International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362
IJARST

WEAPON DETECTION FROM CROWDED ENVIRONMENTS USING DEEP LEARNING

[1] T. Sudha Rani, [2] Yelijala Ruchitha, [3] Aenugu Sruthi Reddy, [4] Yerukala Varsha
[1]Assistant professor in Department of Information Technology, Bhoj Reddy Engineering College for Women
[2,3,4,]UG Scholars in Department of Information Technology, Bhoj Reddy Engineering College for Women
[2] ruchithagoud1215@gmail.com , [3] sruthiredddy980@gmail.com ,[4] varsha.yerukala@gmail.com

Abstract

Security and safety is a big concern for today's modern world. For a country to be economically strong, it must ensure a safe and secure environment for investors and tourists. Having said that, Closed Circuit Television (CCTV) cameras are being used for surveillance and to monitor activities i.e. robberies but these cameras still require human supervision and intervention. We need a system that can automatically detect these illegal activities. Despite state-of-the-art deep learning algorithms, fast processing hardware, and advanced CCTV cameras, weapon detection in real-time is still a serious challenge. Observing angle differences, occlusions by the carrier of the firearm and persons around it further enhances the difficulty of the challenge. This work focuses on providing a secure place using CCTV footage as a source to detect harmful weapons by applying the state of the art open-source deep learning algorithms. This paper presents a system for gun and knife detection based on the Faster R-CNN methodology. Two approaches have been compared taking as CNN base a GoogleNet and a SqueezeNet architecture respectively. The best result for gun detection was obtained using a SqueezeNet architecture achieving a 85.45% AP50. For knife detection, the GoogleNet approach achieved a 98.68% AP50. Both results improve upon previous literature results evidencing the effectiveness of our detectors. Two approaches are used i.e. sliding window/classification and region proposal/object detection. Some of the algorithms used are VGG16, Inception-V3, Inception-ResnetV2, SSDMobileNetV1, Faster-RCNN Inception-ResnetV2 (FRIRv2), YOLOv3, and YOLOv4. Precision and recall count the most rather than accuracy when object detection is performed so these entire algorithms were tested in terms of them. Yolov4 stands out best amongst all other algorithms and gave a F1-score of 91% along with a mean average precision of 91.73% higher than previously achieved.

I INTRODUCTION

The use of weapons in public places has become a major problem in our society. These situations are more frequent in countries where weapons are legally purchased or their use is not controlled [10]. Crowded places are specially vulnerable. Unfortunately, mass shootings have become one of the most dramatic problems we face nowadays [20]. Video surveillance systems, typically based on classic closed circuit television (CCTV) are especially useful for intruder detection and remote alarm verification [6]. However, these systems need to be continuously supervised by a human operator. In this respect, it is estimated that the concentration of a security guard watching a camera panel decreases catastrophically after 20 minutes. Security can be increased applying artificial vision algorithms on images obtained from video surveillance systems. Another advantage of these algorithms is the possibility of monitoring larger spaces using fewer devices thus requiring less dependence on the human factor. Machine learning techniques have been widely used in the field of video surveillance. The prevalent paradigm of deep learning has but increased the potential of machine learning in automatic video surveillance. The objective of this work is the development of two novel weapon detectors, for guns and knives, applying deep learning techniques and assess their performance.

