Improving Surveillance Efficiency with Resnet-101 in Object Detection

K.Barane Prasath[1], AROKIARAJ CHRISTIAN ST. HUBERT[2], Punithan.M[3], Praveen Kumar.E[4]
[1,3,4]*UG Student, Department of Computer Science and Engineering, Madagadipet, Puducherry, India*
[2]*Associate Professor, Department of Computer Science and Engineering, Madagadipet, Puducherry, India*

***Abstract*—Security remains a global challenge, requiring advanced mechanisms for real-time threat detection. Traditional object detection models like RCNN struggle with accuracy and speed, limiting their effectiveness in surveillance applications. ResNet-101, a deep learning-based convolutional neural network, addresses these issues with its residual connections, enabling deeper network training without vanishing gradients. This architecture enhances feature extraction, improving accuracy while maintaining computational efficiency. Compared to RCNN, ResNet-101 processes images faster, making it suitable for real-time surveillance. Its ability to detect threats with higher precision ensures timely responses, reducing potential security risks. By leveraging ResNet-101, surveillance systems can achieve superior threat identification, enhancing safety across various environments. This advancement in deep learning significantly improves real-time monitoring, making security infrastructure more reliable and responsive. As security threats evolve, adopting efficient models like ResNet-101 becomes essential for robust and proactive surveillance solutions.**

Keywords: Security, Real-time Surveillance, Threat Detection, RCNN, ResNet-101, Deep Learning, Residual Connections, Object Detection, Accuracy, Speed.

I. INTRODUCTION

Security has become a universally recognized concept that involves measures taken to protect people, assets, and information from harm, theft, or unauthorized access. Its importance spans across various sectors, including government, corporate organizations, law enforcement agencies, and individual citizens. As threats evolve, especially in the digital and physical realms, the need for more effective, comprehensive security systems has become paramount. One of the most effective ways to achieve this is through surveillance systems that can monitor, detect, and track potential security threats in real-time. Surveillance systems play a critical role in maintaining security, particularly in high-risk environments. These systems provide the ability to continuously monitor activity, detect anomalies, and respond to potential threats in a timely manner. In many cases, the primary goal of surveillance is to ensure the safety of people and property by identifying unauthorized activities, such as trespassing, theft, or violence, before they escalate into more serious incidents.

b. The Need for Real-Time Surveillance:

A real-time surveillance system is crucial in today's fast-paced, dynamic world. It allows security personnel to respond immediately to potential threats, minimizing the risk of damage or harm. Real-time systems provide instant notifications when suspicious activities are detected, enabling immediate intervention. This is particularly important in environments that are difficult to monitor manually, such as large public gatherings, busy commercial areas, or remote locations. In overcrowded environments, where large numbers of people converge, monitoring and tracking individuals manually becomes almost impossible. Traditional surveillance systems, which rely on human observation, are ineffective in such situations due to limited resources and the complexity of monitoring vast areas. This makes it essential to employ automated surveillance systems that can process large volumes of data quickly and accurately, allowing for the identification of potential threats in real time. With the advancement of technology, there has been a significant increase in the use of visual systems in security applications. These systems leverage cameras, sensors, and software to capture, analyze, and interpret visual data. The ability to process visual information in real-time has revolutionized security, making it easier to detect anomalies, recognize objects, and track movements. Visual systems are now widely used in various sectors, including public safety, traffic monitoring,

