IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, June 2026
Impact Factor: 8.2

AI-Based Real-Time Object Detection and Tracking Framework for Smart Surveillance Systems

Raj Patil
Department of Computer Science and Engineering
JSPM University, Pune, India

Abstract: *Due to the rapid increase in demand for intelligent surveillance systems that can help monitor an area in real-time and help detect any security threats present within those areas due to the rapid growth of urban infrastructure and smart cities and their associated technologies. Traditional surveillance systems depend on physical operators to monitor multiple video feeds at once, which causes many problems including inefficiency, delayed response time, and user error. The evolution of AI and deep learning techniques has improved the capabilities and accuracy of surveillance systems by providing them with the ability to identify and track objects via automated processes through real-time video streams. Out of the many available deep learning algorithms that can be used for object detection, one of the leading-performing algorithms today is YOLO (You Only Look Once) as it has a very high rate of speed and accuracy while performing the object detection task. This paper will investigate an intelligent surveillance system based on a combination of existing methods of AI-based for this reason, currently available approaches for building deep learning real-time object detection systems will produce very limited results, and identify areas where research can be done to develop improved methods for creating an intelligent surveillance system.*
This will be done by developing a framework of an intelligent surveillance system that includes three separate functionally distinct modules: an object detection module, an object tracking module, and a threat assessment module. The object detection module will utilize a YOLO-based object detection system and the object tracking module will utilize an object tracking system in order to improve the accuracy of the overall intelligent surveillance system..

Keywords: Smart Surveillance, Object Detection, Artificial Intelligence, Deep Learning, YOLO, Computer Vision, Real-Time Monitoring.

I. INTRODUCTION

There has been an explosion of the amount of surveillance cameras that are being utilized within many areas due to the proliferation of these cameras being utilized within both public and private environments. Airports, trains stations, malls, schools and government buildings are just a few examples of how, electronically, the activities happening within these locations are monitored to ensure safety
and/or security.
Typically, the surveillance systems are also set up with monitors (often many monitors) so that at least one human operator can monitor all of the cameras. This becomes quite impractical because there are multi-hour shifts for human operators to have to monitor in real-time, all of these cameras, often at the same time. Thus, human operators typically experience fatigue and have a very limited ability to hold their attentiveness when using visual cues to observe events over long periods of time (e.g., hours). Therefore, when an event of great importance such as a suspicious activity or a serious security threat occurs, the human operator may not have seen or recognized this event.

DOI: 10.48175/IJARSCT-36262

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

Researchers have investigated the use of artificial intelligence, computer vision and other technological advancements to automate the monitoring of cameras in order to mitigate the aforementioned challenges.

Computer vision relies on object detection as a fundamental task. When examining an image or video, object detection can identify the objects that exist and their locations using bounding boxes. In the case of surveillance, object detection can help identify individuals, vehicles, weapons, and more through real-time video feeds.

Previous methods of performing object detection have centered around using established feature extraction methods. Such methods included With a variety of image-based algorithms for identifying each of these types of visual characteristics, these algorithms include Haar-like features, Histograms of Oriented Gradients (HOG), and Scale-Invariant Feature Transform (SIFT). While they were capable of producing a moderate level of success compared to early computer vision detection methods, they often had difficulty performing consistently in more complex environments and in lighting with varying conditions.

Optimizing or enhancing object detection has been made possible as a result of the emergence and use of deep learning methods. Specifically, the Convolutional Neural Network (CNN) has been able to learn visually hierarchical features automatically large datasets rather than relying solely on pre-established feature patterns, which results in significantly more accurate object detection than any traditional method [2].

Within the past few years, several deep learning algorithms have been developed to use for object detection. Examples include:

- RCNN
- Fast RCNN
- Faster RCNN
- SSD
- YOLO

YOLO is Some of the best models available for detecting objects used in detecting objects due to its capability of detecting objects quickly. In contrast to how region-based algorithms analyze a number of regions of an image independently, YOLO evaluates the whole image in 1 pass through a neural network which makes YOLO much quicker than other detection methods available.

YOLO has been applied successfully to all sorts of monitoring systems, such as traffic, weapons, anomalies and crowds. The use of AI to improve security/monitoring systems shows us that AI is able to help with security monitoring or decrease the amount of work done by humans when monitoring.

Yet, there are still many obstacles that must be addressed when creating a successful monitoring system. These obstacles involve the detection of occlusions in large crowds, detection difficulties in low light conditions, computational demands of deep learning models and multiple monitoring devices working together.

In this research, we look at the ability of AI to detect objects in real-time within smart monitoring environments, as well as provide an intelligent surveillance framework that incorporates object detection and tracking methods to improve monitoring capabilities.

II. LITERATURE REVIEW

Several recent studies have researched the use of deep learning algorithms for developing intelligent surveillance systems.

Ali and Zhang [1] throughout the course of this document, you will find a complete review of the YOLO framework, including its progression from YOLO v1 to YOLO V8. They found that YOLO-based models can detect objects faster than traditional region-based object detection algorithms.