II LITERATURE SURVEY

The applications of the deep learning paradigm for weapon detection are still rather limited. The seminal work of Olmost et al. [14] presented an automatic handgun detection system for video surveillance. This system was based on a Faster R-CNN with a VGG16 architecture trained using their own gun database. Results provided zero false positives, 100% recall and a precision (IoU=0.5) value of 84,21%. In Valldor et al. [17] a firearm detector for application to social media was presented. The detector employed a Faster R-CNN and an Inception v2 network for feature extraction. A public database of images containing several firearms was manually labelled and used for training. Benchmarking was performed on the COCO dataset obtaining a ROC curve that showed usable results. Verma et al. [18] used the Internet Movie Firearm Database (IMFDB) to generate a handheld gun detector. For that purpose, a Faster R-CNN based on a VGG16 architecture was applied only for feature extraction. Classification was performed using three different classifiers: a Support Vector Machine (SVM), a K-Nearest Neighbor (KNN) and a Ensemble Tree classifier. The best result achieved was 93.1% accuracy, using a Boosted Tree classifier. We have to note that the IMFDB dataset contains mostly profile images of pistols and revolvers at high resolution with homogeneous background, which is not a realistic situation. The work of Akcay et al. [5] presented a detection and classification system for X-ray baggage security imagery. The work explored the applicability of multiple detection approaches based on sliding window CNN, Faster R-CNN, Regionbased Fully Convolutional Networks and YOLO. Their system was composed by images divided into six classes: camera, laptop, gun, gun component, knife and ceramic knife. The best results for firearm detection were achieved with a YOLO architecture obtaining a 97.4% AP50. For knife cases, the best results were obtained using a Faster R-CNN based on a ResNet-101 architecture with a 73.2% AP50. Finally, in Kanehisa et al. [11] the YOLO algorithm was applied to create a firearm detection system. The firearm dataset used for this study was extracted from the IMFDB website. Detection results obtained a 95.73% of sensitivity, 97.30% of specificity, 96.26% of accuracy and 70% of mAP50. Regarding knife detection, the most relevant results have been obtained in the context of the COCO (Common Objects in Context) Challenges. COCO is a large-scale object detection dataset focused on detecting objects in context [13]. Each year COCO launches a challenge based on any of the following artificial vision tasks: detection, segmentation, keypoints or scene recognition. The last object detection challenge using bounding boxes was released in 2017 where the best result for knife detection was obtained by the Intel Lab team. Employing a Faster R-CNN and a HyperNet architecture this team achieved 36.6% AP50. In Yuenyong et al. [19] knife detection was explored using a dataset of 8,527 infrared (IR) images. A GoogleNet architecture was applied to classify IR images as person or person carrying hidden knife. The classification accuracy reported was 97.91%.

In summary, the Faster R-CNN seems to be the prevalent deep architecture for gun and knife detection. This work also focuses on that architecture.

III EXISTING SYSTEM

Different approaches then used for weapon detection using sliding window and region proposal algorithms. HOG (Histogram of oriented Gradient) models were used to predict the objects in the frame. HOG significant work used low-level features, discriminative learning, and pictorial structure along with SVM [35-37]. These algorithms were slow for real-time scenarios with 14s per image. Although these classifiers gave good accuracies, the slowness of the sliding window method was a big problem, especially for the real-time implementation purpose.

This work focuses on the state of the art deep learning network rather SIFT and HOG features which use handcrafted rules for feature extraction, selection, and detection in real-time visual scenario using CCTV cameras. X. Zhang et al. concluded an important finding that helped my work. They concluded that the automatic feature representation gave improved results rather than manual features [38]. Not only the learned features were better in performance, they also had learned the deep representation of the data and reduced a lot of manual work, and saved time and energy.

Disadvantages of Existing System:

Weapon detection in real-time is a very challenging task. As our desired object has a small size so, detecting it in an image is also very challenging in presence of other objects, especially those objects that can be confused with it. Deep learning models faced several below mentioned challenges for detection and classification task: The first and main problem is the data through which

- CNN learn its features to be used later for classification and detection. No standard dataset was available for weapons.

For real-time scenarios, making a novel dataset

- manually was a very long and time-consuming process. Labeling the desired database is not an easy task, as

- all data needs to be labeled manually. Different detection algorithms were used, so a labeled

- dataset for one algorithm cannot be utilized for the other one. Every algorithm requires different labeling and preprocessing operations for the same-labeled database.

- As for real-time implementation, detection systems

- require the exact location of the weapon so gun blocking or occlusion is also a problem that arises frequently and it could occur because of self, inter object, or background blocking.

