International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Multimodal Weapon Detection and Forecasting Application for Visible and Thermal-Based Concealed Threats using RGB and Thermal Data

Mukundan CS[1], Panbarasan D[2], Naveen Kumar R[3], S. Ashok Kumar[4]

[1,2,3,4]Department of Computer Science and Engineering, Sri Venkateswara College of Engineering, Sriperambudur, Tamil Nadu, 602117 India.

Abstract

With growing needs for securing public and private spaces, a new trend of introducing better security systems has emerged. In particular, it gave rise to smart surveillance systems capable of detecting any dangers in real-time. For that reason, we propose the creation of a multimodal weapon detection and forecasting system. By using RGB (visible light) and thermal imagery, it will be able to perform detection tasks in challenging environments, like those of low lighting and/or partial visibility.

To detect weapons in real-time, two models using the YOLO (You Only Look Once) algorithm were implemented. One of them works with RGB data, and the other with thermal imagery. As a means to compensate for the disadvantages of using only one type of data, their outputs are compared, with the predictions being merged at the decision level. The prediction with the highest confidence score becomes final for the current frame.

As far as implementation goes, the system can process both live camera feed and image uploads from users. Temporal filtering is employed to eliminate false positives, which occur through comparing several consecutive detections. Upon detecting the presence of a weapon, the system sends out a signal that involves a sound alarm and an email notification containing the corresponding picture.

In addition to that, there is a module that allows the user to perform analytics and forecasting. Using historical data, this algorithm would identify patterns, calculate potential future threats, and determine peak activity periods. In this way, the detection system will transition into a forecasting system.

Thus, it can be concluded that a multimodal detection system that utilizes RGB and thermal imagery, along with simple analytics, can be quite efficient.

General Terms: Artificial Intelligence, Computer Vision, Machine Learning, Image Processing, Surveillance Systems, Security Systems

Keywords: Multimodal Weapon Detection, RGB and Thermal Imaging, YOLO, Computer Vision, Real-Time Surveillance, Decision-Level Fusion, Predictive Analytics

1. Introduction

Safety is one of the primary concerns in public places, such as airports, train stations, educational institutions, and malls. Despite the presence of surveillance cameras in all of these places, most of the time, monitoring is done manually, which causes some problems: delayed reaction, distraction after a while, as well as errors, which might cause ignoring critical threats.

With the help of Artificial Intelligence and Computer Vision, automated surveillance became popular during the last decade. It uses image recognition methods to detect some interesting things in videos and images in real-time mode. The most popular algorithm for solving this problem belongs to the YOLO class of algorithms (You Only Look Once). They offer a decent level of accuracy and speed.

Nevertheless, nowadays, there exist some limitations for such systems: most of them work only with RGB (red-green-blue) images, which means that there should be enough light to make a good prediction. Such devices cannot operate effectively in darkness and crowded scenes, which is not good news for the users who want to get the best quality of detection.

Thermal imaging might help to solve this problem. As opposed to standard cameras, these gadgets can detect heat signatures, not light. That means that these cameras will be able to operate effectively at night and under other poor lighting conditions. Nevertheless, there is one drawback: thermal cameras do not give any information about textures and colors.

Taking into account these limitations, it was decided to adopt a multimodal system that consists of RGB and thermal cameras. Both types of images will be processed with two YOLO models, and their results will be aggregated via decision-level fusion.

Besides, there are several additional functionalities implemented in the proposed system: real-time monitoring via the camera, detection from pictures, as well as generating alerts and processing data for the next forecast.

If a weapon is detected, it will be notified to the operator with the help of a siren and the message with the detected image will be sent to their email address. All detected threats will be saved for analysis purposes. Moreover, a special forecasting tool will be able to find some correlations based on historical data and show some forecasts concerning potential risks as well as peak activity moments.

2. Literature Review

2.1 Traditional vs. Machine Learning-based Weapon Detection

With the help of the development of technologies of image processing and machine learning, the evolution of surveillance systems was achieved through improvements in methods of object detection in an image or video sequence. Traditionally, weapon detection methods were based on manually developed criteria and characteristics like edges, shapes, and other visual properties. Although those solutions worked well in a controlled environment, their performance was limited in real life.

The methods that involved the use of machine learning were more advanced since they were based on classifiers that were trained using features detected in objects. Nevertheless, the performance of such methods was influenced by the presence of shadows, lighting changes, object occlusions, and, therefore, was not very high either.

2.2 RGB Model for Weapon Detection

At present, almost all weapon detection systems are based on an RGB model that analyses images captured in a visible range of frequencies and determines objects based on their shape, color, and structures. Since such an algorithm is simple enough and works with the most available cameras, it is extensively used by researchers nowadays.

Nevertheless, the main problem with RGB-based detection lies in the heavy dependence of results on environmental factors like low levels of illumination or cluttered backgrounds. In these circumstances, the algorithm may fail to detect some parts of the image, and the entire object will be missed.

2.3 Using the Thermal Model in the Detection Process

As it is clear, the RGB model cannot be used as a tool for detecting objects in low lighting. To overcome this problem, thermal detection was introduced into surveillance systems.

In such a way, objects can be detected based on heat signature rather than visual appearance, thus making detection possible under the mentioned adverse conditions. Despite the benefits of this solution, it does not have any information on the visual features of objects since thermal detection cannot recognize color, shape, etc.

2.4 Combining Different Detection Models and Data Fusion

In order to combine the advantages and eliminate the weaknesses of thermal and RGB models, multimodal detection algorithms were developed recently. Such algorithms integrate information from both the RGB model and thermal detection. The process of data fusion is important in multimodal systems as it allows achieving improved detection accuracy.

There are three types of fusion used in machine learning: early, feature-level, and decision-level fusion. At present, decision-level fusion is widely applied as it is relatively easy to implement and provides good results. In this case, the output predictions of both RGB and thermal models are analyzed, and the one with the highest confidence value is chosen.

2.5 Weaknesses of Current Weapon Detection Approaches

Although recent advances have made progress in this field quite noticeable, there are still many problems in modern detection systems. First, the majority of them work based on just one kind of data and, therefore, do not perform well under complex conditions. Besides, even the multimodal systems focus only on improving detection accuracy.

Such approaches have the following limitations:

- Lower performance in a real-life environment;
- Lack of effective data fusion methods;
- Poor detection of false positives;
- Insufficient alert mechanism;
- Lack of analytics and forecasting tools.

In addition, almost all modern detection systems ignore the benefits of working with the data accumulated during previous operations.

2.6 Research Gap & Proposed Solution

Accordingly to these issues, there is a need for creating an instrument that would incorporate detection, validation, and intelligent analysis procedures in the same software.

Thus, as a solution for overcoming these limitations, it is suggested to use the combination of thermal and RGB detection models that work in parallel and deliver their predictions in accordance with the most confident of the two decisions.

