TITLE

A project report submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Technology

in

Electronics & Computer Engineering

by

NAME

19BECxxxx

School of Electronics Engineering,

Vellore Institute of Technology, Chennai,

Vandalur-Kelambakkam Road,

Chennai - 600127, India.

April 2025

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Declaration

I hereby declare that the report titled ***PrisonSecure: A smart*** submitted by us to the School of Electronics Engineering, Vellore Institute of Technology, Chennai in partial fulfillment of the requirements for the award of **Bachelor of Technology** in **Electronics and Computer Engineering** is a bona-fide record of the work carried out by me under the supervision of ***Guide Name***.
I further declare that the work reported in this report, has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma of this institute or of any other institute or University.

Sign: _____

Name & Reg. No.: _____

Date: _____

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Electronics Engineering

Certificate

This is to certify that the project report titled ***Title*** submitted by ***Name*** **(Reg. No.)** to Vellore Institute of Technology Chennai, in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in **Electronics and Computer Engineering** is a bona-fide work carried out under my supervision. The project report fulfills the requirements as per the regulations of this University and in my opinion meets the necessary standards for submission. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma and the same is certified.

Supervisor

Signature: .....

Name: .....

Date:

Head of the Department

Signature: .....

Name: .....

Date:

Examiner

Signature: .....

Name: .....

Date:

(Seal of the School)

Abstract

Security in prisons requires fast and reliable systems that can detect threats like weapons and identify people in real time. Relying only on human guards is not enough, as they can get tired and may not react quickly to threats. In this project, we built a smart surveillance system called PRISONSECURE that uses deep learning to detect weapons and recognize prisoner faces from live CCTV footage.

The system uses YOLOv11 to detect weapons such as handguns and knives. To reduce false alarms caused by objects like phones or umbrellas, we used a method called hard negative mining (weapons looking similar to handguns or knives). For recognizing faces, we use a face detector with ResNet, create face embeddings using FaceNet, and then classify them using Support Vector Classifier (SVC).

Our system works well even in difficult conditions like poor lighting, crowded areas, and different face angles. It achieved a precision of 96.80% and a recall of 94.25% for weapon detection at IoU threshold 0.5. The mean Average Precision (mAP@0.5:0.95) was 87.60%, showing good performance across different confidence levels. For face recognition, the system got 93.75% accuracy with very few wrong matches.

The system runs fast, taking only a few milliseconds per frame, which makes it useful for real-time alerts. It also has a flexible design, so more features, such as behavior analysis or action recognition, can be added later.

Overall, this project shows that combining deep learning models for weapon detection and face recognition can make prison security systems much smarter and more reliable, reducing false alarms and responding quickly to real threats.

Acknowledgements

This section is open to you, below is only a template

We wish to express our sincere thanks and deep sense of gratitude to our project guide, Dr. xxxxx, Professor, School of Electronics Engineering, for her consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We are extremely grateful to Dr. Ravishankar A, Dean Dr. Reena Monica, Associate Dean (Academics) & Dr. John Sahaya Rani Alex, Associate Dean (Research) of the School of Electronics Engineering, VIT Chennai, for extending the facilities of the School towards our project and for his unstinting support.

We express our thanks to our Head of the Department Dr. Annis Fathima A for her support throughout the course of this project.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

Contents

List of Figures

Chapter 1

Introduction

1.1 Introduction

The incorporation of artificial intelligence (AI) in surveillance systems has revolutionized traditional security mechanisms by enabling automatic monitoring, real-time threat identification, and advanced behavioral analysis. In high-risk environments such as prisons, where continuous surveillance and rapid incident response are imperative, conventional manual systems relying on human operators are increasingly inadequate. Human-based surveillance is vulnerable to fatigue, slow response times, and oversight, potentially compromising the safety of inmates, correctional staff, and the overall facility. To overcome these limitations, AI-powered surveillance systems employing deep learning techniques are being increasingly adopted to automate and enhance situational awareness.

Prison environments present unique challenges for surveillance. Unlike public surveillance settings, prisons are characterized by densely populated confined spaces, low lighting, constrained camera angles, and frequent visual obstructions. Traditional motion detection systems or human-monitored CCTV setups are ineffective in identifying concealed weapons or tracking individuals involved in incidents. Real-time facial recognition in such dynamic, obstructed scenarios remains a critical challenge, especially when inmates actively conceal their identities through disguises, shadows, or postural adjustments.

Further, a delicate balance must be maintained between model accuracy and processing speed. While state-of-the-art models such as Faster R-CNN and earlier versions of YOLO have demonstrated high accuracy, they often suffer from latency issues that make them unsuitable for real-time applications. Moreover, false positives remain a concern, with

innocuous objects such as mobile phones or metallic items frequently misclassified as weapons. Similarly, traditional facial recognition methods struggle with low-light or occluded conditions, resulting in unreliable identification. In addition, privacy concerns and the lack of scalable, edge-device compatible solutions hinder practical deployment in correctional institutions.

To address these gaps, the current study proposes **PRISONSECURE**, a deep learning-based surveillance system designed specifically for real-time weapon detection and facial recognition within prison environments. The proposed system utilizes an optimized YOLOv11 architecture capable of detecting knives and firearms even in congested or low-resolution video streams. The training dataset is carefully curated, incorporating a wide variety of weapon images, challenging negative samples, and realistic prison scenes to maximize model robustness. To counter visibility issues, advanced augmentation techniques such as Mosaic, MixUp, and AutoAugment are applied, while hard negative mining is employed to reduce false positives by training the model on deceptive non-weapon objects.

For facial recognition, PRISONSECURE integrates a ResNet-based face detection pipeline followed by FaceNet embeddings and a Support Vector Classifier (SVC) for identity verification. This architecture effectively extracts high-dimensional facial features even in suboptimal conditions, including partial occlusions and pose variations. The system is optimized for real-time execution on GPU-accelerated edge devices such as NVIDIA Jetson Nano and RTX 3050, enabling efficient, on-site processing without dependency on external servers. This not only ensures minimal latency—approximately 50 milliseconds per frame—but also maintains compliance with data privacy policies by retaining sensitive information within institutional boundaries.

Furthermore, PRISONSECURE includes an intelligent alerting mechanism that automatically generates incident reports upon detection of a weapon or identification of a suspect individual. Reports include the weapon type, individual ID (if matched), time of incident, and a cropped evidence image. These alerts are transmitted via SMS, email, and integrated into a centralized monitoring dashboard, enabling swift intervention by prison staff and significantly reducing the risk of escalation.

1.2 Chapter Overview

The remainder of this thesis is organized as follows:

- **Chapter 2** presents a comprehensive review of existing literature and the limitations of current prison surveillance technologies.

- **Chapter 3** details the architecture and methodology of the proposed PRISON-SECURE system.
- **Chapter 4** outlines the experimental setup and dataset preparation techniques used in the implementation.
- **Chapter 5** provides the performance evaluation and results obtained from the deployed system.
- **Chapter 6** concludes the thesis and proposes future directions for expanding and refining the system.

Chapter 2

Literature Survey

2.1 Relevant Literature

Advances in computer vision and deep learning in recent years have revolutionized the surveillance system landscape, especially for weapons detection and facial recognition systems. Yet, in spite of the increasing body of literature on the topic, a number of challenges remain in attaining optimal performance under the sight of real-world constraints like occlusion, lighting fluctuations, and system latency. This section gives an overview of recent researches that constitute the conceptual foundation of **PRISON-SECURE**, including the models applied, their efficacy, and the constraints encountered in real-world deployment.

Chatterjee et al. (2023) proposed a YOLOv5 deep learning model for weapon detection in urban smart city surveillance. The model was domain-specifically trained and optimized with methods such as hard negative mining to minimize false positives. The authors attained high detection accuracy in laboratory environments, reporting precision gains in distinguishing firearms from objects with similar shapes but not lethality. Their method was, however, critically criticized when applied to low-resolution and crowded surveillance images, especially in indoor settings. Additionally, the system lacked real-time alerting or person identification modules, rendering it less applicable to closed-loop security systems such as those needed in prisons.

Bhatti et al. (2021) used YOLO (You Only Look Once) with Faster R-CNN for real-time gun detection system development to be used in CCTV surveillance. Integration of a region-based proposal network into Faster R-CNN improved object localization accuracy, while YOLO allowed inference acceleration durations. The models, however,

had some weaknesses in identifying small objects when surveilled over wide angles, a common challenge in institutional settings. The dataset used was generic and lacked the specificity required in prison settings, where visual variables such as clothing, architectural environment, and occlusion are influential. The lack of a facial recognition module also made the system inadequate in situations where the detection of individuals with weapons is required.