IV PROPOSED SYSTEM

This article presents an automatic detection and classification method of weapons for real-time scenario using state of the art deep learning models. For real-time implementation relating the problem question of this work “detecting weapons in real-time for potential robbers/terrorist using deep learning”, detection and classification was done for pistol or knife, revolver and other shot handheld weapons as in binary class called pistol or knife and related confusion objects such as cell phone, metal detector, wallet, selfie stick in not pistol or knife class. A major reason behind this was our research done on weapons used in robbery cases and it further motivated us to choose pistol or knife and revolver as our target object.

Different approaches are used in this work for weapon classification and detection purpose but all have deep learning and CNN architecture behind them because of their state of the art performance. Training from scratch took very much time so the Transfer learning approach was used and ImageNet and COCO (common objects in context) pre-trained models are used. Different datasets were made for classification and detection. For real-time purposes, we made our dataset by taking weapon photos from the camera, data was extracted manually from robbery CCTV videos, downloaded from imfdb (internet movie firearm database), data by university of Granada and other online repositories. All the work has been done to achieve results in real-time.

The main contributions of this work are: presentation of a first detailed and comprehensive work on weapon detection that can achieve detection in videos from real-time CCTV and works well even in low resolution and brightness because most of the work done earlier is on high definition training images but realtime scenario needs realtime training data as well for better results, finding of the most suitable and appropriate CNN based object detector for the application of weapon detection in real-time CCTV video streams, making of a new dataset because real-time detection also needs real-time training data so we made a new database of 8327 images and preprocessed it using different OpenCV filters i.e. Equalized, Grayscale and clahe that helped in detecting images in low brightness and resolution, introducing the concept of related confusion classes to reduce false positives and negatives, training and testing of our novel database on the latest state of the deep learning based classification and detection models.

To achieve high precision, increase number of frame per seconds and improve localization, we moved to the object detection and region proposal methods. The different state of the art deep learning models for object detection were used and the results were compared in terms of precision, speed, and standard metric of F1 score. State of the art deep learning based SSDMobileNetv1 [52-54], YOLOv3 [55], FasterRCNN-InceptionResnetv2 [56-58], and YOLOv4 [59] were trained and tested.

V IMPLEMETATION

Image Acquisition

The first step of the Dangerous Object Classification is image acquisition. High-quality Dangerous Objects dataset used which are obtained from open Github repository. The entire sample set is divided into three parts: training samples and validation samples in the training phase and testing samples in the testing phase. Moreover, the sample set is divided into positive and negative samples—a positive sample is an image showing patient behaviors, whereas a negative sample is a background image.

Annotated Dataset Collection

A Knowledge-based dataset is created by proper labeling of the collected images with unique classes.

Image Processing

The obtained images that will be engaged in a preprocessing step are further enhanced specifically for image features during processing. The segmentation process divides the images into several segments and utilized in the extraction of Dangerous Objects from dataset.

Feature-Extraction

This section involves the convolutionary layers that obtain image features from the resize images and is also joined after each convolution with the ReLU. Max and average pooling of the feature extraction decreases the size. Ultimately, both the convolutional and the pooling layers act as purifiers to generate those image characteristics.

Classification

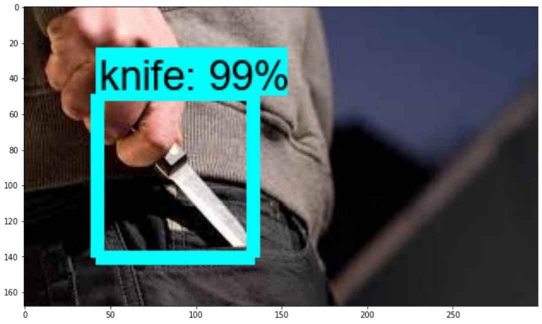
The final step is to classify images, to train deep learning models along with the labeled images to be trained on how to recognize and classify images according to learned visual patterns. The authors used an open-source implementation via the TensorFlow module, using Python and OpenCV including the Faster R-CNN model.

