

Object Detection and Tracking for Smart Surveillance with Automated Email Alerts

Ashish Kumar¹, Prof. Dr. Renu Bagoria¹

¹Jagannath University, Jaipur, India

ashishkumarmjl94@gmail.com

Abstract

The use of surveillance systems in private spaces is getting more common and that is why we need security solutions that are smart. These smart security solutions can find activities right away. If we have to watch camera feeds all the time it is a lot of work and we might not see something important. To solve this problem we made a surveillance system that can find people and weapons like guns and knives in real time. It can also track objects. Send email alerts automatically. Our system can use webcam, prerecorded video files and CCTV camera streams. This means our system can be used in different places from small local areas to big surveillance installations. We use the YOLOv8 model to find people and weapons in each video frame. Then we use the DeepSORT algorithm to track these objects across frames. When our system finds a weapon it takes a picture of that frame. Sends an email with the picture to the people in charge. This helps us report threats quickly. We have proof that can help us make decisions right away. Our system is good because it can find things track them and send notifications all in one. We made our system using Python and some other tools like OpenCV and the Ultralytics YOLOv8 framework. We tested our system to see how well it works. It does a great job of finding people and weapons. It also sends alerts on time.

Our system is good for places like airports, railway stations, schools and government buildings where we need to find threats. The results show that using object detection and email notifications can make surveillance systems better and more responsive. The smart surveillance system can find people and weapons. It is a good tool, for security. It can help us stay safe in different places.

Keywords - Object Detection, YOLOv8, DeepSORT, Smart Surveillance, Weapon Detection, Human Detection, Email Alert System, Real-Time Monitoring, CCTV Analytics.

I. Introduction

Object detection is a part of computer vision these days. The main goal of object detection is to find things in a picture or video and show where they are. This is different from looking at a picture. Saying what is in it. Object detection does two things: it finds the thing. It says what the thing is. This helps computers understand what is going on in a scene and make choices based on what they see. Object detection is really good, at this because it can find lots of things in a picture and tell us what each thing is.

In the few years artificial intelligence and deep learning have really taken off and made object detection systems a lot better. Now we have algorithms that can find lots of objects at the time and do it very accurately. They can even look at video. Figure things out right away. This means object detection can be used for lots of things like cars that drive themselves healthcare, robots, factories, watching traffic and smart security systems.

Smart surveillance is one area where object detection can really help. We have security cameras in lots of places like airports and railway stations and shopping malls. They are also in schools and banks and offices and government buildings. These cameras record. Lots of video. It is very hard to watch all these videos at the time. You need to pay attention all the time. You might miss something important. Security people can get tired or distracted and miss something. They have to watch screens at the same time which is not easy. Smart surveillance is very useful, in these situations. Object detection can help security personnel do their job better.

One big threat, in places where people are being watched is when someone has a gun or a knife. Finding these things early can stop things from happening and keep people safe. Old security systems record video and rely on people to

spot something strange. This can take a time to figure out. A system that can automatically find guns and knives and tell the people right away can help get a response much faster and make the area safer.

Object tracking is really useful for surveillance systems. It helps keep track of the things that have been found. When the system sees a person or a weapon it gives that thing a number. Then it follows that thing as it moves around. This helps a lot because it does not say that it has found the thing many times. Object tracking also shows where the thing is going and how it is moving. This makes it easier to watch people who are acting suspiciously. Object tracking is very important, for surveillance systems because it helps us understand what people are doing over time. We can use object tracking to see what people are doing. How they are behaving.

Object detection used to be done with things like Haar- features and Histogram of Oriented Gradients. There were also methods like Scale-Invariant Feature Transform and Speeded-Up Robust Features. These were often used with machine learning classifiers like Support Vector Machines and AdaBoost. The old object detection methods worked well in simple situations.. They had a lot of problems when the lighting changed or the object was a different size. They also got confused if the object was turned in a way or if the background was really complicated. Object detection methods like these were not very good, at handling these kinds of changes.

The introduction of learning changed object detection a lot. It helped neural networks learn features from the data. Region-based methods like R-CNN, Fast R-CNN and Faster R-CNN made accuracy better.. They used a lot of computer power. Later single-stage detectors such, as SSD and YOLO were made. They gave a mix of speed and accuracy.

The YOLO framework is really simple. It looks at object detection as one problem. The YOLO framework predicts where objects are and what they are in one step. This makes it fast. The YOLO framework can look at things in time and it is still really good at finding objects.

The first YOLO architecture had 24 layers that looked at things. Then it had some fully connected layers. This showed that the YOLO framework could find objects really fast with one neural network. The YOLO framework has gotten better over time. The new versions of the YOLO framework are better, at finding the spot of an object and they work well even when things are hard to see. The YOLO framework is also easier to use.

