Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.021061
Article

Tech Science Press

Deep Neural Networks for Gun Detection in Public Surveillance

Erssa Arif[1,*], Syed Khuram Shahzad[2], Rehman Mustafa[1], Muhammad Arfan Jaffar[3] and Muhammad Waseem Iqbal[4]

[1]Department of Computer Science, Superior University, Lahore, 54000, Pakistan
[2]Department of Informatics and Systems, University of Management and Technology, Lahore, 54000, Pakistan
[3]Faculty of Computer Science and Information Technology, Superior University, Lahore, 54000, Pakistan
[4]Department of Software Engineering, Superior University, Lahore, 54000, Pakistan
*Corresponding Author: Erssa Arif. Email: erssaarif1@gmail.com
Received: 21 June 2021; Accepted: 26 August 2021

Abstract: The conventional surveillance and control system of Closed-Circuit Television (CCTV) cameras require human resource supervision. Almost all the criminal activities take place using weapons mostly handheld gun, revolver, or pistol. Automatic gun detection is a vital requirement now-a-days. The use of real-time object detection system for the improvement of surveillance is a promising application of Convolutional Neural Networks (CNN). We are concerned about the real-time detection of weapons for the surveillance cameras, so we focused on the implementation and comparison of faster approaches such as Region (R-CNN) and Region Fully Convolutional Networks (R-FCN) with feature extractor Visual Geometry Group (VGG) and ResNet respectively. Training and testing are done on database that consists of local environment images. These images are taken with different type and high- resolution cameras that minimize the idealism. Some metrics also defined to reduce the false positives which are specific to the solution of problem. This research also contributes to the constitution of a hybrid CNN model of both faster-based R-CNN and R-FCN. Both hybrid and existing models experimented to reduce false positive in weapon detection. Result represented in graph with calculation during and after training with confusion matrix and hybrid model results better than other models.

Keywords: Gun detection; convolution neural networks; video surveillance

1 Introduction

The population of the world is increasing exponentially, so it becomes impossible to keep conventional security methods in practice any further. As national and public security is the focus of every country in the era technology is getting enormous attention in policing. The immense crime rates resulting from using gun have led government to seek solution to deal with such terrorist incident. These incident have a negative impact on public security [1]. There might be many departments working to provide security but with the manpower, it can't be done at every second, so it requires a modern system that fills the hole. The main

point that can elaborate the requirement of technology involvement in policing is the reactive approach of conventional security methods.

All over the world, a huge number of people resolve gun-related violence issues every year. A fully automated computer-based system is developed in this work, specifically basic weapons like rifles and handguns [2]. Today, a huge number of criminal activities are taking place using handheld arms e.g., guns, pistols, revolver, and semi machine guns or shotguns also in some cases [3]. These activities can be reduced by monitoring and identifying them at an early stage. The manual surveillance system still need human eye to detect the abnormal activities and it take a sufficient amount of time reporting security official to tackle the situation [4].

The way to minimize the violence is by early detection of suspicious activity so that law enforcement can take necessary action. Current control and surveillance still need human supervision and interference. CCTV surveillance cameras are broadly in use for monitoring and other security purposes [5]. There might be negligence, misconception so the most suitable solution of this particular domain is to the deployment of surveillance cameras along with automatic hand-held weapon detection with the alert trigging system.

These observations forced the researchers on a need for an active surveillance system that works on automated weapon detection algorithms. These algorithms help the operator in detecting and recognizing the presence of guns in real-time. In this way, these algorithms are helping the authorities to save lives by alerting them to take immediate action according to the situation. Deep learning approaches [6] are rapidly used nowadays because of the capability of giving data-driven solutions to such problems. Extraordinary results of image classification using deep neural networks have exceeded human performance [7].

The R-CNN model works by combining region proposals using different CNNs. The R-CNN uses a small amount of detection data to localize and train the objects using a deep network. For the classification of objects, it uses ConvNet to achieve a remarkable accuracy of object detection. The R-CNN can scale large-scale object classes and it does not need to use approximate techniques like hashing [8]. A fast R-CNN algorithm is developed to fix the disadvantages of SPP-net and R-CNN by improving the system's accuracy and speed.

The R-CNN detector crops the region proposals to resize them but, in fast R-CNN, entire image is used to process. Its, detection rate is also higher than other two by training in a single stage with the help of multi-task loss. By sharing computations of overlapping regions in an image, fast R-CNN also shows efficiency over R-CNN. Moreover, it does not require any disk storage to store the image features [9].

The R-FCN can have a costly subnetwork like fast R-CNN and Faster R-CNN and is entirely convolutional through all computations that are shared on a complete image. In other words, R-FCN is a combination of fully shared convolutional architectures where all layers are convolved to classify the Region of Interests (ROIs) into different object types and backgrounds. So, R-FCN can reduce the total amount of work, which is needed for each ROI. Moreover, the feature maps of R-FCN do not depend on ROIs and it is possible to calculate them outside of ROIs [10].