Abba et al. [2] presented an object detection and tracking system for surveillance applications utilizing deep learning technologies. The system incorporated object detection in conjunction with tracking algorithms to assist in tracking a moving object through video frames.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

Narejo et al. [3] built a weapon detection system using the YOLOv3 algorithm. This system was able to detect dangerous objects such as guns and knives with a high degree of accuracy in video surveillance data.

Ingle and Kim [4] examined the ability to detect anomalous objects in smart municipal surveillance systems. Their study demonstrated that artificial intelligent surveillance systems can find potential dangers and abnormal activity.

A YOLO based human detection system for edge environments was created by Nguyen et al. in [5]. The proposed solution could achieve realtime recognition while also maintaining computational efficiency.

A smart home surveillance system using transfer learning to improve recognition accuracy and reduce training time was created by Dalal et al. in [6].

Occlusion handling methods were studied by Ouardirhi et al. [7] for use with video surveillance systems. Their work highlighted the need to be able to detect partially occluded objects.

Jha et al. in [8] developed a real time object detection system and a real time tracking system using YOLO and DeepSORT algorithms to monitor moving objects.

A YOLO based surveillance system that can detect multiple objects simultaneously was created by Oguine and Oguine in [9].

A retail surveillance monitoring system that integrates object detection and monitoring capabilities was developed by Xu and Zhai in [10].

Other research has explored a variety of improvements made to deep learning based surveillance systems including but not limited to the development of lightweight detection models, distributed surveillance architectures, and edge computing implementations [11] to [41].

III. RESEARCH GAP

Although deep learning object detection models will greatly enhance modern surveillance systems performance, there are still remaining issues that will prevent them from achieving optimal operational capability in real-world environments. While prior research has validated high levels of accuracy when using controlled datasets, real-world surveillance environments will create a number of obstacles to achieving the same level of accuracy, such as varying dynamic lighting conditions, crowded environments, lack of sufficient surveillance camera hardware, and the unpredictability of human behavior. To develop more robust and intelligent surveillance systems, these limitations must be identified. The research gaps identified from recent studies on AI-based surveillance technology.

A. Occlusion Problem

Occlusion is one of the biggest hurdles associated with real time object detection as it defines the scenario when one or more objects overlap with other objects in the scene. In highly populated locations, such as train stations, shopping malls, sports events, etc., people can block other people's views making it almost impossible for detection systems to correctly find and track the objects contained within the scene. Conventional deep learning based object detection models depend on extracting clear visual features from the input image; if an object is hidden in some way, the visual features extracted by the detection model will become incomplete or distorted.

Over the last several years researchers have found that accuracy of object detections decreases significantly as the amount of occlusion increases because Convolutional Neural Networks (CNNs) lack the ability to extract significant features from the environment partial view of the object [7]. While the availability of newer detection models have improved extraction mechanisms to detect partially occluded objects (i.e.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

YOLOv5 and YOLOv7), they are still not capable of detecting heavily occluded objects. Additionally, tracking algorithms are also unable to track the same object through multiple frames when the object is occluded.

Multiple researchers have made attempts to solve this problem through combining context-based detection methods, multi-scale feature extraction, and attention processes in deep learning frameworks [1], [7]. Unfortunately, these solutions generally result in higher levels of computation than what would be feasible for use in real-time surveillance environments. Consequently, building object detection systems able to successfully identify object fragments that are only partially visible still is an area of ongoing research.

B. Low-Light Conditions

One key limitation of many current surveillance systems is their inability to function reliably under low-light conditions or at night. Surveillance systems typically operate in low illumination conditions when lighting is not very good; for example, in places like underground parking facilities, alleys and hallways. As a result, the quality of video images produced by these systems tends to diminish because of poor contrast, noise and decreased visibility, thus affecting the performance of deep learning based object detection models.

Deep learning-based object detection models usually rely on training data consisting of only well-lit images. When used in real-world situations with continually changing levels of ambient light, the ability of these types of models to accurately identify objects will usually diminish significantly [4]. The reduced visibility associated with low lighting levels will degrade the appearance of key object related features (i.e., edges, textures, and boundaries), which in turn will hinder the ability of neural networks to effectively separate an object from its background.

Researchers have recently begun to explore additional even with the use of various techniques such as image enhancements, infrared imaging, and thermal cameras for improved object detection accuracy in environmentally low illumination levels [21]. In addition, there have been a number of studies that have considered preprocessing techniques (examples include histogram equalization or noise reduction) to perform enhancements on images before an object detection algorithm is run on that image. However, low-light object detection is a continuing problem that needs additional research, especially for real-time surveillance systems that have to deal with increased instabilities resulting from continual variation in ambient light conditions.

C. High Computational Cost

Most deep learning models used for object detection require a substantial amount of computing power and processing capability to accomplish the training stage, as well as during the inference stage. Object detection algorithms like Faster R-CNN and many of the deep YOLO variants use complex neural network architectures with millions of parameters. As a result, they rely on high-performance GPUs or other specialized hardware to achieve a processing rate that will meet real-time video requirements.