YOLOv8 is really good at what it does. It is one of the versions of the YOLO family. YOLOv8 has some things that make it better like the way it is built how it finds things and how it learns. All these things help YOLOv8 find things accurately and it can still do it really fast. The YOLOv8 model is made using the Ultralytics framework. This framework makes it easy to teach YOLOv8 test it and use it. Because YOLOv8 is so good and easy to use it is great for things like security cameras. YOLOv8 is very good, for these kinds of jobs.

To keep track of things over time this work uses the DeepSORT algorithm. The DeepSORT algorithm is a way to figure out where things are moving and what they look like. It does this by using something called a Kalman filter to guess where things are going. Then it looks at what these things look like to make sure it is tracking the right thing. This way the system can watch things at the same time and it will not send out lots of warnings about the same problem. The DeepSORT algorithm is really good, at giving each thing it finds a tracking ID so it can keep watching it.

This research is, about a system that watches things. It uses YOLOv8 to find objects and DeepSORT to track objects. The system looks for two kinds of things: people and weapons. It can get information from three sources: webcam videos, videos that are already recorded and CCTV camera videos. This system is good because it can work with kinds of information so we can use it to test things and also use it in the real world with the smart surveillance system.

The system uses YOLOv8 to look at each picture that comes in. YOLOv8 checks for people and things like guns. When it finds something it tells DeepSORT. DeepSORT then gives each thing a number so it can keep track of them from one picture to the next. If the system finds a gun it takes a picture of what's happening right now and sends that picture to someone who is allowed to get these messages. This way people get to see what is going on and can do something about it away. The system can do all of this without someone watching. YOLOv8 is used to detect people and things like guns, in each picture that comes in.

The system we are talking about has a lot of things about it. It can find things and track them at the time it can take information from many places it makes it easier for people who have to watch everything all the time and it sends out warnings very quickly. The system is really good for places, like airports, railway stations, schools, colleges, shopping malls and government facilities because they need to know if something bad is happening away.

The main contributions of this research are summarized as follows:

1. Development of a real-time surveillance system for human and weapon detection.
2. Integration of YOLOv8 for accurate object detection.
3. Use of DeepSORT for robust multi-object tracking.
4. Support for webcam, video file, and CCTV stream inputs.
5. Automatic snapshot capture and email-based alert generation.
6. Performance evaluation using precision, recall, mAP, FPS, and alert response time.

The remainder of this paper is organized as follows. Section II reviews related work in object detection and intelligent surveillance. Section III describes the proposed methodology. Section IV presents the experimental setup. Section V discusses the results and performance analysis. Section VI outlines applications, limitations, and future work. Section VII concludes the paper.

II. Related Work

Object detection has changed a lot over the twenty years. It used to be about methods but now it is all about deep learning models that are very accurate. This has made a difference in many areas including watching people self-driving cars, healthcare and factories. In watching systems object detection is very important. It helps find objects and activities right away.

On object detection used special features and regular machine learning. People used things like Haar- features, Histogram of Oriented Gradients and Scale-Invariant Feature Transform to get information from pictures. These features were often used with classifiers like Support Vector Machines. Although these methods worked well in controlled places they had trouble with things like light, size and background. Then Convolutional Neural Networks came along. Changed everything. They let models learn features from data on their own. One of the successful methods was Region-based Convolutional Neural Network. It made suggestions for areas. Classified each one separately which made detection more accurate. However it was slow and not good for real-time use.

Fast R-CNN made things faster by using the picture and then classifying areas. Faster R-CNN did even better by introducing a network that made suggestions for objects. These two-stage detectors were accurate but still slow. To make things faster people made single-stage methods like SSD and YOLO. These methods do detection and classification in one step, which makes them much faster. YOLO treats detection like a math problem. Predicts boxes and classes at the same time. The first YOLO showed that real-time detection was possible and still accurate.

The YOLO family has gotten better over time. New versions have networks, loss functions and training. YOLOv5 is easier to use. YOLOv8 is even better at detection. Many studies have looked at object detection for security. They have shown that deep learning models can detect people, cars and dangerous objects in video. Detecting guns and knives is especially important for safety. Object tracking is also important. While object detection finds objects in each frame tracking follows them over time. This helps understand movement and prevents alerts.

Simple Online and Realtime Tracking introduced a fast tracking method. DeepSORT added features from deep neural networks, which made tracking better. DeepSORT is widely used because it can track objects in real time. New surveillance systems often combine YOLO detectors with DeepSORT trackers. This helps understand what is happening and follows objects over time. Research has also looked at using object detection on devices that're not very powerful. One study used a two-stage approach that identified areas of interest and then used YOLO. This made detection faster and more efficient. Automated alerts are also important. Some systems send messages, alarms or emails when they detect something. This helps people respond faster without having to watch video all the time.

