Deep Classification of Gun Carried by Moving Persons Using Proposed TUVD-CSA Dataset

Rajib Debnath and Mrinal Kanti Bhowmik

Abstract The ability to detect handheld gun or gun on the other body parts is an ordinary human skill; the same detection problem presents an exceptional challenge for computer vision. Very few works has been reported in the area of real-time detection of scene where persons carrying gun, although it has several implication in the area of security and surveillance. Using real-time gun detection to improve surveillance methods is a promising application of Convolutional Neural Networks (CNNs). In the present scope of this article, we are particularly interested in the real-time scene detection where person with gun appears. As a result, a comparison is made between the existing state-of-the art classification techniques based on CNNs architecture using our created mimic dataset **Tripura University Video Dataset for Crime Scene Analysis (TUVD-CSA)**. For an endways fine-tuning using contemporary architectures, we witness the direct proportion of performance and network complication.

Keywords Deep CNN · Gun classification · Transfer learning · Security and Surveillance

1 Introduction

In this digital world of security and surveillance, the number of Closed-Circuit Television Systems (CCTV) are installed in public and private areas such as parking places, malls, places of worships, entrance of buildings, different security zones, etc., are increasing exponentially. Therefore, it makes challenge for a human operator to

R. Debnath (✉) · M. K. Bhowmik
Department of Computer Sceince & Engineering, Tripura University A Central University, Suryamaninagar 799022, India
e-mail: mrinalkantibhowmik@tripurauniv.ac.in

R. Chaki et al. (eds.), *Advanced Computing and Systems for Security: Volume 13*,
Lecture Notes in Networks and Systems 241,
https://doi.org/10.1007/978-981-16-4287-6_8

inspect and analyze the video feed from the remote camera and take any appropriate action thereon; this repeatedly burdens an unworkable amount of observance, also can be expensive and unproductive when several video streams are present. Different studies [1–3] suggested that the human operator suffers video blindness after 20–40 min of active monitoring and misses the screen activity as high as 95%, which drastically reduce the detection accuracy up to 83%. Real-time system for automatic crime detection including armed assault and robbery becomes imperative for achieving comprehensive security system. Such an automated system is liable to raise the alarm or indication whenever any abnormal activity is encountered under CCTV surveillance, due to which the operator will prioritize his attention on the video feed and will initiate appropriate action thereon [4]. Such dangerous scenario is the Active Shooter Event such as the Colorado Theatre Shooting (USA), Oslo (Norway), and Paris (France) shows that rapid detection and identification of Armed Shooter is essential in reducing the number of casualties [5].

When any individual carries a gun or other weapon in hand, it is a strong indicator of a possibly risky situation, this is because the gun is operative by hand only while committing any crime with it. Automatic detection of person with gun is an imperative task in the field of computer vision. Video surveillance in public spaces and the **proliferation of body cameras for police** can possibly be leveraged for gun detection system. This vision-based system could generate an alarm that are able to alert surveillance human personnel and police in real time, resulting in prompter action.

Recent days, CNN-based classification and detection are mostly used in machine learning and computer vision. Very few works, reported in literature, used CNN for classification of scenes based on the presence or absence of gun. Same amount of work are available for gun localization too in scenes. As per our knowledge, olmos et al. [6] first used Faster-RCNN for localization of gun in images. They used VGG-16-based F-RCNN for localization of gun and attained 84.21% accuracy. Following this work, other literature are reported different accuracy for F-RCNN based on different classification architecture. [7, 8] shown that F-RCNN performs well with VGG-16, whereas [9–11] shown complex architectures such as $ResNet_{101}$, SqueezeNet, and MobileNet attained better accuracy than the VGG-16. Only [12] showed performance of VGG-16 in classification of input video streams based on presence of gun.

Motivated by [7, 8, 11], we conduct an extensive set of experiments to evaluate the strength of CNN-based classifiers in detection of scenes. We compare almost all the CNN base classifiers by implementing them from the scratch and fine-tuned them for the used dataset. The overall flow of the manuscript has shown in Fig. 1, it represents the different experimentation framework used in this manuscript.