Furthermore, a study concluded that the detector is performed his/her activities admirably to detect handguns in various scenes with diverse rotations, scales, and shapes. The results of the perceptible study are shown that YOLO-V3 can be used for gun detection as an alternative to faster R-CNN. The YOLO-V3 can be used in a real-time environment to provide much faster speed and to get nearly identical accuracy [11].

Two types of feature extractors that have been used in this research are VGG and ResNet as baseline models for feature extraction. Both models are CNN-based and their pre-trained weights are publicly

available and trained on millions of images, so there is no need to train both models. These pre-trained models are being used and modified to get significant results in this research.

2 Literature Review

The functionality of the architectures of CNN by giving the complete description of CNN models and the research is focused mainly on three important applications of deep learning neural architectures namely small arms detection. For every model, a summary of a detailed review composed of the system's application, its database, and accuracy claimed was explained as a guideline to help in future work [12]. United Nations Office (UNO) reported statistics on crimes and drugs revealed that most of the crimes involve guns. It described an automatic handheld gun detection system for surveillance, video, and control purposes. One pioneering solution to this issue is that there must be a precise and automatic detection of handheld arm/gun and alert systems in surveillance cameras. In the last few years, deep learning CNNs have achieved good outcomes for machine learning methods. It formulated the detection issue into problem of reducing the false positive and resolve it with the development of the training dataset which is led by the outcomes of a classifier based on deep CNN. The study also described some challenges while detecting pistols in videos automatically. They claimed that their work is the first automated weapon detection and alarm system which uses deep ConvNet models. The research work is focused on evaluating the near real-time solution, accurate detectors, fast detectors, assess performance, and suitability were evaluated using a new metric as automated gun detection alarm system, alarm activation time per interval [13].

The training result confirm that YOLO V3 object detection model by training it on our customized dataset. Applying this model can attempt to save human life and can be implemented in high end surveillance and security robots to detect a weapon or unsafe assets to avoid any kind of assault or risk to human life [14].

A very large-scale dataset consisting of different handguns was introduced. The images of handguns were taken by CCTV cameras. 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 the gun. The experimental results indicated that by using the proposed model, the average accuracy of weapon detection can be increased up to 18% when compared with the previous approaches [15]. Further, research represented a digital framework by using Google's Tensor flow API for the identification of handguns or firearms. To train their system for the successful identification of handguns of different sizes, shapes, and orientations, it used a neural network of MobileNetV1. The experimental results have shown the efficiency of the early gun detection system [16].

Another model to detect anomaly using deep learning algorithms showed better results of minimum time when they used the features of both anomalous and normal surveillance videos to get rid of high training time. Furthermore, they have also explained the significance of locality by showing that for long videos, it is better to locate anomalies to obtain improved performance [17]. Moreover, a study was conducted to classify images based on the level of threat detection using X-ray imagery from baggage security. By using BoVW system that has descriptors and detectors for features, supported by Random Forest Classification (RFC) and Support Vector Machine (SVM), they have illustrated the accuracy capability of these methods following this model of image classification over a large-scale data set of X-ray baggage imagery [18].

To expand the research work in this field, they used a dataset called SIX ray, which was consisted of a million X-ray images captured from real-world situations and therefore these images covered many complex scenarios. They manually marked 6 types and twenty thousand illegal items. The dataset was nearly 100 times bigger than the already available datasets [19]. The method was used in this research by splitting the original image into different sub-images before implementing the chamfer matching

algorithm on them. Two methods were used such as machine learning and voting to detect whether an image had a gun or not. After that, they used SVM to study the importance of subdivisions but that method was also a failed one. However, by using SIFT method of feature descriptors for the input of the machine learning algorithm; they achieved likely better results [20].

Furthermore, to ensure security information, Intrusion Detection System (IDS) has proved very helpful, which works on the principle of accurately identifying the various attacks in a system. The study explored an intrusion detection network established on deep learning by using Recurrent Neural Networks (RNN-IDS). Multiclass classification, binary classification, multiple neurons, and impacts of various learning rates were used to evaluate the performance of the castoff approach [21]. Crime scene estimation without the intervention of humans can have a remarkable impression on computer vision. This study used to represent the CNN model to detect blood, a gun, and a knife from the image. By detecting harmful objects from an image, it can be predicted that whether a particular crime happened or not and what is the location of the crime scene. It is mainly focused to detect accuracy to lessening the probability of giving wrong alerts by ensuring the efficient use of this system. At first, only used 50 to 100 images and gained an accuracy of 25% to 30% but after increasing the training dataset, the accuracy rate improved remarkably [22].

Moreover, a comparison between two different models like VGGNet19 and Google Net Inception V3 was also made in this research to show that the results obtained from VGG19 were more accurate with regard to training accuracy. Fast RCNN and RCNN methods were also used to mark the objects in the CCTV images like knife, gun, pistol and person.