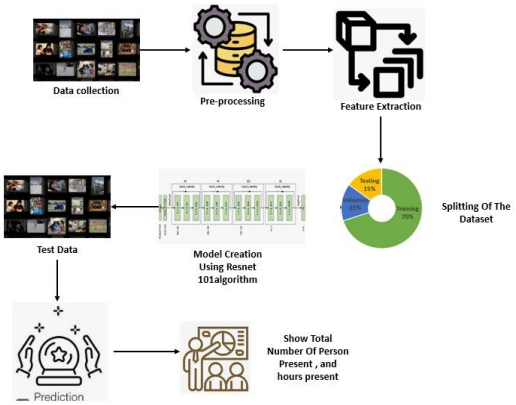
retail, and even healthcare. Visual systems have numerous applications in security, ranging from video anomaly detection to object recognition and tracking. Video anomaly detection involves identifying unusual behavior or events in surveillance footage, such as loitering, aggressive actions, or other suspicious activities. Object recognition allows systems to identify specific items or individuals, which is particularly useful in areas such as access control, baggage screening, and facial recognition. Object tracking, on the other hand, enables systems to follow the movement of people or objects across multiple cameras, ensuring continuous monitoring of potential threats. To effectively utilize visual systems in security, it is crucial to define clear goals and objectives. These systems must be designed to meet specific security needs, whether it's ensuring public safety, protecting property, or monitoring sensitive areas. For example, a surveillance system in a public space might prioritize detecting suspicious behavior, while a system in a restricted area might focus on identifying unauthorized access. The goals of a system should be aligned with the security requirements of the environment it is monitoring.

II. LITERATURE SURVEY

Sani Abba , Ali Mohammed Bizi A , Jeong-A Lee B, Souley Bakouri [1] Global security challenges necessitate efficient real-time surveillance systems for object detection, tracking, and monitoring in dynamic environments. This study proposes a comprehensive framework integrating approximate median filtering, component labeling, background subtraction, and deep learning to enhance detection accuracy. The system is developed using Python for algorithm implementation and C# for a user-friendly interface, ensuring seamless integration via Microsoft Visual Studio (2019 edition). Experimental validation with MOT-Challenge datasets (MOT15, MOT16, and MOT17) demonstrates superior accuracy and precision compared to existing methods. Designed as standalone software, the framework enhances usability and practical deployment in security applications. Future improvements will focus on scalability, enabling adaptability to complex scenarios such as overcrowded areas. Additional integration of multiple data sources will help address variations in time, location, and weather conditions, further strengthening real-time surveillance capabilities. This approach ensures a robust and effective solution for security monitoring across diverse environments. *R.Aarthi, K.Kiruthikadevi [2]* This paper surveys techniques to enhance video surveillance by improving moving object detection and tracking. Accurate detection is crucial for effective tracking, but challenges such as lighting variations, object speed, and background clutter require robust methods. The survey explores detection approaches like Background Subtraction with alpha, Statistical Methods, Eigen Background Subtraction, and Temporal Frame Differencing. Each method offers advantages depending on surveillance needs, from dynamic background updates to probabilistic modeling and motion-based detection. For object tracking, the paper examines point tracking, kernel tracking, and silhouette tracking. Point tracking follows feature points across frames, kernel tracking uses appearance models for occlusion handling, and silhouette tracking captures object shape variations. By integrating these detection and tracking methods, surveillance systems can achieve greater accuracy and efficiency. This study provides valuable insights for developing advanced video surveillance solutions capable of adapting to diverse environments and challenges. *Nuha H. Abdulghafoor, Hadeel N. Abdullah [3]* This study addresses challenges in real-time object detection and tracking by proposing a robust algorithm that combines Principal Component Analysis (PCA) with deep learning networks. Traditional monitoring systems struggle with varying conditions and limited computational resources, reducing their effectiveness. PCA enhances feature extraction by reducing dimensionality while preserving essential information, while deep learning networks excel at detecting complex patterns. By integrating these methods dynamically, the algorithm improves accuracy, tracks multiple moving objects, and adapts to environmental changes in real time. Experimental evaluations show that the proposed approach outperforms existing systems in detection and classification accuracy, even under constrained resources. Its adaptability to diverse real-world scenarios makes it a promising solution for enhancing security and monitoring efficiency. The study demonstrates how combining traditional and modern techniques ensures a responsive, scalable surveillance system suitable for various applications, improving real-time threat detection and situational awareness. *Malik Javed Akhtar, Malik Javed Akhtar,*