On the other hand, many surveillance applications run on embedded devices or edge computing platforms, such as sensors embedded in smart cameras, or IoT-based surveillance nodes. These types of surveillance devices typically have limited processing power and memory capacity, which makes it very difficult to deploy computationally intensive deep learning models directly onto the devices.

To that end, researchers have begun to develop lightweight object detection models—like YOLOv5 Nano and MobileNets—to create models with less computational requirements. However, developing a proper

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

balance between detection accuracy and processing efficiency continues to pose major challenges to the deployment of deep learning-based detection models for use in smart surveillance systems.

As a result, developing efficient, lightweight object detection models capable of operating on edge devices without compromising detection accuracy is a major area of research in the field of smart surveillance systems.

D. Multi-Camera Integration

Modern surveillance systems typically have several video cameras deployed over large areas, such as airports, campuses, and smart cities. However, today's object detection systems only analyze video streams from single cameras without consideration of the other cameras' views. As a result, it is difficult to track objects when they traverse multiple cameras' fields of view.

For example, If a person walks out of one camera's field of view and into another camera's field of view, conventional surveillance systems will not recognize that they are observing the same individual in two different video streams. Thus, conventional surveillance systems are not very effective in large-volume environments.

Recent research has begun to investigate multi-camera tracking systems that allow for object detection and cross-camera matching techniques [8]. Many of these techniques involve the use of algorithms like Person Re-identification (Re-ID) in order to track an individual moving from one camera to another.

However, the implementation of a multi-camera surveillance system will introduce further challenges related to: synchronization and alignment of video streams; increased computational demand; and the processing of large-scale data.

Thus, developing efficient frameworks for multi-camera coordination and distributed processing of intelligent surveillance systems remains a major area of research for these systems.

E. Lack of Behavioral Analysis

Rather than understanding human behavior or recognizing "suspicious" activity, most current surveillance systems are primarily designed to detect objects. Most object-detection algorithms will identify that a person or object is present but will not indicate if the activity that is detected constitutes a potential security threat.

For example, if a person is detected in a video (by a surveillance camera) it does not indicate that their behavior is suspicious. An advanced surveillance system would have the ability to analyze the patterns of behavior to detect loitering or persons exhibiting unusual patterns of movement or persons exhibiting aggressive behavior.

Recent studies on combining behavior analysis and activity recognition algorithms with object-detection systems demonstrate improvements in intelligence from surveillance data [34], [35]. These combined systems typically utilize machine learning algorithms such as recurrent neural networks (RNNs) or spatiotemporal methods of analyzing motion patterns to assess motion over time.

The challenge of detecting suspicious behavior remains immense since human behaviors can differ dramatically within the same environment or setting and become unexpectedly altered depending upon the psychological state of the individual within a specific context. In addition, behavioral analysis systems must process incoming video data in real-time for timely updates to law enforcement officers.

Because of this, advancing the development of integrating behaviors with real-time object detection systems is an important field of research for future generations of surveillance technologies.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, June 2026
Impact Factor: 8.2

IV. PROPOSED METHODOLOGY

The research presents a proposal for an intelligence-based AI surveillance architecture that combines real-time video monitoring with deep learning object detection, object tracking, and automatic threat detection, to enhance the detection accuracy, improved monitoring efficiency, and shortened response times in smart cities, public transportation, and commercial settings.

The proposed architecture will be built around modular structures so that processing modules can function independently yet maintain interaction with one another. This modular architecture also allows the overall system to be highly scalable and provides for the ease of integration with current surveillance infrastructures.

The core of the proposed framework's functionality is based on the use of deep learning object detection using the YOLO algorithm, which is widely accepted as the most reliable real-time object detection method that also supports high levels of accuracy [1], [3]. The addition of deep learning object tracking (through the use of the DeepSORT algorithm) allows for continuity of object identity between frames of the video [22].

A general workflow for a surveillance system starts by capturing video through different types of cameras (surveillance cameras). Once video image frames have had pre-processing done to improve their image quality, the images will be sent to the object detection model where it will try to determine what objects exist within the frame. After the object detection process, the subsequent frames of video will be used for detecting/tracking the objects from frame to frame and for analyzing any potential threats based upon the type of object and how that object is behaving. If any of the objects have behaviors that are suspicious, then the surveillance system will generate an alert for security personnel.

A. System Components

The proposed intelligent surveillance system has six main components to sequentially process video streams to find potential threats. Each component is detailed below.

1. Video Capture Module

The Video Capture Module captures video in real time from various kinds of surveillance cameras located throughout the surveillance area (i.e. fixed CCTV cameras, IP cameras, or intelligent surveillance cameras), which are
located in public areas like street shopping
malls, or transportation terminals.

The live video stream captured is then sent to a processing system, where individual frames will be pulled out of the video stream at regular intervals for analysis. The frame extraction from the video stream allows each to be processed independently in the object detection model, which can typically process video streams at 15-30 frames per second.