The first block of the overall flow represents the mentioned classification using the SVM classifier with the traditional feature set. The feature set contains edge-based features such as SIFT, Surf, Harris corner, and HOG. We also used bag of features with SIFT, SURF, Harris corner, and HOG feature. The next experiment framework shown in the flow represents the CNN training with layer freezing. As shown in Fig 1, $\mathrm{DFS}_1, \mathrm{DFS}_2, \ldots, \mathrm{DFS}_{1-N}$ are the features extracted by increasingly fine-tuning the layers of CNN. More specifically, in the second block the last fully convolutional layer is fine-tuned and except that pre-trained weights are used for

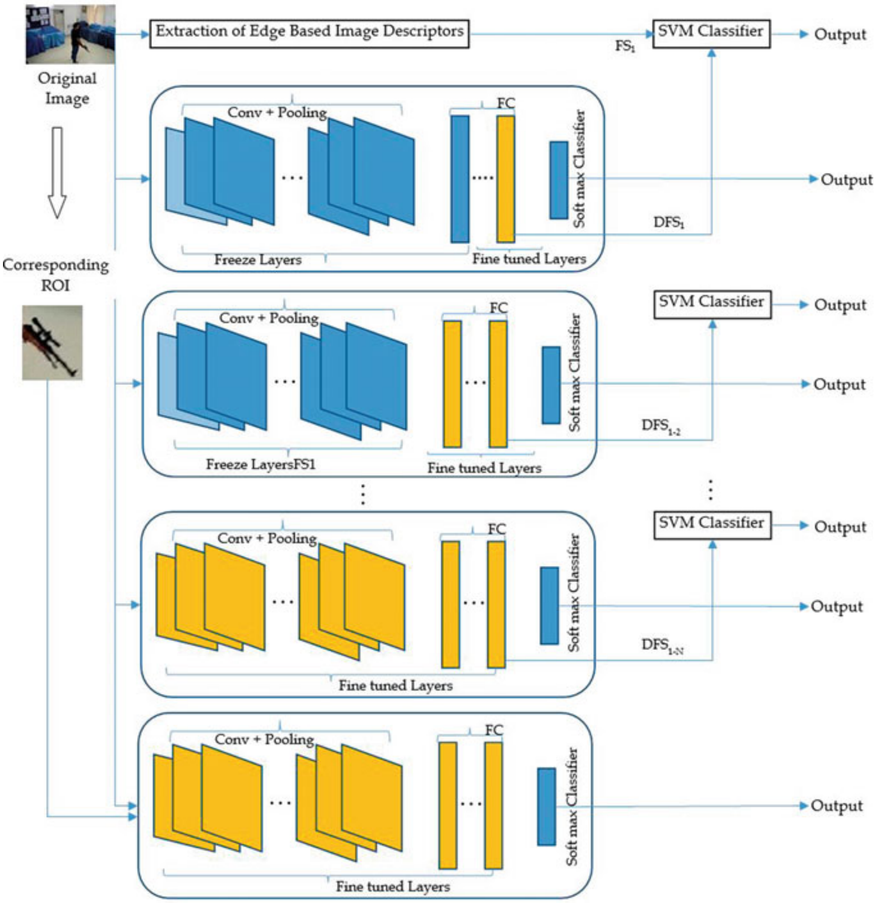

Fig. 1 Overall flowchart of the Proposed system

other layers. Likewise, in the next block the last two fully convolutional layers are fine-tuned and so on up to the second last block. In Fig 1, the yellow-colored layers are fine-tuned layers and blue-colored layers are the freezed layer. Freezed layer refers to the use of pre-trained weights. Afterward, features from each block are fed into the SVM classifier. Each block except the first block generated CNN classification result by using softmax classifier after the last fully convolutional layers. From this experimentation, we can observed how fine-tuning the layer's weights at varying points in the network influence the performance. To increase the performance of the CNN classification, we have modified the input of the CNN architecture. Instead of using only the input frame, we also fed the corresponding gun part (ROI) as an input to the CNN architecture. This contributional tweak in the input layer is able to produce more accurate results as shown in the result part. The last block of Fig. 1

represents the end-to-end training of the CNN architecture with the tweak input. The contributions of this paper are:

i. We proposed a scenario of classification, where along with the input image, ROI of that image is also used as input to train the network.
ii. The exhaustive evaluation of most important conventional classification architectures against prior works [7, 9–11].
iii. The feature space comparison of the CNN classification result with the traditional features. Contrasting performance results are obtained against the prior published studies of [7, 8] over proposed dataset of **99030** examples making this the largest scene detection with gun study in the literature.

The rest of the paper is organized as follows. Section 2 discusses the proposed dataset and description of different classes used for the experiment purpose. In Sect. 3, we discuss the Methodology, the experiments and the results are presented in Sect. 4. We conclude the paper in Sect. 5.