Malik Javed Akhtar [4] This study enhances object recognition in surveillance by improving YOLOv2 for detecting small objects, particularly vehicles. Traditional algorithms like R-CNN, Faster R-CNN, and YOLO struggle with low precision for tiny objects. To address this, the proposed approach integrates DenseNet-201 into YOLOv2, optimizing feature extraction through direct layer connections. This architecture reduces redundancy, lowers parameter count, and enhances detection accuracy, especially for distant or partially obscured vehicles. The model was trained on Kaggle and KITTI datasets and cross-validated with MS COCO and Pascal VOC datasets. Experimental results show superior precision and accuracy compared to existing models while maintaining computational efficiency. The compact design ensures suitability for real-time applications, improving vehicle detection in surveillance videos. By leveraging DenseNet-201, this study advances small-object recognition, making real-time monitoring systems more robust and effective for security and autonomous applications. *Dr. (Mrs.) S.S Vasekar , Ms. Sakshi Patil , Mr. Rahul Ban , Ms. Meghanasonawane [5]* This study enhances security video surveillance by automating object detection and tracking in critical locations like banks, roads, and public spaces. The proposed system leverages background subtraction to isolate moving objects, serving as a foundation for intelligent video analysis. A mask sampling technique refines detection by reducing noise and irrelevant elements, ensuring higher accuracy. To further improve recognition, classifiers are integrated, making the system adaptable to diverse surveillance scenarios. The research evaluates multiple state-of-the-art object detection algorithms under real-world driving conditions using a custom-built platform. Extensive testing highlights the strengths and limitations of these methods, providing insights into their applicability for different environments. By combining background subtraction, mask sampling, and classification techniques, the study presents a robust framework for efficient video surveillance. This multi-layered approach enhances speed, reliability, and adaptability, offering an advanced solution for real-time monitoring and threat detection in dynamic settings.

b. ARCHITECTURE DIAGRAM:



The provided architecture diagram represents a pipeline for a person-counting system using deep learning. The process begins with data collection, where images are gathered for analysis. These images undergo pre-processing, which typically includes operations such as resizing, noise reduction, and normalization to improve model performance. The pre-processed data then goes through feature extraction, where key attributes are identified to facilitate accurate classification. Next, the dataset is split into three parts: training (70%), validation (15%), and testing (15%) to ensure the model generalizes well. A ResNet-101 deep learning model is then trained using the training dataset to learn patterns and detect people in images. Once trained, the model is tested using new, unseen test data, and predictions are made based on the learned patterns. Finally, the system outputs the total number of people detected along with the duration of their presence, providing valuable insights for applications such as surveillance and crowd monitoring.

III. PROPOSED SYSTEM

ResNet-101, a deep learning model with residual connections, significantly enhances object detection by improving both accuracy and processing speed. Traditional deep networks often suffer from vanishing gradients, which weaken their learning ability as layers increase. However, ResNet-101 overcomes this challenge by using shortcut connections that allow gradients to flow efficiently through the network. This design enables the model to capture intricate patterns in images, making it highly effective for detecting objects in real-time surveillance applications. By leveraging residual learning, ResNet-101 ensures faster and more reliable threat detection, which is crucial for security monitoring. Its ability to process frames efficiently

allows surveillance systems to quickly identify potential threats with high precision. This leads to timely alerts and improved situational awareness, reducing response times in critical environments. As a result, ResNet-101 serves as a robust solution for real-time security applications, ensuring enhanced safety and monitoring in various settings, from public spaces to high-security areas.

a. Data Collection:

Data collection is the first step in building an effective object detection system for real-time surveillance. The primary objective here is to gather a diverse and comprehensive dataset that covers different environmental conditions, such as variations in lighting, weather, and angles. For surveillance systems, this typically involves capturing video footage from various sources, including CCTV cameras, drones, and sensors. The dataset should contain annotated data for both moving and stationary objects, such as people, vehicles, and animals. The goal is to create a dataset that accurately represents real-world scenarios, ensuring the system's ability to perform under various conditions. Additionally, it's crucial to collect data that includes multiple frames over time to capture the motion of objects, as this helps in the tracking and detection process. For large-scale deployments, collecting data continuously from cameras in critical areas, like streets or airports, would help the system learn the dynamics of crowded environments.