3 Methodology

Data collection in different sizes and varieties is important to get significant results in research. It is a very crucial part because the model training is depending on it where data describes that how good and precise the model will work.

In this research, the data collection (indoor and outdoor images) is done at different times and circumstances. The dataset consists of different images that are captured from different types of cameras having different specifications. Different types of images like colored, greyscale, and black & white with different sizes and pixels density. The objects which are present in gathered images are guns or weapons. An image contains at least one gun and a maximum of 4 guns that are used in our research work. Total images that are used, 10,000, and the data set splitting is done into test data, training data, and validation data. The data set is divided into test data and training data with the ratio of 30% and 70% respectively. and the ratio also possible 20% and 80% depends on image quality and accuracy.

Dataset preprocessing is also done in which the dataset images are resized to a common scale of 1280 × 720 splits. Tab. 1 given below shows the classes and total images in a data repository.

Table 1: Data repository

Database	Classes	Total images	Images of weapons	Rest of images
1	1	10,000	9000	1000

In Tab. 1 explain the total 10000 images used and 9000 images used related to we open of different kinds of gun.

The preprocessing is done for practical works and resizing images to a common aspect ratio and data annotation. Images are annotated to 1280 × 1080 aspect ratio, and then annotation is done in which images are labeled. Labeling is done by using software "labeling". Data is annotated on the PASCAL VOC standard. A feature extractor is also a CNN architecture that is used to extract features from the input images which are multilayer matrices [23].

3.1 Visual Geometry Group

A CNN is a multilayer special type of neural network which are designed to directly recognize the visual pattern with minimal preprocessing from pixel images. The VGG−16 is one of the most famous feature extractors which was developed by Xue et al. [24]. It is a CNN and as its name is consists of 16 convolutional layers. Its architecture is very uniform with 3x3 convolutions with a high number of filters. The weight configuration of VGG16 is available publicly that is being used as a baseline by many other feature extractors. VGGNet has 138 million parameters that make it challenging to handle.

The VGG-16 is being used by many researchers [6,10,16,19,21,23] as a feature extractor along with Faster CNN. There is no need to train a new CNN for this purpose because the pre-trained CNN weights are publicly available and the first 5 convolutional layers are mostly used to fulfill the requirements. The VGG−16 uses a very small 3 × 3 filter size in the convolutional layer which is 5 with 1-pixel padding with convolutional stride fixed to 1 pixel. Deep CNN like VGG-16 is trained in the loss minimization prediction. Let say there is a 2D input image x and y and the equivalent output class labels, the purpose of the training is to iteratively reduce the overall loss defined as:

$$J(w)= \frac{1}{N}\sum_{i=1}^{N} L(f(w;\ xi),\ yi)+ \lambda R(w) \tag{1}$$

The VGG-16 is used in our research for feature extraction. VGG-16 extracts feature from an image that could be black and white (grayscale) or colored (RGB). Eq. (1) illustrates on RGB input image convolution is performed with a smaller width of 64 channels in the first layer and it increases by the multiple of 2 after performing max-pooling every time till the width reaches 512 channels. Over 2 × 2 pixel window max-pooling is performed with stride 2.

$$G[m,\ n]= (f * h)[m,\ n]= \sum_{j}\sum_{k} h[j,\ k]f\ [m-j,\ n-k] \tag{2}$$

After a stack of convolutional performed in layers, there comes fully connected layers and two of them have 4096 channels and the third one containing 1000 channels in which each channel represents a single class of an object. In Eq. (2), at the end here comes the soft-max layer.

$$Y\ =\ ReLU(X) \tag{3}$$

In VGG-16 architecture, there may be many hidden layers are equipped with ReLU shown in Eqs. (3), (4).

$$ReLU(X)=\ \max\ (0,\ \ x) \tag{4}$$

3.2 Resnet

Efficiently trained networks with 100 layers and 1000 layers also and the main reason for using residual block is to minimize the issue of vanishing gradient by employing the activations from previous layers until the next layer learned the weights. Residual blocks work perfectly when a single non-linear layer is stepped, or all middle layers produce the same size of feature set. Fig. 1 given below shows the residual block [24].

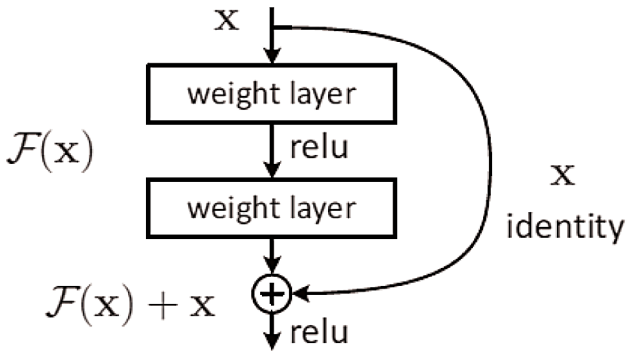

Figure 1: Residual block used in residual network

We used ResNet along with the R-FCN model for our practical work (an ensemble model). ResNet has won first position for classification competition in the ILSVRC-2015, by getting the highest error rate of 3.57 percent. ResNet has also earned first position in ILSVRC and COCO-2015 contests in ImageNet detection, ImageNet localization, Coco detection, and Coco segmentation. To detect better improvement up to 28 percent, the replacement of VGG-16 layers done with Faster R-CNN and ResNet-101.