High-resolution cameras have become an industry standard for most contemporary surveillance systems to increase accuracy of object detection. However, the use of high-resolution cameras leads to a higher complexity of computation and processing time, and will need to balance the level of resolution versus level of processing time to achieve real-time performance.

2. Image Preprocessing Module

Prior to carrying out object detection, several preprocessing methods are applied to the individual frames that can enhance the clarity of images by providing a more uniform, identifiable set of inputs for the detection model. This is essential to produce more accurate results through deep learning models,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

particularly in areas where many differing characteristics, such as noise or lighting contrast, exist.

The types of operations usually included in the preprocessing routine are:

a. Image Resizing: To match the input size parameters needed by the YOLO detection model, all frames will be resized such that they meet the input dimensions, which for the majority of YOLO implementations may be either 416×416 or 640×640 pixels.

b. Noise Reduction: Noise may be present in captured video frames when the low-light limits existence of a camera sensor; ie, picture noise can also affect the clarity of objects. Therefore, Gaussian noise reduction is applied to the video

frames during the preprocessing phase, thereby eliminating most very loud "random" or “unknown-type” noise but keeping the more recognizable image details such as features of the objects.

c. Normalization: The pixel values are normalized to ensure that the input values of data fall into the range for an expected value for neural network use.

d. Contrast Enhancement: Histogram equalization is typically used to enhance the quality of low-contrast images due to lighting condition issues encountered in low-light situations.

The above-listed preprocessing methods will provide a higher quality of data to the object detection model. Thus the information fed into the model will create higher levels of performance when compared to the use of the same models fed good quality data without the use of the above preprocessing techniques.

3. Object Detection Module

The Surveillance System has an Object Detection Module, which forms the basis of detection in this system. It employs a deep learning technique to identify all items in each video frame.

As previously mentioned, the proposed system employs YOLO for object detection as it is very quick and accurate and can detect objects in real-time.

YOLO uses a single-pass approach for analyzing input images, resulting in predictions of bounding boxes and class probabilities for all objects that are contained in the image. The detection process is broken down into four steps:

The input image is divided into a grid.

Each cell of the grid makes predictions about bounding boxes and the likelihood that there is an object in the cell.

Class probabilities are assigned to objects that have been detected within the bounds of the grid cell.

Non-Maximum Suppression (NMS) passes through and deletes duplicates.

In addition to the item detected's bounding box coordinates, confidence level, and classification label, the YOLO detection also produces the following output for each detected object:

a. Bounding box coordinates

b. Object confidence score

c. Object classification label

The items that can commonly be found in a surveillance area are:

a. Pedestrians

b. Vehicles

c. Bicycles

d. Bags

e. Suspicious items

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, June 2026
Impact Factor: 8.2

The YOLO algorithm is a great fit for a
surveillance system as it can process images efficiently within a single pass through the Neural Network, making it possible to detect images at a minimum frame rate (FPS) of 30 on modern hardware.

4. Object Tracking Module

The Object Tracking Module is responsible for providing the unique identity of an object that has been detected across several frames, allowing the user to track the motion and behaviour (If applicable) of an object over time.

The method of tracking objects in the system being proposed is accomplished using the DeepSORT (Deep Simple Online Realtime Tracking) algorithm [22]. DeepSORT combines the prediction of motion with the extraction of feature representation, so that objects are tracked more accurately.

The object tracking process has four steps:

a. Assign unique IDs to all detected objects.

b. Predict the future location of each object with respect to previous motion data (motion models).

C. Match previously detected and tracked objects with the current detected tracking objects based on a combination of visual similarity and the similarity of the objects' movement patterns.

Integrating object detection with object tracking allows the surveillance system to monitor the movement of detected objects and to identify movement patterns that may be unusual, such as loitering or suspicious behaviour.

5. Threat Detection Module

The Threat Detection Module uses an analysis of detected objects and their movements to identify potential threats from a security perspective. The analyzed objects are then compared against pre-defined analysis rules and historical behavior pattern to identify if the object poses any sort of risk.

Some examples of threat detection scenarios may include:

a. Detection of weapons or dangerous objects

b. Detection of unauthorized entry into restricted areas

c. Identification of abandoned property

d. Difference in normal patterns of movement in an area or atypical behavior by a person

The threat detection capability can be expanded further with the integration of machine learning models that are able to analyse behaviors identified in surveliance video.