Moreover, this approach will include temporal validation to decrease the number of false positives, a real-time alert system, and an analytics and forecasting module.

3. Proposed System Architecture

The proposed system is a multimodal weapon detector and predictor that is capable of recognizing weapons in both visible (RGB images) and hidden (thermal images) states. It operates in two modes – live camera and static image upload, and consists of both front and back end, which are implemented via React

JS frontend and Python Flask backend correspondingly. As shown below, the architecture of the application is structured into six distinct layers – Input, Preprocessing, Multimodal Detection, Fusion and Decision, Alert and Response, and Intelligence and Analytics.

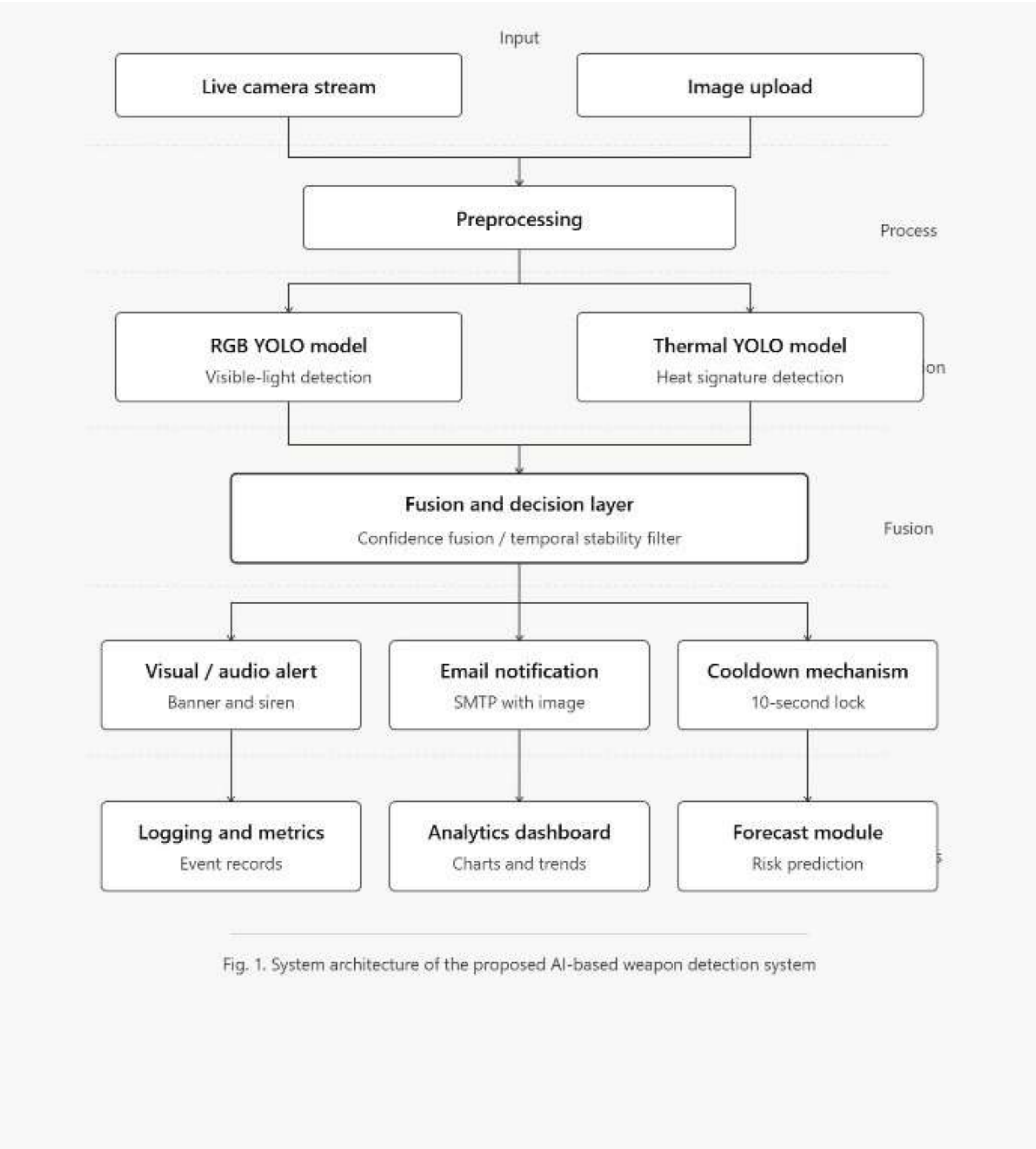


Figure 1: System Architecture Diagram

3.1 Architectural Overview

As per common practice, the system utilizes the standard client-server communication protocol. The frontend takes care of user experience, visualization of detections, and alerts. The backend processes all the heavy calculations including model inference, fusion logic, alert generation and dispatching, logging, and computations related to analytics and forecasting. Moreover, it serves as a data hub where records of detections are stored in relation to detection confidence, source type, and time. The communication between frontend and backend is performed via HTTP with the aid of Axios library on the frontend side.

3.2 Input Layer

There are two ways the system accepts input data.

3.2.1 Live Camera Stream

In live operation mode, the system receives the input stream of images from a connected webcam. In this case, frames represent a live view of the surveillance site in RGB or thermal representation depending on the specific camera used. The frames are passed further to the preprocessing without any additional modifications.

3.2.2 Static Image Upload

In addition, a static image can be uploaded by the user via frontend. Once it is sent to the backend, the image will be subjected to preprocessing as well and then processed by both detection models at the same time.

3.3 Preprocessing Layer

Prior to sending an image to a detection model, it needs to be prepared for processing. In this regard, both types of inputs are subject to resizing to 416 x 320 pixel dimensions. The resizing technique allows maintaining a consistent input size while significantly speeding up the inference process. In addition, in the live feed pipeline only every second frame is actually sent to the model to reduce the computational burden. As such, live input is not affected by the resizing.

3.4 Multimodal Detection Layer

As it was stated above, the detection pipeline consists of two YOLO models working with either RGB or thermal images respectively. Both of these models operate on the basis of YOLOv8 architecture and are fine-tuned to work with their specific input modality.

3.4.1 RGB YOLO Model

The RGB model is trained on standard visible light images containing pictures of weapons. The weapon is represented by either a gun or a knife, and in total there are two classes available. The model detects weapons by analyzing their shape, color, and texture patterns and produces a bounding box, the class label, and a confidence value for each detection. In live detection mode, detections having confidence less than 0.65 are filtered out, which decreases the false positive ratio. As such, the model is used in live and upload detection pipelines.

3.4.2 Thermal YOLO Model

In addition to RGB image-based detections, the system uses a thermal image detection model. Its main advantage is that it works with invisible heat signatures of weapons, which makes it suitable for detecting weapons under poor visibility conditions. In the upload detection pipeline, both models process the input images independently, and their results are further analyzed in the fusion step.