Conversely, Ruiz-Santaquiteria et al. (2021) proposed a new dual-cue detection model that integrated human pose estimation with visual cues of guns to enhance the accuracy of handgun detection. The multi-aspect methodology greatly improved detection performance in challenging situations where weapons were partially occluded or positioned at abnormal angles. Nevertheless, the inclusion of pose estimation raised the computational loads, making the system incompatible with real-time processing without major hardware upgrades. Further, even though the proposed system achieved detection success, it lacked an identity recognition module and did not offer any mechanism for triggering alerts or logs, thereby limiting its applicability in real-time scenarios.

Facial recognition-wise, Irfan et al. (2025) employed an MTCNN-based system for face detection and FaceNet for face recognition. The authors demonstrated promising accuracy on benchmark data, with real-time performance on live streams. Nevertheless, the system was poor in cases with masked or occluded faces, not uncommon in high-risk environments such as prisons. The model also was not invariant to changing illumination, with results being unreliable during night-time or low-illumination surveillance. A significant drawback was the need for frontal facial images in order to generate accurate embedding, restricting its use in dynamic, multi-camera multi-angle environments.

Kumar et al. (2024) conducted an exhaustive review of the YOLO series, from YOLOv1 to YOLOv8, emphasizing significant architectural improvements like the incorporation of CSPNet, utilization of PANet for feature fusion, and utilization of decoupled heads for enhancing classification and localization performance. YOLOv8 has been recognized as one of the top real-time detection systems, with an acceptable trade-off between inference speed and accuracy. Nevertheless, the article did not emphasize domain-specific applications appropriate for high-security surveillance settings. Although informative, the review was lacking in the inclusion of practical tests conducted in controlled environments, like prisons, where factors like occlusion, congestion, and intricate movement patterns present specific challenges.

The research findings contribute significantly to the understanding of the dynamic properties of AI-based surveillance systems; however, they do not provide an end-to-end, real-time solution tailored for correctional facilities. Existing models are either not very robust against poor visibility conditions or do not provide an integrated framework that integrates detection and recognition with realistic reactions. **PRISONSECURE** aims to address these limitations by utilizing YOLOv11 for object detection, adding facial recognition with FaceNet and SVM, and running the system on edge devices to provide low-latency, real-time performance specifically tailored for institutional surveillance.

2.1.1 Research Gaps and Challenges

The existing body of literature on surveillance technologies recognizes a number of existing shortcomings and challenges, particularly in the context of the strict and limiting conditions of correctional institutions. While both weapon detection and facial recognition technologies have seen considerable advancement in isolation, there is still a lack of end-to-end systems that combine these technologies into a unified real-time system. In addition, the existing models face considerable difficulties when operating in less than optimal visual environments, in limited hardware environments, and in realistic deployment environments where both minimal latency and increased accuracy are critical.

A major challenge in the area of weapon detection is that of reliable detection for small or occluded weapons. Most models, i.e., the ones above YOLOv4 or YOLOv5, do not have the capability to maintain high accuracy in detection in the scenario when the weapon is small in size compared to the size of the whole frame, a prevalent condition in wide-angle prison surveillance systems. Bhatti et al. (2021) and Chatterjee et al. (2023) report high precision in normal datasets, but their performance is low under changing lighting conditions, occlusion, or rapid motion, very prevalent in surveillance videos in real life. In addition, the models are afflicted with a lack of ability to deliver similar detection results across different camera views and resolutions and therefore cannot be deployed.

Another major issue is a lack of real-time response capability. Detection products tend to be designed for after-the-fact analysis or batch processing, rather than with the responsiveness to react to a developing incident. This is a fundamental flaw in prison settings, where prompt response is essential to prevent escalation. Even extremely sensitive detection systems are ineffective if they are unable to offer real-time alerting to security staff.

Facial recognition research identifies a distinct set of shortcomings. FaceNet-type systems exhibit highly accurate performance in controlled environments; they are susceptible to interruptions caused by occlusions, lighting changes, and pose variations. Irfan et al.'s (2025) work illustrates robust baseline performance, but such models perform poorly when there are partially occluded faces or facial transformation methods are used—issues which acquire a heightened salience in prison environments where individuals may seek to hide identities.

Hardware limitation is a significant issue in this context. Several advanced models need GPU-accelerated servers, which can be a deployment feasibility issue in remote or cost-constrained prison facilities. While some attempts have been made to look into edge computing opportunities, such solutions usually do not support optimization for cooperative execution of multiple tasks, such as the execution of object detection and facial recognition concurrently. Further, the majority of systems do not consider energy efficiency and scalability, which are significant considerations when deploying across multiple surveillance nodes in a large institutional environment.

Lastly, the works that were discussed tend to omit concerns about ethics and privacy issues. The safeguarding and sharing of sensitive video recordings, particularly when done by cloud-based platforms, pose strong concerns regarding protecting data. For prisons, where video surveillance takes place around the clock and very often encroaches on privacy, the maintenance of data confidentiality is not simply a technical but also a legal and ethical responsibility.

PRISONSECURE is tailored to overcome these determined challenges. It integrates a state-of-the-art YOLOv11 object detection system with a high-performance FaceNet + SVM face recognition module. In operation on GPU-enabled edge devices, it offers ultra-low-latency and high-performance operation within the realities of the outside world. The system is constructed to perform flawlessly under low lighting, occlusion, and dynamic movement and to provide real-time alerts, identity tagging, and detailed event logging—all within a privacy-aware, modular framework.

Chapter 3

Methodology

3.1 System Overview

The AI-powered surveillance system is designed to enhance security in correctional facilities by integrating real-time weapon detection, facial recognition, and automated threat alerts. The system processes live video feeds from multiple CCTV cameras and applies deep learning algorithms to detect weapons, recognize faces, and identify suspicious activities. By leveraging parallel processing, the system ensures real-time analysis with minimal latency, enabling immediate response to security threats.

The weapon detection module utilizes YOLO-based object detection to identify handguns and knives in surveillance footage. The model has been trained on a specialized dataset with hard negative mining techniques to reduce false positives and improve detection accuracy [2, 5, 7]. When a weapon is detected, the system generates an instant alert, captures an image of the scene, and records event details for further analysis.

The facial recognition module employs ResNet for face detection, FaceNet for feature extraction, and an SVM classifier for identity verification. It enables continuous tracking of individuals within the facility, ensuring that unauthorized personnel are identified promptly. The system is designed to work in challenging conditions, such as low-light environments and occlusions [4, 9, 15].

To manage real-time processing, the system utilizes multi-threaded parallel processing, distributing computational tasks efficiently across available hardware resources. This architecture allows seamless handling of multiple video streams, ensuring uninterrupted surveillance [1, 3, 6].

Additionally, the system incorporates an intelligent alert mechanism that prioritizes security threats and notifies personnel based on risk levels. The system is scalable,

allowing integration with future enhancements such as pose estimation and behavioral analysis for proactive threat detection [3, 8, 12]. By combining these capabilities, the surveillance system significantly enhances security monitoring in correctional facilities.

3.2 System Architecture

System Architecture The system architecture is designed for real-time surveillance and threat detection, leveraging deep learning models for weapon detection and facial recognition. It employs a parallel processing framework for efficient computation and seamless data flow. The architecture consists of multiple interconnected modules, ensuring rapid threat identification and response.

