Proceedings of the 7th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2023)
IEEE Xplore Part Number: CFP23OSV-ART; ISBN: 979-8-3503-4148-5

2023 7th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC) | 979-8-3503-4148-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/I-SMAC58438.2023.10290401

Next-Gen Security: YOLOv8 for Real-Time Weapon Detection

Deepali Deshpande
Department of Information Technology
Vishwakarma Institute of Technology
Pune, India
deepali.deshpande@vit.edu

Manas Jain
Department of Artificial Intelligence and Data Science
Vishwakarma Institute of Technology,
Pune, India
manas.jain21@vit.edu

Adhip Jajoo
Department of Artificial Intelligence and Data Science
Vishwakarma Institute of Technology,
Pune, India
adhip.jajoo21@vit.edu

Devika Kadam
Department of Artificial Intelligence and Data Science
Vishwakarma Institute of Technology,
Pune, India
devika.kadam21@vit.edu

Harshvardhan Kadam
Department of Artificial Intelligence and Data Science
Vishwakarma Institute of Technology,
Pune, India
harshvardhan.kadam21@vit.edu

Aryan Kashyap
Department of Artificial Intelligence and Data Science
Vishwakarma Institute of Technology,
Pune, India
aryan.kashyap21@vit.edu

Abstract*—The swift and accurate identification of weaponry holds paramount importance in military operations to ensure the safety of personnel and the effectiveness of missions. In recent times, deep learning models have emerged as robust solutions for object detection tasks, rendering them valuable tools for enhancing military security. This research study delves into the realm of weapon detection by presenting a novel approach utilizing YOLOv8-Small, a streamlined variant of the renowned You Only Look Once (YOLO) detection framework. The study's primary objective revolves around harnessing the capabilities of YOLOv8-Small for precise weapon detection within military contexts. Through a meticulous design process and rigorous training, the proposed model demonstrates its competence in identifying a diverse range of weapons with remarkable accuracy and efficiency. The experimental results validate the potential applicability of YOLOv8-Small in bolstering military operations, underscoring its utility as a force multiplier on the battlefield. Moreover, the research delves into the model's adaptability to varying environmental conditions, a critical factor in real-world military scenarios. The findings reveal the model's capacity to maintain consistent performance across different terrains, lighting conditions, and weather situations. This adaptability significantly enhances its operational viability, ensuring reliable weapon detection capabilities even under challenging circumstances. The implications of this research extend to broader military strategies and tactics, where rapid and accurate weapon detection can tip the scales in favor of mission success. The potential integration of YOLOv8-Small with existing military systems holds promise for enhancing situational awareness and proactive threat mitigation. In conclusion, this research study presents a pioneering contribution to the field of military weapon detection by leveraging YOLOv8-Small's efficiency and adaptability. The study's insights provide valuable guidance for military stakeholders seeking innovative solutions to enhance security, thereby paving the way for more effective and safeguarded military operations.

Keywords*—Object Recognition, Deep Learning, Convolutional Neural Network, Computer Vision, Single Shot Detection, You Only Look Once version 8-Small (YOLO v8s).

I. INTRODUCTION

In order to ensure public safety and stop potential violent acts,weapon detection is a vital responsibility. The threat of mass shootings and other violent crimes is growing in today's society. Particularly in congested or challenging locations, traditional physical inspection and surveillance approaches are not always successful in finding firearms. The automationof the weapon detection process has shown tremendous promise when using deep learning techniques.[1],[6],[8],[14]. YOLO (You Only Look Once) is one of the most well-liked deep learning object identification frameworks [4],[5],[12]. The most recent version of YOLO, known as v8, provides a number of improvements over earlier iterations, including increased accuracy, speed, robustness, and scalability. Because it may combine these benefits, YOLOv8 is relevant for the job of detecting weapons. YOLOv8 is fast enough to be utilized in real-time applications, accurate enough to reliably detect weapons, andscalable enough to be used to detect a variety of weapons. YOLOv8 is a dependable tool for weapon detection in a variety of contexts since it is resilient to changes in illumination, posture, and occlusion. The research investigate the usage of YOLOv8 for weapon detection in security applications. On a range of datasets and in a variety of environments, we will assess YOLOv8's performance. The study also talks about YOLOv8's drawbacks and potential upgrades in the future. The findings of this study will contribute to enhancing the resilience, accuracy, and speed ofweapon detecting systems. This will enable the deployment of this model on hardware systems because of its robustness and light weight nature.

In the following sections, the research will delve into the methodology, experimental setup, results, and discussions to provide a comprehensive analysis of the proposed weapon detection approach using YOLOv8-Small. The research objectives of this study revolve around the development and assessment of a real-time weapon detection system utilizing YOLO v8. The primary goal is to create a robust and efficient system capable of identifying weapons in real-time scenarios. To achieve this, the system will be thoroughly evaluated using diverse datasets, ensuring its effectiveness across a range of conditions and scenarios. Additionally, this research aims to contribute to the field by comparing the performance of the developed system with other contemporary state-of-the-art methods for weapon detection, thereby providing valuable insights into the system's capabilities and potential advancements in the domain of security and surveillance technologies.

II. LITERATURE SURVEY

In recent years, there has been a growing concern regarding gun violence rates and the use of lethal weapons in various settings. To address these concerns and enhance security measures, researchers have turned to artificial intelligence (AI) and deep learning techniques for weapon detection in security applications. This paper aims to provide an overview of several research papers in this domain, shedding light on their methodologies, findings, and limitations, while also drawing connections between them.

P. Shanmugapriya's paper [1] begins by acknowledging the challenges associated with weapon detection, such as the diverse appearance of weapons, clutter and occlusions, and rapidly changing security scenarios. The paper reviews various AI methods, including Support Vector Machines (SVMs), Random Forests (RFs), and Deep Neural Networks (DNNs), applied to weapon detection. The authors propose a DNN-based approach that involves feature extraction using a Convolutional Neural Network (CNN) and subsequent classification. However, a notable limitation of this study is the absence of real-time testing, which is crucial for practical security applications. Dr. N. Geetha 's paper [2] shares a similar concern over gun violence and presents a machine learning-based firearm identification system. The study utilizes a Support Vector Machine (SVM) to classify images containing firearms based on characteristics like size, color, and shape. While this approach demonstrates promise, it overlooks the impact of environmental conditions and adversarial attacks, which are critical factors for real-world deployment. Harsh Jain's research [3] focuses on enhancing security through video surveillance systems capable of detecting aberrant actions, including weapon detection. Two CNN-based algorithms, Single Shot Detector (SSD) and Faster R-CNN, are employed in this study. The results show fair accuracy, but the trade-off between speed and accuracy suggests that specific requirements and constraints must be considered. The inability of Faster R-CNN to detect firearms in real time and the challenge of gathering a sizable dataset of weapon photos are notable limitations. Sanam Narejo's work [4] aims to create an accurate gun detection smart surveillance security system using the YOLOv3 model. This system not only distinguishes between types of firearms but also records incident details for future reference. The YOLOv3 model consists of a CNN for feature extraction, a Region Proposal Network (RPN) for object location proposals, and a classifier for weapon classification. The study's innovative approach includes simulating a real-world scenario with socket programming. However, future work is needed to train the model on a larger dataset and reduce false positives. Haitong Lou's paper [5] introduces the DC-YOLOv8 algorithm for small size target detection, addressing the limitations of human observation and judgment errors in complex environments. This algorithm improves detection accuracy for small size objects while maintaining accuracy for larger targets. Experimental results using public datasets demonstrate superior performance compared to YOLOv8. This research contributes to the advancement of target recognition techniques, particularly for small size objects.

In conclusion, these research papers collectively contribute to the field of weapon detection using AI and deep learning techniques for security applications. They highlight the importance of addressing real-time detection, environmental conditions, adversarial attacks, and the trade-off between speed and accuracy in practical implementations. Further advancements in these areas are essential to enhance security and protect public safety effectively.

III. APPROACH TO MODEL SELECTION

Modern object detection algorithms can be divided into two categories: Two Shot object detection and Single Shot object detection.

In Two -Shot object detection, an image is passed twice in thealgorithm before predicting its output. During the first pass, aset of suggestions or potential object positions are generated. The second pass then further develops these hypotheses to generate the final predictions. Compared to single-shot object detection, this approach requires more computation but is more accurate. Region-based convolutional neural networks (RCNN), Fast RCNN, and Faster RCNN are a few of the well-known models in this area. [3]

Single stage object detectors have a simplified architecture that is specifically built for object detection in a single step while taking into account all region proposals. These detectors are computationally efficient since they produce thebounding boxes and probability of an object belonging to a particular class by taking into account all the spatial sizes of an image at once. But compared to other approaches, single-shot object detection typically performs less accurately. In situations with limited resources, these strategies can be used to quickly detect objects. A fully convolutional neural network (CNN) is used by the single-shot detector YOLO to analyze a picture. Generally speaking, single-shot object detection is better for real-time applications than two-shot object detection for applications that prioritize accuracy.

Numerous single stage object detection-based algorithms have been developed recently, including Deconvolution Single Shot Detector (DSSD), M2Det, RetinaNet, and RefineDet++ [3][4] . But, due to the complexity and strength of Two Stage detectors, they typically outperform Single Stage detection algorithms. However, since the development of You Only Look Once (YOLO) and its sequels, efforts to complete object detection in a single step have earned excellent reviews. Deep neural networks are used in these techniques to tackle the localization problem, which is framed as a regression problem. It is observed that YOLO isgiving both earlier single staged detectors and two staged detectors a stiff battle in terms of accuracy and prediction time. It is only one of the most widely used solutions in production because of its simple architectural design, low level of complexity, and simple implementation. As a result,the YOLO architecture was chosen sinceit offers the speed and accuracy needed for real-time object identification. YOLOv8 is the newest and most accurate real-time object identification model, and it

Proceedings of the 7th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2023)
IEEE Xplore Part Number: CFP23OSV-ART; ISBN: 979-8-3503-4148-5

best serves our goals because of its balanced configuration between speed and accuracy. YOLOv8 comes in a variety of model variations to meet userneeds. These versions have been refined for usage and available computing power.

Table 1. Comparison of different versions of YOLO v8

Model	Size (pixels)	mAP Value (50-95)	Speed CPU ONNX (ms)
YOLOv8n	640	37.3	80.4
YOLOv8s	640	38.3	128.4
YOLOv8m	640	39.2	234.7
YOLOv8l	640	39.7	375.2
YOLOv8x	640	41.3	479.1

After evaluating these different variations, it was concluded that the YOLOv8s model offers a good frame rate with excellent accuracy despite operating on a modest computational device. Due to the fact that the proposed algorithm must be able to detect weapons using autonomous cars or cameras placed throughout diverse locations, which may not have powerful processing capabilities, YOLO v8s seems a reliable option.

IV. ARCHITECTURE

It is unable to directly assess the direct research methods and ablation studies used to develop YOLOv8 because it does not yet have a published paper. Having stated that, the repositoryand information on the model was examined that was accessible to begin documenting what is new in YOLOv8. The following elements make up YOLOv8-Small'sarchitecture details:

Backbone:

- YOLOv8-Small is built on the Darknet architecture, which is composed of several convolutional layers.
- It collects features from the input image atvarious sizes to capture object details.

Neck:

- As a neck component, YOLOv8-Small uses feature pyramid networks (FPNs).
- Multi-scale feature fusion is made possible by FPNs, allowing the model to recognize objects of varying sizes.
- Skip connections are used to connect low- level features with higher-level layers, aiding in the detection of small objects.

Detection Heads:

- Within YOLOv8 Small, there are several detection heads assigned the duty of forecasting bounding boxes and probabilities associated with various classes.
- Each detection head is composed of convolutional layers followed by post- processing operations.
- Anchor boxes, pre-defined bounding box shapes of different scales and aspect ratios, are utilized to enhance object localization.

Training:

- YOLOv8-Small is trained using annotatedtraining data, which includes bounding boxannotations and corresponding class labels.
- Classification loss functions, such as cross-entropy loss, are employed to train the model for accurate object classification.
- Localization loss functions, such as thesmooth L1 loss, are used to train the modelfor precise object localization.

Some special features of YOLO v8s are:

Anchor Free Model:

The YOLOv8 model differs from older YOLO models by being anchor-free, meaning it predicts an object's center directly instead of calculating its offset from pre-defined anchor boxes. The use of anchor boxes in previous YOLO versions posed challenges, as they might represent the box distribution of a standard benchmark but not necessarily the custom dataset being used. With anchor-free detection, the model makes fewer box predictions, leading to a faster and more efficient Non- Maximum Suppression (NMS) process. NMS is a post-processing step that filters through candidate detections after the model's inference to retain the most accurate and relevant detections.

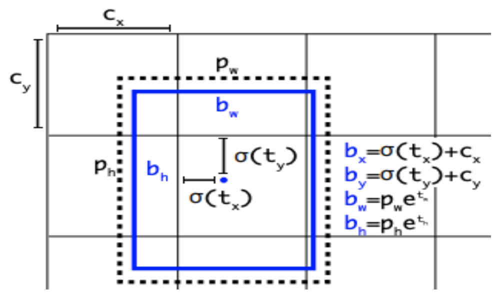

Fig 1. Visualization of Anchor Box in YOLO

New Convolutions:

The YOLO v8 model uses a total of 23 convolutional layers. The first 14 layers are used in the backbone network, which is responsible for extracting features from the input image. The remaining 9 layers are used in the neck and head networks, which are responsible for generating the output predictions. The convolutionallayers in the YOLO v8 model use a variety of kernelsizes and activation functions. Small kernel sizes (3x3 or 5x5) and Rectified Linear Unit (ReLU) activation functions are used in the first few layers. With the help of these layers, low-level elements from the source image, like edges and textures, are extracted. The later layers in the model use larger kernel sizes (7x7 or 11x11) and ELU activation functions. These layers are used to extracthigh-level features from the input image, such as object parts and object shapes.

In YOLOv8, there were some modifications made tothe fundamental building block. Specifically, C2f (acomponent) was used in place of C3 from previous versions. Additionally, the initial 6x6 convolution in the stem was replaced with a 3x3 convolution. Despite the increase in the kernel size from 1x1 to 3x3, the bottleneck in YOLOv8 remained similar tothat of YOLOv5. These changes suggest that YOLOv8 might be moving back towards the ResNet block, which was originally introduced in 2015. In the neck of YOLOv8, features are directly combined without enforcing strict channel proportions. As a result, there is a reduction in the overall number of parameters and the size of the tensors in the model. This design choice likely contributes to better efficiency and optimization in the YOLOv8 architecture.

Closing the Mosaic Augmentation:

While model architecture is often the focus of deep learning research, YOLOv5 and YOLOv8's training procedure is crucial to their effectiveness. YOLOv8 enhances photos while you're training online. The model views a slightly different variety of the images it has been given at each epoch. Mosaic augmentation is one of the augmentations. The model is compelled to learn things in novel places, in partial occlusion, and against varioussurrounding pixels as a result of the stitching together of four photos.

The YOLOv8-Small architecture is designed to be lightweight and efficient, making it suitable for real-time object detection applications. It strikes a balance between accuracy and computational efficiency, leveraging techniques such as FPNs, anchor boxes, skip connections, and optimized loss functions to achieve robust and effective object detection performance.

V. METHODOLOGY

Modern object detection algorithms can be divided into two categories: Two Shot object detection and Single Shot object detection.

A. Data Collection and Pre-processing

The first phase of our research involved constructing a customized dataset tailored to the specific requirements of our weapon detection model. There are various steps that have to be followed in the preprocessing of the dataset:

1. **Data Collection and Annotation:** The project began by curating a diverse dataset containing images captured from various scenarios. Present dataset contains 9633 images which are further divided into 5 classes that are Pistol, Missile, Gun, Grenade, and knife [4],[2].Each image was meticulously annotated with bounding boxes around weapons. [2]
2. **Data Cleaning:** The annotated dataset underwent athorough quality control process to identify and rectify labeling errors, missing annotations, and inconsistencies.
3. **Data Augmentation:** To enhance the model's ability to generalize, augmentation techniques wereapplied to range of data. [8]
4. **Data Splitting:** The dataset was split into three subsets: a training set, a validation set, and a testing set. The 75-15-10 split ratio was chosen to ensure sufficient data for training while enabling robust evaluation. [1]

B. Detection Procedure of YOLO v8s

The detection procedure in the YOLOv8s model is as follows:

The input image is resized to a fixed size of 640x640 pixels. Subsequently, the image is partitioned into a grid comprisingcells arranged in a 13x13 layout, with each cell tasked with detecting objects within its designated image area. For each cell, the model predicts 5 bounding boxes, each of which hasthe following attributes:

- x, y: The coordinates of the center of the boundingbox relative to the cell.
- w, h: The width and height of the bounding box.
- Confidence: The probability that the bounding boxcontains an object.
- Class_id: The ID of the object class that the bounding box is most likely to contain.

The model then applies a non-maximum suppression algorithm to the bounding boxes to remove any overlapping boxes that are unlikely to contain objects. The remaining bounding boxes are then ranked by their confidence scores, and the top-scoring boxes are returned as the detection results. The YOLOv8s model is a single-stage object detection model, which means that it predicts all of thebounding boxes and class labels for an image in a single pass.This makes it faster than two-stage object detection models

The dataset, which consisted of approximately 9633 images, was accompanied by annotation files. To generate these annotations, a cloud-based platform was leveraged that allowed the researchers to draw bounding boxes around weapons in each image and assign corresponding labels to them. This meticulous annotation process ensured that the model would be trained on accurately labeled data, critical for achieving high detection accuracy.

Subsequently, the YOLOv8 small architecture was chosen for the object detection model, as it offered real-time detection capabilities, making it well-suited for our application. This architecture allows us to detect weapons in images and video streams with remarkable efficiency essential for security applications that require quick response times.

During the training phase, model was inputted with the dataset into the neural network and conducted thorough experiments to optimize the model's performance. To obtain the best possible results, the YOLOv8 small model was trained for a range of 25 to 40 epochs. This training duration allowed us to fine-tune the model and achieve optimal performance on our specific dataset. The objective was to minimize the loss function, focusing on both box_loss (regression loss for bounding boxes) and class_loss (categorical classification loss). This ensured the model acquired the ability to precisely anticipate bounding boxes and accurately categorize weapons within the images.

C. Validation of the Model

After achieving a well-trained model with minimum loss, the model was proceeded to validate its performance using a separate validation set, which was previously reserved for this purpose. During validation, the hyperparameters were fine-tuned to further enhance the model's generalization capabilities. Through this process, a balance was achieved between avoiding overfitting on the training data and maintaining good performance on unseen data.

D. Testing of the Model

Finally, with a well-optimized and validated model, inferences were run on the test dataset to evaluate the efficiency of the selected approach. This step involved passing the test images through the model to detect and visualize the presence of weapons accurately. YOLOv8 first predicts a set of bounding boxes around objects in an image.Then, it uses a confidence score to determine which boundingboxes are likely to contain objects. Finally, it removes overlapping bounding boxes to ensure that only one boundingbox is predicted for each object in the image. The remaining bounding boxes with high confidence scores are the predictedlocations of objects in the image. Also, the key performance metrics such as precision, recall, and Intersection over Union(IoU) were measured to quantitatively assess the model's accuracy and effectiveness in detecting weapons.

E. Factors to Improve Efficiency of the Model

To further increase the efficiency of the model following points can be taken into consideration:

- **Using a large dataset:** A larger dataset will help to train YOLO models to be more accurate and robust,at present the dataset had 9633 images, and it can be increased to approx. 25,000 to improve the efficiency to its max. [10]

- **Using quantization technique:** Quantization is a technique that converts the weights and activations of a neural network from floating point values to integer values. This can make the model much smaller and faster, without sacrificing too much accuracy, post-training quantization can be used to make our model faster.

- **Using a faster hardware accelerator:** YOLO models can be very computationally expensive to train and inference. Using a faster hardware accelerator, such as a GPU or TPU, can significantlyimprove performance efficiency. [4]

- **Optimize Hyperparameters**: Experiment with different hyperparameters such as learning rate, batch size, epochs, regularization, etc. Hyperparameter optimization techniques like grid search or Bayesian optimization can help you find the best set of hyperparameters for your task.

The combination of the meticulously curated dataset, the choice of YOLOv8 small architecture, extensive training, and careful hyperparameter tuning resulted in an efficient and reliable weapon detection model. By focusing on real-time detection capabilities, our model is well-suited for security applications, contributing to enhanced safety and threat detection in various real-world scenarios.

VI. RESULTS

The proposed project developed a YOLOv8-based weapon detection model for web API deployment and hardware integration. The YOLO v8 small object detection model has a detection rate of 48.6% mAP on the custom test-set. This means that it can correctly detect small objects 48.6% of the time, with a confidence of at least 50%. This is a significant improvement over previous YOLO models, which had difficulty detecting small objects. Real-time testing showed an average speed of 25 frames per second on CPU and couldbe much higher depending on the efficiency of the GPU. Thisswift processing makes it highly suitable for deployment in real-world applications where real-time response is critical. Notably, the model showed excellent performance in identifying concealed weapons and distinguishing them fromnon-threatening objects, reducing the risk of false alarms andensuring reliable detection.

To validate the adaptability of our model for web deployment and hardware integration, the successfully implemented it asan API service, enabling users to access weapon detection functionalities easily. Additionally, model was made in sucha way that it can be easily integrated into a hardware system with a camera module, showcasing its potential for seamless integration into security setups and autonomous surveillancesystems.

Following are the images which depict the model's capabilityto accurately identify and locate various types of weapons within complex and cluttered scenes. The model effectively highlights instances of firearms, knives, and other potentially dangerous objects, demonstrating its utility in enhancing public safety and security measures. The result images below show the model’s robustness across different lighting conditions, angles, and perspectives, solidifying its potential as a valuable tool for real-world weapon detection applications.