2 Database Description

The data is composed of two classes, the first class contains the images of the person who carrying the gun in hand and gun alone (cropped from the person with gun images) and the second class contains the images of person without having gun and other objects alike guns are bottle, mobile phones, etc. The images of each class has been obtained from our own created **TUVD-CSA**, which is available for research community in (http://www.mkbhowmik.in). The database totally contains 150 video clips (65 videos in indoor and 60 videos in outdoor condition and 25 videos are downloaded from different web sources) that contains more than 5 Lakh frames. These 25 clips are downloaded to include some real-time scenario, which will make the dataset as versatile as possible. Other 125 video clips are created and collected in parking places, building premises, corridors, garden, open fields, different crossings (three-way, four-way), lobby, laboratory, class room, etc., of Tripura University campus. Nikkon D5100 camera with 30 fps (Frame per second) was deployed in this work. Camera is positioned on a tripod stand of height 9–12 ft. from the ground level and at an angle of 60^0 to the camera tripod. We ask few students to annotate actions of a suspected person with guns in hand. Effects of illumination change, occlusion, rotation, pan, tilt, scaling of gun are effectively demonstrated in this dataset. The reason behind to take images containing of person carrying the gun in hand along with gun alone is done to provide the deep Neural Network (NN) with images that are similar to those that the network will face in its operation, where the gun appears in complex environments with multiple objects around it.

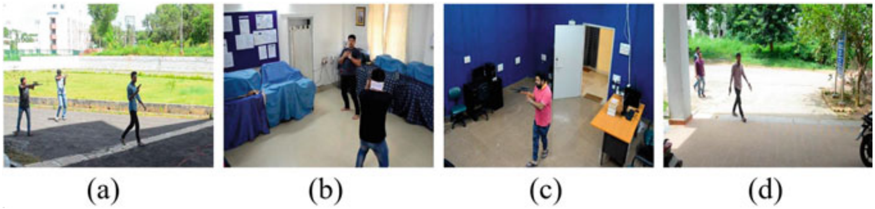

Fig. 2 Different situations **(a)** mass firing scenario; **(b)** crime scene with partial occlusion of gun; **(c)** robber is aiming the gun at waist height; **(d)** robber is not aiming

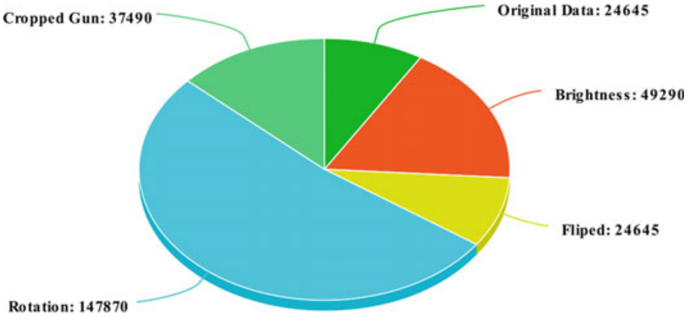

Fig. 3 Structure class A

2.1 Class A

The first class of this dataset contains 24645 images of person carrying guns collected in indoor and outdoor conditions considering different challenges like sudden illumination change, partial occlusion of guns, etc. This class also covers different situations like mass firing scenario, crime scene, images where the robber is aiming the gun at waist or shoulder height, and finally, with images where the robber is not aiming the gun. Few sample images are shown in Fig. 2.

These images were resized into 640 × 480 pixels. In order to increase the accuracy of the system data augmentation technique has also been applied to the dataset. The aim is to perform transformations that simulate realistic views of the object to be detected. The structure of the class is shown in Fig. 3;

- Increasing and decreasing the brightness by (20%) in order to simulate the luminosity change; shown in Fig. 4(a).
- Flipping and rotations ($\pm 10^0$, $\pm 20^0$, $\pm 30^0$) to create the different canonical views of the object; shown in Fig. 4(b).
- Cropping the gun region from the whole frame to simulate different shapes and colors of the gun; shown in Fig. 4(c).

With these above augmentation techniques, we increase the original database from 24,645 images to a total of 283,940 images.