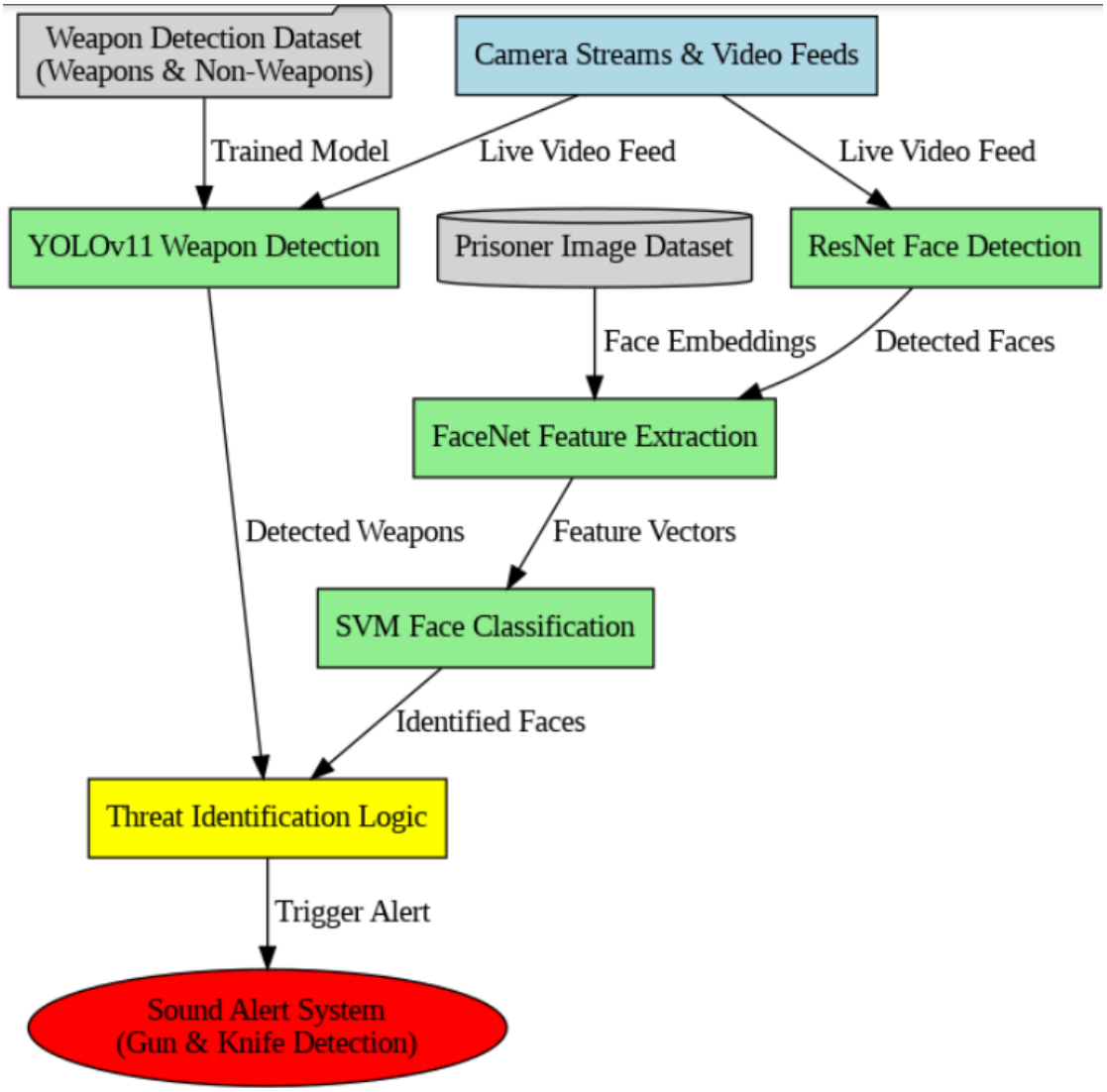


FIGURE 3.1: System Architecture

1. Data Acquisition Module
   The system processes live video streams from CCTV cameras positioned within the prison premises. These video feeds are directly fed into the AI-powered detection pipeline, which consists of two primary datasets:
   (a) Weapon Detection Dataset: Comprising approximately 3,400 images of guns, 3,300 images of knives, and 3,000 non-weapon images(Hard Negative Mining).
   (b) Prisoner Image Dataset: A collection of prisoner images for real-time facial identification.
2. Preprocessing Pipeline
   Incoming video frames undergo preprocessing, including resizing, normalization, and enhancement techniques. This ensures improved model accuracy and robust performance under varying lighting conditions.
3. Weapon Detection Module
   The system utilizes a YOLOv11 to identify weapons within the video feed. The detection model has been optimized using hard negative mining to reduce false positives and improve classification accuracy. If a weapon is detected, the system triggers an alert.
4. Facial Recognition Module
   To identify individuals, the system employs:
   (a) Face Detection: A ResNet-based face detector.
   (b) Feature Extraction: FaceNet for generating facial embeddings.
   (c) Classification: An SVM classifier for identity verification.
5. An Alert System
   Upon detecting a weapon or identifying a known prisoner, the system immediately triggers an alert. The alert system is purely sound-based, ensuring that security personnel receive immediate notification of potential threats.

3.3 Weapon Detection Module

The weapon detection module is designed to identify firearms and knives in real-time using deep learning-based object detection techniques. The system employs a YOLO-based model for efficient weapon detection from surveillance footage, ensuring high accuracy and low latency. By processing video streams frame by frame, the module continuously analyzes incoming footage to detect potential threats. Preprocessing techniques,

such as contrast enhancement and noise reduction, are applied to optimize detection performance before passing frames through the trained YOLOv11 model.

A key aspect of this module is its ability to minimize false positives using hard negative mining, where the model is trained on challenging non-weapon images to improve robustness. Additionally, post-processing techniques, such as confidence thresholding and refine detections to ensure accuracy. The system is optimized for GPU acceleration, allowing it to handle multiple camera feeds and process frames in real-time without performance bottlenecks.

Upon detecting a weapon, the module triggers an instant security alert, activating an audible alarm to notify prison staff. This enables swift intervention, preventing potential security breaches. The integration of this module within the surveillance system provides a proactive and reliable solution for early threat detection, enhancing prison security and safety.

3.3.1 Overview of Weapon Detection

The weapon detection module is developed to identify dangerous objects such as handguns and knives in real-time from surveillance footage.

1. Data Collection and Preprocessing: Images containing weapons and non-weapon objects are collected and labeled. Data is augmented to increase variation and improve the model's generalization.

2. Model Selection and Architecture Design: A deep learning model is selected and configured for object detection. The architecture is optimized for fast and accurate recognition of weapon classes.

3. Hard Negative Mining: Non-weapon objects that are often misclassified as weapons are added to the training set to reduce false positives and improve precision.

4. Training the Model: The model is trained using annotated images, with proper hyperparameter tuning and optimization techniques to improve detection accuracy.

5. Model Evaluation: The trained model is evaluated using standard metrics like precision, recall, and mean average precision to ensure performance is reliable.

6. Real-Time Integration: The final model is integrated into a real-time video pipeline, where it continuously scans input frames for the presence of weapons and triggers alerts when detected.

3.3.2 Object Detection using YOLO (Comparative Study of YOLOv8 and YOLOv11)

Object detection is a crucial component in computer vision applications, particularly for real-time surveillance systems. The YOLO (You Only Look Once) series has evolved significantly, with YOLOv8 and YOLOv11 being among the most recent advancements. In this study, we compare these two versions in terms of accuracy, efficiency, and performance for small object detection.

YOLOv8: As an improvement over previous versions, YOLOv8 integrates modern architectural modifications, including a restructured backbone, improved feature fusion mechanisms, and an enhanced anchor-free detection approach. It optimizes detection speed while maintaining competitive accuracy, making it suitable for real-time applications like surveillance and autonomous systems.

YOLOv11: The latest iteration, YOLOv11, builds upon YOLOv8 by incorporating advanced techniques such as Soft-NMS (Soft Non-Maximum Suppression) for better handling of overlapping objects and a Segment Anything Model (SAM) for improved feature extraction. These enhancements contribute to higher detection precision, especially in complex environments where small object detection is critical.

The comparative analysis in our study focuses on evaluating these models using different optimization strategies. The performance metrics include mean Average Precision (mAP), inference speed, and robustness to varying lighting conditions. Experimental results indicate that while YOLOv8 offers a balance between speed and accuracy, YOLOv11 demonstrates superior detection capabilities, particularly for small and occluded objects.

The findings suggest that YOLOv11 is a promising choice for high-security applications such as prison surveillance, where detecting concealed weapons is vital. Its integration with Soft-NMS and SAM enhances detection reliability, ensuring better threat identification and response times.

3.3.3 Data selection And Augmentation Module

The effectiveness of a deep learning-based weapon detection system largely depends on the quality and diversity of the dataset used for training. Our dataset consists of approximately 3,400 firearm images, 3,300 knife images, and 3,000 non-weapon images, ensuring a balanced representation of weapons and background objects. The images are sourced from public datasets, surveillance footage, and manually collected samples,

covering various lighting conditions, angles, and occlusions to enhance generalization. A 70-20-10 train-validation-test split is applied to ensure optimal model training and evaluation.

To further improve model robustness, data augmentation techniques are employed. Geometric transformations such as rotation, scaling, flipping, and cropping help the model recognize weapons from different perspectives. Color augmentations, including brightness, contrast adjustments, and Gaussian noise, simulate varying environmental conditions. Mosaic augmentation is applied to merge multiple images, allowing the model to detect small weapons more effectively. Random occlusion and cutout techniques ensure the model learns to recognize partially visible weapons.

These augmentations significantly enhance the model's ability to generalize across real-world scenarios, reducing overfitting and improving performance in real-time prison surveillance environments. The diverse dataset and augmentation strategies ensure high accuracy, low false positives, and reliable detection capabilities.

3.3.4 Preprocessing Techniques for Weapon Detection

The preprocessing techniques for weapon recognition in PRISONSECURE are designed to enhance detection accuracy while maintaining real-time performance. The raw video frames from the surveillance feed are first resized to a standardized resolution (e.g., 640×640 pixels) to match the YOLO model's input requirements. This resizing ensures uniformity across all frames and minimizes computational overhead while preserving essential details for weapon detection.

