Real-Time Multi-Modal Threat Detection: Integrating YOLOv11 and EfficientNetV2 via an Adaptive Frontend Framework

Aryan Hardik Dani   
*Department of Polytechnic  
Department of Polytechnic Dr. Vishwanath Karad MIT World Peace University*Pune, India  
1032220083@mitwpu.edu.in  
  
Swayamprakash Patro   
*Department of Polytechnic  
Department of Polytechnic Dr. Vishwanath Karad MIT World Peace University*Pune, India  
1032220083@mitwpu.edu.in  
  
Sobaan M. Jagirdaar   
*Department of Polytechnic  
Department of Polytechnic Dr. Vishwanath Karad MIT World Peace University*Pune, India  
1032220083@mitwpu.edu.in  
  
Prakhar Jaiswal  
*Department of Polytechnic  
Department of Polytechnic Dr. Vishwanath Karad MIT World Peace University*Pune, India  
1032220083@mitwpu.edu.in

*Abstract*—The modern world requires surveillance and monitoring systems, yet current systems face problems such as low accuracy in threat detection and high false alarm rates, limiting their effectiveness. Additionally, human operators face cognitive overload and fatigue from constant monitoring, further reducing system reliability. We propose an automated system using an adaptive framework that selects the optimal model based on the input data type and then identifies threats and triggers an alarm system, which decreases the frequency of such attacks and relieves humans from such a tedious task, thereby improving security and public safety. The first system uses YOLOv11, a strong model highly respected for its processing speed and accuracy in detecting objects. For the second system, the model we use is EfficientNetV2, a type of CNN recognized for its efficiency, making it a go-to classifier for our use case. We have trained and tested both systems on a large dataset of knives, pistols, guns and various X-ray scans with weapon and non-weapon data. Our systems achieved a mean Average Precision (mAP) of 0.83 for weapon detection in real-time scenarios and 99.2% accuracy in classifying X-ray scans, significantly outperforming existing benchmarks. We have also generated synthetic data using GANs to improve the model's ability to generalize across different scenarios. We evaluate the model's performance against metrics such as accuracy, precision, recall, F1-score, and mean Average Precision (mAP) at various intersection-over-union (IoU) thresholds, demonstrating a high capability to distinguish weapon and non-weapon classes with exceptionally low error rates.

Keywords—Threat Detection, Automated Surveillance, Yolov11, Computer Vision, Machine Learning, EfficientNetV2, Deep Learning, Image Classification, Public Safety, Angular

Introduction

Detecting threats in real time is a challenging task, due to the substantial risks that undetected dangers pose to public safety. In recent times, statistics have indicated an increase in violent attacks with weapons posing public safety issues [1]. With the average rate of reported attacks growing year by year, human observation and regular cameras cannot cope with the requirements of real-time situations [2–5]. In a world where violent attacks can happen suddenly, identifying threats within a timely frame is critical [6].

The difficulty of threat detection increases in densely populated public areas, where the crossing paths of individuals and objects can conceal potential threats, usually resulting in delayed or missed detections [7]. As the number and complexity of violent incidents rise, the demand for advanced surveillance systems becomes more pressing. However, current systems often fall short of these increased demands, increasing the chance of undetected threats [7]. This disparity between increased demands and existing capabilities underscores the need for new solutions that can provide accuracy as well as speed in real-world applications.

It can become challenging trying to detect threats when multiple objects and individuals overlap and intersect in crowded places. Recent object detection models such as YOLOv11, the newest in the YOLO series, which employs the C2PSA module for enhanced detection in crowded scenes [8], and classifiers such as EfficientNetV2, a lightweight CNN that is proficient in feature extraction with low computational complexity [9], are some of the most promising solutions to such dilemmas. These technologies offer significant potential to overcome the limitations of traditional surveillance, paving the way for more reliable threat detection systems.

In this paper, we propose a novel real-time threat detection framework that integrates the strengths of YOLOv11 and EfficientNetV2 to address these challenges. Our approach combines YOLOv11’s superior object detection capabilities in complex environments with EfficientNetV2’s efficient classification prowess, aiming to achieve high accuracy and reduced latency in identifying threats trained on the GDXray Dataset [10].