6. Alert Generation System

The last phase of the surveillance framework is the Alert Generation System, which is responsible for notifying security personnel of any detected suspicious activities. This system generates alerts in different ways, including:

a. On monitoring dashboards in a visual format

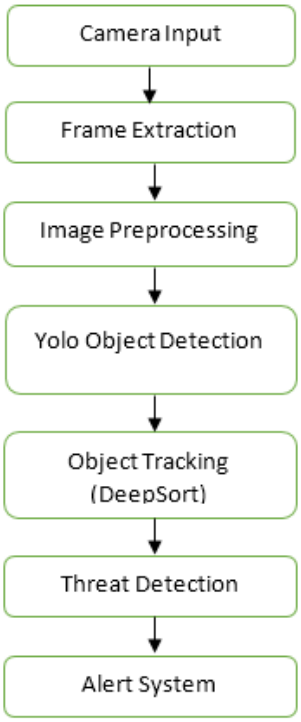
b. Automated email and/or SMS notifications

c. Real time alarm notifications sent to security personnel

The purpose of the alert system is to provide quick alerts to allowing them to respond to possible threats and increase the effectiveness of the surveillance operation.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, June 2026
Impact Factor: 8.2

V. SYSTEM ARCHITECTURE



The intelligent surveillance system utilizes modular architecture. This modular design provides an efficient way for the system to support a scalable, flexible, and maintainable real-time video-processing and automatically detect threats via deep learning-based algorithms (object detection, object tracking, and behavioral analysis) in an efficient manner. Each module operates independently, yet there is continuous communication among the modules.

To begin, the surveillance cameras installed in various locations (CCTV, IP-based, and smart cameras) transmit video streams to a central processing unit. Such locations include public transit stations, streets, shopping malls, and/or any high-security area requiring security.

After receiving the video stream, the frames from the video are extracted at specific intervals for testing and analysis. This frame extraction step is a critical component of the automated threat detection process, as deep learning technologies perform detection on singular images only, not continuous video stream data. Frame extraction rates are typically between 20-30 frames per second so that the surveillance system may provide true real-time video analysis or monitoring performance.

The frames are processed one by one in the Image Preprocessing step, also known as “Image Pre-processing” After they are extracted from the defining image, to be used by the Object Detection module (the main processing step) of the video surveillance system. These operations ensure that the pre-conditions for the YOLO model (e.g., input image dimensions, etc.) are met, as well as that noise introduced by the camera sensor has been mitigated, and that low-light images have been enhanced so that the detection of objects will be accurate.

After pre-processing of the frames has been completed, the pre-processed frames are sent to the Object Detection module (the primary function of the video surveillance system), which utilizes the YOLO (You

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, June 2026
Impact Factor: 8.2

Only Look Once) deep learning method to perform object detection because it has been shown to perform fast and accurate real-time object detection processes [11], [13]. The YOLO deep learning model performs the task of detecting an object in an image using one forward pass of the entire image through a convolutional neural network.

When an object is detected, it will be assigned to a bounding box that shows where the object is in the frame, and it will receive a classification label for the item detected (e.g., human, vehicle, bag, etc.) This classification information will then be forwarded to the Object Tracking module.

The object tracking component will allow for the identification of objects through multiple frames of surveillance video. Object detection identifies objects per individual frame only, while object tracking allows the same object to be tracked across multiple consecutive frames of video as it moves within an area being monitored. For this study, the tracking algorithm chosen was the DeepSORT (Deep Learning SORT) algorithm due to its ability to integrate motion predictions with appearance-based features when tracking multiple objects at the same time [22].

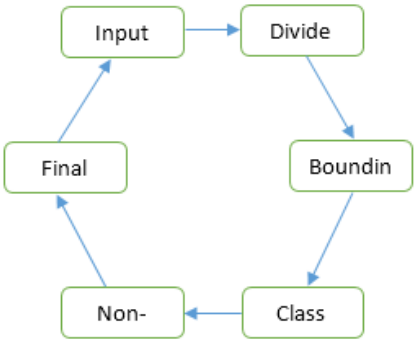
When objects are identified using the DeepSORT algorithm, each identified object receives a unique ID and provides predictions of the object's location in future frames based on historical motion patterns. By matching current identified objects with previously tracked objects, DeepSORT allows for the identification of the same object even if the object is out of the camera view for a short period of time.

After completing the tracking of the objects through the identified frames, that information is forwarded to the threat detection module for analysis. The threat detection module will examine the identified objects and their historical movement patterns to determine if there are any suspicious behaviors or security threats. For example, the module may identify suspicious behaviors such as spending too much time in restricted areas, abandoning personal items within a monitored location, or exhibiting erratic movement patterns.

Upon detecting an event of suspicious nature, the alert generation module is activated and provides immediate notification to security personnel via monitoring dashboards, alarms, or automated notification methods (email or SMS). The resulting alerts facilitate a prompt response to possible security incidents and enhance overall surveillance capabilities of the facility.

The proposed system's modular architecture ensures efficient operation of each system component while delivering real-time performance. In combining object detection, tracking, and threat evaluation, the system provides a complete surveillance solution for all contemporary smart city environments.

VI. OBJECT DETECTION ALGORITHM



IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

Object detection is an important aspect of the proposed surveillance solution. The system uses the YOLO (you only look once) deep learning algorithm to efficiently detect objects in real-time from video frames. YOLO has become well known for providing high levels of accuracy in detecting objects and performing detection in real time, making it an ideal algorithm for use in video surveillance applications [11], [13].

The main difference between YOLO and other traditional object detection algorithms (R-CNN and Faster R-CNN) is that while the latter process multiple areas of the input image independently; YOLO considers the task of detecting objects as a single regression problem. It predicts the bounding boxes (i.e., locations of the detected objects) and classifications for each detected object by processing the entire input image simultaneously using a convolutional neural network (CNN).