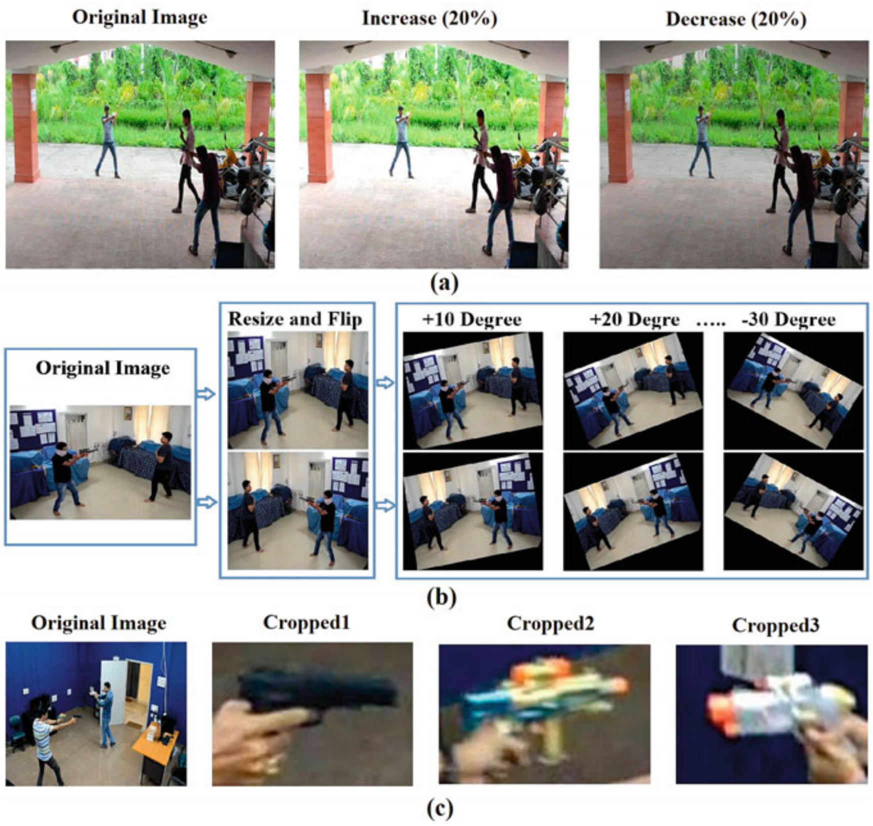

Fig. 4 Augmentation process; **(a)** Luminosity scale; **(b)** process of flipping and rotation to increase the data; **(c)** process of cropping ROI from the whole image

2.2 Class B

The second class of this dataset contains 74385 images of people without having gun. The first type of images of this class corresponds to images of people who are in different positions and places like building premises, corridors, garden, open fields, different crossings (three-way, four-way), lobby, laboratory, class room, etc. of Tripura University. The second type of images in this class are the images of person who carries gun-like objects such as mobile phones, batons, bottle, etc. Two types of augmentation techniques have also been applied for this class; the brightness change and flipping of image. The structure of this class is shown in Fig. 5.

With these above augmentation techniques, we increase the original database from 74,385 images to a total of 297,540 images.

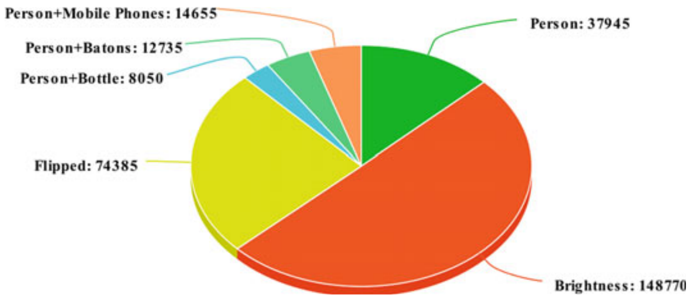

Fig. 5 Structure class B

3 Methodology

Aim of this work is to quantify efficiency of deep learning architectures for classifying video frames into gun or non-gun. According to a brief survey of existing literature. We define two approaches for classification of gun and non-gun classes.

- Holistic-based classification.
- ROI- and Holistic-based classification.

Holistic feature is mostly used for implementation of deep learning-based classification irrespective to its applications. Holistic feature referred to the raw video frame as an input to the deep architectures. Classification of the incoming scenes from surveillance camera can be obtained only by inputting the raw image. From the entire scene, local small features are extracted to generate low-level features for classification. Relative shape of gun in the incoming scene becomes challenge in this case. There are cases, where guns are captured in very small shape in the scene. In addition, guns are of different color and different shape; color and shape are primary features extracted by the deep architecture.