3.5 Fusion and Decision Layer

Once the predictions of the models were obtained, there is a need to select the appropriate one that would serve as the final detection. The selection strategy varies depending on the input modality used.

3.5.1 Decision-Level Confidence Fusion (Image Upload Mode)

While dealing with uploaded images, the system compares the highest prediction confidence of the RGB and thermal models and selects the winner of the two. Thus,

Final Prediction = argmax (Confidence_RGB, Confidence_Thermal)

Then, the image with annotations created by the selected model, corresponding class label, and confidence are considered as final predictions. Also, the source type is logged. If no weapons were found, the image is classified as a non-weapon image.

3.5.2 Temporal Stability Filtering (Live Camera Mode)

When detecting weapons in live feed, one detection per single frame is not sufficient to confirm the finding. For this reason, a special stability counter is used that increments upon successful detection and decrements when no weapon is found on the given frame. The prediction is regarded as confirmed only when the counter reaches 3. Then, it triggers an alert and displays the bounding box until it reaches five consecutive frames since the last successful detection. Such an approach allows for reducing the number of false positives triggered by temporary disturbances in the scene.

3.6 Alert and Response Layer

Upon detecting a weapon, three types of responses are initiated.

3.6.1 Visual and Audio Alert

First of all, an alert banner pops up on the frontend screen. Simultaneously, an audio siren sound starts to play. However, to avoid generating an alert repeatedly, a trigger is generated only once per detection event and does not repeat until the alert ends. For this purpose, the state of the alert flag is checked to see if the transition happened from off to on.

3.6.2 Email Notification

An automatic email notification is generated with the information regarding the weapon class, detection confidence, model type, and the time of detection. Also, an image with the annotation applied to it is appended to the message as an attachment. All emails sent as notifications are tied together using the same thread ID for easier management.

3.6.3 Cooldown Mechanism

To prevent multiple emails from being sent during one detection event, a ten-second cooldown timer is used to prevent re-sending alerts to the recipient.

3.7 Intelligence and Analytics Layer

Data received from detections is processed by two modules – analytics and forecast. The data flow is provided by the backend, and it includes the last 100 detection records consisting of timestamp, detected class, confidence, and detection model name.

3.7.1 Logging and Metrics

Logs are kept in reverse chronological order and added after every detection. Four types of data are derived from the logs, namely the total number of detected weapons, non-weapon frames, and RGB and thermal detections. Metric cards reflecting those metrics are displayed on the dashboard and are refreshed every second.

3.7.2 Analytics Dashboard

On the analytics dashboard, there are three graphs displayed in sequence. The first one represents the trend in confidence score values for both weapon types in the last 20 detections. The second graph is dedicated to the comparison of weapon detections and non-weapon detections. Finally, the third graph depicts the hourly activity level of the system.

3.7.3 Forecast Module

Using the weapon detection data, a forecast module is built to predict the expected number of weapons in the future, assign a risk level, and identify the peak threat hour. The number of expected detections is estimated as a proportion of detected weapons to total recorded events. Then, the risk level is assigned as LOW (<6), MEDIUM (6 – 15), or HIGH (>15). The peak hour with the largest number of detections is determined and reported as the peak threat hour.

3.8 System Architecture Summary

Table 1 provides a list of architectural layers of the system, their components, and their functions.

Table 1: Architectural Layers of the System

Layer	Component	Function
Input	Live Camera / Image Upload	Captures RGB or thermal input for processing
Preprocessing	Frame Resizer / Frame Sampler	Standardizes input size and reduces inference load
Multimodal Detection	RGB YOLO Model	Detects weapons from visual features
Multimodal Detection	Thermal YOLO Model	Detects weapons from heat signature features
Fusion and Decision	Confidence Fusion Engine	Picks higher-confidence model output for uploads
Fusion and Decision	Temporal Stability Filter	Confirms detection across multiple frames for live feed
Alert and Response	Visual / Audio Alert	Shows an alert banner and plays a siren on detection
Alert and Response	Email Notification	Sends image-attached alert to security personnel
Alert and Response	Cooldown Mechanism	Blocks repeated alerts for 10 seconds after each trigger
Intelligence	Logging and Metrics	Records events and computes detection count summaries
Intelligence	Analytics Dashboard	Shows confidence trends, detection distribution, and hourly activity

Intelligence	Forecast Module	Predicts risk level, expected count, and peak activity hour

4. Implementation

In this section, we detail the practical implementation of the multimodal weapon detection and forecasting system described above. The project is designed as a full-stack application that implements an end-to-end process of data acquisition, fusion of predictions, and triggering of appropriate actions such as alerts. This is achieved by employing the RGB model, thermal model, and fusion module in conjunction with other functionalities.

4.1 System Environment

The following technologies were used to implement the system:

- Backend: Python programming language; Flask framework
- Frontend: React JS programming language
- Computer Vision: OpenCV
- Machine Learning Models: YOLOv8
- Communication: REST API
- Visualization: Recharts

The backend is responsible for model inference and decision fusion, while the frontend provides a convenient visualization of the data and performs user interaction functions. The computer vision library (OpenCV) is employed to capture and process incoming frames or images, and machine learning models infer their content. Predictions obtained in this manner are fused at the decision level, and appropriate actions are undertaken depending on whether a weapon is detected. Finally, visualization libraries provide a way of representing the results of the analysis.

4.2 Model Integration

The system uses two different YOLOv8 models. Specifically, the RGB model infers content in frames captured from the visible light spectrum, whereas the thermal model infers content in thermal images. When integrating models into the system, these models are imported and loaded in the backend upon launch. The same frames/images should be supplied to both models. Results produced by both models are sent to the decision-making and alerting components of the system.

4.3 Real-Time Detection Pipeline

In live surveillance mode, the application constantly captures frames from the camera using the OpenCV library. Frame capturing happens in a continuous mode. However, due to resource limitations and the need for maintaining acceptable real-time performance, the system uses frame sampling – not all captured frames are analyzed and used. Instead, frames sampled at regular intervals are chosen for analysis.

Each frame undergoes preprocessing in the form of input resizing, after which inference using the RGB model and the thermal model takes place. The resulting bounding boxes, class labels, and confidence values are then sent to the fusion module.

4.4 Image Upload Processing

Offline mode allows analyzing already-captured images. In this case, the application enables users to select images using a frontend interface. These images are then sent to the backend for inference and decision-making.

The same procedure as in live mode is followed: both models analyze the same image separately and produce detection results that are later combined into a single output.

