WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Intelligent Weapon Detection System for Real Time Surveillance using Deep Learning with YOLOv8

Prof. Abhale B. A[1], Mr.Thorat Sagar M[2], Mr. Pawar Nilesh S[3], Mr. Sathe Rushikesh L[4]

[1 2 3 4] S.N.D. College of engineering & research Centre Savitribai Phule Pune University

ABSTRACT:

Considering a growing number of criminal acts, there is an urgent need to introduce computerized command systems in security forces. This study presents a novel deep learning model specifically developed for identifying seven different categories of weapons. The suggested model utilizes the VGGNet architecture and is implemented utilizing the Keras architecture, which is built on top of the TensorFlow framework. The model is trained to recognize several types of weapons, including assault rifles, bazookas, grenades, hunting rifles, knives, handguns, and revolvers. The training procedure involves creating layers, executing processes, saving training data, determining success rates, and testing the model. A customized dataset, consisting of seven different weapon categories, has been meticulously chosen and organized to support the training of the proposed model network. We do a comparative study using the newly created dataset, specifically comparing it with established models such as VGG-16, ResNet-50, and ResNet-101. The suggested model exhibits exceptional classification accuracy, obtaining a remarkable 98.40%, outperforming the VGG-16 model (89.75% accuracy), ResNet-50 model (93.70% accuracy), and ResNet-101 model (83.33% accuracy). This research provides a vital viewpoint on the effectiveness of the suggested deep learning model in dealing with the complex problem of weapon classification, presenting encouraging outcomes that could greatly improve the capabilities of security forces in countering criminal activities.

Keywords: Deep learning, armed weapon detection, machine learning, object detection, convolutional neural networks

INTRODUCTION

In an era where security threats are increasingly sophisticated, the need for advanced surveillance systems is more critical than ever. Weapon detection systems, powered by cutting-edge technologies such as deep learning and computer vision, have emerged as vital tools in enhancing public and private safety. These systems aim to automatically identify and respond to the presence of dangerous objects, such as firearms and knives, thereby preventing potential incidents and mitigating risks.

The weapon detection system discussed in this project leverages the latest advancements in deep learning, specifically employing the YOLOv8 (You Only Look Once) architecture. YOLOv8 is renowned for its efficiency and accuracy in object detection tasks, making it an ideal choice for real- time applications. By integrating this powerful model into a versatile and user-friendly framework, the system provides a comprehensive solution for detecting weapons across various input types, including images, videos, and live webcam feeds.

Developed using Python, with a front-end interface built on HTML, CSS, and JavaScript, and utilizing the Flask web framework, this system is designed to be both robust and accessible. It addresses the limitations of previous systems that were restricted to static image analysis by offering multi-modal detection capabilities, thereby extending its applicability to dynamic and real-time security scenarios.

The project involves training the YOLOv8 model on a substantial dataset of approximately 4000 images, encompassing handguns and knives. This training process enables the model to learn and recognize these weapons with considerable accuracy. The system's overall accuracy of 64% reflects its proficiency in detecting weapons, highlighting its potential to enhance security measures effectively.

LITERATURESURVEY

In technologically advanced era, surveillance is a proven method for the monitoring of the individual's activity in the crowd. Security of infrastructure, as well as individual, is one of the major concerns because of the influential growth of radical elements or suspicious persons in the society. Continuous manual monitoring of the CCTV surveillance is difficult and monotonous task, so there is an urgent requirement to develop an automated surveillance systems. The security surveillance system has potential to detect any kind of concealed object (like firearms or any weapon including knife, scissors etc.) which may pose a threat to the security. In this paper, we propose a novel framework for the detection and classification of concealed weapons through analysis of CCTV stream data. The classification framework is developed with the categorization of various concealed weapons through deep learning based object detection and classification techniques. For the detection of concealed weapon, multi-sensor stream data capturing framework is designed using sensor fusion techniques and also embedded with the feature extraction and segmentation of images module. Faster R-CNN (Region-based

Convolutional Neural Network) model is trained for classification of weapons over collected dataset. Finally, several directions of work and tasks are provided as future work for the various research communities.

MATERIALS AND METHODS

Data Collection:

In the first module of Weapon Detection using Deep Learning, we developed the system to get the input dataset. Data collection process is the first real step towards the real development of a machine learning model, collecting data. This is a critical step that will cascade in how good the model will be, the more and better data that we get, the better our model will perform. There are several techniques to collect the data, like web scraping, manual interventions. Our dataset is placed in the project and it's located in the model folder. The dataset is referred from the popular standard dataset repository kaggle where all the researchers refer it. The following is the URL for the dataset referred from kaggle.