b. Pre-processing:

Pre-processing is essential to prepare raw data for analysis and improve the efficiency of the model. This step involves several tasks, including resizing the images to a consistent resolution, normalizing pixel values to a standard range, and augmenting the data through transformations like rotation, flipping, and scaling to simulate different environmental conditions. Pre-processing also includes removing irrelevant noise, such as irrelevant background elements, to ensure that the algorithm focuses only on the objects of interest. Additionally, the images might be converted to grayscale or normalized in terms of color values, depending on the algorithm requirements. Another key part of pre-processing is splitting the dataset into training and testing sets, ensuring the model is trained on a variety of data and can generalize effectively. By refining the data in this step, the model is provided with clean and consistent inputs, which leads to better detection and faster processing.

c. Feature Extraction:

Feature extraction is a critical component in any machine learning or deep learning-based object detection system. It involves extracting meaningful patterns from the images that help the model identify and classify objects. For ResNet-101, feature extraction is achieved through its convolutional layers, which automatically detect low-level features such as edges and textures, as well as higher-level patterns like shapes and objects. The architecture of ResNet-101 is designed to optimize this extraction through residual connections, enabling the network to capture increasingly complex patterns as the data flows deeper into the layers. The benefit of this process is that it reduces the need for manually crafted features, allowing the model to learn features directly from the data. These learned features are then used for tasks such as object recognition and classification, making the algorithm more adaptive and efficient. The extracted features serve as the input for subsequent steps in the model, including object detection and tracking.

d. Model Creation Using ResNet-101 Algorithm:

Model creation using ResNet-101 involves setting up the network architecture and training it to learn from the pre-processed data. ResNet-101's deep learning architecture is based on residual learning, which introduces shortcut connections that bypass one or more layers. These shortcuts help the network learn better and deeper features without the risk of vanishing gradients that can slow down training. The architecture consists of 101 layers, allowing it to capture highly complex patterns in large datasets. During training, the model processes images through its convolutional layers, pooling layers, and fully connected layers, adjusting the weights based on the error calculated by a loss function. The residual connections allow the network to propagate gradients more effectively, which improves learning and speeds up convergence. The model is trained with backpropagation and stochastic gradient descent to minimize the error and improve detection accuracy. By fine-tuning the model with different hyperparameters, such as learning rates and batch sizes, the system is optimized for high performance.

e. Test Data:

Once the model is trained, it is time to evaluate its performance using test data. The test data consists of images or video frames that were not seen by the model during training. This data serves as a way to assess how well the model generalizes to new, unseen data. The test set is typically split from the original dataset and represents a variety of real-world conditions, such as different object types, movement patterns, and environmental factors like lighting or weather. During testing, the model is evaluated using metrics such as accuracy, precision, recall, and F1-score. These metrics help determine how well the system detects objects, both in terms of true positives and minimizing false positives or negatives. For surveillance applications, the test data might also be annotated to track specific objects like people or vehicles across video frames, providing a comprehensive view of the model's ability to detect and track moving objects accurately.

f. Prediction and Notifications:

Once the model has been trained and tested, it is ready to make predictions on new, unseen video data in real-time. As video frames are processed by the system, ResNet-101 identifies and classifies objects based on the features extracted during training. The prediction process involves detecting moving objects, categorizing them into predefined classes (e.g., people, cars), and then providing additional analysis, such as counting the number of people or vehicles present in the scene. Additionally, the system is capable of generating notifications based on specific thresholds or events. For example, if the system detects a high concentration of people in an area, an alert could be triggered for security personnel. The model can also estimate the total time an object stays within a certain area, providing valuable insights into the behavior of detected entities. Notifications can be sent in real-time via email or mobile alerts, enhancing situational awareness and response times. This capability is crucial for dynamic, high-stakes environments where timely action is needed to maintain security and safety.

