International Journal of Computer Applications (0975 – 8887)
Volume 186 – No.20, May 2024

Development of a Mathematical Model for Crime Detection based on YOLO Network Architecture

Oghenevovwero Zion Apene
Department of Computer Science,
Nasarawa State University, Keffi,
Nigeria

Nachamada Vachaku Blamah
Department of Computer Science,
University of Jos, Nigeria

Gilbert Imuetinyan Osaze Aimufua
Department of Computer Science,
Nasarawa State University, Keffi,
Nigeria

Morufu Olalere
Department of Cybersecurity,
National Open University, Jabi, Abuja, Nigeria

ABSTRACT
Advancements in computer vision and deep learning have led to significant progress in automated crime detection systems. This study focuses on the development of a novel mathematical model for crime detection based on the You Only Look Once (YOLOv5) network architecture. The proposed model utilizes state-of-the-art object detection techniques to identify, classify, and detect criminal activities in surveillance footage, including images and videos, focusing on critical crime categories such as weapons and violent behaviour. The model's performance is evaluated on seven classes of weapon objects and violent scenes, achieving a precision (P) of 0.842, recall (R) of 0.77, and mAP of 0.811. These results demonstrate the model's efficiency in accurately identifying and categorizing criminal activities, thereby contributing to enhancing public safety and security through the utilization of cutting-edge deep learning technologies in crime prevention and detection.

General Terms
Crime Detection Model

Keywords
Keywords: Crime detection, Public Safety, Computer Vision, Deep Learning, YOLO model

1. INTRODUCTION
Detecting and preventing crime is essential for maintaining public safety and security. Governments worldwide strive to address challenges like food security, unemployment, and crime to create a safe environment where businesses can thrive, investors feel secure, and individuals can freely express their cultural rights without fear of violence [1].

Machine learning has the potential to revolutionize security, similar to its impact on industries like healthcare, education, finance, and weather forecasting. By leveraging advanced algorithms and large datasets, machine learning can identify patterns, detect anomalies, and enhance predictive capabilities in security applications, leading to more proactive and effective security measures [2].

Advanced machine learning algorithms now provide detailed predictions and guidance for autonomous vehicles [3]. These technologies hold promise for addressing security challenges faced by Nigeria and the global community. Specifically, in security applications, algorithms are developed to detect fights, violence, and aggressive behaviour using strategies like object detection, character recognition, and various local feature extraction methods [4].

In terms of hardware technology, the use of video recording devices by individuals and security agencies has allowed the collection and utilization of video evidence in addressing and deterring criminal activities. Video surveillance is recognized as a vital resource for identifying individuals within a setting, monitoring their actions, and analysing behaviours to detect or prevent potential criminal activities [5].

Video surveillance systems have become prevalent in urban environments, transportation hubs, retail spaces, and other public areas [6]. These systems generate vast amounts of video data, which manual monitoring alone cannot efficiently process. Automated crime detection systems powered by computer vision and machine learning technologies have emerged as critical tools for law enforcement agencies to monitor and analyze video feeds effectively.

The rapid expansion of video surveillance technology has led to an increased need for automated crime detection systems capable of analyzing video feeds in real time and alerting authorities to suspicious activities. However, despite the recent surge in video surveillance research, efficiently processing the vast amount of video data captured by these systems remains a significant challenge, particularly in monitoring large areas that require more cameras and human operators [5]. Traditional surveillance methods often struggle to provide real-time insights, and the escalating volume of data presents obstacles for manual analysis. To overcome these limitations, the integration of artificial intelligence (AI), computer vision and object detection models has emerged as a transformative solution.

Among the various object detection models, the You Only Look Once (YOLO) architecture has gained prominence for its exceptional speed and accuracy in identifying objects within images and video frames [7]. This advancement has significantly reinforced real-time monitoring capabilities, effectively addressing the growing need for efficient data analysis in modern surveillance systems. The YOLO model is a popular object detection model that treats object detection as a regression problem, where the model predicts the bounding boxes and class probabilities for objects in the image or frame. The YOLO model is known for its speed and accuracy, making it a popular choice for real-time object detection applications.