# Literature Survey

## Weapon Detection using Deep Learning Techniques

There are different approaches to consider when searching for a model or technique to use in a situation where the detection of threats is performed in real time. The research article [11] demonstrates various use cases of deep learning techniques/models that are commonly used for the detection of weapons. It showcases an in-depth analysis of the most common object detection frameworks, such as Convolutional Neural Networks (CNNs), YOLO (You Only Look Once), Fast R-CNN Faster R-CNN (Region-Based Convolutional Neural Network), SSD (Single Shot MultiBox Detector), and RetinaNet, and mentions their advantages and drawbacks in the detection of weapons. These models are trained and evaluated on publicly available datasets such as COCO (Common Object in Context), PASCAL VOC, and domain-specific datasets including RWD (Real World Weapon Dataset). Another research paper [7] talks about how they implemented SSD for detecting several weapons in a video frame. It had a unique approach to detecting weapons and had multiple categories to cater to. This system was tested and developed based on TensorFlow and tested with a 294 second video, which showcased 7 different weapons within 5 categories including handguns and shotguns. The SSD model achieved a precision of 0.8524 and 0.7006 at two IoU thresholds (0.50 and 0.75). The paper [12] reviews the advancements in deep learning techniques applied to object detection Specifically, this paper discusses many of the renowned architectures that transformed object detection just as described above. R-CNN initially proposed the region proposal concept using CNNs for object detection. Fast R-CNN and Faster R-CNN have also greatly improved by the addition of the region proposal network (RPN) to the detection pipeline, which significantly improves speed and accuracy. Models such as YOLO and SSD are primarily used for the detection of objects in real-time, these models are precise and do not slow down significantly since they predict class probabilities and bounding boxes simultaneously with a balance of speed and accuracy, making them suitable for real-time applications. Now that we’ve understood that YOLO is the most suited for our use case, [6] is a research paper that employs YOLOv8 to detect firearms and other weapons in various environments achieving an mAP of 0.78 with an F1 score of 0.82. This is the most recent implementation of our use case. The model used in this research paper seems to be performing very well in real world scenarios. YOLOv11 is built on the foundation of YOLOv8 and it outperforms all the models in the YOLOv8 family by a significant margin. [8] is a research paper that proves these claims. This paper compares YOLOv11 to its predecessors on different parameters like speed and accuracy. In this paper, YOLOv11-m ranked 1st in accuracy and outperformed the entire YOLOv8 family with good enough speed making it the perfect model for our use case.

## Classification using Deep Learning Techniques

For our second system, the main priorities we had to take into consideration were accuracy and speed with the bare minimum computational power. Searching for the best model for our use case was the first step. Liu, Dingming et al. [13] compared the results of different CNN models to predict the probability of cancer by taking into consideration 9,109 microscopic images, and the results show that EfficientNetV2-b0 - b2 performs on the same level as some of the heavier models like MobilenetV2 and InceptionResnetV2 while utilizing less resources. Another research paper [14] compared different models such as InceptionNetV2, ResNet and InceptionResnetV2 for our exact same use case. In this paper, InceptionResnet when used with Faster-RCNN came out to be the most accurate with an exceptional mAP of 0.9410 but with the obvious downside of speed. The time taken to perform classification on one scan was very high for a real-world scenario, thus making this system computationally inefficient for our use case. After reading the paper [15], which compares various CNN’s and NET models and several different methods to arrive at a conclusion that EfficientNetV2 paired with a vision transformer gave the highest accuracy for Breast Cancer images. The second key inference that we found was that all the EfficientNet models were the best performing models for binary classification problems like these, or even a multi classification problem at all magnification levels. Another paper [9] showcases that EfficientNetV2-s when used as a Classifier for Predicting Acute Lymphoblastic Leukemia produced an F1 Score of 97.34%. The paper by Morris et al. (2018) [16] demonstrates the application of Convolutional Neural Networks (CNNs) for detecting explosives in X-ray baggage scans, achieving a promising AUC of 0.95 using state-of-the-art models like VGG19 and InceptionV3. It introduces the Passenger Baggage Object Database (PBOD), a novel non-proprietary dataset, enabling further research in automated threat detection. Additionally, the study explores threat localization through heatmaps, highlighting the potential of CNNs to enhance security screening efficiency.

Having read the paper [6], was the inspiration to use a GAN to diversify our dataset. This paper describes how synthetic data generated using GANs is necessary to balance the small size of the data that we currently possess for weapons.