3.3 Region Proposal Network

The Regional Proposal Network (RPN) takes the output feature maps from the previous CNN as input. It slides the 3x3 filters with 512 output channels, over the feature maps to make region proposals. Fig. 2 shows the basic concept of RPN containing 3×3 filters with 512 output channels.

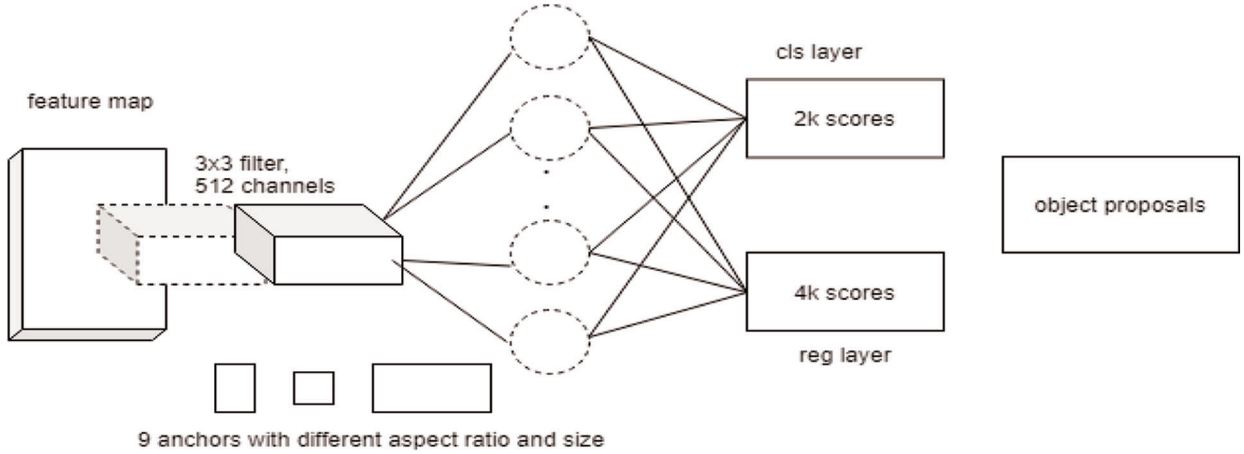

Figure 2: Region proposal network

For each location on the features map, RPN makes multiple anchor boxes. RPN takes all the reference boxes and outputs a set of good proposals up to a certain threshold and the rest are rejected. It does by having two different outputs for each of the anchors. 1st is the probability score that an anchor contains an object or not or the predicted anchor is background or foreground. It is also called binary classification e.g., 2k scores. 2nd output is bounding box regression for adjusting the anchor to better fit the object; it is predicting e.g., 4k coordinates. RPN proposes object proposals only. It does not try to classify any object.

3.4 Region of Interest Pooling

ROI pooling is a neural-net layer that can be used for object detection tasks to maintain high-level accuracy. For this purpose, it can pool to extract fixed-size feature maps to get good results for every ROI. to maintain the accuracy for object detection tasks, the layer takes two inputs such as:

1. A fixed-size feature map was obtained from a CNN.
2. The N × 5 matrices representing a list of regions of interest, where N is the number of ROIs. In 5, the first column represents the image index and the remaining 4 are the coordinates of the top left and bottom right corners of the region [14].

The regions proposed by the RPN have arbitrary sizes. To make them uniform an ROI pooling layer is used which accepts the regions proposed by the RPN and makes them uniform I, in other words, make them of the same size. The main objective of ROI pooling provides the fix length output to fully connected layers.

3.5 Max-Pooling

The sample-based discretization procedure is another name of the max-pooling process. The importance of the max-pooling process is not only to down sample an input image, hidden layer matrix but also to reduce dimensionality. It also allows for the assumptions to be made about the features contained in the sub-regions binned [14,21]. Max pooling used and its length is fixed.

Max-pooling also helps to over fit. It also reduces the computational cost by minimizing the parameters to learn. It is tendered by put on a max filter to the non-overlapping primary representation of sub-regions as shown in Fig. 3 below:

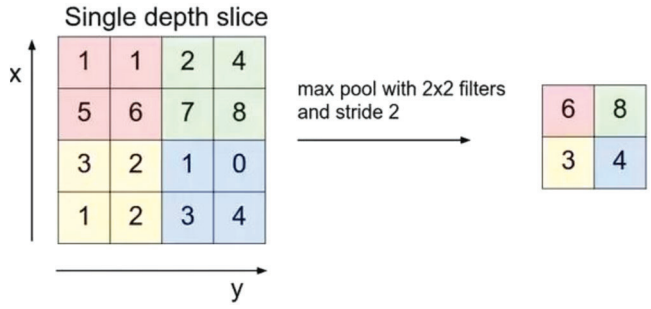

Figure 3: Max-pooling with stride 2

Max pooling mathematical representation is described below in Eqs. (5), (6):

$$Y = \text{Max} - \text{pooling}\ (\text{ixy}) \tag{5}$$

$$\text{Max} - \text{pooling}\ (\text{ixy}) = \max\ (\text{i11},\ \text{i12},\ \text{i13},\ \ldots\ \text{inn}) \tag{6}$$

3.6 Classification

The fixed length of output produced by the ROI pooling is further used by the fully connected layers. There are two pipelines of fully connected layers. One to predicting the region of interest for the object and the second for predicting the class of objects present in the ROI box [21,25].

3.6.1 Cross Entropy

To measure the amount of performance of a classification model, the cross entropy or log loss can be used. Cross-entropy loss, or log loss, is enabled to produce the output whose probability value between

0 and 1. Cross-entropy loss increases as the predicted probability deviate from the genuine label. So, predicting a probability of .012 when the actual observation label is 1 would be bad and result in a high loss value. A perfect model would have a log loss of 0. The Fig. 4 describes the predicted probability and loss class which shows that the predicted probability decreases.

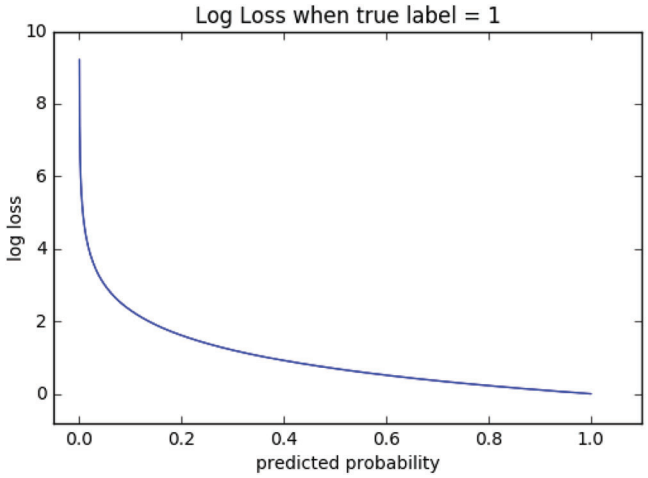

Figure 4: Cross entropy

3.6.2 The Range of Cross Entropy and Log Loss

Fig. 4 given above shows the cross-entropy and the range of log loss. In this graph, the weapon = 1 is considered as a true reflection for the possible range of loss values. It can see that log loss gradually decreases when projected probability approaches 1. However, it is shown clearly that the log loss decreases rapidly when the predicted probability decreases. Log loss penalizes both types of errors, but especially those predictions that are confident and wrong. Cross-entropy and log loss are slightly different depending on the context, but in machine learning when calculating error rates between 0 and 1 they resolve to the same thing. In binary classification, where the number of classes M equals 2, cross-entropy can be calculated as shown below in Eqs. (7), (8):

$$M = (y \log(p) + (1 - y) \log(1 - p)) \tag{7}$$

If M>2 (i.e., multiclass classification), we calculate a separate loss for each class label per observation and sum the result.

$$a \sum_{C=1}^{M} Yo, \ clog(Po, \ c) \tag{8}$$

M-number of classes (dog, cat, fish)

- log - the natural log
- y - binary indicator (0 or 1) if class label cc is the correct classification for observation O
- p - predicted probability observation is of class c

3.7 Optimization

After the loss calculation, the optimization of the parameters is done by using an optimization function. As the training data was very large and it requires a huge amount of memory to load all data

at once. The solution to this problem is to split the data into mini-batches and optimize the parameters of the model.

3.7.1 Stochastic Gradient Descent with Momentum

A Stochastic Gradient Descent (SGD) optimizer is used for model optimization using a mini-batch. SGD optimizer updates parameters by using given data (x(i)) and label (y(i)), also shown below in Eq. (9).

$$\theta = \theta \ \eta.\nabla \theta \ J(\theta; \ x(i); \ y(i)) \tag{9}$$

The mini-batches are used to optimize the parameters instead of one sample. So, the optimization of parameters is performed using Eq. (10).

$$\theta = \theta \ \eta.\nabla \theta \ J(\theta; \ x(i; \ i + n); \ y(i.i + n)) \tag{10}$$

The gradients calculated by the data's mini-batch, partially update the parameters of the model using the learning rate. The value of the learning rate is between 0 and 1. The 0 means, parameters are not updated at all, and 1 means the parameters are completely updated. Value of the learning rate shows the performance of the model. A smaller value of learning rate makes the model more accurate but requires much time and a larger value of learning rate optimizes the model fast, but it will not perform well at testing time. So, we use a large learning rate at the beginning of training and gradually decrease it along the epochs.

