Deep Learning Based Small Object (Weapon) Detection in Complex Scene Using YOLO-V8

Navuluri Hemanth Srivathsav[1], Vegesh Sai Boppana[1], Rajib Debnath[2] ✉, and Kakali Das[2]

[1] Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad, Telangana, India-500075.

[2] Department of Computer Science and Engineering, GITAM (Deemed to be University), Hyderabad Campus, Hyderabad, Telangana, India - 502329

hemanthsrivathsav@gmail.com, saivegeshbopp@gmail.com, rajibdebnath.cse@gmail.com ✉, kakalids54@gmail.com

Abstract. In today's world, where security concerns, are on the rise it has become more important than ever to detect weapons in environments. Our focus is on identifying individuals who are carrying handguns in forms of media such as images, videos, and CCTV footage. In this work, we offers a solution by utilizing learning techniques specifically using the YOLOv8 Nano model. To train and validate the model we have created a dataset consisting of 16,000 images featuring handguns and people holding them. The principal purpose of this study is to assess the ability of existing model to quickly and precisely detect threats in surveillance settings. The proposed work paves the way to enhance the capabilities of security systems allowing law enforcement agencies to promptly identify weapons and minimize response times while mitigating risks. More specifically, we proposed a dataset that was collected from several internet sources and the dataset posses most of the real time challenges in weapon detection such as blurrness of small weapons, partially occluded weapons, weapons having shape similarity with safe objects. The dataset collection is an ongoing process, in this work we have taken a part of the dataset and validate the dataset using YOLOV8 Nano Deep learning model.

Keywords: Small Object, Weapon Detection, Deep Learning, YOLOV8.

1 Introduction

In an era marked by growing security concerns [1,2] and the need for vigilant surveillance, detecting small, concealed weapons within complex, crowded scenes remains a formidable challenge [1,2]. Existing weapon detection systems often struggle to accurately identify these compact threats, leaving critical security gaps[5, 8-12]. In this work, we propose a weapon-specific dataset designed to accommodate all possible real-time challenges in weapon detection. The aim of this work is to develop a highly effective solution capable of robustly and rapidly detecting small weapons in diverse and challenging environments, providing a vital tool to enhance public safety and security measures. To accomplish this, the foremost requirement is a dataset.

Currently available object detection and recognition datasets [4-7] do not focus on weapons. Therefore, methodologies based on these datasets may lack the capabilities to detect weapons adequately. While methodologies trained for shape detection [10,11] may detect weapons, they may fail in classifying different weapons and may also struggle to detect weapons with different or similar shapes. The proposed dataset focuses on incorporating real-time challenges in weapon detection, including occlusion of weapons, similarity between weapons and safe objects, weapons occupying very small areas in captured scenarios, blurriness, and image quality.

In this manuscript, we discuss the dataset collection, creation of ground truths, and all other aspects of the dataset. We also implement the YOLO-V8 nano model on the dataset and record the results for analysis. Our proposed approach aims to address the critical need for enhancing security through advanced computer vision techniques. In today's complex surveillance environments, detecting concealed weapons is a major concern. By utilizing the latest advancements of the YOLO deep learning model with the newest v8 version, we aim to achieve high accuracy in identifying small weapons within crowded and challenging scenes, contributing to improved public safety and security measures.

Hence the contribution of this works are listed as follows:

- 1. Creation and development of a weapon dataset that accommodate all real time challenges in weapon detection application.
- 2. YOLO-V8 nano model has been implemented on the dataset to analyse the efficacy and validity of the dataset and deep based object detection model.

Table 1. Overview of Key Characteristics in Gun Detection Datasets

Dataset Used	**Key Attributes**	**Environment Condition**	**Ground Truth**
Gun Movies Dataset [8]	Captured in indoor settings, a single object holding a gun at a time, constant background	Indoor	No
IMFDB [13]	Freely available, dynamic backgrounds, includes various gun images from movies, obstructions and shadows, motion blur, appearance changes	Indoor, Outdoor	No
Dataset of R. Olmos et. al [14]	Sourced from the internet, manually annotated, features only pistol images, complex background, shadows, and camouflage foreground objects	Indoor, Outdoor	No

2 Related works

This manuscript presents a two-way approach: proposing a dataset and implementing the YOLO-V8 nano model on the same dataset. This section will discuss existing works related to weapon datasets and the YOLO-V8 nano model.

