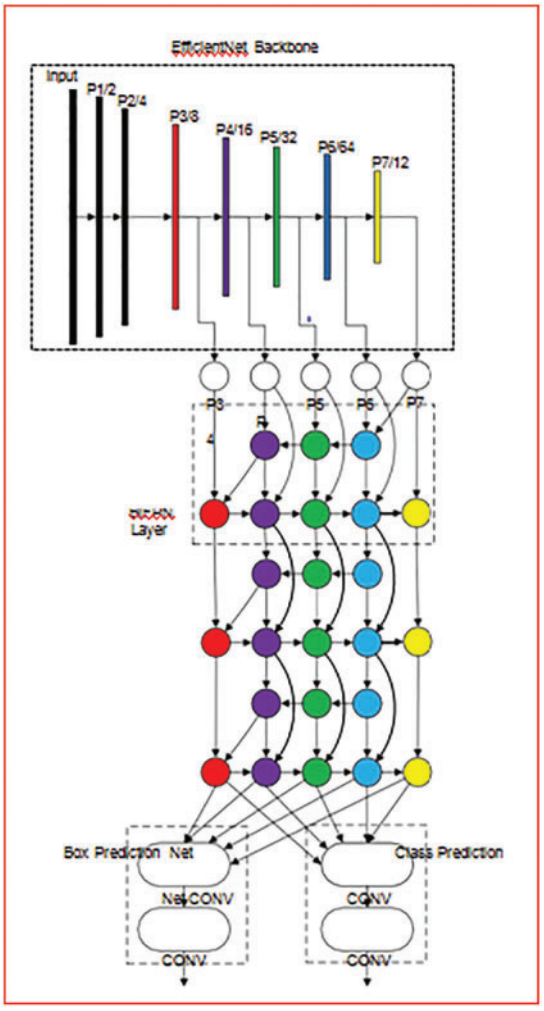

Figure 4: Level of features of efficient backbone network

Bidirectional FPN proposed by us is serve as the network which is responsible for feature extraction. Which utilize feature from level 3–7 from efficient net and then in loop applies the technique of bottom-up and top down two-way feature fusion (as shown in Tab. 1 below).

Table 1: Level of features of network

Symbol	Input size	Backbone network	BiFPN channel	Box layers	Class layers
D0(ϕ=0)	512	B0	64	3	3
D1(ϕ=1)	640	B1	88	4	3
D2(ϕ=2)	768	B2	112	5	3
D3(ϕ=3)	896	B3	160	6	4
D4(ϕ=4)	1024	B4	224	7	4
D5(ϕ=5)	1280	B5	288	7	4
D6(ϕ=6)	1280	B6	384	8	5
D7	1536	B6	384	8	5

Such fused features are fed to a class and box network respectively to generate predictions of object type and bounding boxes. Moreover, weights of box-network and class are used in all level of features [17,19].

3.2.7 Compound-Scaling

To maximize both accuracy and performance, we would like to build a collection of models capable of meeting a broad range of resource constraints. One main question here is how to expand the baseline of our proposed model. Earlier methodology mostly used deeper networks as backbone to scale up their models i.e., ResNet, Amoeba Net [7].

These models require the large number of layers and huge number of FLOPs. Moreover, one of the drawbacks of these networks is they support limited scaling dimensions. However, in the recent timeline a number of networks show comparable performance and effective results on classification of image. As they carried out in depth analysis of features by enhancing all dimensions of the proposed networks (width, depth, height). By studying those networks, we proposed a novel scaling method i.e., 'Compound Scaling' for detection of different objects. The proposed technique of scaling utilizes φ (which is simple compound coefficient) to enhance all the dimensions of backbone, bidirectional FPN, resolution and box/class network together [21,22].

For scaling we used heuristic based technique in order to prevent the large number of scaling dimensions. Because an object detector has a large number of scaling dimensions as compared to image classification. But we keep using our ideas of scaling up all the dimensions jointly.

3.2.8 Backbone-Network

To use the pertained checkpoints of ImageNet easily we employ the efficient-net with the same coefficients of width/depth.

3.2.9 H. Class/Box Prediction Network

To use pertained checkpoints of ImageNet easily we employ the efficient-net with the same coefficients of width/depth. To make the width same as bidirectional FPN and to increase the depth linearly we use the equation [19,23].