3.7.2 Stochastic Gradient Descent Loss Behavior

$$\upsilon t = \gamma \upsilon t - 1 + \eta \nabla \theta \ J(\theta) \tag{11}$$

SGD optimizer navigates to global minimum loss by taking step towards where the loss decreases. But sometimes the SGD stuck into local minima because there is no next point of loss advances. This type of problem is solved by momentum through Eq. (11) because it accelerates the SGD to move in the desired direction.

3.8 Hardware and Software Resources

Training and implementing deep learning-based solutions require a lot of computational resources and storage for storing models and datasets. We have implemented our system on a desktop-based computer having Ubuntu 16.04 as the core operating system. The desktop computer used for training was powerful with 7 Central Processing Unit (CPU) cores and 16 Gigabytes (GB) Random Access Memory (RAM) for fast performance. Considering our detection model, a combination of two different networks, it requires high computational resources. In the current era, deepest learning-based systems run on the Graphics Processing Unit (GPU) to get enhanced performance. The GPU performs computation much faster than CPU. We have utilized Nvidia GeForce GTX 1080 Ti GPU with 11GB of memory for training our GAN-based system. Utilizing this GPU allowed us to develop and train the proposed system nearly 10 times faster than on a simple CPU. The complete detail of hardware and software resources is shown in Tab. 2.

Table 2: Hardware and software resources used in experiments

OS	Ubuntu 16.04
CPU	Intel Xeon E5640
GPU	GeForce GTX 1080 Ti 11GB RAM, 3680 CUDA cores
RAM	16 GB
Software	Tensor flow version 1.8.0 Python 3.6

4 Evaluation and Results

The proposed system (hybrid model) to reduce false positives in the weapon detection system is evaluated in this section. Both systems such as the existing and the proposed models are evaluated separately to produce significant results. In this section, the overall evaluation is discussed to improve the weapon detection using CNN and R-FCN model with max-pooling. Evaluation steps to assess the complete system are described below:

1. Evaluation of weapon detection system with metrics defined on local data.
2. Evaluation of system with max-pooling with matrices on local data.

4.1 Experimental Setting

The experimental setting provides the detail about final datasets where both systems are trained. It is also described the evaluation matrix.

4.2 Data Training

The proposed system is trained on a local dataset that contains quality images that are based on all matrices. After this, the labeled the training dataset, based on the defined matrices for the reduction of false positives. For evaluating the system, we have employed a self-generated dataset.

4.3 Evaluation of Hybrid R-FCN

Extensive experiments have been conducted for training and evaluation of our weapon detection system. The proposed system is evaluated by following the 70–30 split rule. The local dataset contains 10000 images of local weapons. The detailed graphical representation is given of the training phase of the faster R-CNN, R-FCN, and hybrid R-FCN model. Training accuracy and loss graphs are based on 50000 iteration data values which are divided into 3 chunks (two chunks of 20000 and one chunk of 10000) to show the improvement during training. The class loss and box loss are also shown in graphical form. Further, the results are described in the form of a confusion matrix separately. Performance evaluation matrices are also shown separately along with mathematical representation. Fig. 5 below shows the comparison between accuracy and loss.

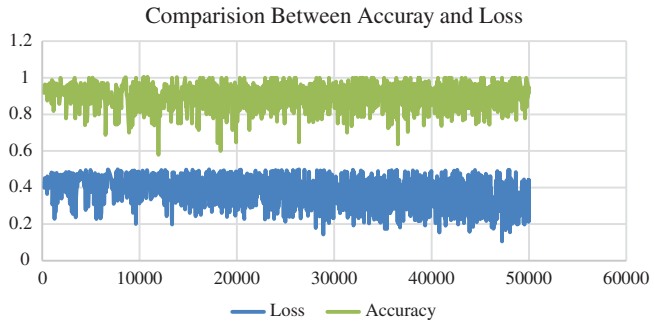

Figure 5: Comparison between accuracy and loss

4.3.1 Loss Class

The local dataset contains 10000 images of local weapons and graphical representation is given of the training phase of the faster R-CNN, R-FCN, and hybrid R-FCN model. The loss graphs are based on 50000 iteration data values which are divided into 3 chunks (two chunks of 20000 and one chunk of

10000) to show the improvement during training. The class loss is shown in graphical form and the results described in the form of a confusion matrix separately describe in Fig. 6.

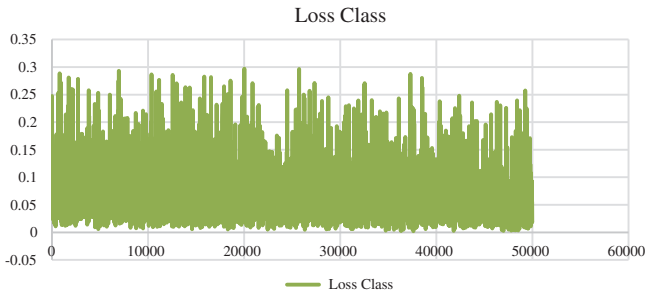

Figure 6: Loss class

The Loss class in Fig. 6 describe the 50000 iteration and totally based to show the improvement. The loss class figure represents in graphical form.