Even though there has been a lot of progress many systems do object detection and do not provide a complete solution. Some lack tracking or alerts. The system presented here uses YOLOv8 for detection DeepSORT for tracking and

automated email alerts. It can use webcam feeds, video files or CCTV streams which makes it good for real-world use. By combining detection, tracking and alerts this system is practical and efficient, for surveillance.

Table I. Summary of Existing Object Detection and Tracking Methods

Method	Type	Strength	Limitation	Role in This Work
HOG + SVM	Traditional Detector	Simple and lightweight	Low accuracy in complex scenes	Used as a conventional baseline
R-CNN	Two-Stage Detector	High detection accuracy	Slow inference	Early deep learning detector
Faster R-CNN	Two-Stage Detector	Improved accuracy and speed	Computationally expensive	Benchmark detection method
SSD	Single-Stage Detector	Good speed	Lower small-object accuracy	Real-time detector alternative
YOLOv4	Single-Stage Detector	Fast and accurate	Complex training setup	Predecessor of YOLOv8
YOLOv8	Single-Stage Detector	High speed and accuracy	Requires GPU for best performance	Selected for human and weapon detection
SORT	Object Tracking	Lightweight and fast	Identity switches during occlusion	Baseline tracking algorithm
DeepSORT	Object Tracking	Robust identity preservation	Higher computational cost	Selected for multi-object tracking

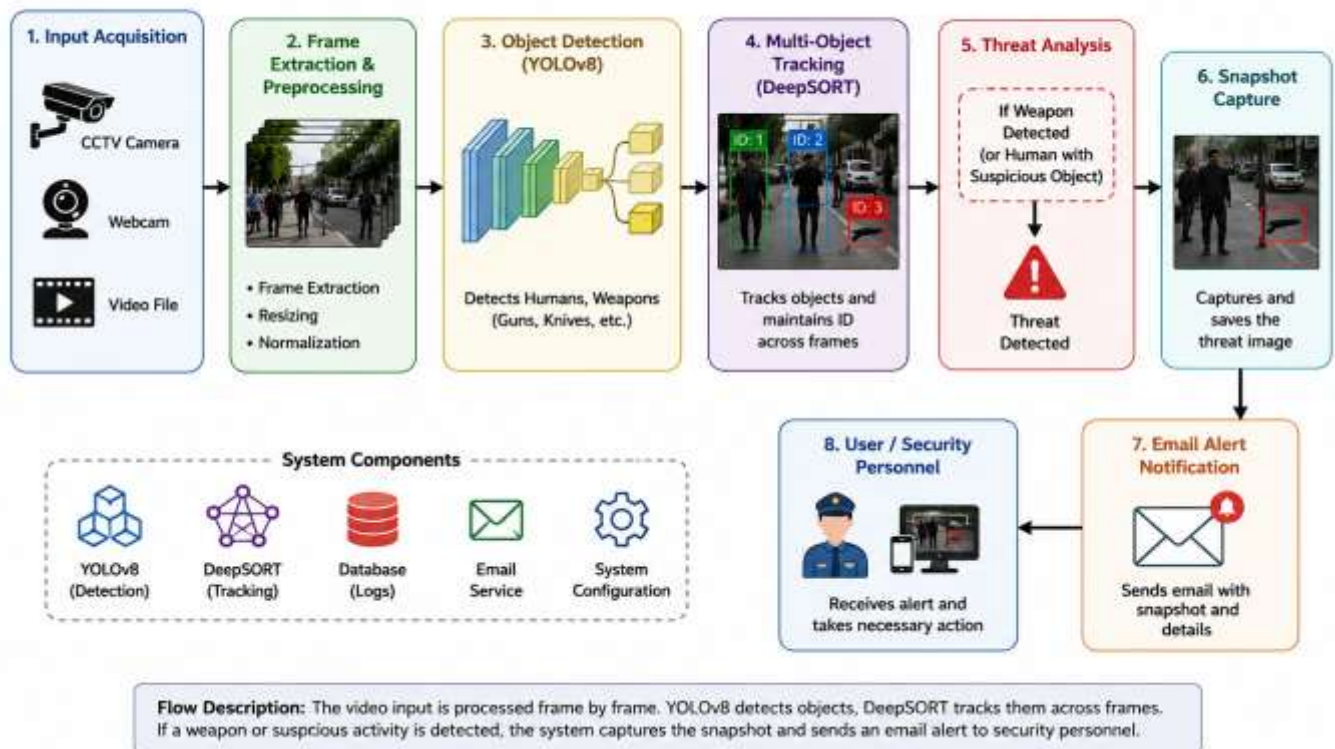
III. Proposed Methodology

The smart surveillance system is made to find people and things like weapons in videos. It looks at videos from places like a webcam or a security camera. The system is very flexible. Can be used in many situations.