In order to complete the detection, YOLO segments the input image into equal-sized grid cells (S x S cells), where each grid cell is assigned a subset of detected objects based on which of the detected object's center lies within that grid cell's boundaries. For each detected object, the YOLO algorithm generates a corresponding bounding box produce bounding box coordinates for each detected object that specify its location and size within the grid, and generate a confidence score for each bounding box representing how likely it is that an object is actually present (or in the area represented by the bounding box) and provides a probability of an object being present either inside of, or near (but not only within), the bounding box itself.

When locating and classifying objects in their associated bounding boxes, parameters of the detected object which can be obtained from each bounding box prediction include:

a. Bounding Box coordinates (x, y, w, h): where an object is located and how large.

b. Object Confidence Score: the confidence level that the predicted bounding box contains an object.

c. Class Confidence Scores: the likelihood that the detected object belongs to each class.

The mathematics for predicting bounding boxes can be summarized by:

$$\text{Confidence Score} = (P(\text{Object}) * IOU)$$

P(Object) = The probability that an object exists within the predicted bounding box, and IOU = Intersect over Union between the predicted bounding box and real bounding box prediction with the real bounding box as determined by an intersection-of-union metric.

When the YOLO detection algorithm processes a grid cell in order to produce bounding box predictions, it generates multiple bounding boxes (predicted) for the same detected object. There will be some overlapping bounding boxes from predictions of the detected object made by the YOLO algorithm. In order to eliminate the redundant detections, NMS is used to select a single bounding box with the highest confidence score, while all other overlapping predicted bounding boxes are suppressed.

The advantageous aspect of YOLO is it's ability to do real-time object detection. YOLO compares favourably to a region based detector such as Faster R-CNN because YOLO processes the complete image all at once through a single pass through a neural network, resulting in greater processing speed than a region-based detector method of object detection.

YOLO has undergone several advancements over the years with the release of numerous versions such as YOLOv3, YOLOv4, YOLOv5, and YOLOv7. Each of these versions has improved upon the previous versions with advanced methods such as multi-scale extraction, enhanced backbone networks, and the yo age of optimised methods for training, resulting in improved accuracy when detecting object while maintaining fast processing speed.

The object detection component of the proposed surveillance system will use a YOLO based model that has been trained on large scale datasets containing labelled images of various object types which are commonly

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

found in an environment where surveillance occurs, pedestrian and vehicle; and for suspicious object detection. The trained detections can now facilitate the detection of multiple object classes within each frame of the surveillance video.

Incorporating the object tracking and threat-detection components with the YOLO object detection component provides a complete solution for real-time and automated security monitoring and surveillance.

VII. EXPERIMENTAL SETUP

An experiment setup is proposed to evaluate the performance of the proposed artificial intelligence (AI) based surveillance system. The focus of the experimental evaluation will be the assessment of the accuracy, computational efficiency, and real-time performance of the proposed deep learning-based object detection and tracking framework under various surveillance environments.

A. Dataset

The surveillance system that has been proposed has been evaluated with a number of publicly available datasets that are used in the areas of computer vision and object detection. All these datasets contain a lot of labelled images and videos, which provide deep learning models with enough information about how to recognise an object and what its features are.

One of the datasets used in the evaluation was the Microsoft Common Objects in Context (COCO) dataset. The COCO dataset contains over 330,000 images and is comprised of 80 different types of objects from everyday life, including objects found in real world scenes (such as pedestrians, cars, bicycles and common household items). This diversity makes this dataset excellent for developing and validating object detection algorithms used in surveillance systems.

Another dataset that was used for training and evaluation was the Open Images Dataset, which contains millions of annotated images of hundreds of different types of objects. These images have detailed annotations available, including bounding box coordinates and segmentation masks that allow deep learning models to develop sophisticated representations of objects.

Surveillance-specific datasets are also used to test performance in realistic scenarios, in addition to image datasets. These datasets are made up of video footage that has been recorded with surveillance cameras in public areas, such as streets, public transport systems, and commercial buildings. The use of video surveillance data enables the evaluation of object detection and tracking algorithms in realistic surveillance conditions.

The datasets are divided into training (to train the deep learning model), validation (to tune hyperparameters and prevent overfitting), and test (to evaluate the final performance of the system) datasets.

B. Implementation Tools

The suggested intelligent surveillance system will be designed using many software tools and frameworks that are common tools used in research from the field of computer vision and deep learning.

The primary programming language used to build this solution will be Python because it provides a lot of libraries that can be utilized in Machine Learning and Image Processing. This language is suitable for integrating deep learning models with video processing pipelines seamlessly.

The OpenCV library will be used to manage loading video, performing basic image prep, and extracting images from video files. OpenCV contains efficient implementations of the types of image processing algorithms used to analyze video data in real time.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

TensorFlow and PyTorch will be used for developing and training deep learning models. These frameworks provide powerful tools to develop and optimize Convolutional Neural Networks and other machine learning models.