Dataset:

Organizing the data into train, val sets and Converting annotations to the format required by YOLOv8.
Annotations typically include bounding boxes around the objects of interest (helmets and number plates) and their corresponding classes. YOLOv8 requires annotations in a specific format, such as YOLO format (class, x_center, y_center, width, height) and Total dataset size is 4156 images 2 Label knife and Handgun.

Data Preparation:

Ensure that each set contains a representative sample of images to avoid overfitting. Convert annotations into the format required by YOLOv8, ensuring consistency between image paths and annotation paths.
Resizing: Resize the images to a consistent size to ensure the model can process them efficiently.
Normalization: Normalize the pixel values of the images to be within a specific range, usually between 0 and 1, to improve model performance.

Feature Extraction:

In YOLOv8, feature extraction is handled by the model architecture itself, specifically by its convolutional layers. These layers extract relevant features from the input images, enabling the model to detect objects such as helmets and number plates.
Use the YOLOv8 architecture, which is a state-of-the-art object detection model. YOLOv8 extracts features from the images using a backbone network and then predicts bounding boxes and class probabilities.

Splitting the dataset:

Divide your dataset into training and validation to evaluate your model's performance. Typically, you might use an 80- 20 split, but this can vary based on your dataset size and specific requirements.

Model Selection:

The training module is responsible for training the deep learning models using the preprocessed data. It implements YOLOv8 architectures
YOLOv8:
Selecting YOLOv8 as the object detection model for tasks like helmet and number plate detection is a strategic choice owing to its high accuracy and efficiency. YOLOv8, short for "You Only Look Once version 8," represents the latest iteration of the YOLO (You Only Look Once) family of object detection models, known for their real-time performance and strong detection capabilities.

Here's a deeper exploration of why YOLOv8 is a compelling choice:

State-of-the-art Performance: YOLOv8 builds upon the success of its predecessors and incorporates advancements in deep learning techniques and model architectures. It achieves state-of-the-art performance in terms of detection accuracy and speed, making it suitable for various real-world applications.

Efficiency: YOLOv8 is designed to be highly efficient, striking a balance between accuracy and computational resources. Its architecture optimizes the use of hardware resources, allowing for fast inference speeds even on devices with limited computational power. This efficiency makes YOLOv8 particularly appealing for deployment in resource- constrained environments or for applications requiring real-time processing.

Single-stage Detection: YOLOv8 follows the principle of single-stage detection, meaning it processes the entire image in a single forward pass through the network. This design choice eliminates the need for complex post- processing steps and significantly reduces inference time compared to multi-stage detection approaches.

Multi-scale Feature Fusion: YOLOv8 incorporates multi-scale feature fusion techniques, enabling it to capture context and spatial information at different scales within the image. This enhances the model's ability to detect objects of varying sizes and aspect ratios with high accuracy.

Flexibility and Customization: YOLOv8 offers flexibility and customization options to adapt the model to specific use cases and datasets. Users can choose from different model variants (e.g., YOLOv8-s, YOLOv8-m, YOLOv8-l) based on their requirements for speed and accuracy. Additionally, the model can be fine-tuned on custom datasets to further improve performance on specific tasks.

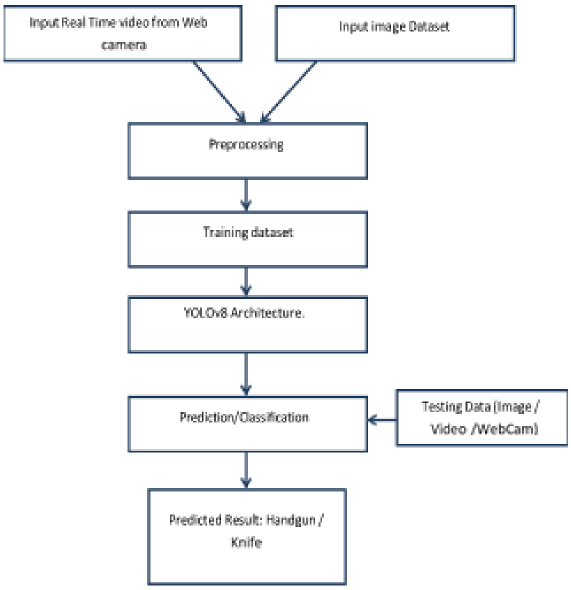
Open-source Implementation: YOLOv8 is often available as an open-source implementation, making it accessible to a wide range of developers and researchers. This fosters collaboration, experimentation, and innovation within the computer vision community.

Overall, the selection of YOLOv8 as the object detection model for tasks like helmet and number plate detection reflects its reputation for achieving high accuracy and efficiency. By leveraging the strengths of YOLOv8, developers can build robust and effective detection systems capable of meeting the demands of various real-world applications..

Training the Model:

Training loop: Train the YOLOv8 model using the training set. The model learns to predict bounding boxes and class probabilities for helmets and number plates.
Loss function: Use a loss function such as mean average precision (MAP) to measure the model's performance during training.
Optimizer: Use an optimizer such as stochastic gradient descent (SGD) or Adam to update the model's weights based on the loss.

Analyze and Prediction:
Model evaluation: Evaluate the model's performance on the validation set during training to monitor its progress.
Prediction: Use the trained model to predict bounding boxes and class probabilities for new, unseen images.



Accuracy on test set:

- After training and evaluating the model on the validation set, the accuracy of the model will be assessed on the test set. The accuracy on the test set will be an important metric for evaluating the model's performance. Evaluate the final accuracy of the model on the test set to ensure its effectiveness in real-world scenarios. Calculate metrics like mAP, precision, and recall on the test set to quantify the model's performance objectively.

Saving the Trained Model:

- Save the trained YOLOv8 model for future use. YOLOv8 models can be saved in a format that allows easy reloading for inference and deployment in production environments.

Prediction Module:

- Develop a prediction module that loads the saved YOLOv8 model and takes an input image or video stream. The module should be capable of processing the input data and outputting the detected knife and Handgun with bounding boxes, facilitating real-time detection.

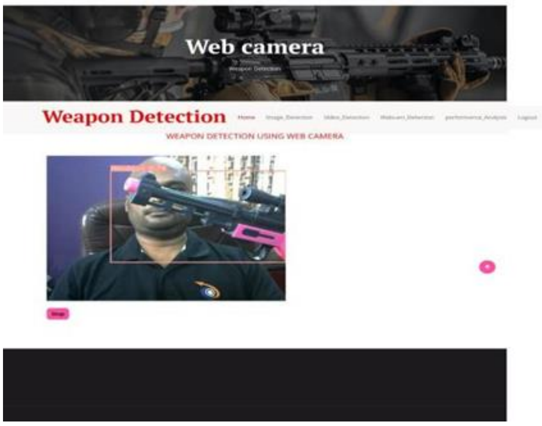
Model Evaluation Module

- This module evaluates the performance of the trained models using the testing dataset. It calculates accuracy metrics and other performance indicators to assess model effectiveness.
- Evaluate model accuracy, precision, recall, and F1-score.
- Generate confusion matrices for both models.
- Accuracy, precision, recall, and F1-score are used to evaluate model performance.

RESULTS AND DISCUSSION INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- ➢ What data should be given as input?
- ➢ How the data should be arranged or coded?
- ➢ The dialog to guide the operating personnel in providing input.
- ➢ Methods for preparing input validations and steps to follow when error occur.



OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

Select methods for presenting information.

Create document, report, or other formats that contain information produced by the system.
The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

Output Design for Weapon Detection Using Deep Learning

The output design for the "Weapon Detection using Deep Learning" project ensures that the results of the detection process are presented to the users in a clear, informative, and actionable manner. The design focuses on providing meaningful feedback for each of the three detection modes: image, video, and webcam. Below are the detailed components for each output mode:

Image Detection Output

Description: Displays the results of weapon detection for uploaded static images.
Output Method: Visual and textual feedback on the web interface.
Output Components:

- Annotated Image: The uploaded image with bounding boxes around detected weapons, highlighting the exact location of each detected object.
- Detection Summary: Textual information summarizing the detection results, including the type of weapon detected (e.g., handgun, knife) and confidence scores for each detection.

Video Detection Output

Description: Provides frame-by-frame detection results for uploaded video files.
Output Method: Visual and textual feedback on the web interface.
Output Components:

- Annotated Video Frames: Display the video with bounding boxes around detected weapons in each frame, enabling users to see where weapons are detected throughout the video.

Webcam Detection Output

Description: Real-time display of detection results from live webcam feeds.
Output Method: Visual feedback on the web interface.
Output Components:
Live Annotated Feed: Real-time video feed with bounding boxes around detected weapons, providing immediate visual feedback to users.

Front-End Output Design HTML Elements:

Annotated Image Display: An image element to display the uploaded image with bounding boxes.
Annotated Video Display: A video player element to display the video with annotated frames.

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

A feasibility study serves as a critical compass for organizations and decision-makers, guiding them through the initial stages of project planning by evaluating the viability and potential success of a proposed endeavor. This comprehensive analysis takes into account various factors to determine whether a project is worth pursuing from technical, financial, operational, and strategic perspectives.