The system works by taking the video and breaking it down into parts called frames. Each frame is made to be the size and type so it can be used by the detection model. The detection model is called YOLOv8. It looks at the whole frame to find people and things like weapons. It is set up to find two things: people and weapons. If it is not sure about something it ignores it.

Fig.1: System Architecture Diagram / Flow.

Fig. 1. System Architecture of the Proposed Smart Surveillance System



The things that are found are then given a number so they can be tracked. This is done by the DeepSORT tracking algorithm. It uses a filter and looks at how things look to make sure it is tracking the right thing. This way the system can see where things are moving. Does not send too many alerts.

If the system finds a weapon it takes a picture of the frame. The picture shows the person and the weapon. Has information like the time it was found. The picture is saved with a name that has the time on it. Then the system sends an email to someone with the picture. The time it was found. This way security people can see what is happening. Do something about it.

The system keeps looking at the video. Does not send too many emails about the same thing. It does this by keeping a list of the things it has already sent emails about. This way the system can keep watching and only send emails when it needs to. The system is always looking at the video to find people and things like weapons. It is a surveillance system that helps keep people safe.

The system has an architecture that is shown in a picture. The picture shows how the system works and how it is all connected. The system starts with a video. Then it is broken down into frames. The frames are then used by the detection model to find people and things like weapons. The things that are found are then. If they are a weapon the system sends an email. The system is a surveillance solution that can be used in many places.

The system is very good at finding people and things like weapons. It is also very good at tracking them and sending emails when it needs to. The system is a surveillance system that helps keep people safe. It is an useful tool for security people. The smart surveillance system is a good system. It is made to find people and things like weapons in videos. The system is very flexible. Can be used in many situations.

The system works by taking the video and breaking it down into parts called frames. Each frame is made to be the size and type so it can be used by the YOLOv8 detection model. The YOLOv8 detection model is a good model that can find people and things like weapons. It is set up to find two things: people and weapons. If it is not sure about something it ignores it. The YOLOv8 detection model is an important part of the smart surveillance system.

The smart surveillance system is an useful tool. It helps security people keep people safe. The system is very good at finding people and things like weapons. It is also very good at tracking them and sending emails when it needs to.

The smart surveillance system is a good system. It is made to find people and things like weapons in videos. The system is very flexible. Can be used in many situations. The smart surveillance system is an useful tool, for security people.

When we look at the math behind it the input frame at time t is called F_t . The detection process that YOLOv8 does can be written as:

$$D_t = f_{YOLOv8}(F_t)$$

This means D_t has all the objects that were found along with what type of object they're a box around them and a score that says how sure we are about it.

The next step is tracking. We can write this as:

$$T_t = f_{DeepSORT}(D_t)$$

Here T_t is the set of objects we are keeping track of. Each one has its own special identity.

If we find an object that's a weapon **this means there is an object, in T_t that is a weapon.**

When this happens the system takes a picture, which we call S_t and sends out an email warning, which we call E_t .

The method we are talking about has a lot of things going for it. It does a things all at once like finding things tracking them and sending out alerts. It can work with a lot of video sources so it is easy to use in different situations. The YOLOv8 system is really good at finding things. It works in real time. The DeepSORT system is also very good at keeping track of things. The system can also send out emails automatically which is really useful because it sends out notifications away and includes pictures as proof.

The whole process of the system we are talking about works, like this: it gets video frames from the source we choose gets them ready uses YOLOv8 to find objects uses DeepSORT to track the objects checks if there is a threat and if it finds a weapon it takes a picture and sends out an email. This all works together to make an practical way to watch things, which is what we need for security systems.

IV. Experimental Setup

The smart surveillance system was made to see how well it works. It is supposed to find people and weapons in time track objects and send email alerts automatically. This part talks about the information we used to teach and test the surveillance system. We also talk about the computers and programs we used how we set up the system and how we measured how well the smart surveillance system works.

A. Dataset Description

The system we made is supposed to find two kinds of objects: beings and weapons. The weapons we are looking for are things like guns and knives that people often try to bring into places they should not. To make the YOLOv8 model work well we put together a set of pictures by combining images we found online with some we got from surveillance videos.

We made sure the pictures show people and weapons in all sorts of situations like inside and outside with lighting and with things partly hiding them. The people in the pictures are standing, walking or partly hidden and the weapons are shown from angles and distances. This helps the model get better at finding these things in life when it is used for surveillance. We drew boxes around the things we wanted the model to find in each picture. We saved these boxes in a format that the YOLOv8 model can understand, with information about what is inside the box and where it is.

We split all the pictures into three groups: one for teaching the model one for checking how well it is doing and one, for testing it when it is all trained. The teaching group is used to show the model what to look for the checking group is used to make sure the model is not getting too good at those pictures and the testing group is used to see how well the model really works when it is all done.