g. Introduction to ResNet-101

ResNet-101, or Residual Network with 101 layers, is a deep convolutional neural network (CNN) architecture designed to address the challenges faced by traditional deep networks, particularly the degradation problem. As the depth of neural networks increases, their performance often plateaus or even degrades due to the difficulty in training very deep models. ResNet-101 solves this issue by introducing residual connections or "skip connections," which allow the model to learn residual mappings rather than directly learning the desired output. This makes it easier to train deeper networks while preserving accuracy. The key innovation behind ResNet is the introduction of residual connections, which essentially bypass one or more layers of the network. These connections allow the input to skip some layers and be added to the output, creating a shortcut path. This helps combat the vanishing gradient problem and allows for more efficient training of deep neural networks. Instead of learning the direct mapping from input to output, ResNet learns the difference (residual) between the input and output, making it easier for the network to optimize. This architecture enables the network to effectively train hundreds of layers without suffering from degradation in performance.

h. Architecture Of Resnet-101:

ResNet-101 consists of 101 layers of convolutional, batch normalization, and activation layers, making it significantly deeper than traditional CNNs. The architecture is composed of several building blocks, each containing multiple convolutional layers. The basic block is called a residual block, which includes a shortcut connection that adds the input to the output of the block. This enables the network to focus on learning residuals instead of directly learning the output. ResNet-101 is structured with a series of such residual blocks, followed by fully connected layers at the end for classification tasks. Due to its depth and residual connections, ResNet-101 is capable of extracting complex features from images.

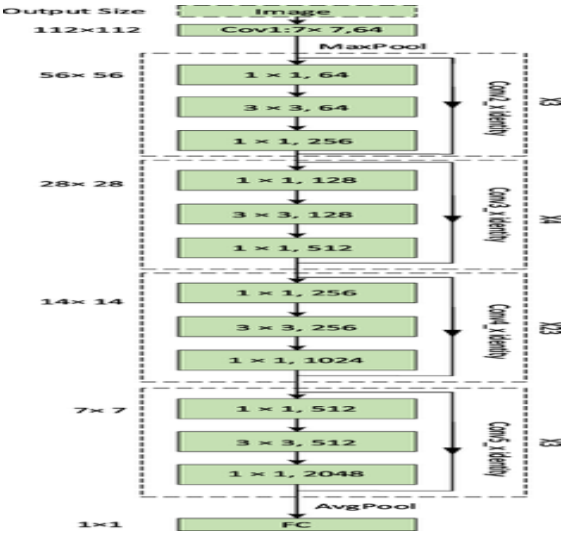
i. Applications And Performance:

ResNet-101 has achieved state-of-the-art performance in a variety of computer vision tasks, such as image classification, object detection, and segmentation. Its deep architecture allows it to learn detailed hierarchical features from images, which is particularly useful in tasks requiring high accuracy and fine-grained recognition. The model has been widely used in real-world applications, including

facial recognition, medical image analysis, and autonomous driving. Thanks to its high accuracy, ResNet-101 is considered one of the go-to architectures for complex vision tasks, where other shallower models might struggle to achieve comparable performance.

IV. RESULT AND DISCUSSION

The integration of ResNet-101 with its deep learning architecture and residual connections significantly enhances object detection performance. Residual connections allow the network to learn deeper, more complex features by preventing the vanishing gradient problem, which is common in traditional deep learning networks.



This enables the model to handle intricate patterns in real-time surveillance scenarios, where both speed and accuracy are critical. As a result, the network can process video frames more quickly, reducing detection times and enabling rapid response in security-critical situations. In real-world applications, this enhancement ensures that threats are detected and identified with higher precision, leading to improved security outcomes. The ability of ResNet-101 to maintain accuracy while processing data faster is especially important for dynamic environments where the conditions may change rapidly, such as crowded areas or high-speed scenarios. This makes it an ideal choice for real-time surveillance systems, ensuring that security personnel can take timely action based on reliable information.

a. Accuracy:

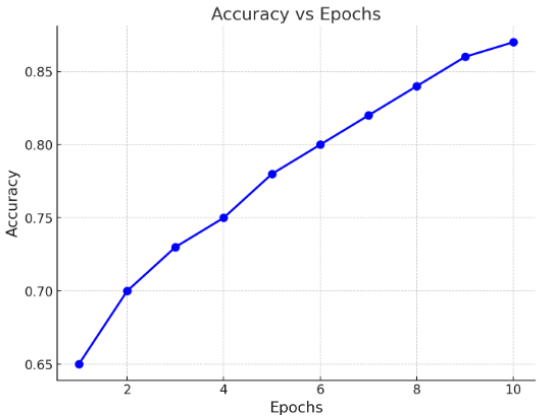
Accuracy is a fundamental metric for evaluating the performance of object detection algorithms. It measures the proportion of correctly identified objects (both true positives and true negatives) to the total number of objects. The formula for accuracy in object detection is:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Number of Predictions}} = \frac{TP + TN}{TP + TN + FP + FN}$$

Where:

- **True Positives (TP)** represent correctly detected objects.
- **True Negatives (TN)** represent correctly identified background regions (no objects).
- **False Positives (FP)** represent incorrectly detected objects (false alarms).
- **False Negatives (FN)** represent objects that were missed by the algorithm.

In object detection, accuracy is a valuable metric as it provides an overall measure of how well the model is distinguishing between objects and the background. However, for a more detailed evaluation, especially in imbalanced datasets, accuracy may not be sufficient on its own. For instance, if the background occupies most of the image, a model may achieve high accuracy by simply predicting no object in most cases. Thus, additional metrics such as Precision, Recall, and F1-Score are often used to provide a more nuanced understanding of the algorithm's performance.



Accuracy alone can sometimes fail to highlight issues like false positives or false negatives, which are particularly important in surveillance systems. Therefore, it's crucial to complement accuracy with other metrics, such as Precision (the proportion of true positive detections among all detected objects) and Recall (the proportion of true positive detections among all actual objects), to ensure that the object detection system is not only fast but also precise in its identification.

b. Loss:

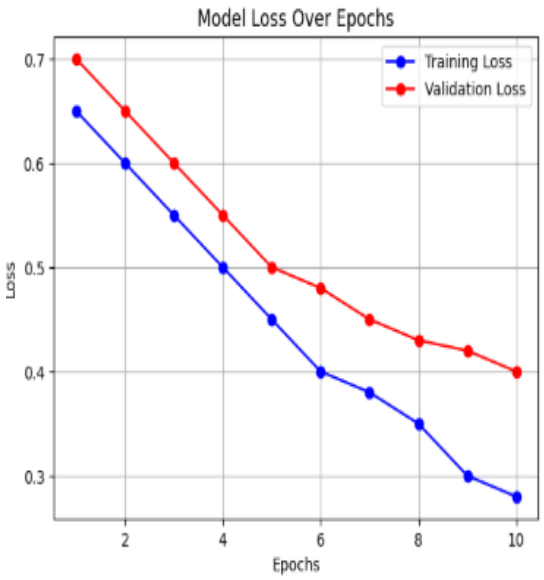
In the context of integrating ResNet-101 for real-time surveillance and object detection, the loss

function plays a crucial role in training the model to accurately detect and classify objects. For classification tasks, Cross-Entropy Loss is commonly used. It measures the difference between the predicted class probabilities and the actual class labels, with the goal of minimizing this difference to ensure accurate object classification. In parallel, Smooth L1 Loss is applied for bounding box regression, which adjusts the predicted coordinates of the bounding boxes (location and size of the detected objects).

$$\mathcal{L}_{cls} = -\sum_{i=1}^{C} y_i \log(p_i)$$

Where:

- C is the number of classes.
- y_i is the true label (binary: 0 or 1).
- p_i is the predicted probability of class i.