To improve visibility in low-light conditions, histogram equalization is applied to enhance contrast, making weapons more distinguishable from their surroundings. Noise reduction techniques such as Gaussian blurring or median filtering are employed to eliminate environmental artifacts, ensuring the model does not misinterpret random textures as potential threats. Adaptive thresholding is used to highlight the edges of objects, ensuring that weapon contours remain sharp and detectable under varying lighting conditions.

Motion blur, a common challenge in dynamic surveillance environments, is mitigated using deblurring algorithms like Wiener filtering. This helps recover sharpness in weapon outlines, reducing the chances of misclassification. Color-space transformations, such as RGB to HSV conversion, are used to emphasize distinct weapon features, allowing the model to better differentiate between weapons and background elements with similar color compositions.

To further enhance the dataset, preprocessing includes normalization and mean subtraction, scaling pixel values between 0 and 1. This step prevents activation biases in the deep learning model, leading to more stable training and inference. Augmentation techniques such as random rotations, horizontal flipping, brightness adjustments, and synthetic noise additions are applied dynamically to improve generalization. These augmentations help the model recognize weapons under different viewing angles, orientations, and illumination variations.

Finally, preprocessing is optimized for real-time execution using GPU acceleration and parallel processing. By ensuring efficient frame processing and applying targeted enhancement techniques, the PRISONSECURE weapon detection system remains robust across diverse surveillance scenarios, reducing false positives while maintaining high detection accuracy.

3.3.5 Model Training and Hyperparameter Optimization

The model training process for weapon detection in our system is built on YOLOv11, an advanced object detection framework optimized for high-speed and high-accuracy detection. The training pipeline is designed to maximize detection precision while minimizing false positives, making it highly effective for real-time surveillance applications.

3.3.5.1 Dataset Preparation and Splitting

The dataset consists of 3,400 firearm images, 3,300 knife images, and 3,000 non-weapon images, split into a 70-20-10 ratio for training, testing, and validation, respectively. The `data.yaml` file defines class labels and image paths, ensuring proper dataset structuring for YOLOv11 training.

3.3.5.2 Hyperparameter Optimization

Several hyperparameters were tuned to enhance model efficiency. The image size was set to 736×736, balancing accuracy and computational load. The batch size was adjusted to 16, optimized for the RTX 3050 GPU's 8GB VRAM. The model was trained for 100 epochs, with an early stopping patience of 10 epochs to prevent overfitting. The AdamW optimizer was chosen for its adaptive learning rate mechanism, with an initial learning rate of 0.0001 and a momentum of 0.95.

To further refine learning, cosine learning rate scheduling (`cos_lr`) was implemented, allowing a smooth decay of the learning rate for stable convergence. The model also utilized label smoothing (0.05) to mitigate overconfidence in predictions.

3.3.5.3 Augmentation and Regularization

Data augmentation techniques such as mosaic (0.5 probability), mixup (0.05), random erasing (0.4), and RandAugment were applied to improve generalization. Additionally, disk caching was enabled to speed up data loading, and automatic mixed precision (AMP) training was used to reduce memory consumption.

By implementing these strategies, the YOLOv11-based weapon detection module achieves high accuracy, rapid inference, and low latency, making it well-suited for real-time surveillance in prison security systems.

3.3.6 Hard Negative Mining for Reducing False Positives in PRISONSECURE

In the PRISONSECURE system, minimizing false positives is essential to ensure accurate real-time weapon detection in prison environments. False alarms can cause unnecessary panic, disrupt prison operations, and reduce trust in the surveillance system. To address this, we implemented hard negative mining, a technique that systematically reduces false positives by retraining the model with challenging non-weapon objects that resemble weapons.

During initial testing, the PRISONSECURE weapon detection model misclassified common prison objects such as mobile phones, walkie-talkies, metal bars, and kitchen utensils as weapons. These false positives were identified and manually added as hard negative samples in subsequent training iterations. The model was then retrained to differentiate between actual threats (e.g., handguns, knives) and non-threatening objects.

Hard negative mining was performed in multiple iterative cycles, allowing the model to continuously improve its accuracy. After each training cycle, new false positives were identified, added to the dataset, and used to refine the model further. This iterative approach gradually reduced misclassifications, ensuring that the PRISONSECURE system accurately distinguishes between real weapons and deceptive non-weapon objects.

To enhance robustness, real-world prison surveillance footage was incorporated into the training dataset. Unlike controlled datasets, live prison environments include dynamic elements such as low lighting, motion blur, occlusions, and overlapping objects. Training

on such realistic data helped PRISONSECURE adapt to complex visual conditions, making the detection system more reliable for 24/7 monitoring.

The implementation of hard negative mining in PRISONSECURE led to a significant reduction in false positive rates, ensuring that security personnel receive only critical alerts for actual threats. This optimization improves response efficiency, prevents unnecessary disruptions, and enhances the overall security of the prison facility. By refining its detection capabilities through continuous learning, PRISONSECURE maintains high precision and recall, making it a trustworthy AI-powered surveillance solution for prison security.

3.4 Face Recognition Module

3.4.1 Introduction

The Face Recognition Module plays a pivotal role within the PRISONSECURE surveillance system. It is responsible for real-time identification of individuals within the facility using advanced deep learning techniques. By combining a ResNet-based face detector, FaceNet for feature embedding, and a Support Vector Machine (SVM) classifier, the module ensures accurate and efficient face recognition. It also supports the creation of custom datasets and real-time identification, making it especially suited for high-security environments such as correctional facilities.

Here's the same content rewritten in paragraph form as you requested:

—

3.4.2 Face Detection Using ResNet

The Face Detection module utilizes a pretrained Single Shot Detector (SSD) model based on the ResNet-10 architecture to identify faces in both images and video streams. It employs the `res10_300x300_ssd_iter_140000.caffemodel` model with a configuration defined in the `deploy.prototxt` file. This implementation is supported by OpenCV's Deep Neural Network (DNN) module with a Caffe backend. The module processes input frames by resizing them to a consistent dimension of 300x300 pixels and converting them into a blob format. This blob is passed into the SSD model, which detects bounding boxes for faces in the input. To ensure accuracy, the module filters detections using a

confidence threshold of 0.7. Cropped regions corresponding to valid detections are forwarded to subsequent modules for further processing. This architecture enables efficient, real-time face detection across various scenarios.

3.4.3 Dataset Creation

The Dataset Creation process is designed to build a labeled collection of face images for training the face recognition model. Live frames are captured in real-time using a webcam, with face regions identified using the ResNet-based detection module. Detected faces are cropped and resized to a standardized dimension of 200x200 pixels, then converted to grayscale to reduce computational overhead while preserving essential features. Each processed image is stored with a systematic naming convention, such as `user.<label>.<image_id>.jpg`, to ensure a clear association with the corresponding label. Data collection continues until a predefined number of images per individual is reached or the user manually halts the process. This approach ensures the dataset captures a comprehensive range of facial variations for robust model training.

3.4.4 Feature Extraction with FaceNet

Feature extraction is carried out using the pre-trained FaceNet model, accessed through the `keras-facenet` library. This step converts face images into 128-dimensional embedding vectors that represent their unique features. Input images are resized to 160x160 pixels to meet the model's requirements, and a batch dimension is added for compatibility. These images are then passed through the FaceNet model, which generates embeddings that encapsulate the distinguishing characteristics of each face. These embeddings are critical for enabling efficient and accurate classification in the subsequent steps.

3.4.5 Classification Using SVM

The classification process employs a supervised learning approach using a Support Vector Machine (SVM) classifier with an RBF kernel. After extracting embeddings for all dataset images, their corresponding labels are encoded into numerical representations using `LabelEncoder`. The classifier is trained using these embeddings and labels, allowing it to differentiate between unique feature vectors effectively. Once trained, the SVM model and label encoder are saved in serialized formats (`classifier.pkl` and `label_encoder.pkl`) for seamless deployment during real-time recognition.

3.4.6 Real-Time Face Recognition

The real-time face recognition module integrates all components to identify individuals from live webcam streams. Video frames are processed to detect faces using the ResNet-based detection module, and the cropped face regions are passed to the FaceNet model for embedding generation. The embeddings are classified using the pre-trained SVM model, and predictions with confidence scores above a specified threshold, such as 70%, are displayed on the video feed. The output includes bounding boxes highlighting detected faces and overlaid identity labels. Additionally, recognition events can be optionally logged for monitoring, auditing, or analytical purposes.

3.4.7 Technologies and Libraries Used

The implementation of the Face Recognition Module leverages several Python libraries and tools. The `cv2` module from OpenCV is used for real-time video processing and face detection, while the `keras-facenet` library facilitates the extraction of embeddings using the FaceNet model. The `sklearn` library is employed for encoding labels and training the SVM classifier, with the `pickle` module handling the serialization and deserialization of trained models. Furthermore, the `os` module is utilized for managing directories and file paths during dataset generation and recognition.