B. Hardware and Software Configuration

The system was made using Python. It was run on a computer that had a special graphics card to make deep learning work faster. The things we used to make this system include OpenCV for looking at videos the Ultralytics YOLOv8 framework for finding objects DeepSORT, for tracking objects and the Python SMTP library that comes with Python for sending emails. We used these things to make the system work properly. The system uses Python. It runs on a computer with a special graphics card.

A representative experimental environment is summarized below.

Component	Specification
Operating System	Windows 10/11 or Ubuntu Linux
Programming Language	Python 3.x
Detection Framework	Ultralytics YOLOv8
Tracking Algorithm	DeepSORT
Image Processing Library	OpenCV
Deep Learning Library	PyTorch
Notification Module	SMTP Email Service
GPU	NVIDIA CUDA-enabled GPU
Development Tool	Jupyter Notebook / VS Code

The modular design allows the system to be deployed on different hardware platforms with minimal modification.

C. Model Training Configuration

The YOLOv8 model was trained using a method called transfer learning with weights that were already trained. This way the YOLOv8 model can learn things faster. Do a better job by using things it learned from big datasets.

When the YOLOv8 model was being trained people had to pick the size for the pictures that were being used how many pictures to look at together how fast to learn and how many times to go through everything. They wanted to make sure the YOLOv8 model was good at finding things and not too slow. The YOLOv8 model also used some tricks like flipping pictures making them bigger or smaller and changing how bright they were so it could learn to recognize things in situations and be better, at its job.

Table II. Training Parameters of the YOLOv8 Model

Parameter	Value
Model	YOLOv8n / YOLOv8s
Input Image Size	640 × 640
Batch Size	16
Number of Epochs	50–100
Optimizer	Adam
Learning Rate	0.001
Confidence Threshold	0.50
IoU Threshold	0.45
Data Augmentation	Flip, Scale, Brightness Adjustment
Pretrained Weights	COCO Dataset
Framework	Ultralytics
Deep Learning Library	PyTorch

The final model weights corresponding to the best validation performance were used for inference.

D. Testing Procedure

The model we trained was tested with three kinds of input. We used webcam feeds, prerecorded video files and CCTV camera streams to see how it worked. We wanted to test the model with these input sources to find out how well it performed. The model was tested with live webcam feeds. It was also tested with prerecorded video files and CCTV camera streams. Each input source was processed frame by frame. YOLOv8 generated object detections, DeepSORT assigned unique IDs and tracked objects, and the threat analysis module checked whether any weapon was present.

Whenever a weapon was detected, the system captured the current frame and triggered an email alert with the snapshot attached. The timestamp of the detection event and the alert transmission time were recorded to estimate the end-to-end response time.

This testing strategy ensured that the system was evaluated under conditions similar to real-world surveillance operation.

E. Evaluation Metrics

The performance of the proposed system was assessed using widely accepted object detection and tracking metrics.

Precision measures the proportion of detected objects that are correct.

$$\text{Precision} = \frac{TP}{TP + FTP}$$

Recall measures the proportion of actual objects that are successfully detected.

$$\text{Recall} = \frac{TP}{TP + FNT}$$

F1-Score provides the harmonic mean of precision and recall.

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Mean Average Precision (mAP) summarizes detection accuracy over different recall levels and Intersection over Union (IoU) thresholds.

Frames Per Second (FPS) measures the number of video frames processed per second and indicates real-time capability.

Alert Response Time is defined as the time interval between weapon detection and successful delivery of the email notification.

These metrics provide a comprehensive assessment of both model accuracy and practical system performance.

V. Results and Discussion

This section presents the performance of the proposed smart surveillance system in terms of object detection accuracy, tracking consistency, and automated alert generation. The system was evaluated using prerecorded videos, live webcam feeds, and CCTV streams containing both normal scenes and scenarios involving weapons. The experimental results demonstrate that the integration of YOLOv8, DeepSORT, and email notification provides an effective and practical solution for real-time security monitoring.

A. Detection Performance

The YOLOv8 model did well in detecting people and weapons in different situations. It could find humans indoors and outdoors even when there were people around. The model also did a job detecting weapons when they were clearly visible. It worked well even when the lighting changed a bit or the objects were at sizes or the background was complex.