The integration of object detection models like YOLO into video surveillance systems represents a significant advancement in automated crime detection. By harnessing the power of computer vision and real-time analytics, law enforcement agencies can leverage video surveillance data more effectively to enhance public safety and respond swiftly to potential threats. In this paper, a mathematical model for crime detection is proposed using the YOLO network architecture. The design and implementation of the crime detection model are discussed.

2. Computer Vision and Object Detection

Computer vision is a branch of artificial intelligence that enables machines to interpret and understand visual information from digital images or videos. It involves a series of tasks such as image classification, object detection, segmentation, and more. The field aims to replicate human visual perception using algorithms and computational techniques. Image acquisition, preprocessing, feature extraction, and recognition are components of computer vision systems [8]

Object detection is a specific task within computer vision that involves identifying and localizing multiple objects within an image or video sequence. Unlike image classification, which assigns a label to an entire image, object detection algorithms locate and classify individual objects while providing their precise spatial coordinates. Traditional object detection methods relied on handcrafted features and algorithms like Haar cascades or Histogram of Oriented Gradients (HOG) [9].

However, deep learning, particularly convolutional neural networks (CNNs), has notably improved object detection accuracy and efficiency [10]. CNNs excel in automatically extracting essential features from data without manual intervention. Notable CNN-based object detection models include R-CNN, Fast R-CNN, Faster R-CNN, YOLO (You Only Look Once), and SSD (Single Shot Multibox Detector). Among these, YOLO has gained attention for its real-time performance and ability to process entire images in a single pass [11] and this current research will focus on the YOLO object detection model.

2.1 YOLO Model

The YOLO (You Only Look Once) model is a popular deep learning-based object detection system known for its efficiency and real-time performance. Unlike traditional object detectors that use multiple stages, YOLO processes the entire image in one pass through a convolutional neural network (CNN), enabling fast and accurate object detection [10].

YOLO divides the input image into a grid and predicts bounding boxes and class probabilities directly from the grid cells. This approach allows YOLO to efficiently detect multiple objects in real time while maintaining high accuracy [10].

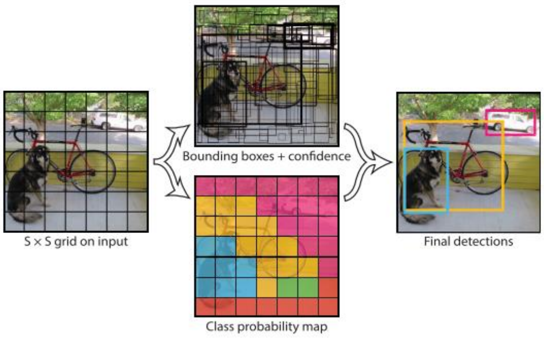

Figure 1: Yolo working principle, Source: [7]

Figure 1 illustrates the operational concept of YOLO, where detection is modelled as a regression problem. The image is divided into an S×S grid, and each grid cell is responsible for predicting B bounding boxes, confidence scores associated with these boxes, and C class probabilities. These predictions are organized into an $S \times S \times (B * 5 + C)$ tensor format.

2.2 YOLO Model Iterations

The YOLO model has undergone several iterations, the evolution of the YOLO (You Only Look Once) series in object detection has seen significant advancements, particularly with the introduction of YOLOv5. Originally pioneered by [7],YOLOv1 revolutionized object detection by proposing a unified neural network approach that predicts bounding boxes and class probabilities directly from entire images in a single evaluation step [7]. Subsequent versions like YOLOv2 (Darknet-19) and YOLOv3 improved upon the original architecture by integrating batch normalization, feature pyramid networks (FPN), and other optimizations to enhance accuracy and maintain real-time performance [10], [12].

In 2020, [13] introduced YOLOv4, which marked a significant advancement in object detection by incorporating advanced techniques like the CSPDarknet53 backbone and spatial pyramid pooling, achieving state-of-the-art accuracy benchmarks. Building upon this legacy, YOLOv5, developed by [14], represents a departure from the original YOLO series but has gained widespread adoption for its simplicity, efficiency, and strong performance.

YOLOv5 presents a more straightforward architecture compared to its predecessors, enhancing accessibility and customization for specific application requirements. Despite its simplified design, YOLOv5 maintains competitive accuracy levels similar to YOLOv4, demonstrating remarkable performance while being user-friendly [14]. The adaptability of YOLOv5, supporting various model sizes such as YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x, enables users to optimize for different trade-offs between speed and accuracy, making it a versatile option for real-time object detection applications. Figure 2 shows the network architecture of YOLOv5.

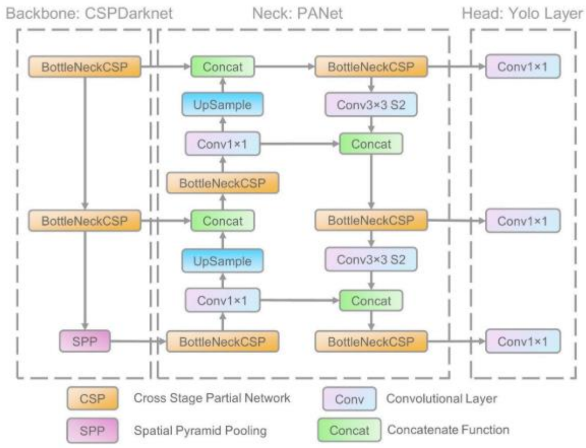

Figure 2: YOLOv5 network architecture, Source: [15]

The YOLOv5 network architecture, depicted in Figure 2, consists of three main components: the Backbone, Neck, and Head. First, the CSPDarknet Backbone extracts features from the data. Subsequently, the extracted features are passed to the PANet in the Neck for feature fusion. Finally, the Yolo Layer in the Head processes the fused features to generate detection results, including class labels, confidence scores, object locations, and sizes. This sequential process enables YOLOv5 to perform object detection tasks effectively.

2.3 Related Literature

Enhancing safety and security while mitigating life-threatening incidents presents significant challenges across diverse environments. As a result, numerous researchers have advanced crime detection and monitoring by utilizing computer vision and object detection technologies.

In an effort to monitor and segregate suspicious individuals and monitor their activities within a mall, [16] implemented a cascade optical flow technique and a modified YOLO (You Only Look Once) model using transfer learning for criminal identification. This model serves as a person detector specifically designed to identify individuals entering places such as supermarkets or restaurants, where perpetrators often conceal their identities using face masks and attempt to evade surveillance cameras.

Similarly, [17] developed an imaging model capable of generating synthetic X-ray images to aid in threat object detection using deep neural networks. The framework employed Convolutional Neural Network (CNN) techniques, specifically You Only Look Once (YOLO) and Faster Region-based CNN (FRCNN), to detect threats within airport baggage.

In [18], the authors utilised deep learning with CNN (VGG-16) and LSTM networks to monitor and classify suspicious activities in campus CCTV footage. Features are extracted from video frames and classified as suspicious or normal using LSTM for sequence analysis. The system distinguished between suspicious behaviors as defined, like using mobile phones, and normal activities, such as walking and running, in surveillance footage.

Also, [19] developed a gun detection model using computer vision and deep learning techniques. They manually collected a customized image dataset of 50 images per weapon class from Google. The YOLOv3 model trained on this dataset outperformed YOLOv2 and traditional Convolutional Neural Networks (CNNs) with lower computational costs.

3. MATERIALS AND METHODS

3.1 Proposed Method

The primary objective is to develop a mathematical model adapted from the YOLO network architecture, specifically tailored to detect seven predefined classes of criminal activities. The proposed mathematical model includes problem formulation, integration of the YOLO model, and a mathematical representation of the crime detection task. After the theoretical formulation, the model was implemented and experimentally evaluated using a custom dataset. The study sought to enhance the accuracy and robustness of crime detection while minimizing false positives and negatives, ultimately providing an efficient and reliable solution for identifying potential criminal activities through the analysis of surveillance camera footage and other image data sources.

3.2 Mathematical Model Formulation

The mathematical model presented describes a YOLO-based crime detection system designed to classify and detect seven distinct crime-related objects and activities: Grenade, Knife, Pistol, Rifle, Snatching, Stabbing, and Vandalism. The system's architecture consists of a Backbone Network, Neck Network, and Head Network, mirroring the YOLO model's architecture (adapted from Figure 2) for efficient feature extraction, and detection of potential crime. Evaluation of this model is based on performance metrics such as Mean Average Precision (mAP), precision, and recall, demonstrating its effectiveness and practical applications in real-world scenarios. The detailed mathematical model is shown as follows:

$$C = \{\text{Grenade, Knife, Pistol, Rifle, Snatching, Stabbing, Vandalism}\}$$

Model Parameters:

Backbone Network $\left(\boldsymbol{F}(\boldsymbol{x})\right)$:

Extracts high-level features from input image and video frames.

Neck Network $(\boldsymbol{P}(\boldsymbol{x}))$

Generates feature pyramids at different scales.

Head Network $(\boldsymbol{b}(\boldsymbol{x}), \boldsymbol{c}(\boldsymbol{x}))$

Predicts bounding boxes $(b(x))$ and 7-dimensional class probabilities

$$\boldsymbol{b}(\boldsymbol{x}), \boldsymbol{c}(\boldsymbol{x}) = \boldsymbol{f_{head}}((\boldsymbol{P}(\boldsymbol{x}))$$

Where:

$\boldsymbol{b}(\boldsymbol{x})$**:** Predicted bounding boxes for input

$\boldsymbol{c}(\boldsymbol{x})$**:** Predicted bounding boxes for input for x, with each element corresponding to the probability of a specific crime class $C_{1,} C_{2,}, \ldots, C_{7.}$

The function $\boldsymbol{f_{head}}$ takes the features generated by the Neck Network $P(x)$ as input and produces the predictions for bounding boxes and class probabilities.

Loss Function:

Combines BCE and Smooth L1 losses:

1. **BCE Loss:**
    - Penalizes incorrect predictions of object presence and class labels for each crime category.
    - Calculated separately for objectness score and each class probability.
2. **Smooth L1 Loss:**
    - Minimizes the difference between predicted and ground truth bounding boxes.
    - Encourages accurate object localization within video frames.

And it is given mathematically as follows:

$$\left(b, c, c_{gt}\right) = \sum_{i=1}^{7} BCE(c_i, c_{igt}) \ + SmothL_1(b, b_{gt})$$

Where:

$\boldsymbol{b}$**:** Predicted bounding boxes

$\boldsymbol{c}$**:** Predicted class probabilities (7-dimensional vector).

$\boldsymbol{b_{gt}}$**:** Ground truth bounding boxes.

$\boldsymbol{c_{gt}}$**:** Ground truth bounding boxes (7-dimensional vector)

$\boldsymbol{BCE(c_{gt} c_{igt})}$**:** Binary Cross-Entropy loss for class c_i

Relationships and Decision Rule:

- The backbone network extracts high-level features from input video frames, which are then passed to the neck network for hierarchical feature extraction. This process establishes a hierarchy of features to capture meaningful representations.

- The neck network generates feature pyramids at various scales, enabling the detection of objects of different sizes within the video frames.
- The head network utilizes these features to predict bounding boxes and class probabilities, producing the final output tensor for potential crime event detection. This output includes information about detected crime classes, bounding box coordinates, and confidence scores.
- The output layer consolidates predictions from the head network, providing structured information such as detected crime class, bounding box coordinates, and confidence scores for each crime class detected.
- **Decision Rule**: Objects are detected based on the bounding box predictions and are classified into one of the 7 crime classes based on the highest-class probability.

Total Loss:

Weighted sum of BCE and Smooth L1 losses, with adjustable weights for prioritizing classification and detection. It is given as:

$$L_{total} = \omega_{obj}L_{obj} + \omega_{class}L_{class} + \omega_{loc}L_{loc}$$

Where:

L_{total}: Total loss of the model

ω_{obj} : Weight for the objectness prediction loss L_{obj}

L_{obj} : Objectness prediction loss, part of the total loss, penalizing incorrect predictions of object presence.

ω_{class} : Weight for the class prediction loss ((L_{obj})

L_{class}: Class prediction loss, part of the total loss, penalizing incorrect predictions

ω_{loc} : Weight for the bounding box localization loss L_{loc}

L_{loc} : Bounding box localization loss, part of the total loss, minimizing the difference between predicted and ground truth bounding boxes.

Optimization

$\theta = \theta - \alpha\Delta L(\theta_1)$

Where:

θ: Model parameters

α: Learning rate

$\theta = \theta - \alpha\Delta L(\theta_1)$: Gradient of the loss function with respect to the model parameters

Evaluation Metrics:

The model will be evaluated using:

Mean Average Precision (mAP):

Mean Average Precision (mAP) is a metric used to assess the accuracy of a detector across all classes within a dataset. It combines precision and recall at various confidence thresholds, providing a holistic evaluation of how well a model can accurately detect a variety of objects. This metric is essential for evaluating the effectiveness and adaptability of YOLO-based object detection models.

The Model performance is evaluated using metrics for each of the 7 classes:

$$mAP = mean(AP_{IoU=0.5,....,}AP_{Iou=0.95})$$

Expressed as:

$$mAP = \frac{1}{N}\sum_{I=1}^{N} AP_i$$

$$mAP = \frac{(AP_{grenade} + AP_{knife} + AP_{pistle} + AP_{rifle} + AP_{snatching} + AP_{stabbing} + AP_{vandalism})}{7}$$

3.3 Dataset

The study utilized custom datasets tailored for crime detection, covering seven critical criminal activities: (grenade, knife, pistol, rifle, snatching, stabbing, and vandalism) using a structured process involving data collection, preprocessing, YOLO format annotation, and augmentation. This approach ensured the custom datasets were accurately labelled to facilitate training of the crime detection model. The dataset includes images of weapons (rifle, pistol, knife) and individuals engaging in criminal activities, such as holding and using weapons. Additionally, violent actions like stabbing and grenade use were included, along with non-violent actions activities like vandalism and occasional snatching. Video data in MP4, AVI, and MKV formats were processed to extract frames in JPEG format. The dataset consists of 7006 images, with 80% used for training, 10% for validation, and 10% for testing, averaging 844 images per class in the training set. Annotation was done in YOLO format (.txt) to label objects in the images for training the crime detection model.

3.4 Model Training Configuration

3.4.1 YOLO Architecture Selection

For the development of the crime detection system, the YOLOv5 architecture was selected as the foundational model due to its state-of-the-art real-time object detection capabilities and its ability to handle multiple objects simultaneously in a single forward pass. YOLOv5 incorporates advanced architectural enhancements such as Cross-Stage Partial Network (CSPNet) and Spatial Pyramid Pooling (SPP), enabling rapid and robust object detection in images or video feeds. These features empower YOLOv5 to identify potential criminal activities with high efficiency and accuracy by reliably distinguishing and localizing different objects and situations of interest. The architecture's real-time performance and multi-object detection capabilities make it well-suited for crime detection scenarios, where timely and accurate identification of suspicious activities is crucial for effective intervention and prevention.

3.4.2 Transfer Learning

To accelerate training and utilise prior knowledge, transfer learning was employed by initializing the YOLOv5 model with pre-trained weights from the Darknet framework, originally trained on the COCO dataset - a widely-used benchmark for computer vision tasks. Leveraging these pre-trained weights allowed the model to benefit from learned representations, reducing training time and computational requirements while enabling efficient fine-tuning for the specific crime detection task.

3.5 Implementation

In this study, the proposed model was implemented using the PyTorch deep learning framework, leveraging the premium version of Google Colab for training and testing purposes. Input images underwent resizing to 456x456 pixels and were subjected to data augmentation before serving as network inputs. The stochastic gradient descent (SGD) optimizer in PyTorch framework was employed with default loss parameters, a batch size of 16, and the model was trained with 80 epochs. This approach facilitated efficient model training and optimization for the intended crime detection application.

4. RESULTS AND DISCUSSION

Figure 3 shows the precision-confidence curve, indicating a consistent precision of 0.924 across all crime classes at a confidence level of 1.0. This uniform precision demonstrates the model's high reliability in accurately predicting various crime instances. The model's robust performance at maximum confidence highlights its effectiveness for crime detection, emphasizing its potential utility in real-world scenarios requiring precise identification and mitigation of criminal activities. This reliability across multiple crime categories underscores its practical value in enhancing crime prevention and law enforcement efforts.

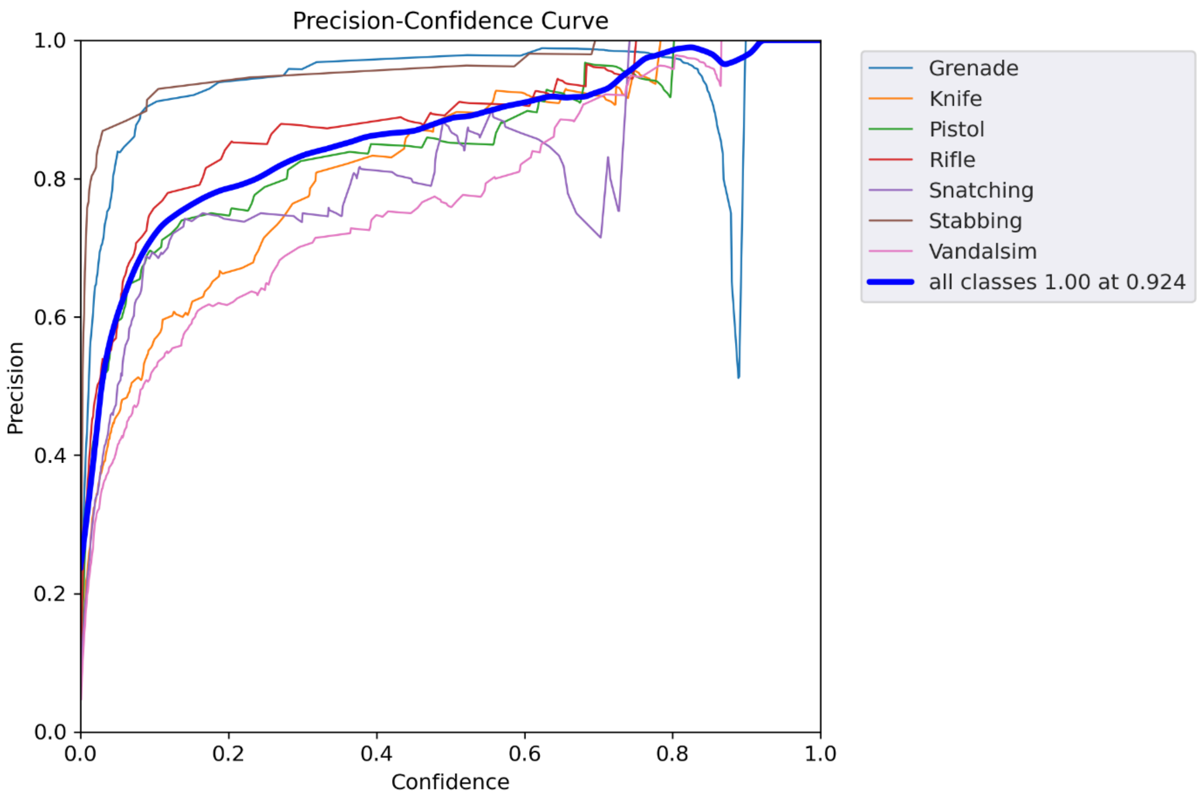

Figure 3: Precision-Recall Curve

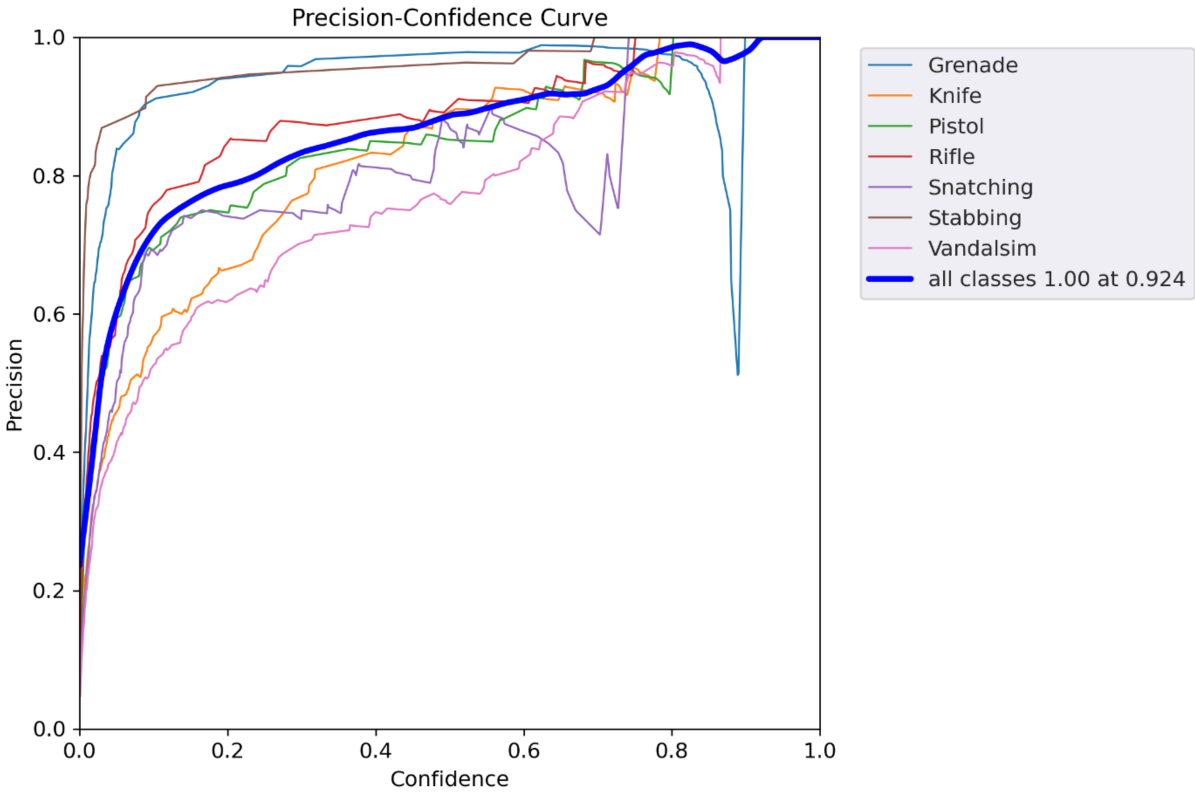

Figure 4: Precision-confidence curve

Figure 5: Detection Results Sample

Figure 4 presents the precision-recall curve, illustrating the tradeoff between precision and recall across different object classes, including Grenade, Knife, Pistol, Rifle, Stabbing, Vandalism, and an aggregated representation of all classes. Precision measures the proportion of true positives among predicted positives, while recall assesses the proportion of true positives among actual positives, making this curve particularly valuable in imbalanced class scenarios. The achieved metrics, highlighted by an aggregated mean average precision (mAP) of 0.811 at a threshold of 0.5, signify a balanced tradeoff between precision and recall. This balance is critical for applications requiring accurate detection of true positives while minimizing false positives, demonstrating the model's effectiveness across diverse scenarios.

Figure 5 depicts the crime detection results samples obtained using our model, trained on our custom dataset. The experimental findings depict successful detections of different crimes, including stabbing, rifles, knives, pistols, vandalism, grenades, and snatching. Each image displays a bounding box around the detected object with a corresponding label and confidence score. This score represents the model's level of confidence in its detection results, providing insight into the reliability and accuracy of the crime classifications and detections.

4.1 Model Evaluation

Table 1: Performance Measure

Class	Precision	Recall	mAP50
All	0.842	0.77	0.811
Grenade	0.969	0.911	0.935
Knife	0.81	0.743	0.777
Pistle	0.83	0.765	0.787
Rifle	0.873	0.795	0.849
Snatching	0.747	0.566	0.667
Stabbing	0.952	0.964	0.966
Vandalism	0.715	0.647	0.694
Overall	0.842	0.77	0.811

In Table 1, 'Class' represents distinct crime categories identified by the model. 'Precision' signifies the accuracy of crime detection for each category, while 'Recall' denotes the model's sensitivity in identifying instances of each crime class. 'mAP@0.5' corresponds to the mean Average Precision (mAP) achieved at a confidence threshold of 0.5 for each crime category. The model achieved strong precision (0.842), recall (0.77), and mAP@0.5 (0.811), demonstrating effective crime detection capabilities across diverse categories. These results emphasize the model's potential for enhancing public safety and security.

5. CONCLUSION AND FUTURE WORKS

The study successfully developed a tailored mathematical model for crime detection based on the YOLO (You Only Look Once) network architecture, focusing on specific criminal activities including grenade, knife, pistol, Rifle, snatching, stabbing, and vandalism. The study utilized the developed custom datasets for each crime category, ensuring the model's accuracy and relevance in identifying these specific criminal behaviors. The developed model demonstrated impressive performance with approximately 81% mean Average Precision (mAP). The developed model can be integrated into surveillance systems like CCTV to detect specific criminal activities such as gun and knife possession, snatching and other crime type studied in this work in real-time. Future research can expand the scope by incorporating additional crime types beyond those studied here, enhancing the model's versatility and applicability in various security scenarios. Additionally, future studies could explore real-world experimentation and alternative evaluation methods to further enhance the model's practical implementation and effectiveness.

6. REFERENCES

[1] O. Z. Apene, N. V. Blamah, and G. I. O. Aimufua, "Advancements in Crime Prevention and Detection : From Traditional Approaches to Artificial Intelligence Solutions," *Eur. J. Appl. Sci. Eng. Technol.*, vol. 2, no. 2, pp. 285–297, 2024, doi: 10.59324/ejaset.2024.2(2).20.

[2] A. Tundis, H. Kaleem, and M. Mühlhäuser, "Detecting and tracking criminals in the real world through an IoT-based system," *Sensors (Switzerland)*, vol. 20, no. 13, pp. 1–27, 2020, doi: 10.3390/s20133795.

[3] G. Bathla *et al.*, "Autonomous Vehicles and Intelligent Automation: Applications, Challenges, and Opportunities," *Mob. Inf. Syst.*, vol. 2022, 2022, doi: 10.1155/2022/7632892.

[4] N. Shah, N. Bhagat, and M. Shah, "Crime forecasting: a machine learning and computer vision approach to crime prediction and prevention," *Vis. Comput. Ind. Biomed. Art*, vol. 4, no. 1, 2021, doi: 10.1186/s42492-021-00075-z.

[5] A. J. Naik and M. T. Gopalakrishna, "Violence Detection in Surveillance Video-A survey Violence Detection in SurveillanceVideo-A survey," no. July 2016, 2017.

[6] Y. Myagmar-ochir and W. Kim, "A Survey of Video Surveillance Systems in Smart City," 2023.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.

[8] R. Szeliski, *Computer vision: algorithms and applications*, vol. 48, no. 09. 2011. doi: 10.5860/choice.48-5140.

[9] R. A. Jaafar, W. A. Jbara, and S. A. Abdulrahman, "A Review on Concept of Object Detection Techniques," no. November, 2019, doi: 10.14445/22312803/IJCTT-V67I8P115.

[10] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv Prepr. arXiv1804.02767*, 2018.

[11] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object Detection in 20 Years : A Survey," pp. 1–39, 2019.

[12] J. Redmon and A. Farhadi, "Better , Faster , Stronger," *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, vol. 7263–7271, pp. 7263–7271, 2017.

[13] A. Bochkovskiy, C. Wang, and H. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv Prepr. arXiv2004.10934.*, 2020.

[14] G. Jocher, A. Stoken, J. Borovec, L. Changyu, and ..., "ultralytics/yolov5: v3. 0," …, 2020.

[15] X. Renjie, H. Lin, K. Lu, L. Cao, and Y. Liu, "A Forest Fire Detection System Based on Ensemble Learning," *Forests*, vol. 12, no. 2, pp. 1–17, 2021.

[16] D. Harshavardhan and D. Swamy, "AN EFFICIENT CRIMINAL SEGREGATION," pp. 636–641, 2021.

[17] Dhiraj and K. J. Deepak, “An evaluation of deep learning based object detection strategies for threat object detection in baggage security imagery ☆,” *Pattern Recognit. Lett.*, vol. 120, pp. 112–119, 2019, doi: 10.1016/j.patrec.2019.01.014.

[18] C. Jyotsna and J. Amudha, “Detection from Surveillance Video,” *ICIMIA 2020 IEEE Int. Conf. Innov. Mech. Ind. Appl.*, no. Icimia, pp. 335–339, 2020.

[19] S. Narejo, B. Pandey, D. Esenarro Vargas, C. Rodriguez, and M. R. Anjum, “Weapon Detection Using YOLO V3 for Smart Surveillance System,” *Math. Probl. Eng.*, vol. 2021, 2021, doi: 10.1155/2021/9975700.