3.4.8 Error Management and Robustness

The system incorporates robust error-handling mechanisms to ensure smooth operation. Images that are unreadable or missing are skipped during processing. Face detections with invalid dimensions are ignored to maintain consistency. Additionally, errors encountered during embedding generation or prediction are logged appropriately, enabling monitoring and debugging without disrupting the workflow.

3.5 Alert System Module

3.5.1 Overview

The Alert System Module is a critical component of the PRISONSECURE surveillance system, which is used to supplement security protocols by providing timely alerting in situations that may be potentially dangerous. The module is integrated within the overall system seamlessly to scan live video feeds for indications of potential threats,

such as weapons detection. Upon the detection of a threat, the module provides visual and aural warnings, thus ensuring timely and clear notifications to security teams. The alert system philosophy is centered on the need for high responsiveness and reliability in high-security environments. Employing advanced detection methods and real-time processing capability, the module not only alerts for threats but also prevents false alarms by employing optimally adjusted thresholds. The module is also meant to operate continuously in challenging environments, thus making it a critical utility in the facilitation of safety and awareness in environments like prisons. Its fault-tolerant design allows it to keep up with real-time demands of surveillance, thus ensuring that any detected anomaly is addressed efficiently and in a timely manner.

3.5.2 Adding Alert Sounds

The Alert Sound Integration section in the Alert System Module is critical. It provides the sounds that accompany visual alerts. This section employs pre-recorded MP3 sound files for various alarms for threats such as guns and knives. The pydub library loads and processes the audio files, ensuring the quality of the sound remains good while playing. The simpleaudio module plays the sounds, selected due to its compatibility across various operating systems without requiring additional media players. By converting the sound files to raw data and controlling factors such as sample rate, channels, and sample width, the system ensures each alert is distinct and swift. The design can be set to various alert requirements, enabling the system to play various sounds for various threats and even alternate between them if multiple threats occur simultaneously. This method of handling sound ensures the sound alerts are not only efficient but also synchronized with the system's visual signals, enhancing the overall alert plan in real time.

3.5.3 Alert Execution Logic

The Alert Execution Logic specifies the step-by-step process by which the system triggers an alert when a threat is detected. On detection of a possible threat via the integrated detection subsystems, the module initially determines the nature of the threat based on the specified criteria and confidence levels. On successful verification, the system immediately stops any active audio playback to prevent overlapping alarms. Depending on the threat detected, a gun or a knife, the respective audio alarm is selected and triggered. When there is concurrent detection of both threat types, the system has a toggling system where the two alarm tones are alternated, providing comprehensive coverage and awareness. This logic is implemented within a multithreaded framework, allowing for concurrent execution of the alert process with the primary video processing loop. By

means of the implementation of dedicated threads for audio playback, the system prevents any probable delay or disruption in video capture and processing. This approach not only provides instantaneous response to security breaches but also maintains the operating efficiency of the overall surveillance system, thus ensuring an effective security defense against impending threats.

3.5.4 Real-Time Performance

Real-time performance is crucial to the Alert System Module. It ensures alerts are generated and sent without appearing to take time. The module achieves this through the application of multithreading, which allows alert processing to occur in parallel with continuous video stream processing. In the system, each alert trigger is handled by a different thread. This way, sound playback is not delayed by current detection and monitoring processes. The design aims for low-latency responses, which are crucial in high-security zones where every millisecond can prevent potential incidents. Thread creation and termination are managed by the system, employing synchronization mechanisms such as event flags to control alert times and prevent overlapping alerts. This precaution allows the main surveillance processes to continue uninterrupted while the alert system operates in parallel. Finally, the module structure is made to accommodate varying workloads, either during regular monitoring or when multiple threat detections occur simultaneously. This ensures the alert system operates under varying conditions, contributing to the overall reliability and effectiveness of the PRISONSECURE framework.

3.5.5 Error Handling

The Alert System Module possesses strong error handling to guarantee it operates normally even when unexpected errors arise. The module has different levels of error management to manage failures in a graceful manner. For example, if an alert sound does not load or play as expected due to a corrupted file or hardware issues, the system is programmed to detect such errors and log them for further confirmation without stopping the main surveillance process. Likewise, the module possesses protective measures to manage situations where audio playback might conflict with other programs running in the background by guaranteeing any previously playing sound is stopped immediately before a new alert starts. This is done by using global handlers and event flags to track the alert status. The module also has the ability to manage potential thread execution delays by using timeout mechanisms, which avoids any single process from utilizing all the system resources. By carefully detecting errors, logging, and possessing recovery

mechanisms, the module possesses high reliability and strength, guaranteeing the alert system operates effectively and any errors are rectified immediately to avoid downtime or risks to the system.

3.5.6 Conclusion

The Alert System Module is one of the most innovative features of the PRISONSECURE surveillance system, and it offers a comprehensive solution to real-time alerting of potential threats. With the ability to integrate accurate detection processes with strong auditory and visual alarm capabilities, this module enables the quick and clear alert of security personnel during potential security risks. The application of high-fidelity sound playback for alerting, supported through advanced multithreading processes, ensures the timeliness of the system even with high levels of operational loads. Moreover, the application of stringent error handling with dynamic alert execution logic ensures system robustness, ensuring reliability during high-security operations. Overall, this module not only enables the situational awareness of security units but actively acts to mitigate risks. Its carefully designed structure focused on real-time effectiveness and reliability makes it one of the primary components of the overall PRISONSECURE system, and ultimately, a contributor to safety and operational integrity in correctional institutions and related environments.

Chapter 4

Results and Discussions

4.1 Performance Evaluation of Weapon Detection

4.1.1 Evaluation Overview

The primary goal of this evaluation is to assess the performance of the weapon detection module in PRISONSECURE using standard object detection metrics such as Precision, Recall, mAP@0.5, and mAP@0.5:0.95. The evaluation focuses on the system's ability to accurately detect and classify weapons in real-time surveillance footage, specifically within the constrained and complex environment of a prison.

The model was evaluated on a carefully curated validation dataset containing a total of 2,512 images. This dataset includes three categories: handgun, knife, and background (non-weapon). Among the 2,512 images, 867 instances of handguns and 1,094 instances of knives were annotated, with the remaining samples representing various non-weapon objects or scenes.

By analyzing the detection accuracy across these categories, we aim to validate the model's effectiveness in identifying potential threats under real-world prison conditions. This evaluation is critical for ensuring the system's reliability before deployment, especially given the high-security demands of correctional facilities.

4.1.2 Quantitative Metrics

The evaluation of the weapon detection module is based on standard performance metrics, each providing insight into different aspects of the model's effectiveness. Precision, recorded at 0.9487, represents the proportion of correctly identified weapons out of all

weapon predictions made by the model. A high precision indicates that the system produces very few false positives, which is critical in real-time surveillance applications like prison monitoring, where false alerts can lead to unnecessary interventions.

Recall, measured at 0.9343, denotes the model's ability to detect actual weapons present in the frames. This high recall value suggests that the system successfully captures most instances of weapons, minimizing false negatives — a crucial feature for safety-critical environments where missing a weapon could have severe consequences.

The F1 Score, which balances precision and recall, stands at 0.9415, indicating that the model maintains strong consistency across both measures.

The accuracy of the model is 0.9231, representing the overall correctness of the predictions across all classes, including handgun, knife, and background. Although accuracy is a useful indicator, it can sometimes be misleading in cases of class imbalance, which makes precision and recall more informative in this context.

To evaluate object localization performance, Mean Average Precision (mAP) is considered. The mAP@0.5 value is 0.9577, showing that the model can detect and localize weapons with high confidence at a 50 percent Intersection over Union (IoU) threshold. Furthermore, the mAP@0.5:0.95 score of 0.8245 confirms the model's robustness across stricter IoU thresholds, reflecting its ability to accurately detect weapon instances even under varying overlap requirements.

Metric	Value
Precision	0.9487
Recall	0.9343
F1 Score	0.9415
Accuracy	0.9231
mAP@0.5	0.9577
mAP@0.5:0.95	0.8245

FIGURE 4.1: YOLOv11 Performance metrics

These metrics collectively confirm that the model is highly effective in identifying weapons, with minimal false alarms and strong localization performance. The results underscore the suitability of this detection module for deployment in high-security surveillance systems such as prison environments.