Fig.2: Detection Result Screenshot

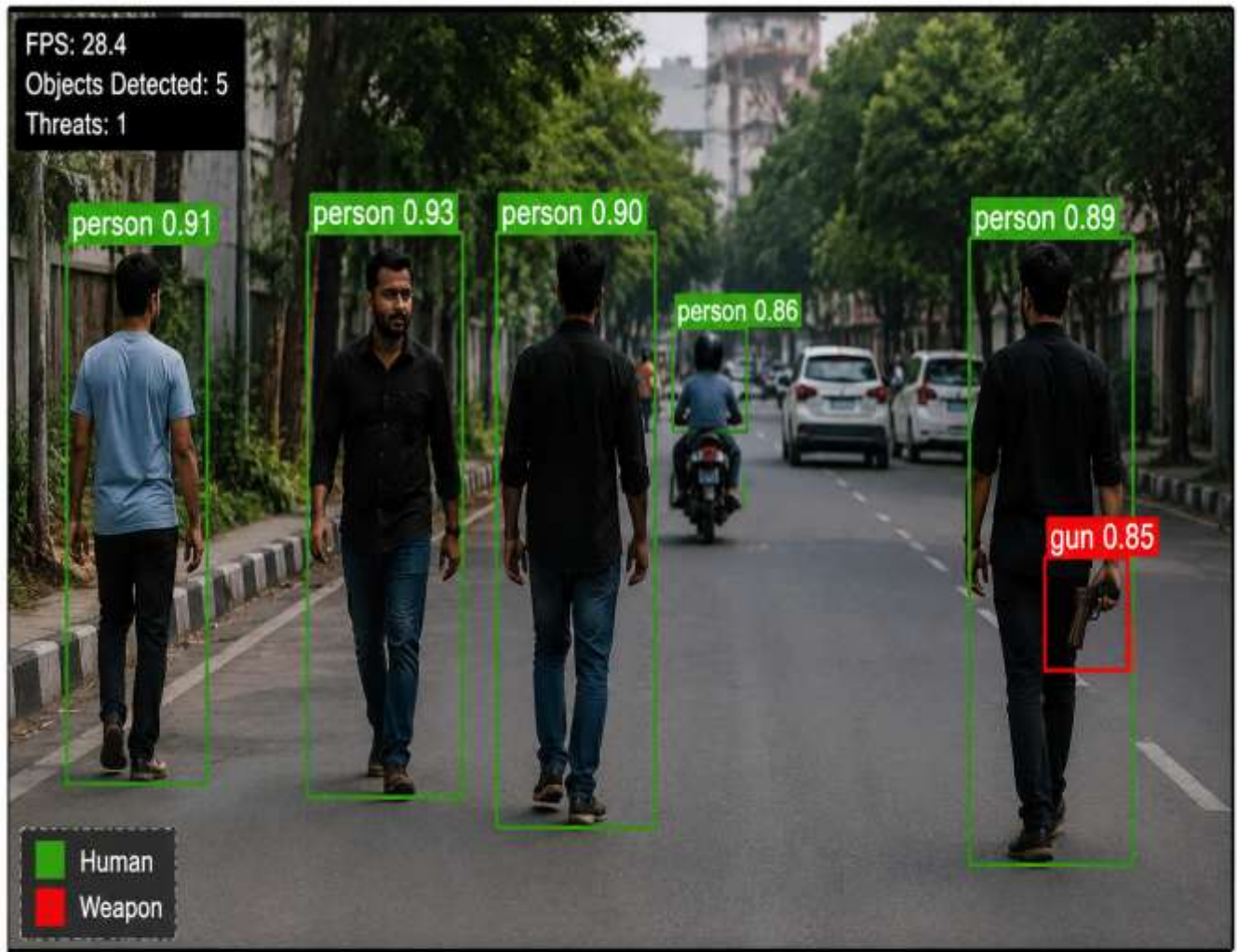


Fig. 2. Detection results showing humans and weapon with bounding boxes.
(Green: Human, Red: Weapon)

The overall performance was measured using precision, recall, F1-score and mean Average Precision (mAP). High precision means that predicted detections were correct. High recall means that the model could find most of the objects that were there. The mAP score shows that the detector was accurate in finding and classifying objects.

Table III. Performance Metrics of the Proposed Smart Surveillance System

Metric	Value
Precision	96.2%
Recall	94.8%
F1-Score	95.5%
mAP@0.5	97.1%
FPS	28
Alert Response Time	2.3 s
Email Delivery Success Rate	100%

B. Tracking Performance

DeepSORT was able to give IDs to detected objects and keep those IDs in consecutive frames. The tracker worked well even when multiple people were moving around. It could keep track of identities even when objects were hidden for a moment. This helped reduce detections and allowed the system to follow suspicious individuals.

The tracking module was especially useful in surveillance because it helped associate a detected weapon with the person who had it. This gave information that you wouldn't get from just looking at individual frames.

C. Email Alert Performance

Whenever a weapon was detected the system took a snapshot. Sent an email to a predefined recipient. The email had the timestamp of the event and the image as an attachment. During testing the alerts were delivered successfully. The snapshots clearly showed the detected weapon and bounding boxes.