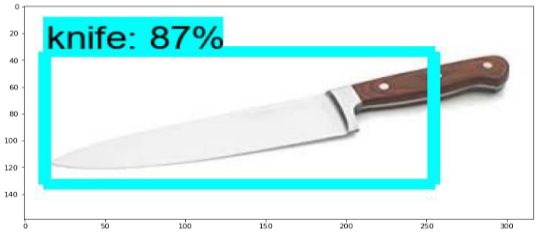
Deployment

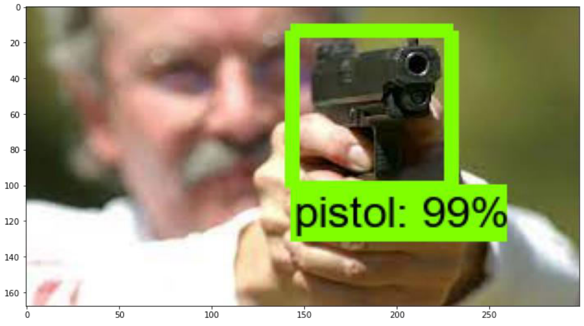
The last step of machine learning life cycle is deployment, where we deploy the model in the real-world system.

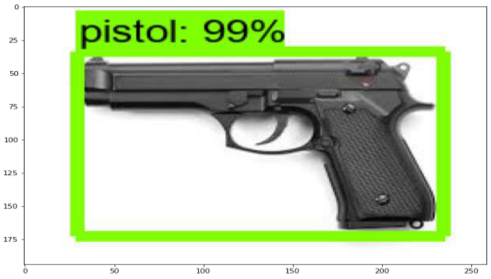
If the above-prepared model is producing an accurate result as per our requirement with acceptable speed, then we deploy the model in the real system. But before deploying the project, we will check whether it is improving ts performance using available data or not. The deployment phase is similar to making the final report for a project.

VI RESULTS ANALYSIS:









In a detection task there are two possible results, positive and negative. Some positive cases can be classified as negative and vice versa. These cases are called false positives (type I error) and false negatives (type II error), respectively. Thus, the following four cases are considered: True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). However, in an object detection task object localization must be considered too. The accuracy achieved in an object detector is commonly evaluated using the mean average precision (mAP). This measurement is defined as the average of the maximum precisions at different recall values. Therefore, the three main concepts considered within this measurement are: precision, recall and Intersection over union (IoU).

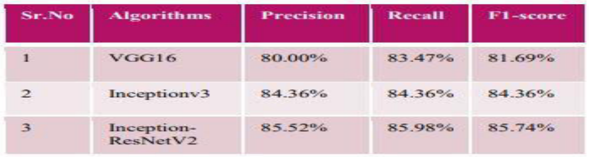
Precision measures the likelihood of a positive case being classified as such. This value is estimated using the amount of real positive cases which were classified as positive. Then, it is the percentage of correct positive predictions.

– Recall (or sensitivity) measures the likelihood of classifying the object as positive. In other words, it measures how good is the network finding positives cases.

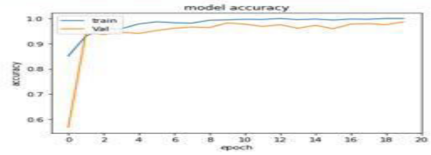
The Intersection over Union (IoU) quantifies the overlapping between 2 regions. This measures how valuable the prediction is with respect to the ground truth (the real object boundary). A prediction is usually considered to be correct when the IoU is equal or greater than 0.5.

– The Precision-Recall Curve summarizes the trade-off between precision and recall values using different probability thresholds. The area under this curve is known as average precision (AP), a value between 0 and 1 which evaluates the quality of the model. When there is more than one object to be detected, the average precision is calculated for each object resulting in the mean average precision (mAP).