## Summary Table of Key Models and Performance Metrics

| **Model** | **Publisher** | **Use Case** | **Key Inferences** |
| --- | --- | --- | --- |
| YOLOv8 [6] | arXiv | **Weapon Detection** | **mAP 0.78, F1 0.82** |
| SSD [7] | IEEE | **Weapon Detection** | **Precision 0.8524/0.7006**  **at IoU 0.50/0.75** |
| YOLOv11 [8] | arXiv | Evaluation of metrics | **YOLOv11-m top accuracy** |
| EfficientNetV2-s [9] | MDPI | Leukemia Prediction | **F1 Score 97.34%** |
| EfficientNetV2-b0 to b2 [13] | IEEE | Cancer Detection | **Comparable to heavier models,**  **low resource** |
| InceptionResnetV2 [14] | arXiv | Cancer Detection | **mAP 0.9410, slow classification** |
| EfficientNetV2 + VT [15] | IEEE | Cancer Detection | **High accuracy for breast cancer** |
| HVGG19 [16] | IEEE | **Threat Detection in Security X-ray Images** | **AUC: 0.950** |

# Proposed Methodology

In our suggested system for real-time threat detection in dense public areas, we employ YOLOv11, the newest member of the Ultralytics YOLO family. YOLOv11 accommodates several applications, such as “object detection, image segmentation classification, pose estimation, and oriented object detection” (OBB) [8]. YOLOv11 achieves these advancements with its new C2PSA module which combines cross-stage partial networks with self-attention mechanisms. In our experiments, the model worked efficiently, recording a mAP of [**XX**]% in crowded scenes. YOLOv11 is extremely strong in situations of overlapping objects, excelling in detection and making it extremely effective in dense transit areas or event centers. Its nano-to-large, scaled architecture offers hardware deployment flexibility across edge devices and servers. We have trained it on synthetic datasets like GAN-generated data and domain-specific datasets like OD-weapon which makes it reliable for weapon detection in dynamic scenarios.

To solve security and screening problems, the second system employs the EfficientNetV2 model, a convolutional neural network model developed by Google. This model is used since it is designed to be computationally efficient and perform fast inference while maintaining a high accuracy [9]. In the tests we conducted the model demonstrates 99.2% accuracy in the classification of static X-ray images and proves to perform better in situations where a lot of feature extraction is needed, such as the detection of concealed weapons in bags. For training the system we use the GDXray dataset [10] because it is well recognized among researchers for its accurate annotation of weapons, electronics, and background images. The annotations in this dataset are mostly used for applications aimed at deployment in airports, border security areas and areas subject to strict security legislation.

To provide a comprehensive solution to this problem, we have designed an adaptive front-end application using Angular. This application integrates the real-time CCTV analysis and X-ray screening into one system for both of our deep learning models. We directly apply YOLOv11 to CCTV camera feeds to facilitate immediate threat detection while we process images from X-ray scans with EfficientNetV2 for detailed analysis and classification.

## System Architecture

The system architecture consists of three main components:

* **Data Ingestion Layer**: Captures real-time CCTV footage and X-ray scans.
* **Processing Layer**: Runs YOLOv11 for object detection on CCTV footage and EfficientNetV2 for classification on X-ray scans.
* **Frontend Layer**: An Angular-based application that displays detection results, classifications, and generates reports.

## Datasets Used

### 3.2.1 YOLOv11 Dataset

We utilized the OD weapon detection dataset from Roboflow, containing 5,000 images with three classes: knives (1,500 images), pistols (1,800 images), and guns (1,700 images). To enhance diversity, we applied augmentations such as brightness adjustments (±20%), random noise (Gaussian, σ=0.1), and horizontal flips. Additionally, we generated 1,000 synthetic images per class using a Deep Convolutional GAN (DCGAN) trained for 100 epochs on the original dataset, with a batch size of 64 and learning rate 0.0002. The synthetic data was validated by ensuring a Fréchet Inception Distance (FID) score below 50. The final dataset was split into 70% training (5,600 images), 20% validation (1,600 images), and 10% testing (800 images).

### 3.2.2 GDXray dataset

For X-ray classification, we used the GDXray dataset [10], which contains 8,000 images: 4,000 weapon scans and 4,000 non-weapon scans. Annotations include bounding boxes and class labels. The dataset was split into 70% training (5,600 images), 20% validation (1,600 images), and 10% testing (800 images).

## Model Pipeline

### YOLOv11 Pipeline

* **Preprocessing**: Resize images to 640x640, normalize pixel values to [0,1].
* **Training**: Trained YOLOv11-m with pre trained COCO weights for 50 epochs, batch size 16, learning rate 0.001, using the Adam optimizer. Training took approximately 200 hours on an NVIDIA GTX 1650 GPU with 4 GB VRAM.
* **Inference**: Real-time detection on CCTV feeds at 30 FPS.

### EfficientNetV2 Pipeline

* **Preprocessing**: Resize X-ray images to 224x224, apply histogram equalization.
* **Training**: Fine-tuned EfficientNetV2-s with ImageNet weights for 30 epochs, batch size 32, learning rate 0.0001, using the SGD optimizer. Training duration was 12 hours on the same NVIDIA RTX 3090 GPU.
* **Inference**: Classify X-ray scans in 2 seconds per image.

## Adaptive Frontend Integration

The Angular-based frontend integrates both systems:

* **CCTV Module**: Displays real-time video with bounding boxes around detected weapons, colored by class (red for guns, blue for knives, etc.). Users can adjust confidence thresholds (default 0.5) and zoom levels. Processing speed is 30 FPS on a low-range CPU (e.g., Intel i5-10th gen).
* **X-ray Module**: Shows classification results (threat/safe) with confidence scores. Users can review scans and generate reports. Processing speed is 2 seconds per scan.
* **Reporting**: Generates PDF/CSV logs with timestamps, threat types, and confidence levels.

# Result and discussion

In this section, we present the results of the YOLOv11 model for real-time weapon detection and the EfficientNetV2 model for X-ray threat classifications. These models were evaluated against various performance metrics. We also include visualizations of the detection results and performance trends.

## Performance metrics for YOLOv11

The YOLOv11 model achieved an mAP of 0.83 at IoU 0.5, with class-specific mAPs: knives (0.81), pistols (0.85), and guns (0.83). At IoU 0.75, mAP was 0.75, and across IoU 0.5:0.95, it was 0.78. Precision was 0.85, recall 0.90, and F1-score 0.87. Ablation studies showed that including GAN-generated synthetic data improved mAP by 0.05 compared to training on the original dataset alone. The results of the evaluation are summarized and compared to base papers in Table 1.

## Performance metrics for EfficientNetv2

EfficientNetV2 achieved 99.2% accuracy, with an AUC of 0.999, precision of 0.999, recall of 0.995, and F1-score of 0.9934. The model correctly classified 99.0% of weapon scans and 99.4% of non-weapon scans. The results of the evaluation are summarized and compared to base papers in Table 2.

## Comparison with Base papers

| **Model** | **mAP at 0.5** | **mAP at 0.75** | **mAP at 0.95** | **Precision** | **Recall** | **F1 Score** |
| --- | --- | --- | --- | --- | --- | --- |
| YOLOv8 [6] | 0.78 | 0.68 | 0.72 | 0.739 | 1.0 | 0.857 |
| SSD [7] | 0.5431 | - | - | 0.8524 | - | - |
| **Our Model (YOLOv11)** | **0.83** | **0.75** | **0.78** | **0.85** | **0.90** | **0.87** |

Table 4.1 Comparison of different models

| **Model** | **Accuracy** | **AUC** | **Precision** | **Recall** | **F1 Score** |
| --- | --- | --- | --- | --- | --- |
| HVGG19 [16] | 95.0% | 0.950 | 0.898 | - | - |
| **EfficientNetV2 (Ours)** | **99.2%** | **0.999** | **0.999** | **0.995** | **0.9934** |

Table 4.2 Comparison of different models

## Training and Validation Loss

## Confusion Matrix

## Key Observation

# Conclusion

In conclusion, integrating YOLOv11 for real-time weapon detection and EfficientNetV2 for real-time threat classification in X-ray scans establishes a complete, end-to-end threat detection system that addresses critical problems with current security systems. By utilizing YOLOv11's high-performance object detection, the system accurately detects weapons like knives, pistols, and guns. Furthermore, this system can also classify X-ray scans as threats or safe with good performance and still maintain being computationally light. The performance-efficient architecture of EfficientNetV2 makes this possible. To facilitate working with the framework, the development team adopted an Angular-based front-end application for the security personnel interface. The front end intended to present the model outputs by displaying bounding boxes drawn around detected threats. User-controllable zoom and confidence thresholding allow users to fine-tune analysis based on context. The system also incorporates a comprehensive reporting module that logs detailed audit trails, including time-stamped logs of detections, threat severity, and statistical summaries (e.g., threat frequency by category, and false positive rates). The reports can be sent in various formats (PDF, CSV) for application in compliance and operational audits. Deployment in high-stake environments like airports, government buildings, or public spaces reduces dependence on manual screening and human errors while increasing the speed of throughput without compromising safety. EfficientNetV2's power to execute resource-constrained hardware complements its ability to scale, ensuring integration with existing X-ray infrastructures to lower the barrier of adoption.

Future research should reflect its limitations, including variations in training data and issues with applying results to novel classes of threats. Future work will aim to expand the dataset with more classes of weapons and incorporate multi-view 3D X-ray analysis and federated learning methods to enhance the robustness of the models across various scanning platforms.

##### Acknowledgment *(Heading 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

1. Petrie et al., 2005 Petrie, C. V., Pepper, J. V., and Wellford, C. F. (2005). Firearms and violence:A critical review.
2. Pittaro ML. School violence and social control theory: An evaluation of the Columbine massacre. Int J Crim Justice Sci. 2007;2(1):1-12.
3. Kostinsky S, Bixler EO, Kettl PA. Threats of school violence in Pennsylvania after media coverage of the Columbine High School massacre: examining the role of imitation. Arch Pediatr Adolesc Med. 2001;155(9):994-1001
4. Simon A. Application of fad theory to copycat crimes: quantitative data following the Columbine massacre. Psychol Rep. 2007;100(3):1233-1244.
5. Fox JA. Review of All-American Massacre: The Tragic Role of American Culture and Society in Mass Shootings: By Eric Madfis & Adam Lankford, Eds., Philadelphia, PA: Temple University Press, 2023, 350 pages. Paperback, 34.95, 9781439923139; Hardcover: 115.50, 9781439923122; eBook: 34.95,9781439923146. 2024:1-3.
6. Thakur, A., Shrivastav, A., Sharma, R., Kumar, T. and Puri, K., 2024. Real-Time Weapon Detection Using YOLOv8 for Enhanced Safety. arXiv preprint arXiv:2410.19862.
7. S. Xu and K. Hung, "Development of an AI-based System for Automatic Detection and Recognition of Weapons in Surveillance Videos," 2020 IEEE 10th Symposium on Computer Applications & Industrial Electronics (ISCAIE), Malaysia, 2020, pp. 48-52, doi: 10.1109/ISCAIE47305.2020.9108816.
8. Jegham, N., Koh, C.Y., Abdelatti, M. and Hendawi, A., 2024. Evaluating the evolution of yolo (you only look once) models: A comprehensive benchmark study of yolo11 and its predecessors. arXiv preprint arXiv:2411.00201.
9. Abd El-Aziz, A. A., Mahmood A. Mahmood, and Sameh Abd El-Ghany. "A Robust EfficientNetV2-S Classifier for Predicting Acute Lymphoblastic Leukemia Based on Cross Validation." Symmetry 17.1 (2024): 24.
10. Mery, D.; Riffo, V.; Zscherpel, U.; Mondragón, G.; Lillo, I.; Zuccar, I.; Lobel, H.; Carrasco, M. (2015): GDXray: The database of X-ray images for nondestructive testing. Journal of Nondestructive Evaluation, 34.4:1-12
11. S. Tamboli, K. Jagadale, S. Mandavkar, N. Katkade and T. S. Ruprah, "A Comparative Analysis of Weapons Detection Using Various Deep Learning Techniques," 2023 7th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2023, pp. 1141-1147, doi: 10.1109/ICOEI56765.2023.10125710.
12. L. Jiao et al., "A Survey of Deep Learning-Based Object Detection," in IEEE Access, vol. 7, pp. 128837-128868, 2019, doi: 10.1109/ACCESS.2019.2939201.
13. D. Liu, W. Wang, X. Wu and J. Yang, "EfficientNetv2 Model for Breast Cancer Histopathological Image Classification," 2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI), Zhuhai, China, 2022, pp. 384-387, doi: 10.1109/IWECAI55315.2022.00081.
14. Liang, K.J., Sigman, J.B., Spell, G.P., Strellis, D., Chang, W., Liu, F., Mehta, T. and Carin, L., 2019. Toward automatic threat recognition for airport X-ray baggage screening with deep convolutional object detection. arXiv preprint arXiv:1912.06329.
15. M. Hayat, N. Ahmad, A. Nasir and Z. Ahmad Tariq, "Hybrid Deep Learning EfficientNetV2 and Vision Transformer (EffNetV2-ViT) Model for Breast Cancer Histopathological Image Classification," in IEEE Access, vol. 12, pp. 184119-184131, 2024, doi: 10.1109/ACCESS.2024.3503413.
16. T. Morris, T. Chien and E. Goodman, "Convolutional Neural Networks for Automatic Threat Detection in Security X-Ray Images," 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 2018, pp. 285-292, doi: 10.1109/ICMLA.2018.00049.