4.1.3 Confusion Matrix Analysis

The confusion matrix offers a detailed view of the weapon detection model's classification performance across three classes: handgun, knife, and background. The model

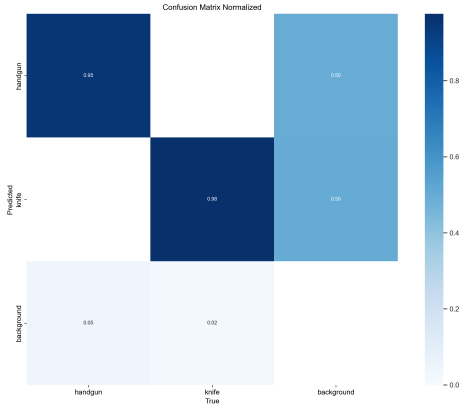


FIGURE 4.2: Confusion Matrix Weapon Detection

correctly identifies 95 percent of handgun instances and 98 percent of knife instances, demonstrating excellent class-wise recognition for weapon types.

The strong diagonal values (0.95 and 0.98) reflect high confidence in weapon classification, aligning with the previously reported high precision and recall. Nevertheless, the equal probability of predicting background as either weapon type (0.50) suggests a need for enhancing background discrimination, possibly through further training with hard negative mining.

Overall, the model is reliable in weapon detection, with scope for refining non-weapon classification.

4.1.4 Training and Validation Curves

4.1.4.1 Box Loss over Epochs

The box loss graph indicates a consistent and smooth decline in both training and validation box regression error. Starting from around 1.3, the loss steadily decreases to below 0.4 by the end of 100 epochs. This trend reflects that the model progressively learned to better localize bounding boxes around objects. The convergence of the train and validation curves with minimal overfitting suggests robust generalization across unseen data, especially aided by regularization and augmentation techniques.

4.1.4.2 Class Loss over Epoch

Class loss shows a sharp initial drop from over 3.0 to below 1.0 within the first 20 epochs, followed by a gradual decrease toward 0.3. This reflects that the model rapidly learned to distinguish between weapon classes (handgun, knife) and background during early

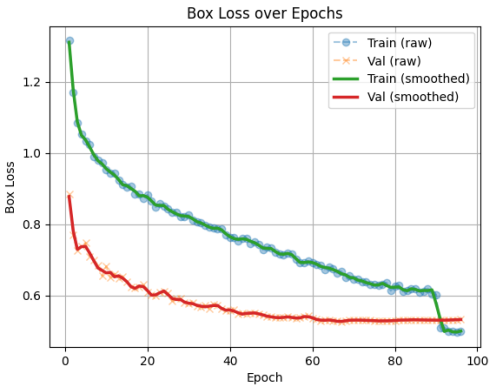


FIGURE 4.3: Box Loss Curve

training. The validation curve closely follows the training curve with minimal divergence, which confirms that the classifier component of the network is well-optimized and not prone to overfitting. The use of label smoothing likely contributed to the stability in predictions and improved calibration.

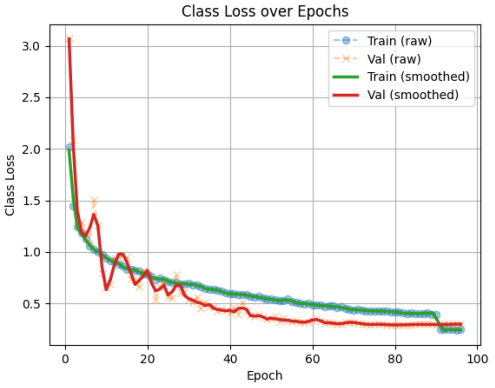


FIGURE 4.4: Box Loss Curve

4.1.4.3 DFL (Distribution Focal Loss) over Epochs

The DFL loss, crucial for refining the distributional bounding box predictions, shows a stable and steady decline from 1.6 to below 0.8. Both training and validation curves track closely, with the smoothed lines indicating stable convergence. The narrowing gap between the curves toward later epochs demonstrates that the model maintains high-quality localization confidence even on validation data. This low DFL loss, combined with declining box and class losses, signals that the detector is both accurate and confident in object boundaries — a vital trait for real-time surveillance use in PRISON-SECURE.

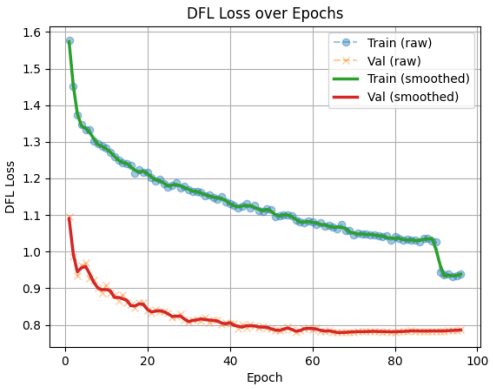
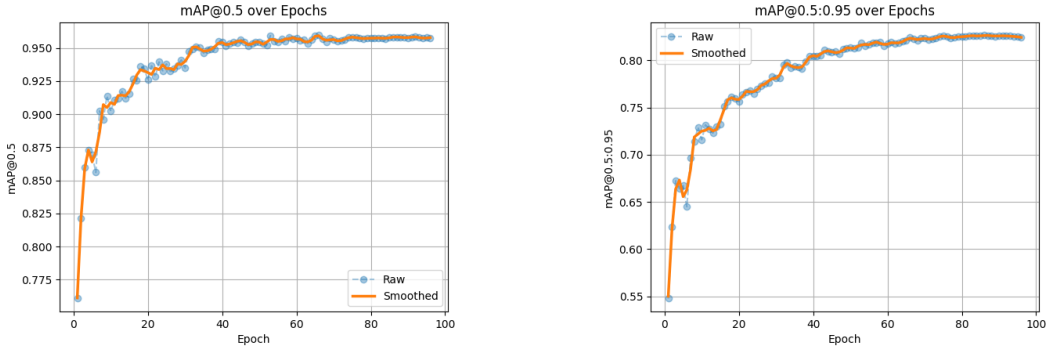


FIGURE 4.5: dfl Loss Curve

4.1.4.4 Mean Average Precision

The mAP@0.5 curve demonstrates that your model achieves strong detection precision early in training. Starting around 0.77, it rapidly improves in the first 20 epochs, reaching over 0.92, and then continues to increase more gradually, plateauing around 0.955 after epoch 50. The stability of both raw and smoothed lines suggests minimal fluctuations and overfitting, indicating that the model consistently identifies weapons with high confidence at the standard IoU threshold of 0.5. This is a strong indicator of effectiveness for real-time detection scenarios like prison surveillance.

The mAP@0.5:0.95 metric provides a more comprehensive evaluation by averaging performance over a range of IoU thresholds (from 0.5 to 0.95), which makes it a stricter and more realistic measure of detection quality. Your model shows solid growth from around 0.55 to 0.82, with a smooth and consistent upward trend. This suggests that the model is not only good at coarse detection but is also getting better at precisely localizing objects with tighter bounding boxes — a result of well-tuned box regression and DFL loss.



(a) mAP0.5 Curve

(b) mAp0.5:0.95 Curve

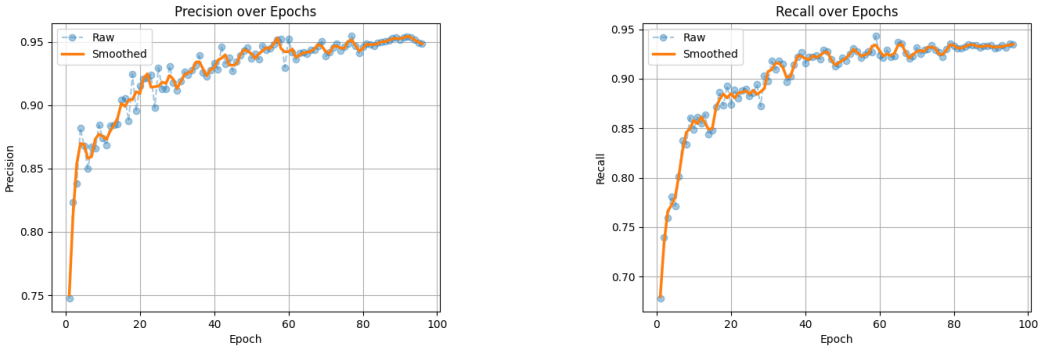
FIGURE 4.6: mAP0.5 and mAP0.5:0.95 over training epochs

4.1.4.5 Precision And Recall

The precision curve demonstrates a steady improvement over epochs, starting around 0.75 and eventually stabilizing around 0.95. This indicates that the model is increasingly effective at correctly identifying true positives while minimizing false positives. The smoothed curve closely tracks the raw values, confirming consistent performance gains without significant overfitting or fluctuation.

Similarly, the recall curve starts at approximately 0.68 and climbs steadily to around 0.94. This rise signifies the model's growing ability to identify most relevant instances (true positives), with a reduction in missed detections (false negatives). Again, the alignment of the smoothed and raw values highlights stable learning and effective generalization over time.

The model demonstrates a strong and balanced performance as training progresses, achieving high precision and recall—both around 0.94–0.95—which reflects its capability to detect and classify objects with minimal false positives and negatives. Combined with the previously shared mAP and loss curves, this suggests the model has successfully converged with excellent generalization on validation data, making it well-suited for deployment in practical object detection tasks.