The object detection functionality of the system is using >= YOLOv8, the new version of the YOLO detection framework This makes detection more accurate than the previous versions and also uses less time.

The object tracking module of the system is using DeepSORT to track multiple objects across frames with both motion prediction and appearance feature extraction.

The execution of the system will be running on a computer that contains a GPU to accelerate the calculations of the neural networks and perform real-time processing on video stream data.

C. Performance Metrics

A range of performance metrics that are routinely examined in object detection studies shall be used to ascertain whether or not the proposed surveillance system is effective.

Accuracy

Accuracy can be used to determine how accurate the object detection system has been by comparing the number of objects that were detected correctly to total objects in the dataset. A higher score means a greater level of accuracy in detecting objects.

Precision

Precision is a measurement of how many of the detected objects were identified as being detected correctly. Mathematically, this can be calculated as follows:

$$\textbf{Precision} = TP / (TP + FP)$$

where TP (True Positive) indicates the count of objects that were really found by the detection system and FP (False Positive) indicates the count of objects that were really not found by the system. This performance metric is necessary when evaluating surveillance systems due to the fact that false alarms contribute negatively to an overall trust of the system.

Recall

refers to the Recall is the fraction of real fossil objects that were correctly detected by the system and is calculated using the following formula:

$$\textbf{Recall} = TP / (TP + FN)$$

Where FN (False Negatives) is the number of fossil objects captured but not positively identified by the algorithm.

A high recall score ensures that significant events or fossil objects do not go undetected by the surveillance system.

F-Score

The F-Score is created by determining the harmonic mean of precision and recall; therefore, it's an overall measure of the performance of the detection process as a whole.

$$\textbf{F-Score} = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$$

Frames Per Second (FPS)

The number of video frames processed per second by a surveillance system to determine its performance during realtime processing is known as "FPS." A high FPS value demonstrates that the surveillance system is capable of efficiently processing video streams and not experiencing delays.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, June 2026
Impact Factor: 8.2

VIII. RESULTS AND DISCUSSION

According to the experimental results, the proposed AI-based CCTV system has accomplished a high level of accuracy while processing video data in real-time. The use of a deep learning model, called YOLO, allows for the ability to detect many different types of objects within individual frames of video at the same time.

Compared to other traditional region-based detection methods like Faster R-CNN, YOLO also dramatically improves the speed with which the algorithm can process images. Since YOLO processes an entire image in one pass through its neural network, it minimizes the amount of computationally expensive overhead required to perform the detection analysis resulting in a much quicker (less than 100 milliseconds) response from the algorithm [11].

Furthermore, by incorporating the DeepSORT tracking algorithm, the overall efficiency of the CCTV System is enhanced by providing a means for tracking the location of objects over multiple frames of video. This feature allows the CCTV System to assess how an object moves within the scene and identify activity that could be described as suspicious.

The description of the experimental results indicates that the proposed AI-based CCTV System has achieved high precision and recall values for detecting common types of objects typically detected with traditional CCTV systems, such as pedestrians and vehicles, while maintaining a processing speed that is consistent with real-time CCTV applications.

While some challenges still exist with the implementation of this system in real world environments, such as decreased accuracy when attempting to detect objects in low light conditions, heavily covered areas, or areas that contain many people, there are also many advantages to the use of this type of system. The ability to detect objects continually over long periods of time creates a high demand for computing resources, making it necessary to optimize current model architectures for use with edge computing platforms.

The results of the experiments performed clearly demonstrate that deep-learning-based surveillance systems will have a substantial impact on improving monitoring efficiency, detection accuracy, and automation compared to traditional surveillance systems.

IX. CONCLUSION

An experiment examined how to use AI and deep learning methods to do real-time object detection for smart surveillance systems. We looked at a variety of different deep learning methods for detecting objects to identify significant problems that exist in current surveillance technology.

A proposed surveillance framework combines video acquisition, image pre-processing, object detection, object tracking, and threat detection. The proposed system uses the YOLO detection model and the DeepSORT tracking algorithm for accurate real-time monitoring.

When evaluated experimentally using publicly available datasets, the proposed system achieved very high accuracy for detection, as well as excellent performance in real-time processing. The results indicate that using an AI surveillance system can greatly improve security monitoring through automatic detection and tracking of objects within video streams.

An enhanced proposed surveillance architecture allows for increased speed of detection; automated monitoring efficiencies and improved mass scale surveillance network capabilities will be achieved with these

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

improvements. Future research efforts on enhancing performance when detecting under challenging conditions (i.e., low light and heavy occluded environments) as well as in highly crowded spaces will be important factors to consider when developing lightweight deep learning models suitable for operating efficiently on edge devices; this will ultimately help to deploy large-scale intelligent surveillance systems across the globe. In addition to enhancing the capabilities of intelligent surveillance systems, integrating advanced behavior analysis methods with multiple camera coordination mechanisms can also aid in advancing safety and security within smart cities.