To resolve this problem, we are proposing to use ROI along with the raw image to train the network. This approach is referred here as ROI and Holistic feature-based classification, considering gun is the ROI of the incoming scene. During training, ROI, gun parts are outlined manually for supplying the features of Gun part, whereas, during testing, a grid-based method has employed to find the gun part from the input image. Features of each grid are compared with the features of trained ROIs to find the ROIs for test images. The holistic (raw image) information and the fine-level information of the ROI (gun) both are used for learning. The proposed system is very efficient with comparison to the previously discussed methods. We implement a number of CNN architectures for both the scenarios. As per our knowledge, few or none works are reported till now which had used ROI features along with the holistic features. Experiment results are also confirm the improvement of CNN performance for the second scenarios (Sect. 4).

We also analyze the performance of pre-trained architectures in classification of scenes in non-gun or gun classes. Freezing of layers is employed to show that fine-tuning of end-to-end architectures performs well and pre-trained models are unable to obtained good accuracy. Freezing means instead of modifying the weights retained the previous weights for classification. Layer freezing showed which features (i.e., low-level features or high-level features) are more efficient in detection of gun. In addition to this, we also conduct a comparison between the tradition features and CNN extracted features. SVM (Support Vector Machine) is used for fair comparison.

4 Evaluation

In order to evaluate the performance of the mentioned deep learning models in gun detection, we conducted a series of tests by varying configurations. We used most of the base deep architectures, AlexNet [13], VGG-16 [14], VGG-19 [14], ResNet$_{34}$ [15], ResNet$_{50}$ [15], ResNet$_{101}$ [15], ResNet$_{152}$ [15], Inception-V1 [16], Inception-V2 [17], Inception-V3 [17], Inception-V4 [18], and XceptionNet [19].

As mentioned earlier, we trained the corresponding deep architectures in three different scenarios:

1. **Input video frames only**. This is the standard application of deep architecture in classifying input frames into two category: with gun and without gun. The goal of this scenario is to evaluate the more raw performance of deep learning classification architecture for gun detection.
2. Raw frame with corresponding gun part. In this scenario, we evaluate the detection performance of our architectural tweak (see Sect. 3).

We considered the same training parameters for all CNN architectures: 20,000 gradient descent iterations, using the Adam optimizer. The loss of each model had stabilized and were showing diminishing improvements with the increasing number of iterations. We used decay of learning parameters for the deep architectures. Initial learning rate of 10^{-4}, a decay of 10^{-4} per iteration is employed. All the images are resized to 640 × 480 for both training and testing. A series of augmentation of images is also carried out described earlier (Sect. 2).

At each iteration, the current model is applied to the validation set, having its performance recorded in terms of accuracy. When the difference between training accuracy and validation accuracy decreases and stabilized is kept as final model. The performance is evaluated by the comparison of True Positive Rate (TPR), False Positive Rate (FPR), Precision (P), Accuracy (A), and F1-score (F1). Results for gun and non-gun class problems for the said two different scenarios are given in Tables 1 and 2. Table 1 shows the results for all the architectures used in this work for both Holistic features, whereas Table 2 showing the performance of CNN architectures with the ROI (the gun part) along with the holistic features. Both the tables also indicate the affect of augmentation on the model performance. Analysis of the result shown in Tables 1 and 2 highlight that model performance has increased abruptly in second scenario (Holistic+ROI-based features) and also dependent on the augmentation. We can see that without augmentation results are not acceptable. Rather

Table 1 Statistical evaluation of varying CNN architectures on TUVD-CSA dataset with **Holistic features**

Classification with holistic feature												
	Methods	TP%	FP%	P	A	F1		Methods	TP%	FP%	P	A	F1
Without augmentation	Alexnet	70.41	78.30	0.44	0.60	0.58	With augmentation	Alexnet	81.08	63.00	0.56	0.76	0.71
	VGG16	74.61	55.00	0.58	0.69	0.63		VGG16	82.59	49.20	0.62	0.77	0.71
	VGG19	76.59	50.30	0.60	0.71	0.68		VGG19	82.35	30.66	0.82	0.77	0.82
	ResNet_{34}	75.81	42.91	0.77	0.71	0.76		ResNet_{34}	84.03	27.38	0.84	0.79	0.84
	ResNet_{50}	79.34	39.12	0.78	0.72	0.78		ResNet_{50}	82.71	24.36	0.87	0.80	0.84
	ResNet_{101}	81.00	31.11	0.80	0.76	0.80		ResNet_{50}	82.71	24.36	0.87	0.80	0.84
	ResNet_{152}	78.55	35.38	0.79	0.79	0.79		ResNet_{152}	92.24	29.59	0.81	0.83	0.86
	InceptionV1	81.38	33.66	0.79	0.79	0.80		InceptionV1	85.16	28.66	0.84	0.80	0.84
	InceptionV2	79.00	31.95	0.80	0.79	0.80		InceptionV2	85.82	26.86	0.85	0.81	0.85
	InceptionV3	82.00	23.00	0.86	0.80	0.84		InceptionV3	86.80	27.63	0.85	0.81	0.86
	InceptionV4	85.52	40.20	0.68	0.83	0.79		InceptionV4	89.01	40.00	0.68	0.85	0.81
	Xception	84.09	44.40	0.65	0.81	0.78		Xception	89.71	28.26	0.85	0.83	0.87