Unlike the traditional L2 loss, Smooth L1 Loss is less sensitive to large deviations, making it more robust for object detection. The total loss function combines these two losses, with the classification loss and bounding box regression loss weighted by a hyperparameter (λ\lambda) to balance their contributions. By minimizing this combined loss, the model learns to both classify objects correctly and predict their precise locations, ensuring reliable and accurate performance in real-time surveillance applications.

c. F1 Score:

The F1 score is the harmonic mean of precision and recall, providing a single metric that balances both false positives and false negatives. It is particularly useful in situations where the data is imbalanced, as it gives a more comprehensive measure of a model's performance compared to using accuracy alone. The formula for the F1 score is:

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

Where:

- **Precision** is the proportion of true positive predictions out of all positive predictions made by the model.
- **Recall** is the proportion of true positive predictions out of all actual positive instances in the dataset.

The F1 score ranges from 0 to 1, where 1 indicates perfect precision and recall. An F1 score of 0 indicates that the model has failed at both precision and recall. It is especially useful in imbalanced datasets, where the negative class significantly outnumbers the positive class. It ensures that the model doesn't just predict the majority class and ignore the minority class, which would lead to misleading performance metrics.

d. Precision:

Precision measures the accuracy of the positive predictions made by the model. It is the ratio of true positive predictions to the total number of predictions classified as positive (both true positives and false positives). Precision is critical when the cost of false positives is high. For example, in a spam detection system, a high precision would mean fewer legitimate emails are incorrectly marked as spam. The formula for precision is:

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

Where:

- TP is the number of true positives (correctly predicted positives).
- FP is the number of false positives (incorrectly predicted as positive).

High precision means that when the model predicts an object as positive, it is likely to be correct. However, it doesn't account for how many actual positive instances the model misses, which is where recall comes in.

e. Recall:

Recall (also known as sensitivity or true positive rate) measures the ability of the model to identify all

positive instances in the dataset. It is the ratio of true positives to the total number of actual positive instances (the sum of true positives and false negatives). Recall is especially important when the cost of false negatives is high. For instance, in disease detection, missing a true positive (i.e., failing to identify a sick patient) can be dangerous. The formula for recall is:

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

Where:

- TP is the number of true positives.
- FN is the number of false negatives (actual positives incorrectly classified as negative).

High recall means that the model can correctly identify most of the actual positives, but it may also include more false positives, which could decrease precision. Therefore, both precision and recall should be considered together for evaluating the model's overall performance.

V. CONCLUSION:

In conclusion, the development of real-time surveillance systems is crucial for enhancing security and ensuring public safety in an increasingly complex and threat-prone world. While traditional algorithms like RCNN have been used for object detection, their limitations in accuracy and processing speed hinder their effectiveness, especially in dynamic and real-time environments. The introduction of the ResNet-101 algorithm offers a significant advancement, as its deep convolutional architecture and residual connections allow for more accurate and efficient detection of objects, making it a suitable solution for real-time applications. By utilizing ResNet-101's enhanced feature extraction and hierarchical learning capabilities, the proposed system can identify and track potential threats more reliably and quickly, ultimately improving decision-making and security outcomes. This advancement in technology represents a vital step toward building smarter, more responsive surveillance systems that can address the evolving security challenges of today's world. Future work can focus on integrating multi-modal data such as audio, thermal imaging, and IoT sensors to enhance detection accuracy in challenging environments. Improving real-time processing through model optimization techniques like pruning and quantization, along with making the system scalable for large-scale surveillance networks, would increase efficiency. Incorporating anomaly detection for identifying unusual behaviors and expanding training datasets to include diverse conditions could further boost performance. Additionally, implementing adaptive learning for continuous improvement would ensure the system evolves to meet new security challenges effectively.

VI. REFERENCE

[1] H. Ulusoy, Revisiting security communities after the cold war: the constructivist perspective, perceptions, J. Int. Aff. 8 (3) (2003) 1–22.

[2] D. Lohani, C. Crispim-Junior, Q. Barth´elemy, S. Bertrand, L. Robinault, L. Tougne Rodet, Perimeter intrusion detection by video surveillance: a survey, Sensors 22 (3601) (2022), https://doi.org/10.3390/s2209360.

[3] F. Dumitrescu, C.-A. Boiangiu, M.-L. Voncila, ˘ Fast and robust people detection in RGB images, Appl. Sci. 12 (1225) (2022), https://doi.org/10.3390/ app12031225.