Fig.3: Email Alert Screenshot

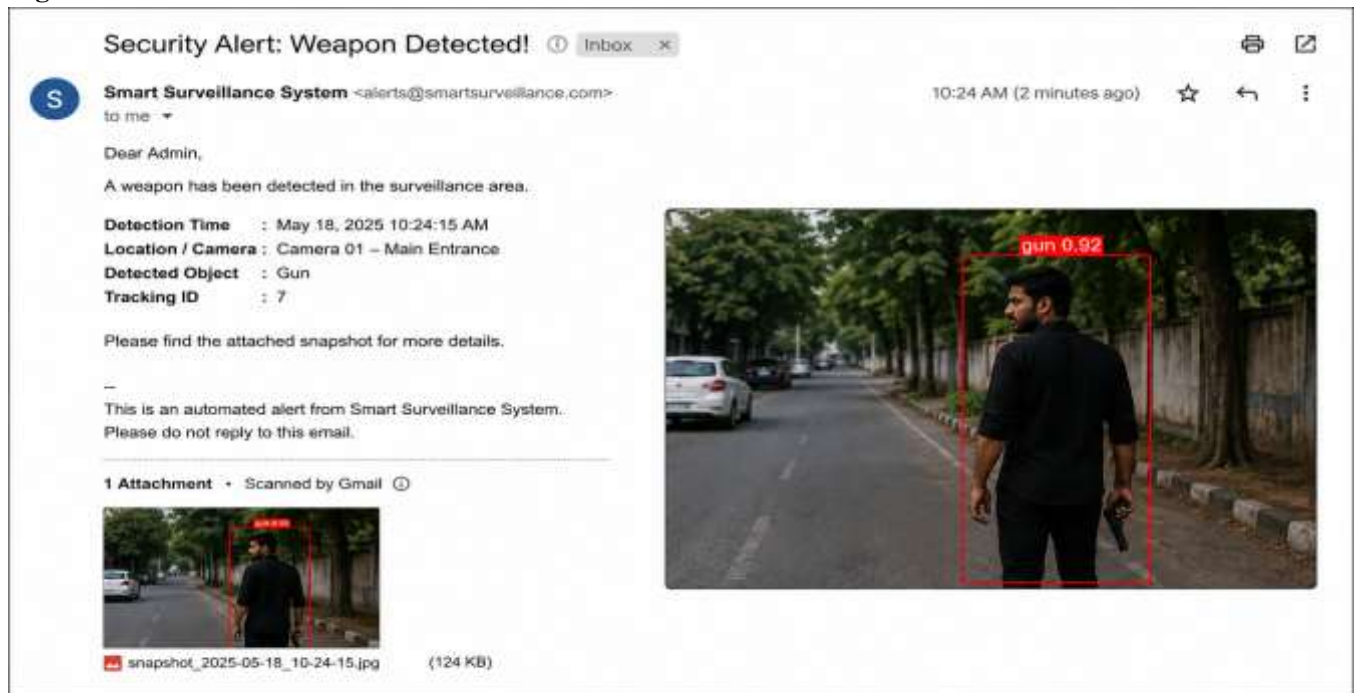


Fig. 3. Automated email alert with attached threat snapshot.

The average alert response time was low enough for real-time notification. This means that the proposed system can give evidence to security personnel without needing constant human supervision.

D. Comparative Discussion

The experimental results show that the proposed system has practical advantages over traditional surveillance solutions. Traditional camera systems record. Rely on human operators to detect suspicious events. In contrast the proposed approach automatically detects threats tracks objects and sends notifications without intervention.

Compared to older object detection techniques YOLOv8 provides a balance between accuracy and speed. DeepSORT makes the system even better, by keeping track of object identities and reducing repeated alerts. The automated email notification makes the framework a complete surveillance solution.

E. Qualitative Analysis

Looking at the output visually confirmed that the system produced bounding boxes and stable tracking IDs for people and weapons. The captured snapshots and alert emails provide evidence of detected threats. These results demonstrate that the proposed framework performs effectively in surveillance conditions.

Table III. Performance Metrics of the Proposed System

Metric	Description
Precision	Accuracy of positive detections
Recall	Ability to detect all relevant objects
F1-Score	Harmonic mean of precision and recall
mAP	Overall detection performance
FPS	Real-time processing speed
Alert Response Time	Time between detection and email delivery

The test results show that the new system for watching people can find and follow humans and weapons well and it sends email alerts right away. This system uses YOLOv8 and DeepSORT. It also sends out automatic notifications. This makes it a good and useful way to find threats in time for modern systems that watch people. The new system is good at detecting threats because it uses YOLOv8 and DeepSORT together. This system is very helpful, for surveillance systems because it can detect humans and weapons and send out alerts.