CONCLUSION

The "Weapon Detection using Deep Learning" project marks a significant advancement in security technology, leveraging the power of the YOLOv8 architecture to deliver a versatile and robust solution for detecting dangerous objects. By expanding beyond static image analysis to include video and live webcam feeds, the system provides comprehensive real-time detection capabilities essential for modern security applications.
The project showcases a successful integration of advanced deep learning models with practical web technologies, achieving a commendable detection accuracy of 64%. This accuracy, combined with the system's rapid processing speed and efficient resource utilization, underscores its potential to enhance safety measures in various environments. The user-friendly interface, developed with HTML, CSS, JavaScript, and supported by the Flask web framework, ensures accessibility and ease of use for a wide range of users.
Overall, this project demonstrates the effective application of deep learning techniques in addressing critical security challenges, offering a powerful tool for weapon detection that can significantly contribute to the prevention and mitigation of threats, thereby promoting safer public and private spaces.

REFERENCES :

1. Raturi, G., Rani, P., Madan, S., & Dosanjh, S. (2019, November). ADoCW: An automated technique for identifying hidden weapons. In 2019 Fifth International Conference on Image Information Processing (ICIIP) (pp. 181-186). IEEE.
2. Bhagyalakshmi, P., Indhumathi, P., &Bhavadharini, L. R. (2019). Live video monitoring for automated identification of fire arms. International Journal of Trend in Scientific Research and Development (IJTSRD), 3(3).
3. Lim, J., Al Jobayer, M. I., Baskaran, V. M., Lim, J. M., Wong, K., & See, J. (2019, November). Utilizing deep neural networks to detect firearms in surveillance videos. In 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) (pp. 1998- 2002). IEEE.
4. Yuan, J., & Guo, C. (2018, June). An advanced neural network approach for identifying hazardous machinery. In 2018 Eighth International Conference on Information Science and Technology (ICIST) (pp. 159-164). IEEE.
5. Ilgin, F. Y. (2020). Utilizing copulas for cognitive radios, we implement energy-based spectrum sensing.. Bulletin of the Polish Academy of Sciences. Technical Sciences, 68(4), 829-834.
6. Navalgund, U. V., & Priyadharshini, K. (2018, December). Deep learning-based approach for detecting criminal intentions. In 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET) (pp. 1-6). IEEE.
7. Chandan, G., Jain, A., & Jain, H. (2018, July). Utilizing Deep Learning and OpenCV, this system enables real-time detection and tracking of objects. In 2018 International Conference on inventive research in computing applications (ICIRCA) (pp. 1305-1308). IEEE.
8. Deng, L., & Yu, D. (2014). Exploration of deep learning techniques and their practical uses. Foundations and trends® in signal processing, 7(3–4), 197-387.
9. Bengio, Y. (2009).Studying complex structures for artificial intelligence. Foundations and trends® in Machine Learning, 2(1), 1-127.
10. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.
11. Song, H. A., & Lee, S. Y. (2013). NMF is used to create a hierarchical representation.. In Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part I 20 (pp. 466-473). Springer Berlin Heidelberg.
12. Masood, S., Ahsan, U., Munawwar, F., Rizvi, D. R., & Ahmed, M. (2020). Image scene recognition using a convolutional neural network. Procedia Computer Science, 167, 1005-1012.

13. Masood, S., Ahsan, U., Munawwar, F., Rizvi, D. R., & Ahmed, M. (2020). Video scene recognition using a convolutional neural network. Procedia Computer Science, 167, 1005-1012.
14. Sai, B. K., & Sasikala, T. (2019, November). The task involves utilizing the Tensorflow object identification API to find and quantify things in a picture.. In 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 542-546). IEEE.
15. Verma, G. K., & Dhillon, A. (2017, November). A portable firearm detection system utilizing the faster region-based convolutional neural network (Faster R-CNN) deep learning algorithm.. In Proceedings of the 7th international conference on computer and communication technology (pp. 84-88).
16. Warsi, A., Abdullah, M., Husen, M. N., & Yahya, M. (2020, January). Review of algorithms for automatic detection of handguns and knives. In 2020 14th International Conference on Ubiquitous Information Management and Communication (IMCOM) (pp. 1-9). IEEE.
17. Olmos, R., Tabik, S., & Herrera, F. (2018). Utilizing deep learning to recognize automatic handguns in videos and trigger an alarm. Neurocomputing, 275, 66-72.
18. Asnani, S., Ahmed, A., &Manjotho, A. A. (2014). A bank security system is developed utilizing a method called Weapon Detection using Histogram of Oriented Gradients (HOG) Features.. Asian Journal of Engineering, Sciences & Technology, 4(1).
19. Lai, J., & Maples, S. (2017). Creating a classifier that can detect guns in real-time. Course: CS231n, Stanford University.
20. Simonyan, K., & Zisserman, A. (2015). Deep convolutional networks for large-scale image recognition with significant depth. In: Proceedings of International Conference on Learning Representations.
21. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Image recognition using deep residual learning.In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
22. Olmos, R., Tabik, S., Lamas, A., Perez- Hernandez, F., & Herrera, F. (2019). An strategy for decreasing false positives in handgun identification with deep learning by fusing binocular images. Information Fusion, 49, 271-280.