[4] H. Masood, A. Zafar, M.U. Ali, T. Hussain, M.A. Khan, U. Tariq, R. Damaˇsevi˘cius, Tracking of a fixed-shape moving object based on the gradient descent method, Sensors 22 (1098) (2022), https://doi.org/10.3390/s22031098.

[5] H. Qiao, X. Wan, Y. Wan, S. Li, W. Zhang, A novel change detection method for natural disaster detection and segmentation from video sequence, Sensors 20 (5076) (2020), https://doi.org/10.3390/s20185076.

[6] J. Wang, S. Simeonova, M. Shahbazi, Orientation- and scale-invariant multi-vehicle detection and tracking from unmanned aerial videos, Rem. Sens. 11 (2155) (2019), https://doi.org/10.3390/rs11182155.

[7] J.-H. Park, K. Farkhodov, S.-H. Lee, K.-R. Kwon, Deep reinforcement learning-based DQN agent algorithm for visual object tracking in a virtual environmental simulation, Appl. Sci. 12 (3220) (2022), https://doi.org/10.3390/app12073220.

[8] R. Opromolla, G. Inchingolo, G. Fasano, Airborne visual detection and tracking of cooperative UAVs exploiting deep learning, Sensors 19 (4332) (2019), https:// doi.org/10.3390/s19194332.

[9] S. Zhang, L. Zhuo, H. Zhang, J. Li, Object tracking in unmanned aerial vehicle videos via multi-feature discrimination and instance-

aware attention network, Rem. Sens. 12 (2646) (2020), https://doi.org/10.3390/rs12162646.

[10] D. Rodriguez, C. Aceros, J. Valera, E. Anaya, A framework for multiple object tracking in underwater acoustic MIMO communication channels, J. Sens. Actuator Netw. 6 (2) (2017), https://doi.org/10.3390/jsan6010002.

[11] F.S. Alsubaei, F.N. Al-Wesabi, A.M. Hilal, Deep learning-based small object detection and classification model for garbage waste management in smart cities and IoT environment, Appl. Sci. 12 (2281) (2022), https://doi.org/10.3390/app12052281.

[12] P. Kowalczyk, J. Izydorczyk, M. Szelest, Evaluation methodology for object detection and tracking in bounding box based perception modules, Electronics 11 (1182) (2022), https://doi.org/10.3390/electronics11081182.

[13] L. Wenhan, X. Junliang, M. Anton, Z. Xiaoqin, Wei Liu, k. T.-Tae, Multiple object tracking: a literature review, Artif. Intell. 293 (2021) 103448, https://doi.org/ 10.1016/j.artint.2020.103448.

[14] G. Ciparrone, F.L. Sanchez, L. Tabik, L. Troiano, R. Tagliaferri, F. Herrera, Deep learning in video multi-object tracking: a survey, (381). https://doi.org/10. 1016/j.neucom.2019.11.023.

[15] Y.D. Li, Z.B. Hao, H. Lei, Survey of convolutional neural networks, J. Comput. Appl. 36 (9) (2016) 2508–2515.

[16] U. Chandrasekhar, T. Das, A survey of techniques for background subtraction and traffic analysis on surveillance video, 1(3), 107–113. Retrieved from: http:// uniascit.in/files/documents/2011_18.pdf, 2011.

[17] Y. Alper, J. Omar, S. Mubarak, Object tracking: a survey, ACM Comput. Surv. 38 (4) (2013) 13.

[18] B. Drayer, T. Brox, Object detection, tracking, and motion segmentation for object-level video segmentation, Retrieved from, http://arxiv.org/abs/1608.03066, 2016.

[19] S. Sen Cheung, C. Kamath, Robust techniques for background subtraction in urban traffic, video, www.vis.uky.edu/~cheung/doc/UCRL-CONF-200706.pdf, 2004.

[20] G.M. Rao, Object tracking system using approximate median filter, kalman filter, and dynamic template matching, 83–89, https://doi.org/10.5815/ijisa.2014. 05.09, 2014.