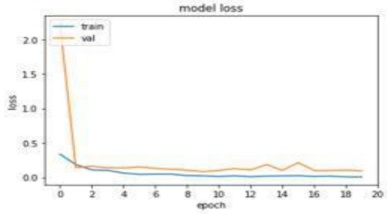
For the problem of gun detection, Faster R-CNN trained using GoogleNet obtained a 55.45% of AP50 (AP at IoU=0.50). Faster R-CNN using a SqueezeNet obtained 85.44% of AP50, a significant difference over GoogleNet. The precisionrecall curve acquired for SqueezeNet is shown in Figure 2. This detector achieved good results, improving upon previous results described in the literature



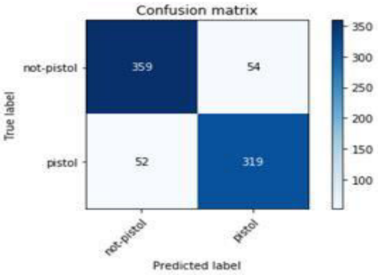
Sr.No	Algorithms	Precision	Recall	F1-score
1	VGG16	80.00%	83.47%	81.69%
2	Inceptionv3	84.36%	84.36%	84.36%
3	Inception-ResNetV2	85.52%	85.98%	85.74%



Model Training Accuracy vs Validation Accuracy



Model Training Loss vs Validation Accuracy



Confusion Matrix

VII CONCLUSION

Public and crowded areas are still the target of many violent acts. Video surveillance can be helped by automatic image analysis using artificial vision. This paper describes the implementation of several weapon detectors for video surveillance based on Faster R-CNN methodologies. For training, gun and knife images from the work of Olmos et al. and COCO dataset have been used. Several transformations such as rotations, scaling or brightness were applied in order to augment the datasets. Detectors were developed using the GoogleNet and SqueezNet architectures as CNN base on a Faster R-CNN. The best result for gun detection was obtained using a SqueezeNet architecture achieving a 85.45% AP50. For knife detection, GoogleNet approach accomplished 98% accurancy. Both detector results improve upon previous literature studies evidencing the effectiveness of our detectors.

VIII FUTURE WORK

The future work includes reducing the false positives and negatives even more as there is still a need for improvement. We might also try to increase the number of classes or objects in the future but the priority is to further improve precision and recall.

REFERENCES

1. COCO dataset 2017. http://cocodataset.org, accessed: 2019-05-04

2. COCO detection leaderboard. http://cocodataset.org/ #detection-leaderboard, accessed: 2019-05-04

3. Open images dataset v4. https://storage.googleapis.com/openimages/ web/ index.html, accessed: 2019-05-04

4. Weapon detection by neural network. https://github.com/Shubham02gupta/ Weapon-Detection-by-Neural-network/tree/master/train, accessed: 2019-05- 04

5. Akcay, S., Kundegorski, M.E., Willcocks, C.G., Breckon, T.P.: Using deep convolutional neural network architectures for object classification and detection within x-ray baggage security imagery. IEEE Transactions on Information Forensics and Security 13(9), 2203–2215 (Sep 2018). https://doi.org/10.1109/TIFS.2018.2812196

6. Dastidar, J.G., Biswas, R.: Tracking human intrusion through a CCTV. In: 2015 International Conference on Computational Intelligence and Communication Networks (CICN). pp. 461–465 (Dec 2015). https://doi.org/10.1109/CICN.2015.95

7. Girshick, R.: Fast r-cnn. In: 2015 IEEE International Conference on Computer Vision (ICCV). pp. 1440–1448 (Dec 2015). https://doi.org/10.1109/ICCV.2015.169

8. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 580–587. CVPR ’14, IEEE Computer Society, Washington, DC, USA (2014). https://doi.org/10.1109/CVPR.2014.81

9. Iandola, F.N., Moskewicz, M.W., Ashraf, K., Han, S., Dally, W.J., Keutzer, K.: Squeezenet: Alexnet-level accuracy with 50x fewer parameters and