4.3.2 Loss Box

Our model's performance is very convincing because only one class is taken. The graphical representation shows a very clear picture. The Fig. 7 shows the high performance of experiments in Loss Box.

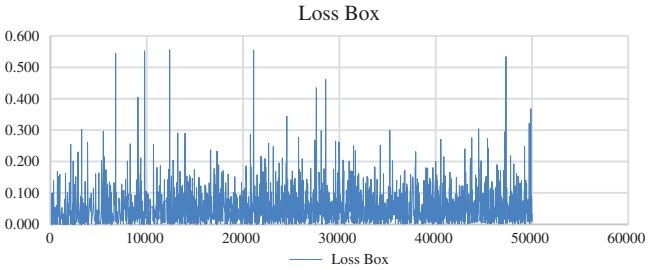

Figure 7: Loss box

4.3.3 Confusion Matrix and Four Outcomes of Classification

A confusion matrix is designed from the four conclusions formed as a result of binary classification.

A binary classifier forecasts all data occurrences of an assessment dataset, either negative or positive. This organization (or prediction) produces four results: true positive, true negative, false positive and false negative.

1. Correct positive prediction: True Positive (TP)
2. Incorrect positive prediction: False Positive (FP)
3. Correct negative prediction: True Negative (TN)
4. Incorrect negative prediction: False Negative (FN)

4.3.4 Confusion Matrix

A confusion matrix of binary classification is a two-by-two table formed by including the number of the four results of a binary classifier which are usually denoted as TP, FP, TN, and FN instead of "the number of true positives", and so on. The Tabs. 3–5 are shown faster R-CNN, R-FCN, and hybrid R-FCN confusion

matrix respectively. Tables describe the confusion matrix of both training and validation and total image 1352% and 70% mean 70% result in these images.

Table 3: Faster R-CNN confusion matrix

	Positive	Negative
Positive	TP (559)	FN (126)
Negative	FP (36)	TN (631)

Table 4: R-FCN confusion matrix

	Positive	Negative
Positive	TP (551)	FN (116)
Negative	FP (20)	TN (665)

Table 5: Hybrid R-FCN confusion matrix

	Positive	Negative
Positive	TP (594)	FN (93)
Negative	FP (25)	TN (640)

In Tab. 4 explain two by two table formed in the confusion matrix of binary classification. In the Tab. 4 describe the TP (551), FN(116), FP(20) and TN(665) respectively.

4.3.5 Performance Evaluation Matrices and Comparison in Tabular Form

There are four matrices involved in performance evaluation such as accuracy, recall, precision, and F1-score as shown in Eqs. (12)–(15)

$$\text{Accuracy} = \text{TP} + \text{TN} \,/\, \text{TP} + \text{FP} + \text{FN} + \text{TN} \tag{12}$$

$$RECALL = TP/TP + FN \tag{13}$$

$$\text{PRECISION} = \text{TP} \,/\, \text{TP} + \text{FP} \tag{14}$$

$$\text{F1 SCORE} = 2 * (\text{Recall} * \text{Precision}) \,/\, (\text{Recall} + \text{Precision}) \tag{15}$$

In Tab. 6 describe the comparison between Faster R-CNN, R-FCN and Hybrid R-FCN. The result of Hybrid R-FCN gives better result as compared to Faster R-CNN and R-FCN. The accuracy Recall rate, precision and F1-score all give better result in Hybrid R-FCN as compared to others models.

Table 6: Comparison between CNN models

Sr. no	Evaluation matrices	Faster-RCNN	R-FCN	Hybrid R-FCN
1	Accuracy	0.88	0.89	0.91
2	Recall	0.81	0.82	0.86
3	Precision	0.93	0.96	0.95
4	F1-score	0.87	0.89	0.90

5 Conclusion

In this study, a hybrid system is presented by combining deep convolutional network models by using pooling technique that is used by the other CNN base model for object detection. Most of the object detection is done with CNN-based neural networks in the modern era. The basic motivation behind this CNN 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, annotation is done by gathering local data set where one matrix is defined. For the baseline CNN or feature extractor we have used VGG-16 for faster R-CNN and ResNet for R-FCN, and hybrid model.

We trained these models and did a comparative study of results. We have 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 good results for the weapon detection system. Overall, hybrid R-FCN gives better result as compared to other models. The accuracy, recall rate, precision, and F1-score all give better results in hybrid R-FCN as compared to faster R-CNN and R-FCN.

In future work, the new century brings with it new challenges in detecting concealed weapon. As criminal justice professional work on the technology and protocol to address these challenges. we will extend our model to cover move objects more efficiently and will drive it to give our model.

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] M. K. Mohamed, A. Taha and H. H. Zayed, "Automatic gun detection approach for video surveillance," *International Journal of Socio Technology and Knowledge Development*, vol. 12, no. 1, pp. 49–66, 2020.