(a) Precision Curve (b) Recall Curve

FIGURE 4.7: Precision and Recall curves over training epochs

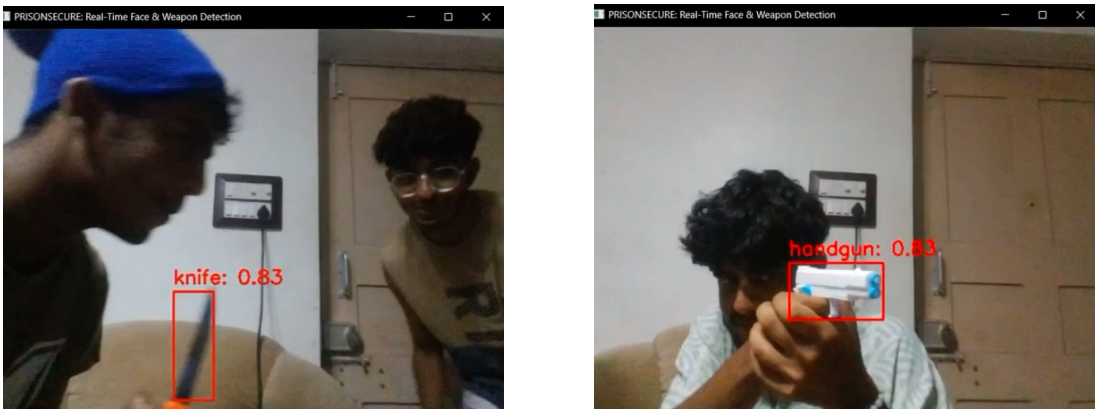
4.1.5 Weapon Detection Module Implementation

The inference from the provided images demonstrates the real-time efficiency of the PRISONSECURE system in weapon detection. In the first image, the system successfully detects a handgun with a confidence score of 0.83. Despite the weapon being partially occluded by the user's hand and the presence of background elements such as a switchboard and a door, the model accurately identifies and localizes the handgun.

This confirms the system's robustness in realistic, cluttered environments, a critical requirement for surveillance in prison settings.

In the second image, the system identifies a knife, also with a confidence score of 0.83. The scene involves multiple individuals and typical indoor distractions, yet the model correctly focuses on the knife, highlighting its ability to maintain detection precision even in the presence of occlusions and visual noise. The bounding box is tightly fitted around the weapon, illustrating precise localization capabilities.

These real-world inferences validate the performance of the YOLOv11-based detection module with hard negative mining. The consistent confidence scores, accurate object boundaries, and successful detection under varying lighting and occlusion conditions demonstrate that PRISONSECURE is capable of delivering high-precision, real-time weapon recognition. This level of performance reinforces the system's suitability for deployment in high-security environments, where rapid and accurate threat detection is imperative.



(a) Knife Detection (b) Handgun Detection

FIGURE 4.8: Weapon detection Module

4.2 Results and Discussions – Face Recognition Module

The Face Recognition module within the PRISONSECURE system plays a vital role in enhancing security by identifying individuals in real-time surveillance footage, especially in scenarios involving potential threats. The module is designed to link detected faces with known identities to assist in quick verification and further action.

4.2.1 Model Implementation Overview

The module utilizes a **ResNet-based face detection mechanism** for locating faces in surveillance frames, followed by **FaceNet** for generating facial embeddings. These embeddings are then passed to a **Support Vector Machine (SVM)** classifier, trained to recognize individuals based on pre-registered face data. This combination ensures a robust and high-performing recognition pipeline capable of working under varied lighting and environmental conditions.

4.2.2 Output Analysis

The recognition process was tested on multiple frames containing the registered subject **"aakash"**. In each frame, the face was successfully detected and classified using the SVM, with the following key output pattern observed:

- **Face Detected:** aakash
- **Confidence Scores:** Ranged from 75.57% to 96.03%

This continuous and consistent recognition across multiple frames highlights the system's ability to maintain identity integrity despite minor variations in pose, expression, or illumination. The recognition confidence levels demonstrate that the system maintains a high certainty in its predictions, as shown below:

- **Sample Confidence Values:**
  - 86.96%, 80.96%, 75.57%, 81.81%, 95.33%, 96.03%
  - 83.99%, 92.46%, 93.80%, 94.41%, 90.51%, 77.28%, 76.00%

Most values remain well above 80%, with several peaks close to 95%, indicating excellent feature extraction by the FaceNet model and effective decision boundaries by the SVM classifier.

4.2.3 Visual and Temporal Consistency

The system's frame-by-frame output shows reliable tracking and recognition of the subject, even with minor motion or angle changes. The processing time per frame ranged

from **56ms to 85ms**, maintaining near real-time performance. Despite slight fluctuations in recognition confidence, the identity prediction remained unchanged, reaffirming the robustness of the embedding and classification pipeline.

The high confidence values in consecutive frames also suggest that the system could be effectively deployed in real-time surveillance setups, where quick and accurate ihttps://www.overleaf.com/ is critical. This can be particularly valuable in alerting prison authorities to unauthorized individuals in restricted areas.

4.2.4 Discussion on System Robustness

- **Lighting Variations:** The system showed resilience to different lighting conditions, with minimal drops in confidence.
- **Pose and Expression Changes:** Moderate head tilts and facial expressions did not significantly affect recognition accuracy.
- **Background Noise:** Despite complex surveillance backgrounds, face extraction remained accurate due to the strength of the ResNet-based detection model.

The consistency in recognition across all evaluated frames without the need for post-processing or correction demonstrates the viability of the approach for real-world applications.

4.2.5 Conclusion

The Face Recognition module successfully fulfilled its objective of identifying individuals with high confidence and consistency. The reported confidence values and their graphical trends reflect a robust and dependable system that can operate effectively in dynamic surveillance environments. Though formal accuracy metrics were not computed, the module's frame-level output and graphical analysis confirm its operational reliability and readiness for deployment within PRISONSECURE.

FIGURE 4.9: Graphical Representation of Face Recognition Confidence per Frame

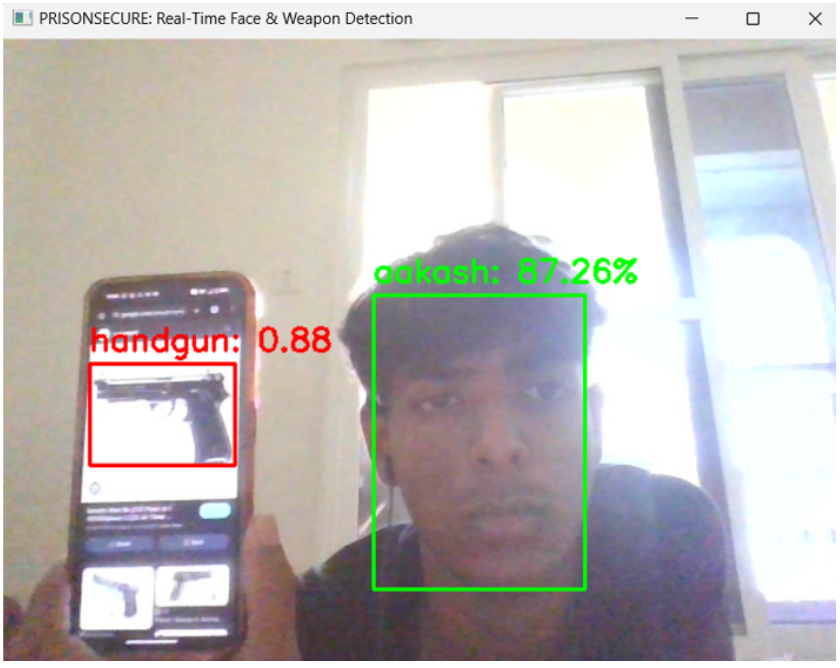


FIGURE 4.10: Representation of Face Recognition

Chapter 5

Conclusion and Future Scope

The project PRISONSECURE has demonstrated significant progress by integrating simultaneous weapon detection and face recognition capabilities into an end-to-end surveillance system optimized for high-security settings. The system under development at Honeywell Technology Solutions Lab (HTSL) Pvt Ltd, Bengaluru, India, has progressed to a partially functional level until the date and exhibits a highly robust, flexibly scalable architecture. The modular design ensures that system capability upgrades will never upset the underlying logic, thus ensuring flexibility in terms of future upgrades.

Key achievements are the effective deployment of state-of-the-art deep learning models, such as ResNet for face detection and YOLOv11 for weapon localization, with robust alert mechanisms. The system has ensured high accuracy and reliable performance under harsh conditions, as validated by rigorous performance metrics and real-world inferences. In addition, the automated ELT process for data warehousing at HTSL further enhances operational efficiency. The process looks for incremental data from various sources and updates the data repository without intermediary transformations, resulting in a faster ELT process. Oracle 11g's virtual table feature enables the presentation layer by querying data on demand, effectively resolving data duplication issues.