Use the following equation Presented in Eq. (3).

$$D_{box} = D_{class} = 3 + \left[\emptyset / 3 \right] \tag{3}$$

In Fig. 5 we clearly show that this technique of scaling significantly enhances the efficiency as compared to single dimension scaling method.

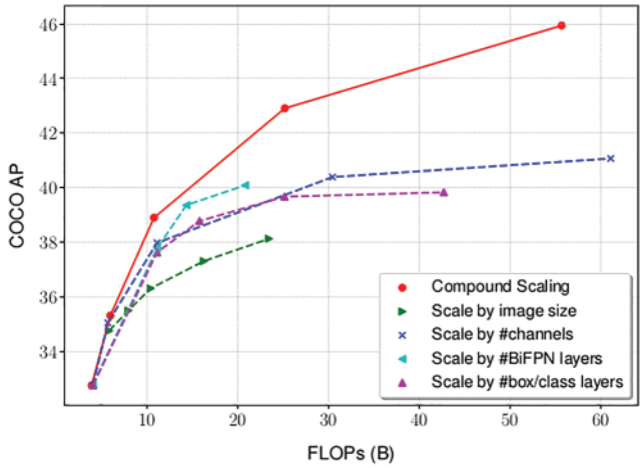

Figure 5: Technique of scaling

In Fig. 5 shown different ratio of FLOPS and COCOAP with all techniques in different color and also shown enhance the efficiency and show the value FLOPS with different color. The compound scale COCO AP is round about 46 and scale by image size COCO AP is 38 etc.

4 Implementation Details

The architecture of the proposed network is designed in Tensorflow framework. The training and the validation process is performed using Tensorflow. The framework is installed on Ubuntu 16.04 with python 3.5 language. The hardware and software specification are as follow.

4.1 Hardware Specification

The training of model is performed on Corei7 (7th generation) CPU has 8 cores with 32 GB DDR3 RAM. NVIDIA GPU 1080 Ti having 11GB DDR5 memory and 3584 CUDA cores are used for parallel processing and matrix multiplication and other math operations. We used the mini batch of 200 samples while training and the validation. The Mean square error (MSE) is used for loss calculation. The mathematical explanation of MSE is described in Eq. (4). SGD is used for optimization of weights along the dynamic learning rate. The dynamic learning rate along the epochs is shown in Tab. 1 [24].

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_i - q_i)^2 + \ldots + (p_n - q_n)^2} \tag{4}$$

Here p is predicted and q actual labels of the annotated dataset provided while training and validation, p1 represents the result of the 1st sample predicted by the trained model and the q1 represents the ground truth of 1st sample annotated by the human.

The above table shows the dynamic learning rate that how much learning rate will decrease after each epoch in fractions between a specified range of the epochs. The model is train for 1000 epochs with mini batch of 200. The learning rate from epochs 1 to 500 is 0.001, from 501 to 800 is 0.0001 and from 801 to 1000 is 0.00001 as shown in Tab. 2. The reason of dynamic learning rate is to accelerate the training process in initial steps. The higher learning rate means the higher learning jumps of classification curve.

Table 2: Dynamic learning rate

Epochs	Learning rate
1–500	0.001
501–800	0.0001
801–1000	0.00001

To handle the over fitting dropout is also used to randomize the feature extraction and selection. We try different values of dropout to check where the best features are selected and extracted. The train and test loss along the dropout ratio is shown in Tab. 3.

Table 3: Dynamic train loss and test loss

Dropout ratio	Train loss	Test loss
0.4	0.00156	0.0325
0.5	0.00164	0.00173
0.6	0.00982	0.0145

We observe for dropout ratio 0.5 model gives us best loss values. The values of train and test loss are much near than for other dropout ratio. So, we continue our whole training for dropout ratio 0.5. The training process for 1000 epochs with this hardware take 50 h to complete [25].

4.2 Challenges

SGD optimizer navigating to global minimum loss by taking step towards where the loss decreases.

sBut sometimes the SGD stuck into local minima because there is no next point there the loss recrudesces. So. It stuck in local minima as shown in Fig. 1 and optimization stops.

This problem is solved by momentum. The momentum accelerates the SGD to move in the desired direction as shown in Fig. 6.

Does momentum work by adding the fraction (lambda) if the last update vector in the currently updated vector as explained in Eq. (5) and Eq. (6).

$$v_t = \gamma v_{t-1} + \eta \nabla_\theta J\ (\theta) \tag{5}$$

$$\theta = \theta - v_t \tag{6}$$

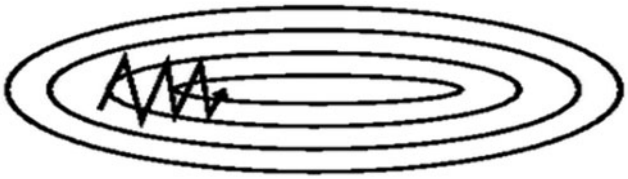

Figure 6: Optimization without momentum

While experiments many people use different values. The most common value of the (lambda) for momentum is 0.9 or near to this. After 1000 epochs the values of MSE shown in graph in Fig. 8. In Fig. 7 we clearly show that this technique of scaling significantly enhances the efficiency as compared to single dimension scaling method.

Figure 7: Optimization with momentum

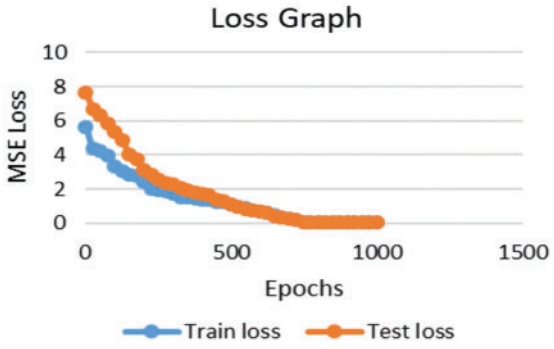

Figure 8: Mean square error graph

We have used real time data in this research like Mock up exercise (demonstration data) of images and videos on several types [20,25].

5 Results

5.1 Confusion Matrix

In this confusion matrix shown in Fig. 9, we use 15000 plus images and videos of live data and the result of our research is better than previous research. Through efficient-net the result better results than previous one and use local and global data and on live videos. In Confusion matrix show the accuracy and loss rate of training and validation.

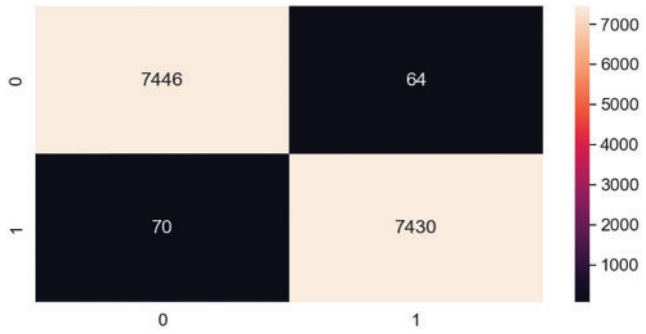

Figure 9: Confusion matrix

Tab. 4 shows result comparison of the proposed model with state-of-the-art deep learning algorithms. Transfer learning on same dataset through faster RCNN with ZFNet and VGG-16 accurately identified weapons with only % and % of accuracy. On the other hand Yolov3 and YOLO v4 achieved % and % of accuracy. The comparison shows that the proposed model outperformed all the previous approaches with 98.12% accuracy.

Table 4: Model comparison

Model	F1 score	Precision	Recall
Faster RCNN	0.9301	0.9611	0.9414
Yolov4	0.9321	0.9134	0.9266
Proposed model	0.9883	1.0	0.9712

5.2 Accuracy

As we increase the ratio of session data the accuracy increases and loss ratio decreases as comparative to previous observation, shown in Fig. 10. The accuracy rate increase increases when the Epochs rate increases.

In Fig. 10 the Epoch size increase and total Epoch in this fig is1500. When Epochs size increase and the accuracy also increase.

5.3 Loss

The Fig. 11 illustrates the training and validation loss decreasing trend similar to mean squared error.

The loss rate decreases when the value of Epochs increases shown in Fig. 11. BY using Efficient-Net the loss rate low as compared to other Algorithms This figure show the training and validation loss and also show Epochs size decrease and the loss rate of training and validation.