Table 2 Statistical evaluation of varying CNN architectures on TUVD-CSA dataset for **Holistic Features + Features from ROI**

Classification with Holistic Feature + Features from ROI													
	Methods	TP%	FP%	P	A	F1		Methods	TP%	FP%	P	A	F1
Without augmentation	Alexnet	74.44	67.70	0.52	0.65	0.61	With augmentation	Alexnet	86.65	39.90	0.69	0.80	0.77
	VGG16	79.92	47.10	0.62	0.70	0.67		VGG16	87.72	28.33	0.84	0.82	0.86
	VGG19	81.01	43.30	0.65	0.73	0.70		VGG19	87.06	27.37	0.84	0.82	0.85
	ResNet$_{34}$	82.59	49.20	0.62	0.77	0.71		ResNet$_{34}$	85.08	4.30	0.66	0.83	0.79
	ResNet$_{50}$	81.20	30.72	0.80	0.82	0.81		ResNet$_{50}$	87.98	35.10	0.71	0.84	0.70
	ResNet$_{101}$	86.28	21.52	0.87	0.83	0.86		ResNet$_{101}$	88.09	28.30	0.84	0.82	0.86
	ResNet$_{152}$	88.46	22.60	0.86	0.84	0.87		ResNet$_{152}$	91.99	38.50	0.70	0.89	0.86
	InceptionV1	91.36	23.10	0.85	0.85	0.88		InceptionV1	87.87	23.21	0.87	0.84	0.87
	InceptionV2	91.58	23.26	0.85	0.86	0.88		InceptionV2	90.14	23.08	0.88	0.85	0.89
	InceptionV3	92.15	20.81	0.86	0.86	0.89		InceptionV3	91.63	23.08	0.89	0.86	0.90
	InceptionV4	94.77	19.93	0.87	0.89	0.91		InceptionV4	93.21	37.80	0.71	0.91	0.86
	Xception	94.68	21.10	0.86	0.88	0.90		Xception	93.01	39.90	0.70	0.90	0.85

than this, Tables 1 and 2 clearly show that the performances of CNN architectures is directly proportional to the network complexity.

Performance of CNN architectures are always questioned for detection of small objects in input images. In the classification of images with gun or without gun, CNN architectures has to deal with this challenge. Due to this challenge, gun-based image classification become different from the online computer vision problems. From the study of the deep architectures, we can conclude that AlexNet, VGG-16, VGG-19, and Inception-V1 networks are exploit spatial relationship of the images.

From Tables 1 and 2, it is evident that these performances of the architectures are not convincing. By analysing the results, we observed that due to the variation of color and size of guns these architectures are failed to classify maximum number of images based on guns. In the images with small-sized gun, partially occluded gun these architecture cannot identify the presence of the gun. Compared to these architectures the depth based architectures, such as Inception V2, Inception V3, and ResNet performed well in the concerned classification problem. Depth-based architectures are based on more enriched feature hierarchies. Hence, these architectures are able to capture more detailed features regarding the guns. After the success of depth-based architectures, width-based architectures are also proposed for better learning of the image features. Width-based architecture, such as inception V4 and Xception attain fair accuracy in classification of images based on guns. Among these, two inception-V4 are width-based as well as depth-based architecture. Therefore, Inception-V4 captured more fine and detailed image features and attain best accuracy compared to other architectures.

Furthermore, Table 3 shows that the number of iterations also played an important role in the classification based on CNN. Most of the architecture acquired best results with the increase in the number of iterations. Table 3 highlights this fact quantitatively, where we can observe that with the increase in the iteration, training accuracy and validation accuracy also increases, whereas losses are decreasing. In addition to this, Table 2 also shows the difference between the training accuracy and validation accuracy. The difference identifies over-fitting and under-fitting of the model. Table 2 represents that implemented models are properly fine-tuned for the aforementioned dataset.