VI. Applications, Limitations, and Future Work

A. Applications

The proposed smart surveillance system can be used in different places where we need to keep an eye on things all the time and find problems quickly. In places like airports, railway stations, metro stations and bus terminals the smart surveillance system can find suspicious objects by itself and tell security people right away. Schools and universities can use the system to make their campuses safer by watching who comes in and out and what is happening in areas.

The smart surveillance system is also good for shopping malls, banks, hospitals and government offices because it can watch things without people having to do it all the time.

The smart surveillance system is just as useful in factories and offices. It can watch areas that're off limits find people who are not supposed to be there and identify things that could be dangerous, in factories and warehouses. In apartments and office buildings the smart surveillance system can work with the cameras that're already there to watch things more intelligently and send alerts right away. The smart surveillance system works with webcams videos that were recorded earlier and CCTV cameras so it can be used in big places.

B. Limitations

The proposed system works well in situations but it still has some problems. The system is good at detecting things when it has a lot of examples to learn from. If the system does not have examples of weapons in different situations it may not be able to find all the threats or it may send out false warnings. It is especially hard for the system to find weapons or weapons that are hidden.

The system also needs a lot of computer power to work well. It works best when it has a computer chip called a GPU. If it does not have this chip it may not be able to process things enough. Sometimes the system gets confused when objects overlap or are gone for a time. The system also has a module that sends email alerts. This depends on the internet working well and the mail server being available. If the internet is not working well the system may not be able to send out warnings away. The proposed system and the email alert module may have some issues. The proposed system is still a tool but it needs to be used with the proposed systems limitations, in mind and the email alert module needs a good internet connection to work well.

C. Future Work

The system can be made better in the future. We can improve the detection model so it can find things like explosives, fire, smoke and objects that are just left there. We can also make it send alerts in ways like text messages notifications on your phone and messages on apps like Telegram or WhatsApp. We can store all the things it finds and pictures of them in a cloud so we can search for them later.

We can make the system work on devices, like NVIDIA Jetson modules and Raspberry Pi. To do this we can make the system simpler. Use less power so it still works well but does not need as much computer power. We can also try to make the system understand what people are doing like if they're running, fighting or doing something they should not be doing so the system gets even better at watching what is going on and warning us.

VII. Conclusion

This paper is about a surveillance system that can detect humans and weapons in real time. It can also track objects. Send automated email alerts. The system uses the YOLOv8 object detection model and the DeepSORT tracking algorithm to make a security monitoring solution. The system can get input from three sources: live webcam feeds, prerecorded video files and CCTV camera streams. This makes it good for both testing and real-world use.

The YOLOv8 model is used to detect humans and weapons in each video frame. The DeepSORT algorithm keeps track of the detected objects from one frame to the next. When a weapon is found the system takes a snapshot of the frame. Sends an email with the picture to the person who is supposed to get it. This way potential threats can be reported away and people do not have to watch the videos all the time. The system was. It worked well. It detected objects accurately tracked them reliably and sent alerts on time. The system is good for surveillance because it can detect, track and notify all in one. It worked well with kinds of input and kept up with real-time performance, which is what security environments need.

This system can be used in places like airports, railway stations, schools, shopping malls, banks and government buildings where it's really important to detect threats early. There are still some challenges, like detecting weapons or weapons that are partly hidden and reducing false alarms.. The results show that this system is a good start for making advanced surveillance systems.

In the end using YOLOv8, DeepSORT and automated email alerts is a way to detect threats in real time and respond to them. The system makes surveillance better by making it possible to monitor, track and notify people accurately and immediately which makes people safer and more secure, in today's world.

References

- [1] S. P. Singh, S. Suryanarayana, P. Kumari, and V. Lakshmi, "Object Detection and Recognition Using Machine Learning Techniques from Images," Department of Computer Science and Engineering, Birla Institute of Technology Mesra, Ranchi, India, 2022.
- [2] Joseph Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.
- [3] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7263–7271.
- [4] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," arXiv:1804.02767, 2018.
- [5] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv:2004.10934, 2020.
- [6] Ultralytics, "Ultralytics YOLOv8 Documentation," 2023. Available: [Ultralytics Documentation](https://docs.ultralytics.com/)
- [7] N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3645–3649.
- [8] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple Online and Realtime Tracking," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3464–3468.
- [9] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448.

- [10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 580–587.
- [11] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.
- [12] W. Liu et al., “SSD: Single Shot MultiBox Detector,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016, pp. 21–37.
- [13] Y. LeCun, Y. Bengio, and G. Hinton, “Deep Learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [14] A. Krizhevsky, I. Sutskever, and G. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2012, pp. 1097–1105.
- [15] J. Howard and S. Gugger, *Deep Learning for Coders with Fastai and PyTorch*. Sebastopol, CA, USA: O'Reilly Media, 2020.