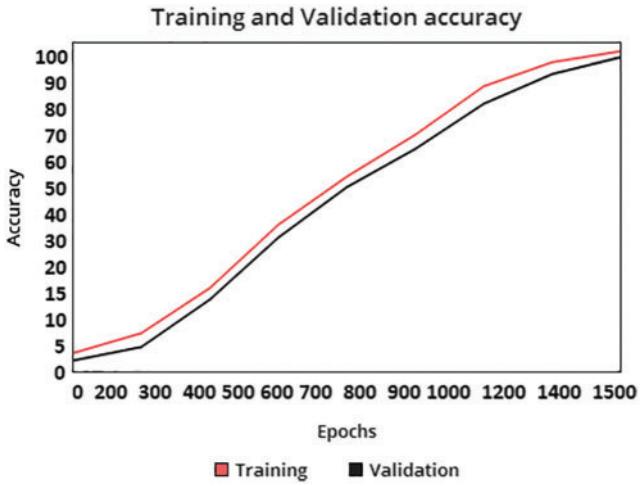

Figure 10: Accuracy ratio

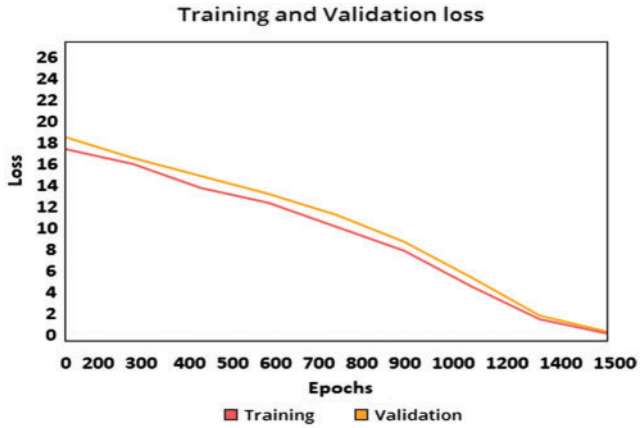

Figure 11: Training and validation loss

6 Conclusion

In this paper, we presented Efficient-net used for object detection. Most of the object detection is done with CNN based neural networks in the modern era. The basic motivation behind this Efficient-net base weapon detection system that employee's region proposal network was to reduce the false positive and make a model with near to real time efficiency that is mainly the center of attention of the researchers.

For increasing the robustness and reducing the false positive of the model we gathered local data set and live videos of cameras that's we defined. We trained these models and used pre-trained feature extractors because it saves a lot of time to fine tune a model according to our problem. Experiments show that our model obtained better results for weapon detection system than the previous research. By using Efficient-Net algorithms the better accuracy achieved as compared to other algorithms.

We will extend our model to cover move objects more efficiently and will drive it to give our own model.

Acknowledgement: We thank our families and colleagues who provided us with moral support.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. Alexandrie, "Surveillance cameras and crime: A review of randomized and natural experiments," *Journal of Scandinavian Studies in Criminology and Crime Prevention*, vol. 18, no. 2, pp. 210–222, 2017.

[2] M. P. Ashby, "The value of CCTV surveillance cameras as an investigative tool: An empirical analysis," *European Journal on Criminal Policy and Research*, vol. 23, no. 3, pp. 441–459, 2017.

[3] Y. LeCun, Y. Bengio and G. Hinton, "Deep learning," in *Int. Society for Behavioral Neuroscience (ISBN) Conf.*, Wuhan, China, pp. 436–444, 2015.

[4] K. He, X. Zhang, S. Ren and J. Sun, "Delivering deep into rectifiers: surpassing human-level performance on imagenet classification," in *IEEE Int. Conf. on Computer Vision (ICCV)*, Santiago, Chile, pp. 1026–1034, 2015.

[5] C. Bagavathi and O. Saraniya, "Hardware designs for histogram of oriented gradients in pedestrian detection: A survey," in *Int. Conf. on Advanced Computing and Communicating Systems (ICACCS)*, Coimbatore, India, pp. 849–854, 2019.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed *et al.,* "SSD: Single shot multibox detector," in *European Conf. on Computer Vision (ECCV)*, Amsterdam, Netherlands, pp. 21–37, 2016.