For further analysis of the performance of CNN in detection of scenes, we use pre-trained AlexNet model and the model is pre-trained on ImageNet dataset. The pre-trained model is fine-tuned layer by layer from the last layer to first layer. Table 4 shows the results for fine-tuning of CNN architectures layer by layer. Furthermore, we start increasing the number of fine-tuned layers. With the increase in the number of fine-tuned layers, increase in the performance can be noticed. It further explained the fact that, when we fine-tuned lower layers along with the higher layers, CNN performs well.

As mentioned earlier, we are training SVM [20] on the CNN features with layer freezing. During this experiment, we start fine-tuning of one layer at first and noted the performance. Afterward, we start tuning more layers after one another and noted accuracies. Pre-trained CNN architectures are used for this experiment purpose. The noticeable fact is that SVM yields relative better performance with CNN features

Table 3 Results obtained with the networks using the TUVD-CSA Dataset for the ROI + Holistic Feature

Methods	Steps	Training		Validation		Methods	Steps	Training		Validation	
		Loss	Acc (%)	Loss	Acc (%)			Loss	Acc (%)	Loss	Acc (%)
Alexnet	12, 000	0.34	79	0.33	80	ResNet_{152}	12, 000	0.24	87	0.25	83
	15, 000	0.29	78	0.34	80		15, 000	0.22	86	0.23	88
	18, 000	0.21	81	0.32	80		18, 000	0.21	89	0.19	89
VGG16	12, 000	0.29	80	0.27	77	Inception V1	12, 000	0.18	89	0.19	84
	15, 000	0.27	82	0.29	81		15, 000	0.20	88	0.18	81
	18, 000	0.28	85	0.26	82		18, 000	0.16	92	0.17	84
VGG19	12, 000	0.26	83	0.27	79	Inception V2	12, 000	0.18	89	0.19	84
	15, 000	0.24	84	0.26	82		15, 000	0.20	88	0.18	85
	18, 000	0.23	86	0.24	82		18, 000	0.16	92	0.17	85
ResNet_{34}	12, 000	0.24	87	0.25	79	Inception V3	12, 000	0.18	89	0.19	85
	15, 000	0.22	86	0.23	81		15, 000	0.20	88	0.18	84
	18, 000	0.21	89	0.19	83		18, 000	0.16	92	0.17	86
ResNet_{50}	12, 000	0.24	87	0.25	80	Inception V4	12, 000	0.18	89	0.19	84
	15, 000	0.22	86	0.23	82		15, 000	0.20	88	0.18	88
	18, 000	0.21	89	0.19	84		18, 000	0.16	92	0.17	91
ResNet_{101}	12, 000	0.24	87	0.25	86	Xception	12, 000	0.21	88	0.22	89
	15, 000	0.22	86	0.23	84		15, 000	0.19	86	0.19	90
	18, 000	0.21	89	0.19	88		18, 000	0.20	89	0.18	90

Table 4 Results of CNN on TUVD-CSA dataset for firearm detection. AlexNet$_{a-b}$ denotes that the network is fine-tuned from layer a to layer b

		P	R	A	F1			P	R	A	F1
CNN with layer freezing	AlexNet$_{1-8}$	0.665	0.653	0.837	0.79	CNN with layer freezing + SVM	AlexNet$_{1-8}$	0.651	0.644	0.822	0.77
	AlexNet$_{2-8}$	0.630	0.612	0.811	0.77		AlexNet$_{2-8}$	0.648	0.641	0.810	0.76
	AlexNet$_{3-8}$	0.613	0.622	0.804	0.77		AlexNet$_{3-8}$	0.644	0.640	0.810	0.76
	AlexNet$_{4-8}$	0.562	0.572	0.776	0.73		AlexNet$_{4-8}$	0.641	0.640	0.810	0.75
	AlexNet$_{5-8}$	0.617	0.611	0.821	0.79		AlexNet$_{5-8}$	0.552	0.529	0.760	0.71
	AlexNet$_{6-8}$	0.544	0.515	0.753	0.70		AlexNet$_{6-8}$	0.498	0.500	0.700	0.73
	AlexNet$_{7-8}$	0.498	0.510	0.712	0.66		AlexNet$_{7-8}$	0.488	0.491	0.690	0.72
	AlexNet$_{8}$	0.454	0.421	0.689	0.62		AlexNet$_{8}$	0.457	0.467	0.691	0.68

Table 5 Results of traditional features on TUVD-CSA dataset for firearm detection

Traditional features + SVM					
Descriptors →	SURF [4]	SIFT + SURF	Harries corner [4]	HOG [21]	BoWss [22]
P	0.544	0.563	0.474	0.518	0.608
R	0.513	0.581	0.458	0.550	0.635
A	0.776	0.796	0.769	0.775	0.834
F	0.727	0.800	0.771	0.756	0.840

than with the traditional features. In addition, a positive increase in the performance of the SVM is observed due to the fine-tuning of more layers. Therefore, we can conclude that SVM trained on the fully fine-tuned CNN able to attain the highest performance on all the metrics.