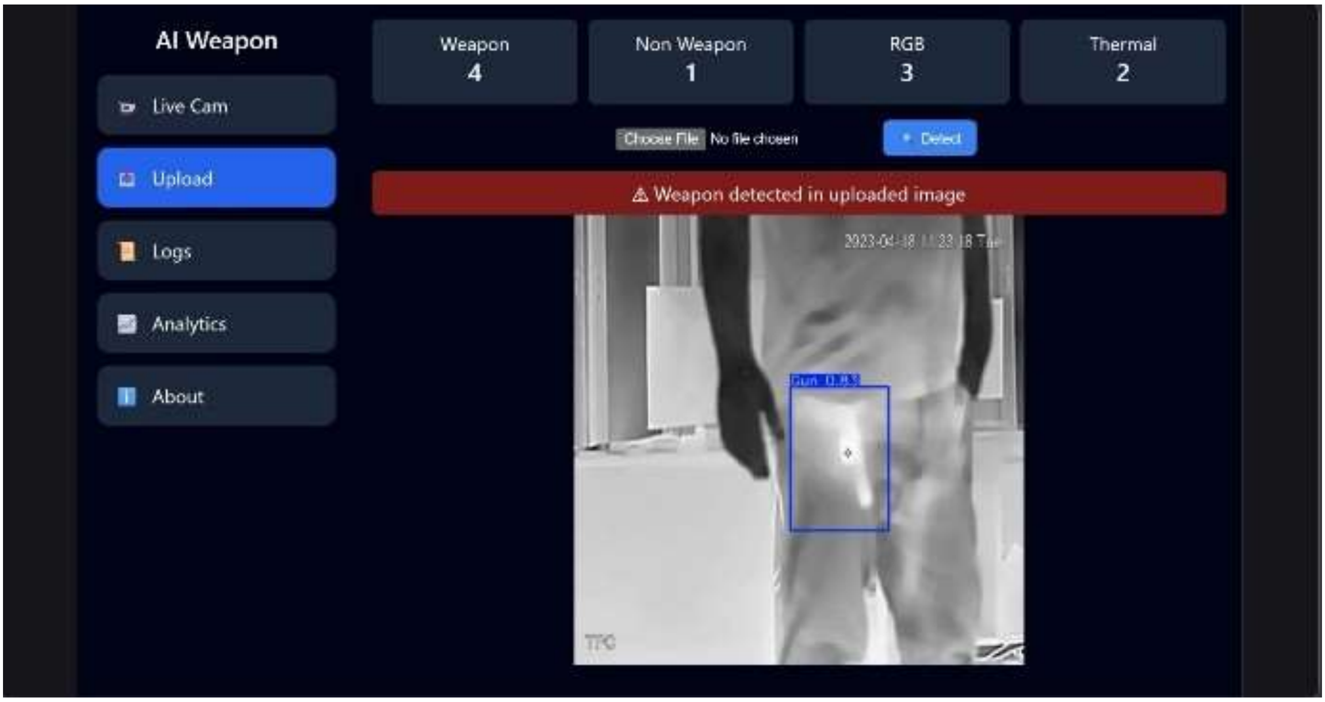


Figure 2: Image Upload Processing Interface

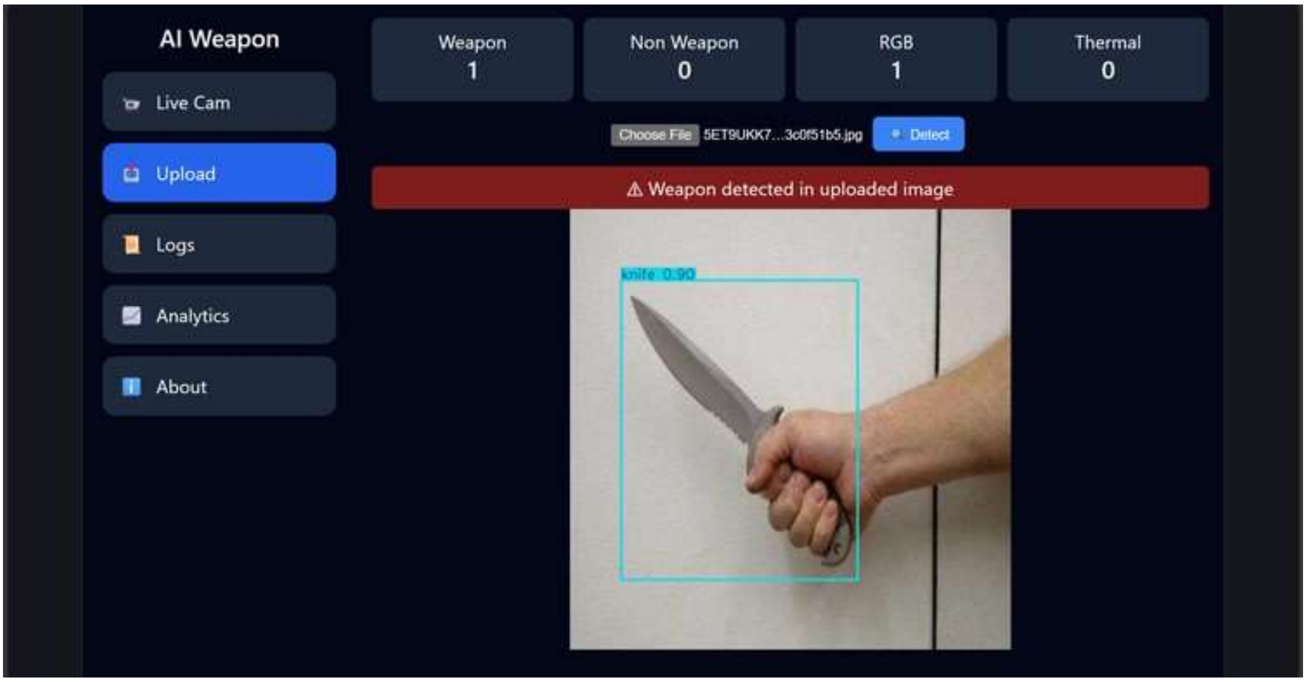


Figure 3: Upload Detection Results

4.5 Fusion Implementation

The application employs decision-level fusion. In particular, predictions are received from both models, and the one with the highest confidence value is chosen as the final one.

D = argmax(C_RGB, C_Thermal)

This procedure is implemented in the backend and does not require any extra features or steps to complete.

4.6 Temporal Validation

Temporal validation of detection is implemented to reduce the possibility of false alarms. Specifically, a certain detection must be seen in at least N number of subsequent frames.

$N \geq 3$

This improves the robustness of the solution and prevents alerts triggered by noise.

4.7 Alert Mechanism Implementation

If the detection is validated, then an alert must be raised through different channels. These are listed below:

- Audio Alert: play a siren sound using the frontend audio player functionality
- Visual Alert: show a message on the frontend screen that a weapon is detected
- Email Notification: send automatic emails with detected object information

More than one alert may be triggered for the same detection; therefore, cooldown measures are taken to prevent this.

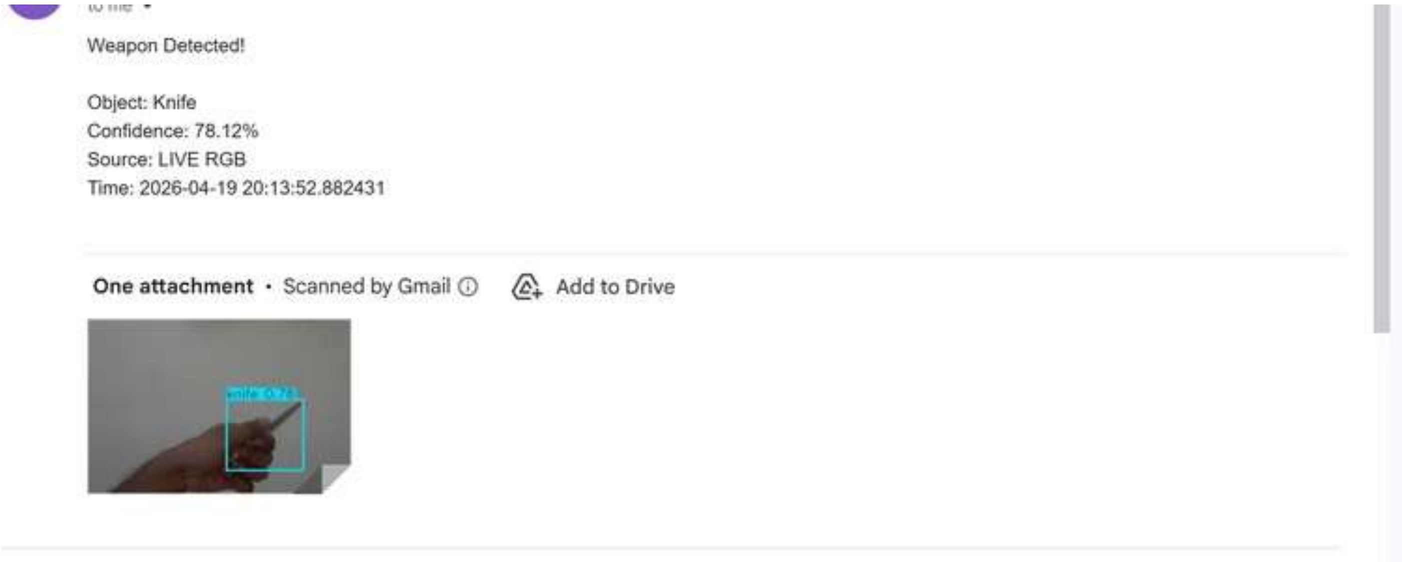


Figure 4: Alert Mechanism Interface

4.8 Data Logging and Storage

For every detection that is successfully made and confirmed, an entry in the detection log is created. Entries include the following attributes:

timestamp, object type, confidence value, model (RGB or thermal)

These records are kept in a rolling manner and can be used for further analysis.

Figure 5: Data Logging Interface

4.9 Analytics Dashboard Implementation

For better insight into detection results, a dashboard is implemented in the frontend. It shows charts that visualize detection information, such as the following:

- total number of weapon and non-weapon detections,
- model-wise detection distribution,
- confidence trends in recent detections,
- hourly detection activity

Figure 6: Analytics Dashboard

4.10 Forecasting Module Implementation

Using the data recorded in the detection log, the forecasting module calculates:

- the expected number of detections,
- risk levels (low, medium, high),
- peak activity periods.

These insights will help make proactive decisions.

4.11 System Performance

The application performs detections of frames in real-time mode. The current frame rate of the system is around 15-20 FPS, and the average latency is about 50-100 milliseconds per frame. This performance is provided by frame sampling and input resizing techniques.

4.12 Summary

The RGB model and thermal model are integrated into a system as part of the decision-making process. Additional modules were implemented to ensure the practicality and reliability of the system, thus making it applicable to real-world scenarios.

5. Results and Discussion

In this section, performance of the proposed weapon detection method will be analyzed. Three methods were analyzed: RGB model, thermal model, and proposed fusion method.

5.1 Experimental Setup

Test images and video streams were used for this evaluation purpose. Dataset was prepared in such a way to contain samples for both cameras with different types of weapons (guns, knives). A set of different testing conditions was provided: normal lighting, poor lighting, object occlusion etc. The performance criteria included the following:

- Precision;
- Recall;
- mAp-50;
- mAp-95;

All models were evaluated in equal conditions.

5.2 Quantitative Performance Evaluation

The following table presents the comparison of performance of all models.

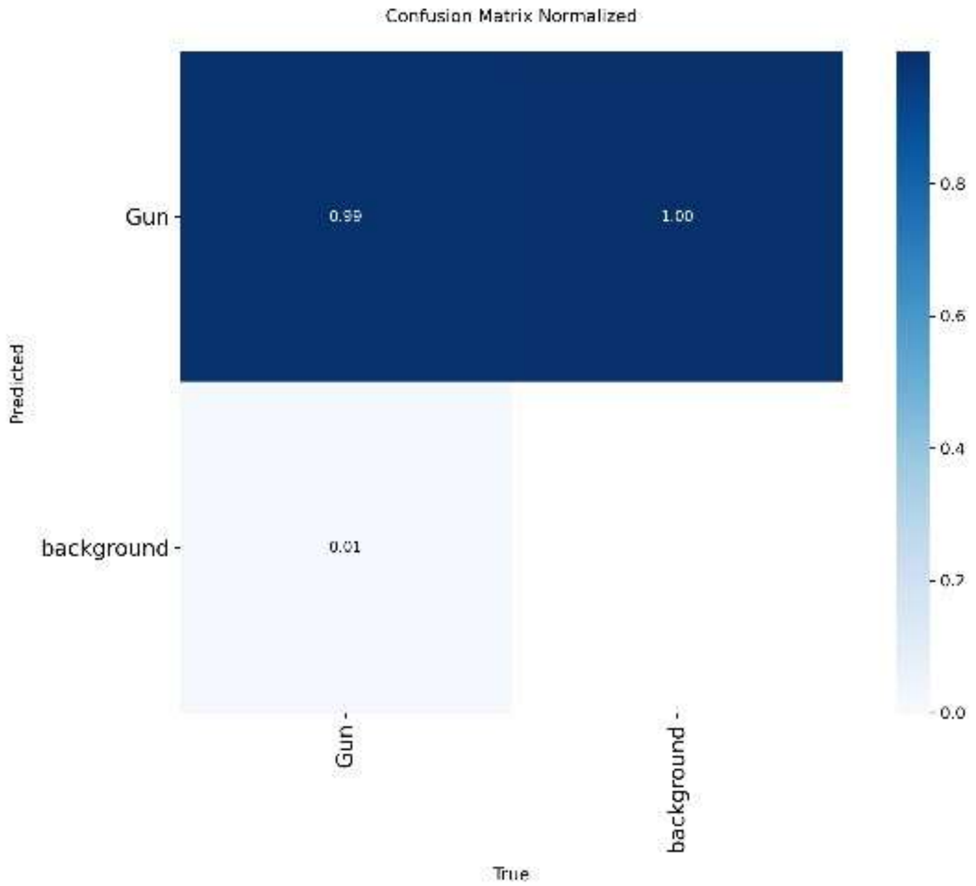
Table 2: Comparison of Performance of Models

Model	Precision (%)	Recall (%)	mAp-50 (%)	mAp-95
RGB Model	88.8	82.2	85.5	60.6
Thermal Model	99.3	99.4	99.3	62.1

As it can be seen from Table 2, the results obtained by the proposed method are superior to those produced by other models in all categories.

International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Thermal Model Results:



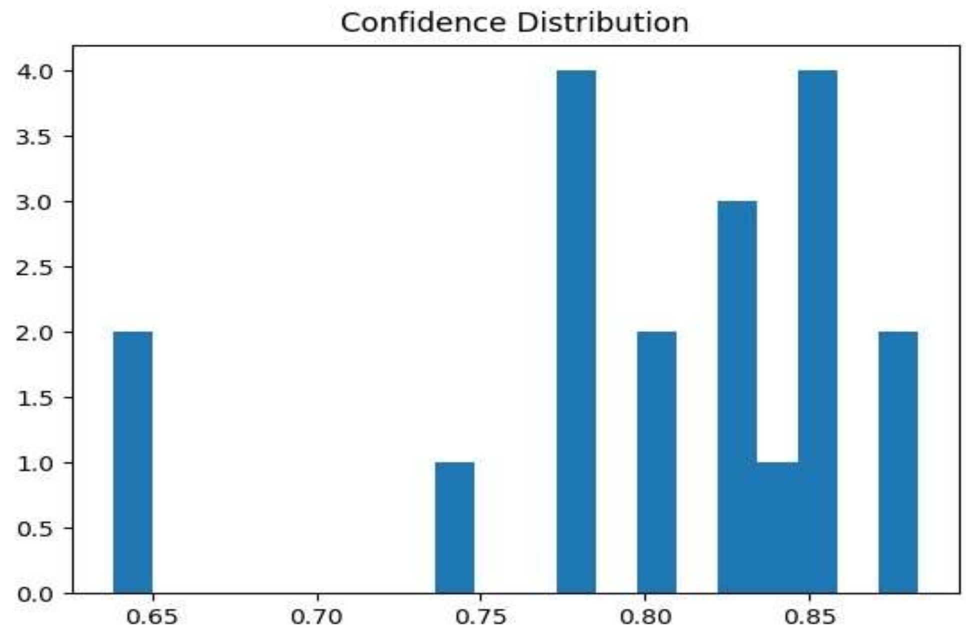


Figure 7: Thermal Model Detection Results

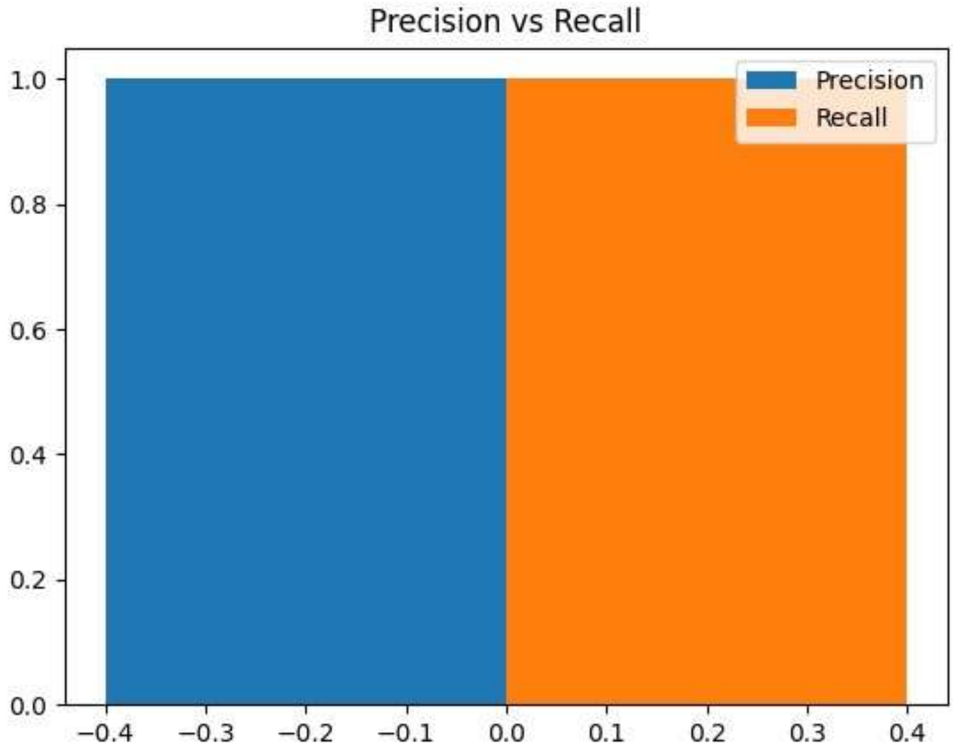
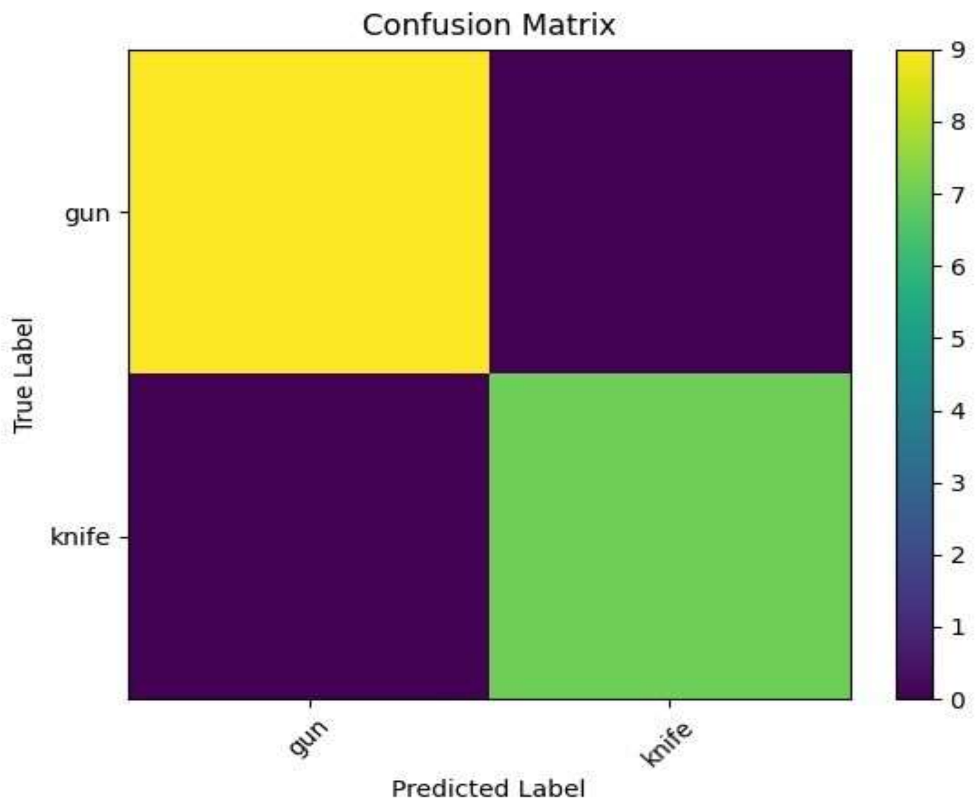
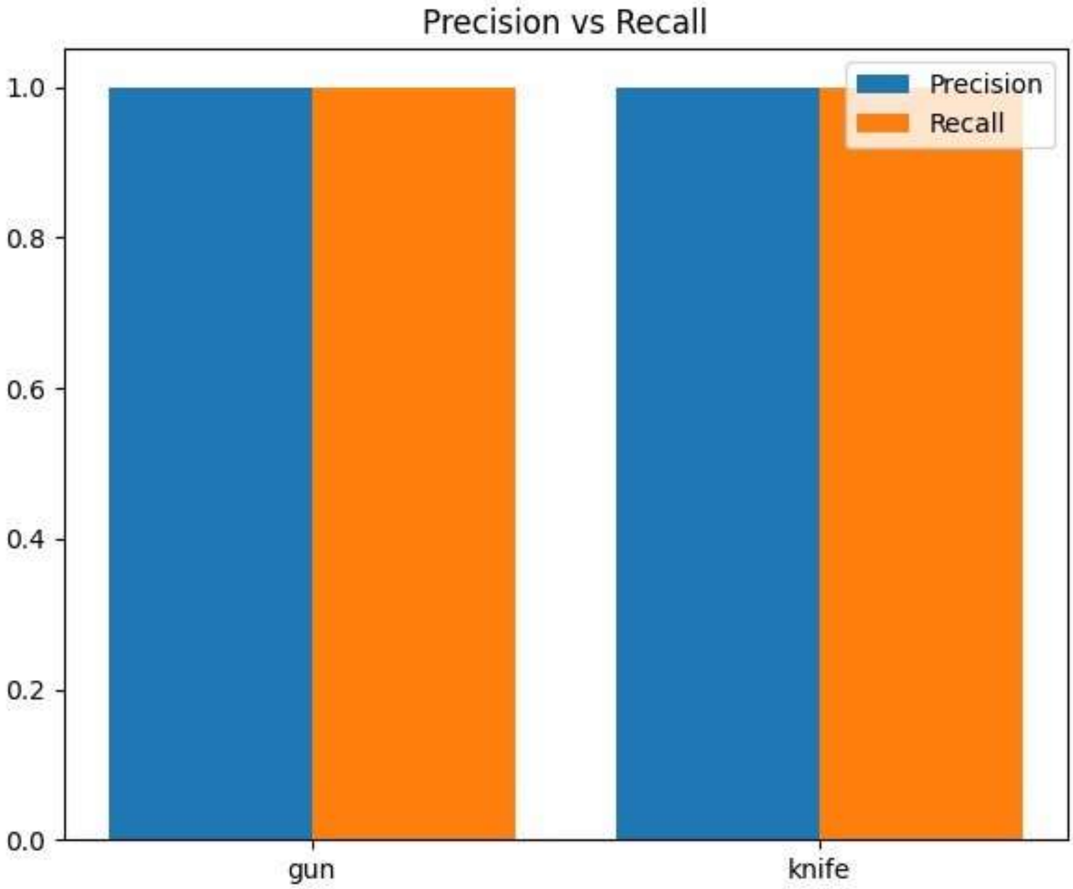


Figure 8: Thermal Model Performance Graphs

RGB Model Results:



International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com



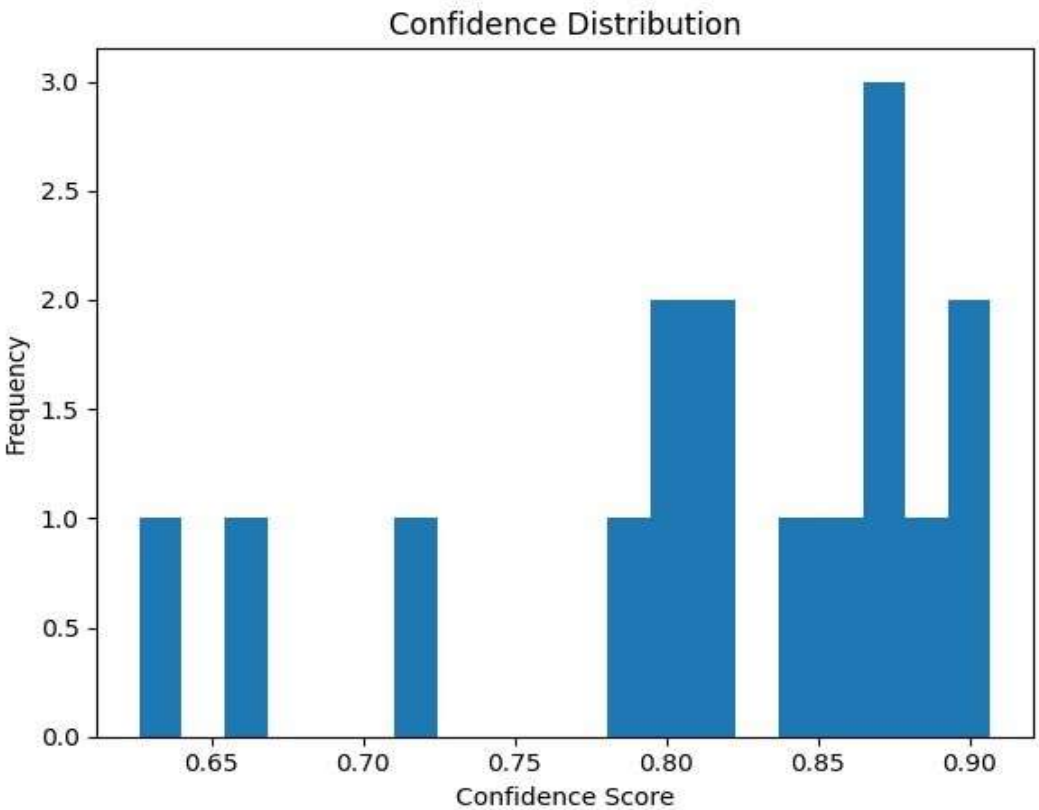


Figure 9: RGB Model Detection Results and Performance Graphs

5.3 Analysis of Model Behavior

Under normal lighting conditions, RGB model demonstrates good performance because it mainly uses visual features of an object such as object structure. Poor lighting reduces the model's performance since some visual characteristics cannot be easily distinguished.

On the contrary, the thermal model does not rely on light and therefore can function under any lighting conditions. Its performance is reduced in normal lighting situations because there is little difference between the temperature of an object and its environment.

Finally, the proposed method utilizes both models and chooses the most reliable output depending on models' confidence levels.

5.4 The Impact of Decision-Level Fusion

Decision-level fusion dramatically improves the system's performance since the system outputs the prediction with the highest confidence value:

$D = argmax(C\_RGB, C\_Thermal)$

Since decision-level fusion does not require additional computations with feature vectors, this method of improving the model's performance is computationally inexpensive.

5.5 Temporal Validation

Temporal validation significantly improves the reliability of object detection since multiple confirmations are required on multiple consecutive frames:

$N \geq 3$ *consecutive frames*

This measure helps avoid errors caused by noise or motion artifacts.

Table 3: False Positive Reduction

Method	False Positives
Without Temporal Validation	48
With Temporal Validation	17

As it can be concluded from the numbers presented in Table 3, the application of temporal validation considerably reduces the number of false positive predictions.

5.6 Real-Time Performance Evaluation

To assess the system's real-time detection abilities, live video stream was passed to the program. Achieved results were the following:

- FPS: 15–20 frames per second
- Latency: 50–100 milliseconds per frame

Those were obtained via frame sampling and appropriate resizing of the video stream.

5.7 Results of Analytics and Forecasting

There are two more features introduced in the system: analytics and risk forecast. Analytic module supplies data about detected objects and confidence level. Forecasting module can predict the number of detections and risk level of the current situation (low, medium, high).

With the use of the system, it becomes possible to estimate:

- Periods with maximum detection rate.
- Average frequency of object detection.

- Risk level of detection.

5.8 Discussion

According to Table 2, a multimodal approach increases the accuracy of object detection compared to each of the models alone. Rich information obtained with the use of RGB model provides additional details about detected objects. Robustness of thermal model allows the system to detect weapons under almost any lighting conditions. Finally, the proposed fusion technique utilizes the benefits of the two methods mentioned above.

Additionally, the application of temporal validation allows making the system more robust against errors caused by noise. Consequently, the system becomes more effective when operating in real-time.

6. Conclusion

In this paper, we discussed a multimodal approach to weapon detection and forecasting based on an RGB model and a thermal model as well as the decision-level fusion technique. The main idea of developing this approach was to combine the capabilities of both models that would enable effective weapon detection at all stages of a video frame analysis.

The proposed methodology suggests applying both models to input and choosing the prediction using the confidence score fusion technique. In addition, temporal validation allows us to decrease the number of false positives. Finally, we have developed an alerting mechanism that informs about detections by sending an immediate message through audio and email services and implements a logging system for storing the results.

The experiments prove the effectiveness of the proposed fusion-based approach compared to RGB and thermal models. As expected, the former works better when there is enough light, while the latter detects objects in adverse conditions. Thus, their combination increases the quality of detection. The inclusion of additional analytic modules such as forecasting allows us to go beyond simple detection since we can determine the most dangerous period in time as well as estimate the risk.

To sum up, the multimodal fusion-based approach to weapon detection is effective, reliable, and easy to implement. The proposed system enables efficient weapon detection in surveillance applications.

The future development of this system might include improving the efficiency of the process and the quality of results by developing novel approaches to fusion and extending the database to include other object categories.

7. References

1. Muñoz J.D., Ruiz-Santaquiteria J., Deniz O., Bueno G., Concealed weapon detection using thermal cameras, Journal of Imaging, 2025, 11(3).
2. Al-Otaibi S., Al-Sameai H.A., Al-Ghamdi A.S., Weapon detection with FMR-CNN and YOLOv8 for enhanced crime prevention and security, Results in Engineering, 2025.
3. Bhardwaj D., Ramamoorthy A., Goyal P., DEF-YOLO: Leveraging YOLO for concealed weapon detection in thermal imaging, ResearchGate, 2025.
4. Kowalski M., Mierzejewski K., Pałys T., Bi-spectral concealed object detection with attention-based fusion, Engineering Applications of Artificial Intelligence, 2025.
5. Salah R., Elngar A.A., Al-Qaness M.A.A., Robust weapon detection in dark environments using YOLOv7-DarkVision, Digital Signal Processing, 2024.

6. Berardini D., Migliorelli L., Galdelli A., Frontoni E., Mancini A., Moccia S., A deep-learning framework running on edge devices for handgun and knife detection from indoor video-surveillance cameras, Multimedia Tools and Applications, 2023.
7. Bonde V., Dharamthok T., Doke S., Choudhari A., Baviskar V., GuardVision: An enhanced weapon detection using computer vision with alert mechanism, International Journal of Emerging Technologies and Innovative Research, 2025.
8. Mukto M.M., Hasan M., Mahmud M.M.A., Haque I., Ahmed M.A., Jabid T., et al., Design of a real-time crime monitoring system using deep learning techniques, Intelligent Systems with Applications, 2024.
9. Redmon J., Divvala S., Girshick R., Farhadi A., You only look once: Unified, real-time object detection, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788.
10. Jocher G., Chaurasia A., Qiu J., Ultralytics YOLOv8, Ultralytics, 2023. https://github.com/ultralytics/ultralytics
11. Olmos R., Tabik S., Herrera F., Automatic handgun detection alarm in videos using deep learning, Neurocomputing, 2018, 275, 66–72.
12. Grega M., Matiolanski A., Guzik P., Leszczuk M., Automated detection of firearms and knives in CCTV images, Sensors, 2016, 16(1), 47.
13. Lim J.S., Al-Masni M.A., Al-antari M.A., Yang S.J., Muhammed T., Kim T.S., A real-time indoor surveillance system based on deep learning for weapon detection, Electronics, 2019, 8(10), 1113.
14. Warsi A., Wani M.A., Ansari A.Q., Real-time gun detection and alert system using YOLOv5 and Flask-based web framework for smart surveillance, Procedia Computer Science, 2023, 218, 1073–1082.
15. Zheng L., Tang M., Chen Y., Zhu G., Wang J., Lu H., Improving multiple object tracking with single object tracking, In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 2453–2462.
16. Dong X., Zheng G., Ma F., Su L., Du J., Few-shot object detection via feature reweighting, In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2022, pp. 8420–8429.
17. Simonyan K., Zisserman A., Very deep convolutional networks for large-scale image recognition, In Proceedings of the International Conference on Learning Representations (ICLR), 2015.
18. He K., Zhang X., Ren S., Sun J., Deep residual learning for image recognition, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.
19. Sreenu G., Saleem Durai M.A., Intelligent video surveillance: A review through deep learning techniques for crowd analysis, Journal of Big Data, 2019, 6(1), 1–27.
20. Aksoy B., Dogan H., Ure N.K., Multi-sensor decision-level fusion for real-time threat detection in surveillance environments, Expert Systems with Applications, 2023, 213, 119151.