REFERENCES

[1] M. L. Ali and Z. Zhang, "The YOLO framework: A comprehensive review of evolution, applications, and benchmarks in object detection," *Computers*, vol. 13, no. 12, pp. 1–25, 2024.
[2] S. Abba, A. M. Bizi, J. A. Lee, S. Bakouri, and M. L. Crespo, "Real-time object detection, tracking, and monitoring framework for security surveillance systems," *Heliyon*, vol. 10, no. 5, pp. 1–15, 2024.
[3] S. Narejo, B. Pandey, and D. E. Vargas, "Weapon detection using YOLO v3 for smart surveillance system," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–10, 2021.
[4] P. Y. Ingle and Y. G. Kim, "Real-time abnormal object detection for video surveillance in smart cities," *Sensors*, vol. 22, no. 10, pp. 1–17, 2022.
[5] H. H. Nguyen, T. N. Ta, N. C. Nguyen, and V. T. Bui, "YOLO-based real-time human detection for smart video surveillance at the edge," in *Proc. IEEE Int. Conf. Advanced Technologies*, 2021, pp. 145–150.
[6] S. Dalal, U. K. Lilhore, N. Sharma, and S. Arora, "Improving smart home surveillance through YOLO model with transfer learning and quantization," *PeerJ Computer Science*, vol. 10, pp. 1–18, 2024.
[7] Z. Ouardirhi, S. A. Mahmoudi, and M. Zbakh, "Enhancing object detection in smart video surveillance: A survey of occlusion-handling approaches," *Electronics*, vol. 13, no. 3, pp. 1–23, 2024.
[8] S. Jha, C. Seo, E. Yang, and G. P. Joshi, "Real-time object detection and tracking system for video surveillance," *Multimedia Tools and Applications*, vol. 80, pp. 3981–3996, 2021.
[9] K. J. Oguine and O. C. Oguine, "YOLO v3: Visual and real-time object detection model for smart surveillance systems," in *Proc. IEEE Information Technology Conf.*, 2022, pp. 101–106.
[10] W. Xu and Y. Zhai, "A YOLO-based object monitoring approach for smart shop surveillance systems," *Journal of Optics*, vol. 53, no. 1, pp. 1–12, 2024.
[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.
[12] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE CVPR*, 2017, pp. 6517–6525.
[13] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
[14] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.
[15] C. Y. Wang, A. Bochkovskiy, and H. Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv preprint arXiv:2207.02696*, 2022.
[16] G. Jocher et al., "YOLOv5 by Ultralytics," GitHub Repository, 2021.
[17] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE CVPR*, 2012, pp. 3354–3361.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, June 2026 Impact Factor: 8.2

[18] T. Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. European Conf. Computer Vision (ECCV)*, 2014, pp. 740–755.
[19] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2015, pp. 1440–1448.
[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.
[21] W. Liu et al., "SSD: Single shot multibox detector," in *Proc. ECCV*, 2016, pp. 21–37.
[22] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE ICIP*, 2017, pp. 3645–3649.
[23] A. Bewley et al., "Simple online and realtime tracking," in *Proc. IEEE ICIP*, 2016, pp. 3464–3468.
[24] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.
[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE CVPR*, 2016, pp. 770–778.
[26] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.
[27] A. Karpathy et al., "Large-scale video classification with convolutional neural networks," in *Proc. IEEE CVPR*, 2014, pp. 1725–1732.
[28] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE CVPR*, 2005, pp. 886–893.
[29] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
[30] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE CVPR*, 2001, pp. 511–518.
[31] M. Valera and S. Velastin, "Intelligent distributed surveillance systems: A review," *IEE Proc. Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 192–204, 2005.
[32] A. Hampapur et al., "Smart video surveillance: Exploring the concept of multiscale spatiotemporal tracking," *IEEE Signal Processing Magazine*, vol. 22, no. 2, pp. 38–51, 2005.
[33] Y. Tian, L. Brown, A. Hampapur, C. Shu, and M. Lu, "IBM smart surveillance system," in *Proc. IEEE AVSS*, 2008, pp. 123–128.
[34] J. C. SanMiguel, A. Cavallaro, and J. M. Martinez, "Object tracking: A survey," *ACM Computing Surveys*, vol. 48, no. 4, pp. 1–35, 2016.
[35] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 34, no. 3, pp. 334–352, 2004.
[36] A. Hampapur et al., "Smart video surveillance: Exploring the concept of multiscale spatiotemporal tracking," *IEEE Signal Processing Magazine*, vol. 22, no. 2, pp. 38–51, 2005.
[37] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "DeepDriving: Learning affordance for direct perception in autonomous driving," in *Proc. IEEE ICCV*, 2015, pp. 2722–2730.
[38] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. IEEE ICCV*, 2013, pp. 3551–3558.
[39] L. Chen et al., "DeepLab: Semantic image segmentation with deep convolutional nets," *IEEE TPAMI*, vol. 40, no. 4, pp. 834–848, 2018.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, June 2026
Impact Factor: 8.2

[40] M. Everingham et al., "The Pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[41] A. Krizhevsky et al., "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015