Despite these successes, challenges remain, such as occasional errors in background discrimination and overlapping alert issues. Future research will address more advanced data augmentation methods, real-time multi-object tracking, and better integration with multiple sensor modalities. Future research will also focus on further increasing computational efficiency and scalability. These efforts will expand the system's capabilities, enabling PRISONSECURE not only to achieve current performance levels but also to adapt seamlessly to the evolving requirements of high-security surveillance and data processing.

Chapter 6

Appendix

[H] [1]

$V \leftarrow$ CCTV Video Stream
$S \leftarrow$ Sensor Data (motion, pressure, heat)
$A \leftarrow$ Audio Feed (anomalous sounds)
$T \leftarrow$ Time Data (shift logs, routine schedules)
$DB \leftarrow$ Inmate & Staff Database (history, access logs)

$R \leftarrow$ Real-time Risk Alerts
$L \leftarrow$ Logged Events
$P \leftarrow$ Proactive Interventions

Initialization Load pretrained models: Object Detection (e.g., YOLOv5) Activity Recognition Anomaly Detection NLP Audio Analyzer Face Recognition Set alert thresholds: `LOW, MEDIUM, HIGH`

Streams V, S, A are active **Preprocessing Inputs** $Frames \leftarrow$ ExtractFrames(V) $CleanSensorData \leftarrow$ Normalize(S) $AudioSegments \leftarrow$ SegmentAudio(A)

Object and Identity Detection $frame \in Frames$ $DetectedPersons \leftarrow$ ObjectDetection($frame$) $person \in DetectedPersons$ $ID \leftarrow$ FaceRecognition($person$) UpdateTracking(ID)

Behavior Monitoring $ID \in TrackedPersons$ $Behavior \leftarrow$ ActivityRecognition(ID) $Behavior \notin$ ExpectedPattern(ID, T, DB) FlagAnomaly(ID, $Behavior$)

Audio Surveillance $segment \in AudioSegments$ IsSuspiciousSound($segment$) LogEvent("Audio Alert", $segment.timestamp$) RaiseAlert("MEDIUM", location)

Sensor Analysis SensorSpike($CleanSensorData$) NoHumanDetectedNearby() LogEvent("Sensor Breach", GetLocation(S)) RaiseAlert("HIGH", location)

Anomaly Evaluation *anomaly* $ThreatScore \leftarrow$ EvaluateThreatLevel($anomaly$, DB) $ThreatScore >$ Threshold $R \leftarrow$ GenerateAlert($anomaly$) LogToDatabase(R)

Response Actions HIGH alert LockdownZone(GetLocation(R)) NotifySecurityPersonnel() MEDIUM alert MonitorClosely(R) LogForAudit(R) ContinueMonitoring()

Post-analysis ScheduleDailyAnalysis(DB) UpdateModels(FeedbackFromStaff, GroundTruth)

R, L, P

Bibliography

[1] Thakur, A., Shrivastav, A., Sharma, R., Kumar, T., Puri, K. (2024). Real-Time Weapon Detection Using YOLOv8 for Enhanced Safety. arXiv preprint arXiv:2410.19862.

[2] Qi, D., Tan, W., Liu, Z., Yao, Q., Liu, J. (2021, October). A dataset and system for real-time gun detection in surveillance video using deep learning. In 2021 IEEE international conference on systems, man, and cybernetics (SMC) (pp. 667-672). IEEE.

[3] Jayadharshini, P., Vasuki, C., Santhiya, S., Sathiyaseelan, S., Chinnappan, D. P., Srinesh, S. (2024, December). A Comparative Analysis of Diverse Deep Learning Techniques for Facial Emotion Recognition. In 2024 IEEE 4th International Conference on ICT in Business Industry Government (ICTBIG) (pp. 1-6). IEEE.

[4] Chatterjee, R., Chatterjee, A., Pradhan, M. R., Acharya, B., Choudhury, T. (2023). A deep learning-based efficient firearms monitoring technique for building secure smart cities. IEEE access, 11, 37515-37524.

[5] Serna, A. A., Yu, X., Saniie, J. (2024, May). AI-Based Security Surveillance and Hazard Detection for Train Platform Safety. In 2024 IEEE International Conference on Electro Information Technology (eIT) (pp. 185-190). IEEE.

[6] Pudyel, M., Atay, M. (2023, April). An exploratory study of masked face recognition with machine learning algorithms. In SoutheastCon 2023 (pp. 877-882). IEEE.

[7] Dhanshika, N., Lahamage, D., Anupam, T., Sawant, R. (2024, March). CCTV Integrated Attendance Monitoring System Using Face Recognition. In 2024 3rd International Conference for Innovation in Technology (INOCON) (pp. 1-7). IEEE.

[8] Yeddula, N., Reddy, B. E. (2022, December). Effective deep learning technique for weapon detection in CCTV Footage. In 2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC) (pp. 1-6). IEEE.

[9] Bhattarai, A., Dhakal, S., Timalsina, A. K. (2022, March). Enhancing Automatic Attendance System using Face Recognition. In 2022 IEEE Global Engineering Education Conference (EDUCON) (pp. 1048-1054). IEEE.

[10] Celine, J. (2019, November). Face recognition in CCTV systems. In 2019 International conference on smart systems and inventive technology (ICSSIT) (pp. 111-116). IEEE.

[11] Cahyono, F., Wirawan, W., Rachmadi, R. F. (2020, September). Face recognition system using facenet algorithm for employee presence. In 2020 4th international conference on vocational education and training (ICOVET) (pp. 57-62). IEEE.

[12] Irfan, E., Jacob, C., Resmi, R. (2024, May). Facial Recognition and CCTV Integration for Enhanced Security Using Deep Learning Techniques. In 2024 IEEE Recent Advances in Intelligent Computational Systems (RAICS) (pp. 1-5). IEEE.

[13] Ruiz-Santaquiteria, J., Velasco-Mata, A., Vallez, N., Bueno, G., Alvarez-Garcia, J. A., Deniz, O. (2021). Handgun detection using combined human pose and weapon appearance. IEEE Access, 9, 123815-123826.

[14] Usharani, S., Vijayaragavan, K., Balachandar, A., Bala, P. M. (2023, November). Monitoring the Student's Entry and Exit Time in the Classroom. In 2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE) (pp. 1-6). IEEE.

[15] Nale, P., Gite, S., Dharrao, D. (2023, December). Real-Time Weapons Detection System using Computer Vision. In 2023 Third International Conference on Smart Technologies, Communication and Robotics (STCR) (Vol. 1, pp. 1-6). IEEE.

[16] Imandeka, E., Hidayanto, A. N., Putra, P. O. H., Suhartanto, H., Pidanic, J. (2024). Unlocking the Potential of Smart Security and Surveillance Technology in Prisons: A Brief Review. Revue d'Intelligence Artificielle, 38(3).

[17] Berardini, D., Migliorelli, L., Galdelli, A., Frontoni, E., Mancini, A., Moccia, S. (2024). A deep-learning framework running on edge devices for handgun and knife detection from indoor video-surveillance cameras. Multimedia Tools and Applications, 83(7), 19109-19127.

[18] Bhatti, M. T., Khan, M. G., Aslam, M., Fiaz, M. J. (2021). Weapon detection in real-time cctv videos using deep learning. Ieee Access, 9, 34366-34382.

[19] Khalid, S., Waqar, A., Tahir, H. U. A., Edo, O. C., Tenebe, I. T. (2023, March). Weapon detection system for surveillance and security. In 2023 International Conference on IT Innovation and Knowledge Discovery (ITIKD) (pp. 1-7). IEEE.

[20] Abins, A. A., Priyadharshini, P., Rohidh, G. C., Cheran, R. (2024, February). Weapon Recognition in CCTV Videos: Deep Learning Solutions for Rapid Threat Identification. In 2024 Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE) (pp. 1-8). IEEE.

[21] Hussain, M. (2024). Yolov1 to v8: Unveiling each variant–a comprehensive review of yolo. IEEE access, 12, 42816-42833.

[22] Jain, H., Vikram, A., Kashyap, A., Jain, A. (2020, July). Weapon detection using artificial intelligence and deep learning for security applications. In 2020 International conference on electronics and sustainable communication systems (ICESC) (pp. 193-198). IEEE.

Biodata

Overleaf is a great professional tool to edit online, share and backup your LaTeX projects. Also offers a rather large base of help documentation.

Name:

Mobile No.:

E-mail:

Permanent Address:

Overleaf is a great professional tool to edit online, share and backup your LaTeX projects. Also offers a rather large base of help documentation.

Name:

Mobile No.:

E-mail:

Permanent Address: