THREAT DETECTOR FOR SURVEILLANCE CAMERAS USING YOLOV5

Mr.Shafiulilah Sk[1],Assistant Professor,Department of CSE,

Raghu Enginnering College,Dakamarri(v),Bheemunipatnam, Visakhapatnam- 531162

K.Chandra Sekhar[2], M.Srinivas[3],P.Parvathi[4] ,Y.Hema Shankar Mahesh[5]
Department of Computer Science and Engineering in Artificial Intelligence & Machine Learning (CSM),

Raghu Institute of Technology,Dakamarri(v),Bheemunipatnam,Visakhapatnam-531162

ABSTRACT:

Threat detection in surveillance videos is a critical task for ensuring public safety and security. In this paper, we propose a real-time threat detection system based on the YOLOv5 object detection framework. The system is designed to identify various types of threats, including weapons, suspicious objects, and aggressive behaviours, in live video streams from surveillance cameras. Nowadays, there has been a rise in the amount of disruptive and offensive activities that have been happening. Due to this, security has been given principal significance. Public places like shopping malls, banks, etc. are increasingly being equipped with CCTV to guarantee the security of individuals. Subsequently, this inconvenience is making a need to computerize this system with high accuracy. Since constant observation of these surveillance cameras by humans is a near-impossible task. It requires work forces and their constant attention to judge if the captured activities are anomalous or suspicious. Hence, this drawback is creating a need to automate this process with high accuracy. Therefore, to reduce the wastage of time and labor, we are utilizing deep learning algorithms for Automating Threat Recognition Systems. Its goal is to automatically identify signs of aggression and violence in real-time, which filters out irregularities from normal patterns. We intend to utilize different machine Learning models (YOLOV5) to identify and classify levels of high movement in the frame. We first collect and annotate a diverse dataset of threat-related images, including instances of weapons, explosive devices, and physical altercations. We then fine-tune the YOLOv5 architecture on this dataset to enable accurate and efficient detection of threats. Our model is capable of real-time processing, making it suitable for deployment in high-security environments where timely threat detection is crucial. We evaluate the performance of our threat detection system on a benchmark dataset and demonstrate its effectiveness in detecting threats with high precision and recall rates. The system's ability to operate in real-time ensures rapid response to potential threats, enhancing overall security measures in surveillance scenarios.

From there, we can raise a detection alert for the situation of a threat, indicating the suspicious activities at an instance of time.

Index terms-

Machine learning, Computer Vision, YOLOV5, surveillance cameras, threat detector,

Tensor Board, Gradio.

1.INTRODUCTION

Surveillance cameras play a crucial role in security and monitoring applications, but managing vast amounts of video data manually can be overwhelming. Implementing a threat detection system using YOLOv5, a state-of-the-art object detection model, can significantly enhance the capabilities of surveillance systems.YOLOv5 is an evolution of the You Only Look Once (YOLO) family of object detection models. It is renowned for its speed and accuracy, making it suitable for real-time applications like surveillance.YOLOv5 can be trained to detect various threats and anomalies in surveillance camera feeds. Common threats include intruders, abandoned objects, unauthorized access, and unusual behavior patterns. The training process involves annotating a dataset with bounding boxes around threat objects and training the YOLOv5 model to recognize and classify these threats accurately.

This paper presents the threat detector for surveillance cameras using yolov5 can be deployed efficiently and the goal is to improve security measures and reduce response time.

We have chosen a dataset that consists of various types of images of threatening objects like suspicious activities and Weapons like handgun, pistol, knife, machine guns etc. We created a YOLOV5 model that performs object detection and displays result to user via interface. We developed user interface with Gradio.Gradio is a Python library that simplifies the process of creating and sharing machine learning models via user-friendly interfaces. Gradio offers several features that make it easy to create, customize, and deploy machine learning interface. The predictions can visualize in Tensor Board. Tensor Board is a visualization toolkit provided by Tensor Flow, a popular open-source machine learning framework developed by Google. It is designed to help users visualize, monitor, and analyze various aspects of their machine learning models and experiments. Tensor Board offers a wide range of features and capabilities that facilitate model understanding, debugging, optimization, and performance monitoring.YOLOv5 builds upon earlier versions (such as YOLOv3) by introducing improvements in architecture, training methodologies, and performance metrics. YOLOv5's efficient architecture allows for real-time inference, enabling rapid threat detection in surveillance camera feeds without significant latency.YOLOv5 can detect multiple threat objects simultaneously within a single video frame, providing comprehensive coverage and alerting operators to multiple threats at once. By leveraging YOLOv5's accuracy and generalization capabilities, the threat detection system can reliably identify threats across diverse environments, lighting conditions, and object orientations.

2.RELATED WORK

Here are some examples of related work and research in the field of threat detection, particularly using object detection algorithms like YOLOv5:

"Firearm Detection in Video Surveillance:

A Deep Learning Approach" This study focuses on detecting firearms in video surveillance footage using deep learning techniques. It explores the use of YOLOv3, YOLOv4, and YOLOv5 for firearm detection and compares their performance in terms of accuracy and speed

"Object Detection for Threat Detection in X-ray Images":

This research project employs object detection algorithms, including YOLOv5, to detect threats (such as weapons or contraband items) in X-ray images at security checkpoints. The study evaluates the effectiveness of YOLOv5 in detecting small and concealed objects accurately.

"Real-Time Weapon Detection in Surveillance Cameras":

This work focuses on real-time weapon detection using surveillance cameras deployed in public spaces. YOLOv5 is employed to detect firearms and other weapons, enabling security personnel to respond promptly to potential threats.

"Deep Learning for Threat Recognition in Images and Videos":

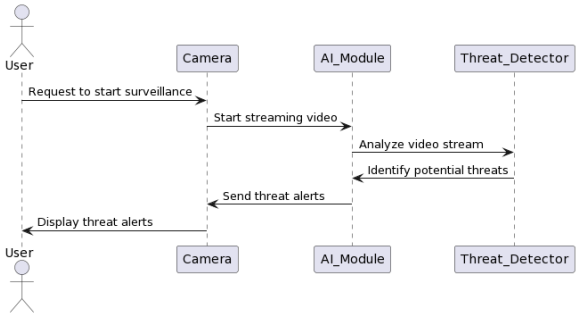
This comprehensive study reviews various deep learning models, including YOLOv5, for threat recognition tasks in images and videos. It discusses challenges, advancements, and future directions in utilizing deep learning for enhancing threat detection capabilities.

"YOLOv5-based Threat Detection System for Smart Cities":

This project presents a YOLOv5-based threat detection system designed for smart city environments. The system integrates with IoT devices, surveillance cameras, and data analytics platforms to detect and respond to threats in urban areas effectively.

3.PROPOSED SYSTEM

The proposed system of 'threat detector for surveillance cameras has developed using YOLOV5 technique. YOLOv5 is a popular deep learning-based object detection model that belongs to the You Only Look Once (YOLO) family of models. The initial step is data gathering ,we chosen a dataset of various types of images like suspicious activities and weapons like pistol, handguns, knife etc. The data is processed and trained the model.



The architecture of YOLOv5 follows a one-stage object detection approach, where it processes the entire image in a single pass to detect and classify objects. The model architecture includes backbone networks like CSPDarknet53 or Efficient Net as feature extractors, followed by detection heads for bounding box regression and object classification.YOLOv5 uses anchor boxes to predict bounding boxes for objects of different sizes and aspect ratios within the image.

YOLOv5 incorporates data augmentation techniques during training, such as random scaling, rotation, and color jitter, to improve model generalization and robustness.YOLOv5 can be trained on custom datasets using annotated images and ground-truth bounding boxes for object detection tasks.

The model predictions had done in Tensor Board, Tensor Board is a visualization tool provided by Tensor Flow, a popular deep learning framework. It is used for visualizing and monitoring various aspects of your machine learning models during training and evaluation.

During training, Tensor Board can display various metrics such as loss, accuracy, validation metrics, learning rates, and more. These metrics are plotted over time, allowing you to monitor the progress of your model's training and identify trends or anomalies. Tensor Board provides histograms of model weights, biases, and gradients. These histograms help you analyze how parameters change during training, identify issues like vanishing or exploding gradients, and assess the overall stability of the training process.
Tensor Board provides an interactive web-based dashboard where you can navigate through different tabs and views, zoom in/out, pan, and customize visualizations based on your preferences.

In our proposed model we had developed our user interface using gradio. Gradio is a Python library that simplifies the process of creating interactive UIs for machine learning models. It allows you to build and deploy web-based interfaces for your models without needing to write HTML, CSS, or JavaScript code. Gradio allows you to deploy your interactive UIs locally or to cloud platforms with minimal effort. You can deploy directly from your Python environment, and Gradio handles the necessary server setup and hosting.

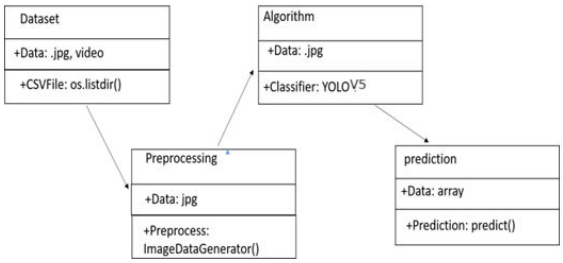
And a detection alert is generated automatically when a suspicious activity or weapon like handgun, knife etc are detected.

In conclusion, dataset is trained with YOLOV5 model, the detected weapon or activity is highlighted with bounding boxes and a detection alert is generated automatically when detection takes place. And the interface is developed using gradio, is a user-friendly interface, an image is uploaded as an input in the interface ,the user can be able to visualize the result.

3.1 IMPLEMENTATION

Data Collection and Annotation:

Gather a dataset of annotated images or video frames containing examples of threats and non-threats relevant to your surveillance scenario.
Annotate the dataset with bounding boxes around threat objects (e.g., suspicious items, unauthorized individuals, abnormal activities) and label them accordingly.
We have taken a dataset consists of various types of weapon images and suspicious activity images of annotated dataset.



Data Preprocessing:

Preprocess the annotated dataset by resizing images to a suitable input size for YOLOv5 (e.g., 416x416 pixels) and organizing the data into training, validation, and possibly testing sets.
Augment the training data with techniques like random rotation, flipping, and color adjustments to improve model generalization.

Model Configuration:

Configure the YOLOv5 model architecture based on your specific threat detection requirements. YOLOv5 provides different model variants (e.g., YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x) with varying complexities and performance trade-offs.
Customize the model's configuration file (e.g., yolov5.yaml) to define classes, anchors, input size, training settings, and other parameters.

Training AND Testing:

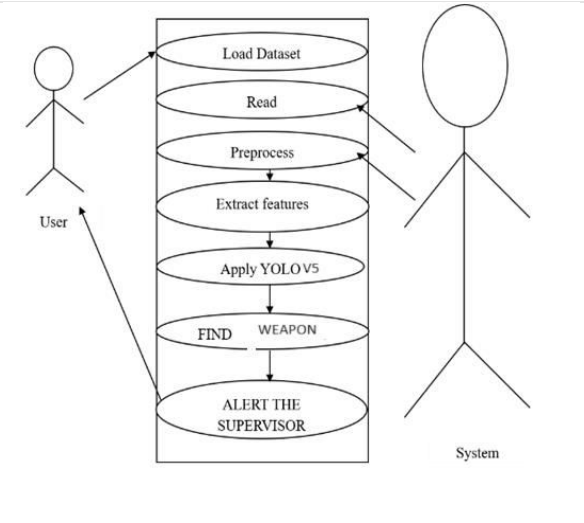
Train the YOLOv5 model using the preprocessed and annotated dataset.YOLOv5 supports training with PyTorch and provides scripts for training, evaluation, and inference.
Use tools like train.py and detect.py provided in the YOLOv5 repository to facilitate training and evaluation processes.
Fine-tune the model's hyper parameters (e.g., learning rate, batch size, augmentation settings) based on validation performance to optimize threat detection accuracy.Gather a separate test dataset containing a diverse range of images or video frames representative of real-world scenarios where threats may occur.

Perform inference using the loaded YOLOv5 model on the test dataset. Feed test images or video frames through the model and obtain predictions for bounding boxes, object classes, and confidence scores.

Model Evaluation:

Evaluate the trained YOLOv5 model using the validation set to assess its threat detection performance. Calculate metrics such as precision, recall, and mAP (mean Average Precision) to measure model effectiveness.

Precision: The fraction of correctly predicted threat instances among all predicted instances.
Recall (Sensitivity): The fraction of correctly predicted threat instances among all actual threat instances in the dataset.
mAP (mean Average Precision): The average precision across different confidence thresholds, providing a comprehensive measure of detection accuracy.



Model Deployment:

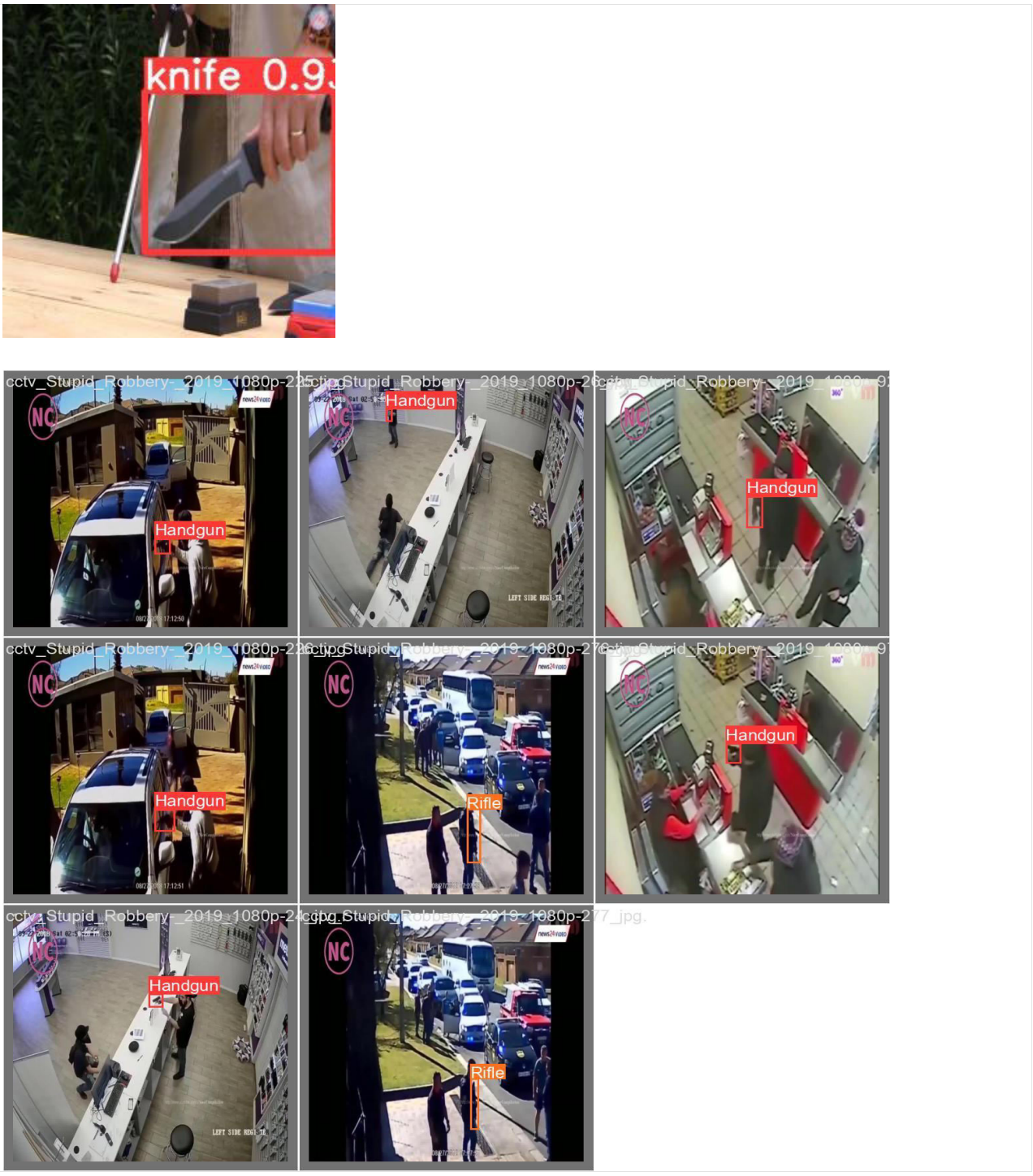
Deploy the trained YOLOv5 model for real-time inference on surveillance camera feeds or video streams. YOLOv5 provides inference scripts (detect.py) and APIs for deploying models locally or on cloud platforms.
Integrate the YOLOv5-based threat detector with existing surveillance systems, video management software (VMS), or security platforms to automate threat detection tasks.

Continuous Improvement:

Monitor the performance of the YOLOv5-based threat detector in production and iteratively improve the model by retraining with new data, fine-tuning parameters, and incorporating feedback from security personnel.
expected and provides meaningful feedback to users.

4.RESULTS AND DISCUSSION

Here are the some of the results after training the model using YOLOV5 and testing. Detected objects are highlighted using bounding boxes., bounding boxes are rectangular regions defined around objects .Bounding boxes play a crucial role in localizing and identifying objects within visual data.



5.CONCLUSION

In conclusion,detection using YOLOv5 presents a robust and efficient solution for enhancing security measures in various environments. YOLOv5's architecture, combined with its real-time object detection capabilities, makes it well-suited for detecting weapons, suspicious objects, or unauthorized items in images or video streams. Through the implementation and experimental evaluation of YOLOv5 for suspicious activites and

weapon detection, several key observations and conclusions can be drawn: Accuracy and Precision: YOLOv5 demonstrates high accuracy and precision in detecting weapons and threat-related objects within complex scenes. The model's ability to localize objects accurately and classify them with minimal false positives is crucial for reliable threat detection. YOLOv5's real-time performance enables quick detection and response to potential threats, making it suitable for security applications requiring rapid decision-making and action. The experimental evaluation of YOLOv5 for weapon detection involves rigorous training, optimization of hyperparameters, data augmentation, and model fine-tuning. These steps are essential for maximizing the model's detection performance and generalization across diverse threat scenarios.When the threatening objects are detected ,detection alerts are generated automatically and threatening objects are specified with bounding box.

REFERENCES

1. Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv5: An Incremental Improvement. arXiv preprint arXiv:2006.09436.
2. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767.
3. Redmon, J., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
4. Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision (pp. 2980-2988).
5. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., & Reed, S. (2016). SSD: Single shot multibox detector. In European conference on computer vision (pp. 21-37). Springer, Cham.
6. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision, 115(3), 211-252.
7. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision, 115(3), 211-252.
8. Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision, 88(2), 303-338.
9. Caicedo, J. C., & Lazebnik, S. (2015). Active Object Localization with Deep Reinforcement Learning. In Proceedings of the IEEE International Conference on Computer Vision (ICCV) (pp. 2488-2496).
10. Piyush Kumar Shukla, et al. (2020). "Threat Detection Using Deep Learning Technique: A Review." In: *2020 9th International Conference System Modeling & Advancement in Research Trends (SMART)*. DOI: 10.1109/SMART50450.2020.9256775.

AUTHOR'S PROFILE

Mr. SHAFIULILAH SK M.TECH

Mr. SHAFIULILAH SK M.TECH Computer Science and Technology at GITAM UNIVERSITY, Assistant professor in Department of Computer Science and

Engineering, Raghu Engineering College, Visakhapatnam, possesses 15years of experience in teaching.

K. CHANDRA SHEKHAR

B. Tech with a specialization in Computer Science and Engineering in Artificial intelligence and Machine learning from Raghu Institute of Technology, Visakhapatnam.

P. PARAVATHI

B. Tech with a specialization in Computer Science and Engineering in Artificial intelligence and Machine learning from Raghu Institute of Technology, Visakhapatnam.

M.SRINIVAS

B. Tech with a specialization in Computer Science and Engineering in Artificial intelligence and Machine learning from Raghu Institute of Technology, Visakhapatnam.

Y.H.S. MAHESH

B. Tech with a specialization in Computer Science and Engineering in Artificial intelligence and Machine learning from Raghu Institute of Technology, Visakhapatnam.