2.1 Weapon Dataset

So far, only a few datasets have been generated to validate and test various weapon/gun detection algorithms for benchmarking purposes. In Table 1, we have compiled existing visual datasets specifically for weapon/gun detection.

This section reports all the critical and essential features of these datasets. It is observed that the Gun Movies dataset[8], collected in indoor conditions, has very restricted factors such as a uniform background, uncluttered weapons, and a restricted number of persons. The other two datasets [13,14]were collected from the internet. The IMFDB dataset [13] is large but not specifically designed for weapon detection, and the dataset of Olmos et al. is also collected from the internet but is limited. The IMFDB dataset is not labeled, while the Olmos et al. dataset [14] is manually labeled. From this discussion, a research gap becomes evident.

2.2 Weapon Detection

In the existing literature, there is a relatively small amount of research dedicated to the field of visual gun detection. Many efforts have focused on detecting the shape of guns, relying on conventional feature descriptors such as FREAK [22], SURF[3], HOG[25], and SIFT[16]. Conventional classifiers like cascaded classifiers[12] and SVMs[16] were used for scene classification with or without guns. Some CNN architectures have also been trained and tested for gun detection. For instance, in [17], a sliding window methodology was employed, but it failed to detect more than one gun present in a scene. Nowadays, deep architectures have gained popularity for their accuracy in detection and classification. In this context, parameters of the F-RCNN architecture were tuned for gun detection in [14], but FRCNN also provided less accuracy for a wide variety of guns and resulted in more false positives. Another algorithm, OAOD proposed in [18], achieved fair results. The method first estimates the object's orientation, which is used to rotate the proposed objects. From these rotated objects, the largest possible rectangles are cropped, which are then classified and localized. The results of their technique, which are reasonable, are presented in Table 2.

The observed fact from these works is that due to the lack of proper dataset validation, the algorithms' applicability for real-time applications is doubtful. Hence, a benchmark dataset is of utmost importance for implementing real-time algorithms for gun detection. Furthermore, previously mentioned studies are unable to tackle all challenges such as Rotation, Illumination changes, Scaling, fluctuations in viewing angles, and Occlusions. Within this framework, we propose a new dataset designed to address and accommodate all these challenges, and a recent object detection algorithm, the YOLO-V8-Nano model, is trained and evaluated on the same.

3 Proposed Dataset

3.1 Data Collection

Table 2. Recent state-of-the-art gun detection based on visual images

Author	**Algorithm Used**	**Reported Accuracy**	**Limitations**
M.Grega et. al [8]	Neural Network, Background subtraction.	99.32%	No features included to handle illumination and issues with rotation and occlusion.
R.K.Tiwari et. al [9]	SURF features, K-means clustering.	84.27%	Only works on images, not video frames; detects gun but not the person holding it.
R.K.Tiwari et. al [3]	Harris point detector	88.67%	Designed for images only, not for video frames
S.Asnani et. al [12]	HOG features and classification.	NP	Ineffective in managing shadows and illumination issues
E.Ardizzone et. al [15]	Graph-Based Visual Saliency.	NP	Used a limited number of medium-quality web datasets
N.B.Halima et. al [16]	BoWSS, SIFT descriptors and SVM classifier.	F-measure: 0.35	Inefficient for live surveillance and time-consuming.
F.Gelana et.al [17]	Sliding window, CNN.	97.78%	Can only detect a single type of firearm.
R.Olmos et. al [14]	Faster RCNN.	Pre:94.17%, Rec:31.91%	Results depend on database and dataset design, with risk of false detections.
J. Iqbal et. al [18]	Faster RCNN.	Pre:0.888	High time complexity.

*NP=Not Provided, Pre=Precision, Rec=Recall

The initial phase involved gathering a dataset for weapon detection in complex scenes from IMFDb, which is available for free use. The IMFDb dataset consists of images collected from movies. However, it may be necessary to create a new dataset for small object weapon detection in complex scenes, as the existing datasets may not fully represent all the variations present in real-world scenarios. Therefore, we collected data from the IMFDB dataset as well as other sources, focusing on videos of "Armed with a Handgun," to create our custom dataset. Attention was given to collecting images of handguns and knives that accommodate various challenges mentioned.

The proposed dataset contains approximately 16,000 images, including handguns, knives, and images of other objects such as money bills, purses, cards, and smartphones, as classes included in the dataset. Of these, 12,870 images were utilized for training, while 2,475 images were for validation to consistently assess and enhance the model's effectiveness. The remaining 990 images form the testing dataset. The dataset contains images taken in both indoor and outdoor environments. The key contributing features of the dataset are highlighted below:

- The primary feature of the dataset is that it contains images where the weapon/weapon occupies a very small portion of the pixels in the entire scene.
- The non-weapon class of the dataset contains almost 40
- Images with guns oriented along various angles are also included in the dataset.
- Scaled, rotated, and partially occluded guns in scenarios are also included in the dataset.
- This dataset considers complex conditions in both indoor and outdoor environments.
- Dataset contains images with more than one weapon, including images of mass firing.

The brief description regarding features are mentioned in the Table 3.

Table 3. Statistical Analysis of Various Features of Proposed Dataset

Features	**No. of Images**	**With weapon**	**Multiple weapon**
Rotation	2853	P	P
Scaling	2906	P	P
Occluded	3576	P	P
Mask	3500	P	P
Multiple weapon	3500	NP	P
Illumination	5000	P	P

*P=Present,NP=Not Present

3.2 Ground Truth Generation

Ground truth annotations (GTs) are essential for evaluating methods applied to the dataset. We carried out three rounds of label annotations: in the first round, moving objects were labeled, the second round focused on annotating shadows, and the third round annotated guns. These annotations adhere to specific attributes outlined in previous work[27], including bounding boxes that enclose the visible extent of the object and classifications like moving objects, shadows, and firearms.

To maintain consistency in the ground truth (GT) annotations, a single annotation team was assigned the task. The annotations are organized in a two-label tree structure, with the root-level annotators adhering to established guidelines. These guidelines specify what elements to annotate, the method of annotation, definitions for each label, and how to address occlusions and rotations. Annotators undergo periodic observation and training to maintain consistency.

After the root label annotations, parent label annotators validate the GTs, checking for false annotations, ensuring proper handling of occlusions, and confirming complete object coverage. The employed GT annotation approach is consistent, mostly correct, and comprehensive, with few annotation errors.

We employed the TSLAB[28] annotation tool to validate our annotation framework, known for its effectiveness in object annotation and handling occlusions. The dataset was annotated using TSLAB and compared against annotations produced by alternative methods, achieving a similarity score of 95%, demonstrating our annotation framework's effectiveness.

Our annotation framework surpasses the limitations of traditional bounding boxes by leveraging the accuracy of human perception. Sample GTs alongside original frames are illustrated in Figure 1. The above methods were employed to improve image quality and preprocess them to prepare for training. We generated ground truth images by labelling the weapon objects in the datasets with the help of TSLAB tool[29] and manually labelling them.

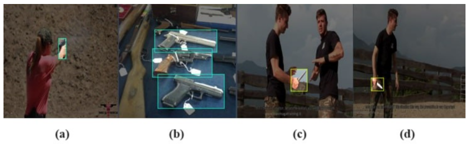


Fig. 1. Samples of few generated GT; (a), (b) refers to the sample GT of Pistol and (c), (d) refers to the sample GT of Knife

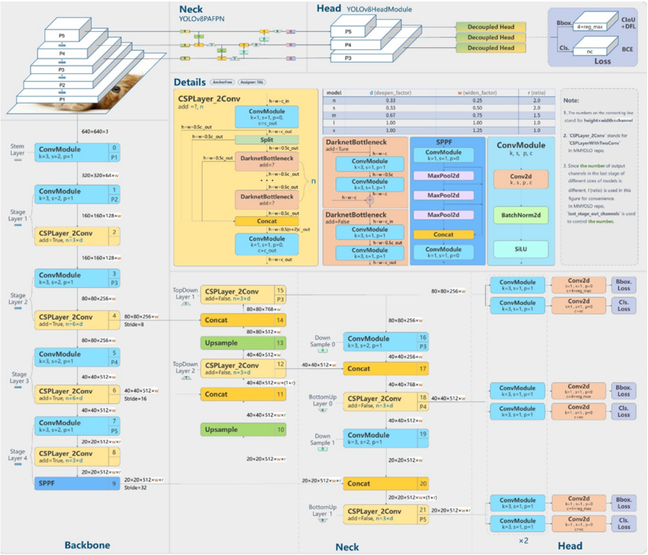
4 Methodology

4.1 YOLOv8

YOLO-v8 [23] is the most recent iteration of the algorithm in January 2023 from Ultralytics a Cutting-edge deep learning-based model expanding upon prior advancements, YOLO has undergone successive improvements to boost the flexibility and performance of the model. YOLOv8 has five variants, the smallest nano model (suffix "n") and the largest extralarge model(suffix "x"), v8n (nano),v8s (small), v8m (medium), v8l (large) and v8x (extralarge) presenting total five different versions based on its size. YOLOv8 is applied across diverse tasks including object tracking, image classification, object detection, and image segmentation.

4.2 YOLOv8 – Architecture

The extraction of features from input images is managed by the backbone network shown in Figure 2. Meanwhile, detection layers undertake the task of forecasting bounding boxes and class probabilities. "YOLOv8 uses a similar backbone as YOLOv5" [21], It incorporates elements from YOLOv5 such as CSP connections, feature fusion methods, and the SPPF module, with additional improvements and modifications to CSP called as C2f module (cross-stage partial bottleneck consisting of two convolution layers). Within the C2F model, the outcomes from the Bottleneck, a phrase for two 3x3 convolutions with residual connections are amalgamated.



model	d (deepen_factor)	w (widen_factor)	r (ratio)
n	0.33	0.25	2.0
s	0.33	0.50	2.0
m	0.67	0.75	1.5
l	1.00	1.00	1.0
x	1.00	1.25	1.0

Note:

1. The numbers on the connecting line stand for height×width×channel
2. 'CSPLayer_2Conv' stands for 'CSPLayerWithTwoConv' in MMYOLO repo.
3. Since the number of output channels in the last stage of different sizes of models is different, r (ratio) is used in this figure for convenience. In MMYOLO repo, 'last_stage_out_channels' is used to control the number.

Fig. 2. Architecture of YOLOv8 [22], where the backbone of the network is followed by detection layers.

YOLOv8 employs an anchorless model with separated head to individually handle objects, classification, and regression tasks. Instead forecasting an object's distance from a predetermined anchor box, C2f module shown in Figure 3(a) explicitly estimates the centre of an object. Anchorless detection shown in Figure 3(b)reduces the quantity of prediction boxes[24].

This architecture shown in Figure 2 enables each branch to concentrate on its designated responsibility, thereby enhancing the model's overall accuracy. The sigmoid function is utilized as the activation function in the output layer of YOLOV8 for computing the objectness score SoftMax function is utilized for calculating the probabilities of each class, indicating the probability of an object being classified into a specific class. v8 uses CIoU[19] and DFL[20] loss functions for box localization error and binary cross-entropy for class prediction error.

5 Experiment Results and Discussion

The proposed model was trained on Google Collab with Tesla T4 GPU over around 2 hours. Experimental results indicate that our model can achieve an impressive precision of 92.8% in detecting weapons while maintaining an overall mean average precision (mAP) of 95.5%, with a recall rate of 90.8%. The box loss is recorded at 0.762 while class loss stands at 0.439; moreover, object loss is measured at 1.62. An accompanying figure presents comprehensive findings from our proposed model which encompass losses as well as precision and recall metrics.

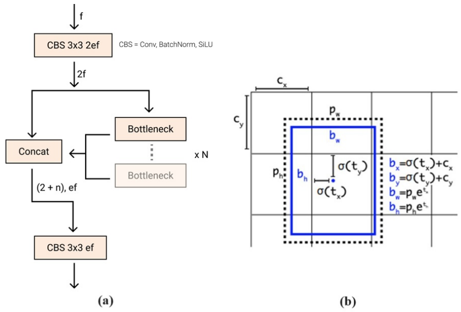


Fig. 3. (a) C2f Module, (b) Visualization of Anchorless detection

5.1 Performance Metrics

Recall indicates the proportion of accurately identified positive predictions among all actual positives. Calculation can be performed using the Recall formula which is $\frac{TP}{TP+FN}$. Precision, on the other hand, represents the percentage of true positive predictions out of all positive predictions and is determined as $\frac{TP}{TP+FP}$. These metrics are considered when computing F-Measure, denoted as FM and calculated using the equation: $FM = 2 \times \frac{recall \times precision}{recall+precision}$

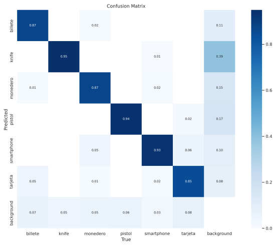


Fig. 4. Confusion Matrix Showing Prediction Summary for 6 Classes classification model's performance across six categories

5.2 Evaluation

To assess the performance of the YOLO-V8-Nano on the proposed dataset and benchmark the dataset, several versions of YOLO were trained and tested on the same dataset. The results are compared and recorded in Table 4. It can be observed that several versions of YOLO were able to attain more than 94% accuracy on average. These results demonstrate that the dataset is indeed challenging, with YOLO-V8 achieving a maximum accuracy of 97%. The corresponding confusion matrix is also presented in Figure 4. From this, it is evident that the YOLO-V8-Nano model performs well for all classes, although it does produce a fair number of false positives.

Furthermore, the performance of the classification model across six diverse categories: weapons, blades, bank cards, currency, wallets, and phones is presented in terms of Recall-Confidence Curve (RC), Precision-Recall Curve (PR),

Table 4. Comparison of the Yolo models on the proposed data set.

Dataset Used	**Model**	**Precision%**	**Recall%**	**F1 Score**
Proposed Dataset	Yolo-v8	**97.7**	**97.8**	**97.75**
	Yolo-v2	97.23	97.74	97.49
	Yolo-v5	94.99	78.28	85.83
	Yolov-6	93.48	28.29	43.44
Gun Movies Dataset	Yolo-v8	**96.34**	**95.43**	**95.89**
	Yolo-v2	95.56	95.78	95.67
	Yolo-v5	94.44	77.45	85.11
	Yolov-6	92.38	78.31	84.77
IMFDB	Yolo-v8	**95.87**	95.83	**95.85**
	Yolo-v2	94.56	**96.23**	95.39
	Yolo-v5	92.69	83.86	88.05
	Yolov-6	94.82	88.93	91.78
R. Olmos Dataset	Yolo-v8	94.92	**97.33**	**96.11**
	Yolo-v2	95.39	94.11	94.75
	Yolo-v5	91.02	88.93	89.97
	Yolov-6	**96.38**	86.81	91.34

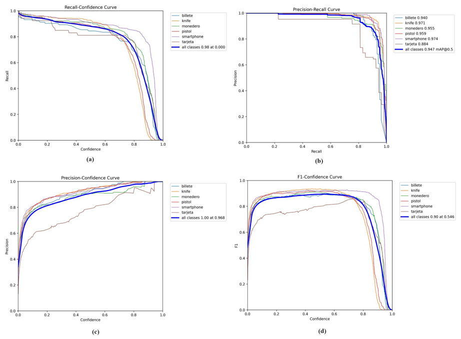


Fig. 5. (a)Recall-Confidence curve (b) Precision-Recall curve (c) Precision-Confidence curve (d) F1-Confidence curve

Precision-Confidence Curve (PC), and F1-Confidence Curve (FC) in Figure 5. These curves demonstrate the efficacy of the YOLO-V8 model in detecting weapons, guns, and knives in the given scenario. The RC, PC, and FC curves present the recall, precision, and F1 value against various confidence threshold values, showing fair results with 0.98 at 0.00.

The training matrices shown in Figure 6 and The Mean Average Precision (MAP) curve shown in Figure 7 is also analyzed to assess the performance of the YOLO-V8-Nano model during training. The MAP curve shows that as the number of epochs increases, the performance of the YOLO-V8 model also increases, indicating that the method retrieves relevant instances with high precision and high recall.

Performance indicators for the YOLO object detection algorithm include a loss function consisting of three components: class loss, object loss, and box loss. The figure shows the loss curves. Class loss, object loss and box loss curves are

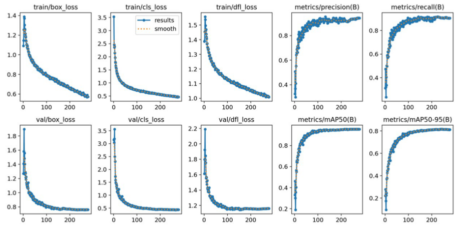


Fig. 6. Visualizations of Trainig Metrics.

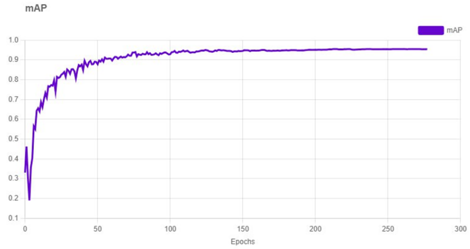


Fig. 7. Mean Average Precision (mAP).

shown in Figure 8. The class loss curve suggests that the model predicts the correct class label with high confidence for each detected object. Box loss calculates the variation between the predicted and actual bounding box coordinates for each detected object, demonstrating the model's ability to precisely predict the spatial extent (location and size) of each object in the image. The object loss combines class loss and box loss and is applied to each detected object. The minimum object loss of the corresponding model implies that the YOLO-V8 model performs effectively on the proposed dataset and validates the proposed dataset. Few sample results are also shown in Figure 9 amd Figure 10.

6 Conclusion

This study utilized YOLOv8 to create a highly effective weapon detection system. By utilizing a varied dataset, the model successfully achieved precise real-time recognition of small weapons in intricate environments, strengthening security protocols. This inventive approach significantly contributes to public safety and improves surveillance capabilities. In this scope of work, we carefully curated a varied set of data and optimized YOLOv8 to create an effective weapon detection system. Yolo-V8 have achieved a high-precision real-time solution for recognizing small weapons in challenging situations, making a meaningful contribution

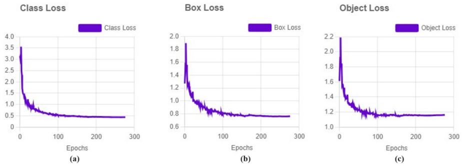


Fig. 8. (a) Class loss. (b) Box loss. (c) Object loss

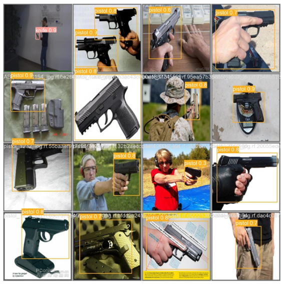


Fig. 9. Sample outputs from the dataset with the detected regions of interest (ROI).

to public safety and security. Enhancing system performance can be achieved by implementing real-time alerts, semantic segmentation, and efficient hardware acceleration. Additionally, adaptability and accuracy improvement over time can be facilitated through transfer learning, cross-domain generalization, and feedback mechanisms. Finally, enhancing compatibility with current security infrastructure will enhance practical application of weapon detection systems.

References

1. Welsh, B. C. and Farrington, D. P.: "Public Area CCTV and Crime Prevention: An Updated Systematic Review and Meta–Analysis", Justice Quarterly, vol. 26, no. 4, pp. 716–745, 2009.
2. Gill, M. and Spriggs, A.: "Assessing the impact of CCTV. London: Home Office Research", Development and Statistics Directorate, 2005.
3. Tiwari, R.K., Verma, G.K.: 'A computer vision based framework for visual gun detection using harris interest point detector', Proc. Comput. Sci., 2015, 54, pp. 703–712
4. Stauffer, C., Grimson, W.E.: 'Adaptive background mixture models for realtime tracking'. Proc. 1999 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (Cat. No PR00149), Fort Collins, CO, USA, 23 June 1999, vol. 2, pp. 246–252
5. Ha, J.E, ., Lee, W.: 'Foreground objects detection using multiple difference images', Opt. Eng., 2010, 49, (4), p. 047201
6. Barnich, O., Van, D.M.: 'ViBe: a universal background subtraction algorithm for video sequences', IEEE Trans. Image Process., 2010, 20, (6), pp. 1709–1724
7. Cheng, F.C., Huang, S.C., Ruan, S.J.: 'Illumination-sensitive background modeling approach for accurate moving object detection', IEEE Trans. Broadcast., 2011, 57, (4), pp. 794–801
8. Grega, M., Matiolanski, A., Guzik, P., et al.: 'Automated detection of weapons and knives in a CCTV image', Sensors, 2016, 16, (1), p. 47
9. Tiwari, R.K., Verma, G.K.: 'A computer vision based framework for visual gun detection using SURF'. 2015 Int. Conf. on Electrical, Electronics, Signals, Communication and Optimization (EESCO), Andhra Pradesh, India, January 2015, pp. 1–5
10. Kmiec, M., Glowacz, A., Dziech, A.: 'Towards robust visual knife detection in images: active appearance models initialised with shape-specific interest points'. Int. Conf. on Multimedia Communications, Services and Security, Berlin, Germany, 31 May 2012, pp. 148–158