Afterward, Table 5 shows the performances of traditional features with SVM in the classification of input frames based on the presence of gun. From the quantitative comparison of the values, it is apparent that CNN features are more effective than the traditional features. BoWss in Table 5 represent the bag of features with SIFT, SURF, Harris corner feature, and HOG feature. SVM is used to make the comparison more consistent.

5 Conclusion

In this work, we exhaustively explore the use of CNN in the tasks of classification of incoming scenes. All the experiments are carried on our proposed dataset, namely, **TUVD-CSA**. The dataset aims to provide the research community with a facility for testing and ranking of existing and new algorithms. The proposed dataset is an only extensive video dataset with the indoor and outdoor scenarios available in this field so far our knowledge goes. We proposed the use of Holistic features along with the ROI features to improve the performance of CNN architecture. We also experiment with pre-trained CNN models. By freezing layer by layer, we also evaluate performance of each layer of CNN models. We also compare these features with the traditional features. Results shown that fine-tuning of end to end model architecture obtained the highest accuracy. Most importantly, we present a scope of classification of image based on guns by highlighting the challenges related to this application. We also showed that depth- and width-based CNN architecture attained the highest accuracy in classification of image based on guns. But the accuracy still can be improved and we will concentrate on it in future works.

References

1. Glowacz A, Kmiec M, Dziech A (2015) Visual detection of knives in security applications using active appearance models. Multimedia Tools Appl 74(12):4253–4267
2. Velastin SA, Boghossian BA, Vicencio-Silva MA (2006) A motion-based image processing system for detecting potentially dangerous situations in underground railway stations. Transp Res Part C: Emerg Technol 14(2):96–113
3. Ainsworth T (2002) Buyer beware. In Security Oz 19:18–26
4. Tiwari RK, Verma GK (2015) A computer vision based framework for visual gun detection using harris interest point detector. Procedia Comput Sci 54:703–712
5. Gelana F, Yadav A (2019) Firearm detection from surveillance cameras using image processing and machine learning techniques. In: Smart innovations in communication and computational sciences, pp 25–34
6. Olmos R, Tabik S, Herrera F (2018) Automatic handgun detection alarm in videos using deep learning. Neurocomputing 275:66–72
7. Olmos R, Tabik S, Lamas A, Pérez-Hernández F, Herrera F (2019) A binocular image fusion approach for minimizing false positives in handgun detection with deep learning. Inform Fusion 49:271–280
8. Verma GK, Dhillon A (2017) A handheld gun detection using faster r-cnn deep learning. In: Proceedings of the 7th international conference on computer and communication technology, vol 275, pp 84–88
9. Fernandez-Carrobles MM, Deniz O, Maroto F (2019) Gun and knife detection based on faster r-cnn for video surveillance. In: Iberian conference on pattern recognition and image analysis, pp 441–452
10. Elmir Y, Laouar SA, Hamdaoui L (2019) Deep learning for automatic detection of handguns in video sequences. In: JERI
11. Castillo A, Tabik S, Pérez F, Olmos R, Herrera F (2019) Brightness guided preprocessing for automatic cold steel weapon detection in surveillance videos with deep learning. Neurocomputing 330:151–161
12. Romero D, Salamea C (2019) Design and proposal of a database for firearms detection. In: The international conference on advances in emerging trends and technologies, pp 348–360
13. Krizhevsky A, Ilya S, Geoffrey EH (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inform Process Syst
14. Simonyan K, Andrew Z (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556
15. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. CoRR arXiv:1512.03385
16. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–8
17. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2818–2826
18. Szegedya C, Ioffe S, Vanhoucke V, Alemi AA (2017) Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-first AAAI conference on artificial intelligence
19. Chollet F (2017) Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1251–1258
20. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297
21. Asnani S, Ahmed A, Manjotho AA (2014) Bank security system based on firearm detection using hog features. Asian J Eng Sci Technol 4(1)
22. Halima HB, Hosam O (2016) Bag of words based surveillance system using support vector machines. Int J Secur Appl 10(4):331–346