[7] K. He, X. Zhang, S. Ren and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[8] B. Abruzzo, K. Carey, C. Lowrance, E. Sturzinger, R. Arnold *et al.,* "Cascaded neural networks for identification and posture-based threat assessment of armed people," in *IEEE, Int. Symp. on Technologies for Homeland Security (HST)*, Woburn, MA, USA, pp. 1–7, 2019.

[9] A. Dhillon and G. K. Verma, "Convolutional neural network: A review of models, methodologies and applications to object detection," *Progress in Artificial Intelligence*, vol. 9, no. 2, pp. 85–112, 2019.

[10] G. K. Verma and A. Dhillon, "A handheld gun detection using faster R-CNN deep learning," in *Int. Conf. on Computer and Communication Technology (ICCCT)*, Allahabad, India, pp. 84–88, 2017.

[11] J. Lim, M. I. Al Jobayer, V. M. Baskaran, J. M. Lim, K. Wong *et al.,* "Gun detection in surveillance videos using deep neural networks," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conf. (APSIPA ASC)*, Lanzhou, China, pp. 1998–2002, 2019.

[12] D. M. El-Din and A. E. Hassanien, "Multi-spectral fusion system based on deep transfer learning and dempster-shafer theory," *Journal of Theoretical and Applied Information Technology*, vol. 98, no. 6, pp. 1817–3195, 2020.

[13] H. Jain, A. Vikram, A. Kashyap and A. Jain, "Weapon detection using artificial intelligence and deep learning for security," in *Int. Conf. on Electronics and Sustainable Communication Systems (ICESC)*, Coimbatore, India, pp. 193–198, 2020.

[14] U. V. Navalgund and K. Priyadharshini, "Crime intention detection system using deep learning," in *Int. Conf. on Circuits and Systems in Digital Enterprise Technology (ICCSDET)*, Kottayam, India, pp. 1–6, 2018.

[15] G. Raturi, P. Rani, S. Madan and S. Dosanjh, "ADoCW: An automated method for detection of concealed weapon," in *Fifth Int. Conf. on Image Information Processing (ICIIP)*, Shimla, India, pp. 181–186, 2019.

[16] R. Mahajan and D. Padha, "Detection of concealed weapons using image processing techniques: A review," in *First Int. Conf. on Secure Cyber Computing and Communication (ICSCCC)*, Jalandhar, India, pp. 375–378, 2018.

[17] F. Gelana and A. Yadav, "Firearm detection from surveillance cameras using image processing and machine learning techniques," in *Smart Innovations in Communication and Computational Sciences*, Springer, Singapore, pp. 25–34, 2019.
[18] Z. Q. Zhao, P. Zheng, S. T. Xu and X. Wu, "Object detection with deep learning: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 1–21, 2019.
[19] S. H. Tsang, "Review-NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Long beach, CA, USA, pp. 7036–7045, 2019.
[20] E. Arif, S. K. Shahzad, R. Mustafa, M. A. Jaffar and M. W. Iqbal, "Deep neural networks for gun detection in public surveillance," *Intelligent Automation and Soft Computing (IASC)*, vol. 32, no. 2, pp. 909–922, 2022.
[21] Y. Cheng, W. Liu and W. Xing, "Weighted feature fusion and attention mechanism for object detection," *Journal of Electronic Imaging*, vol. 30, no. 2, pp. 23015–23031, 2021.
[22] M. S. Hosseini, J. S. Zhang, Z. Liu, A. Fu and J. Su, "CONET: Channel optimization for convolutional neural networks," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Nashville, TN, USA, pp. 326–335, 2020.
[23] J. Nazir, M. W. Iqbal, T. Alyas, M. Hamid, M. Saleem *et al.,* "Load balancing framework for cross-region tasks in cloud computing," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 1479–1490, 2022.
[24] C. C. Chen, J. Y. Ba, T. J. Li, C. C. Chan, K. C. Wang *et al.,* "Efficient net: A low-bandwidth IoT image sensor framework for cassava leaf disease classification," *Sensors and Materials*, vol. 33, no. 11, pp. 4031–4044, 2021.
[25] C. Y. Luo, S. Y. Cheng, H. Xu and P. Li, "Human behavior recognition model based on improved effivient net," *Procedia Computer Science*, vol. 199, no. 1, pp. 369–376, 2022.