11. Maksimova, A.: ‘Knife detection scheme based on possibilistic shell clustering’. Int. Conf. on Multimedia Communications, Services and Security, Berlin, Germany, 6 June 2013, pp. 144–152
12. Asnani, S., Ahmed, A., Manjotho, A.A.: ‘Bank security system based on weapon detection using HOG features’, Asian J. Eng. Sci. Technol., 2014, 4, (1), pp. 23–29
13. Main Page, Internet Movie weapons Database – Guns in Movies, TV and Video Games’. Available at `http://www.imfdb.org/wiki/Main_Page.,accessed:13-May-2019`
14. Olmos, R., Tabik, S., Herrera, F.: ‘Automatic handgun detection alarm in videos using deep learning’, Neurocomputing, 2018, 275, pp. 66–72
15. Ardizzone, E., Gallea, R., La Cascia, M., et al.: ‘Combining top-down and bottom-up visual saliency for weapons localization’. 2014 Int. Conf. on Signal Processing and Multimedia Applications (SIGMAP), Vienna, Austria, 28 August 2014, pp. 25–32
16. Halima, N.B., Hosam, O.: ‘Bag of words based surveillance system using support vector machines’, Int. J. Secur. Appl., 2016, 10, (4), pp. 331–346
17. Gelana, F., Yadav, A.: ‘weapon detection from surveillance cameras using image processing and machine learning techniques’. Smart Innovations in Communication and Computational Sciences, Singapore, 2019, pp. 25–34
18. Iqbal, J., Munir, M.A., Mahmood, A., et al.: ‘Orientation Aware Object Detection with Application to weapons’. arXiv preprint arXiv:1904.10032. 22 April 2019
19. Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., & Ren, D. (2020). Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. Proceedings of the AAAI Conference on Artificial Intelligence, 34(07), 12993-13000. `https://doi.org/10.1609/aaai.v34i07.6999`
20. Li, Xiang, et al. "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection." Advances in Neural Information Processing Systems 33 (2020): 21002-21012
21. Terven, Juan, and Diana Cordova-Esparza. "A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond." arXiv preprint arXiv:2304.00501 (2023).
22. M. Contributors, “YOLOv8 by MMYOLO.” `https://github.com/open-mmlab/mmyolo/tree/main/configs/yolov8`, 2023. Accessed: 22 Nov, 2023.
23. G. Jocher, A. Chaurasia, and J. Qiu, “YOLO by Ultralytics.” `https://github.com/ultralytics/ultralytics`, 2023. Accessed: 22 Nov, 2023
24. https://blog.roboflow.com/whats-new-in-yolov8/#what-is-yolov8 Accessed: 22 Nov, 2023
25. Tiwari, Rohit Kumar, and Gyanendra K. Verma. "A computer vision based framework for visual gun detection using harris interest point detector." Procedia Computer Science 54 (2015): 703-712.
26. Khalid, Shehzad, et al. "Weapon detection system for surveillance and security." 2023 International Conference on IT Innovation and Knowledge Discovery (ITIKD). IEEE, 2023.
27. ‘The PASCAL Visual Object Classes Homepage’, available: `http://host.robots.ox.ac.uk/pascal/VOC/,accessed:13-May-2019`
28. Cuevas, C., Yanez, E.M., Garcia, N.: ‘Tool for semiautomatic labeling of moving objects in video sequences: TSLAB’, Sensors, 2015, 15, (7), pp. 15159–15178