[2] S. Narejo, B. Pandey, D. Esenarrovargas, C. Rodriguez and M. R. Anjum, "Weapon detection using yolo V3 for smart surveillance system," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–9, 2021.

[3] 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.

[4] T. S. S. Hashmi, N. U. Haq, M. M. Fraz and M. Shahzad, "Application of deep learning for weapons detection in surveillance videos," in *IEEE Proc. of (ICoDT2)*, Islamabad, Pakistan, pp. 1–6, 2021.

[5] M. P. J. 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.

[6] Y. LeCun, Y. Bengio and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[7] K. He, X. Zhang, S. Ren and J. Sun, "Delving deep into rectifiers: surpassing human-level performance on image net classification," in *Proc. of ICCV*, Las Condes, Chile, vol. 1, no. 1, pp. 1026–1034, 2015.

[8] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. of ICCVPR*, Beijing, China, vol. 1, no. 1, pp. 1–6, 2014.

[9] M. Malik, M. A. Jaffar and M. R. Naqvi, "Comparison of brain tumor detection in MRI images using straight forward image processing techniques and deep learning techniques," in *Proc. of HORA*, Ankara, Turkey, pp. 1–6, 2021.

[10] R. Wang, R. Qin, J. Zou and L. Zhang, "AGR-Fcn: Adversarial generated region based on fully convolutional networks for single- and multiple-instance object detection," in *IEEE Int. Conf. on Imaging Systems and Techniques(ICIST)*, Abu Dhabi, United Arab Emirates, pp. 1–6, 2019.

[11] A. Warsi, M. Abdullah, M. N. Husen, M. Yahya, S. Khan *et al.*, "Gun detection system using yolov3," in *Proc. of ICSIMA*, Kaula Lumpur, Malaysia, 2019.

[12] 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.

[13] M. R. Naqvi, M. Arfan Jaffar, M. Aslam, S. K. Shahzad, M. Waseem Iqbal *et al.*, "Importance of big data in precision and personalized medicine," in *Proc. of HORA*, Ankara, Turkey, pp. 1–6, 2020.

[14] M. Nakib, R. T. Khan, M. S. Hasan and J. Uddin, "Crime scene prediction by detecting threatening objects using convolutional neural network," in *Proc. of ICCCCME*, Indore, India, vol. 1, no. 1, pp. 1–6, 2018.

[15] J. Lim, M. Istiaque, A. Jobayer, V. M. Baskaran, J. M. Lim *et al.*, "Gun detection in surveillance videos using deep neural networks," in *Proc. of APSIPA*, Lanzhou, China, vol. 1, no. 1, pp. 1–6, 2019.

[16] M. Singleton, B. Taylor, J. Taylor and Q. Liu, "Gun identification using tensor flow, machine learning and intelligent communications," *Springer*, vol. 5, no. 5, pp. 3–12, 2018.

[17] Ruben J. Franklin "Anomaly detect ion in videos for video surveillance applications using neural networks," in *Proceedings of ICISC*, India, vol. 1, no. 1, pp. 632–633, 2020.

[18] M. Mundegorski, S. Akcay, M. Devereux, A. Mouton and T. P. Breckon, "On using feature descriptors as visual words for object detection within x-ray baggage security screening," in *Proc. of ICICD*, London, United Kingdom, pp. 1–6, 2016.

[19] C. Miao, L. Xie, F. Wan, C. Su and H. Liu, "A large-scale security inspection x-ray benchmark for prohibited item discovery in overlapping images," in *Proc. of ICCVPR, Long Beach California*, USA, vol. 1, no. 1, pp. 2119–2128, 2019.

[20] C.Yin, Y. Zhu, J. Fei and X. He, "A deep learning approach for intrusion detection using recurrent neural networks, *IEEE Access*, vol. 5, pp. 21954–21961, 2017.

[21] R. Olmos, S. Tabik and F. Herrera, "Automatic handgun detection alarm in videos using deep learning," *Neuro Computing*, vol. 275, pp. 66–72, 2018.

[22] Z. Zhong, L. Sun and Q. Huo, "An anchor-free region proposal network for faster R-cNN-based text detection approaches," *International Journal on Document Analysis and Recognition*, vol. 22, no. 3, pp. 315–327, 2019.

[23] G. K. Verma, and A. Dhillon. "A handheld gun detection using faster R-cNN deep learning." in *Proc. of ICCCT*, Allahabad, India, pp. 84–88, 2017.

[24] Y. Xue and Y. Li, "A fast detection method via region-based fully convolutional neural networks for shield tunnel lining defects," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 8, pp. 638–654, 2018.

[25] S. Narejo, B. Pandey, D. E. Vargas, C. Rodririguez and M. R. Anjum,"Weapon detection using YOLO V3 for smart surveillance system," *Mathematical Problems in Engineering*, vol. 3, no. 8, pp. 1–9, 2021.