Fig 2. Validation Image generated by the model

Fig 3(a). Result on test data

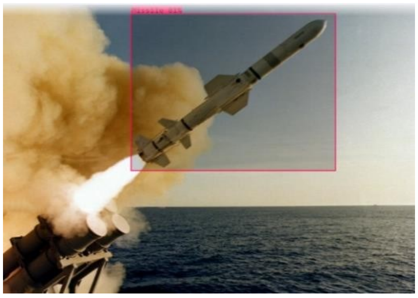

Fig 3(b). Result on test data

Fig 4(a). Real-time testing of rifle for the present model

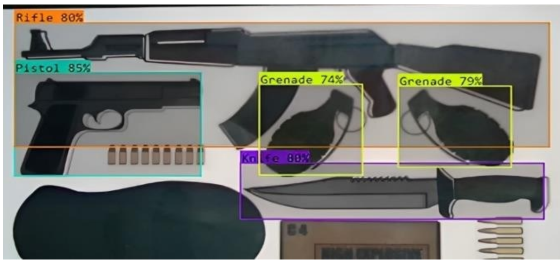

Fig 4(b). Real-time testing of firearms for the present model

VII. CONCLUSION

Weapon detection using the YOLO object detection framework has shown promise in automating the detection and classification of weapons in real-time. The YOLO architecture, with its multi-scale feature fusion and anchor boxes, enables accurate localization of weapons in images or video frames. While YOLOv8 is hypothetical, advancements in the YOLO framework may lead to further improvements in weapon detection accuracy and efficiency. It is important to stay updated with the latest research in the field to explorethe most recent advancements in weapon detection using the YOLO framework. In conclusion, the research presents a robust and accurate YOLOv8-based weapon detection model, showcasing its efficacy for real-time deployment on the web via API and its suitability for hardware integration. With its remarkable accuracy and real-time performance, the model can significantly contribute to enhancing public safety and security in various domains.

REFERENCES

[1] P. Shanmugapriya, Gurram Yugandhar Reddy, Jammalamadaka Mahendra Kumar, "Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications", IRJET 2022

[2] Dr. N. Geetha, Akash Kumar. K. S, Akshita. B. P, Arjun. M Coimbatore Institute of Technology, “Weapon Detection in Surveillance System”, IJERT 2021.K. Elissa, “Title of paper if known,” unpublished.

[3] Harsh Jain, Aditya Vikram, Mohana, Ankit Kashyap, Ayush Jain Telecommunication Engineering, RV College of Engineering® Bengaluru, Karnataka, India. “Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications”, ICESC 2020

[4] Sanam Narejo, Bishwajeet Pandey, Doris Esenarro vargas, Ciro Rodriguez and M. Rizwan Anjum, “Weapon Detection Using YOLO V3 for Smart Surveillance System”, Recent Trends in Advanced Robotic Systems 2021.

[5] Haitong Lou1 , Xuehu Duan1 , Junmei Guo1 , Haiying Liu1 *, Jason Gu2 , Lingyun Bi1 , Haonan Chen “DC-YOLOv8: Small size Object detection algorithm based on camera sensor”, 2023.

[6] Alan Agurto Yong Li Gui Yun Tian Nick Bowring Stephen Lockwood “A Review of Concealed Weapon Detection and Research in Perspective”.

[7] Milind Rane, Manas Jain, Aryan Kashyap, Adhip Jajoo, Harshvardhan Kadam, Devika Kadam, Vishwakarma Institute of Technology, Pune, “Mine Detecting Military Bot using IoT”, ESCI 2023

[8] Krizhevsky, A., I. Sutskever, and G.E. Hinton, “ ImageNet classification with deep convolutional neural networks”, in Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1. 2012, Curran Associates Inc.: Lake Tahoe, Nevada.

[9] Wang, C., K. Huang, “How to use Bag-of-Words model better for image classification”. Image and Vision Computing, 2015.

[10] Okafor, E., “Comparative study between deep learning and bag of visual words for wild-animal recognition”, 2016.

[11] Turaga, “Machine recognition of human activities: A survey”. IEEE Transactions on Circuits and Systems for Video technology, 2008.

[12] Zhang, Y., H. Wang, and F. Xu. “Object detection and recognition of intelligent service robot based on deep learning” , IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), 2017.

[13] Bhatnagar, “An Ensemble of Deep Learning and Feature Based Models for Financial Sentiment Analysis”, 2017

[14] Krizhevsky, A., I. Sutskever, and G.E. Hinton. “ Imagenet classification with deep convolutional neural networks”. in Advances in neural information processing systems, 2012.

[15] Martinez-Martin, “Object Detection and Recognition for Assistive Robots: Experimentation and Implementation”. IEEE Robotics & Automation Magazine, 2017