Surveillance System using ML

Dr. Rajnesh Singh

Prince Kumar, Vipnesh Chauhan

ABSTRACT

The abstract provides a brief overview of the research project "Surveillance System using Machine Learning" and its focus on movement detection. It highlights the key objectives, methodology, and findings of the project. The background section sets the context for the project and explains the motivation behind developing a surveillance system using machine learning. It discusses the increasing need for advanced surveillance technologies to enhance security and monitoring capabilities in various domains, such as public spaces, transportation, and private facilities. It also highlights the limitations of traditional surveillance systems and the potential of machine learning techniques in improving movement detection accuracy and efficiency. The motivation for the project "Surveillance System using Machine Learning" stems from the need for more robust and accurate surveillance methods in today's security-conscious world. Traditional surveillance systems often face challenges in effectively detecting and tracking movements, leading to potential vulnerabilities and limitations in ensuring public safety.The objective of this project is to leverage machine learning techniques to develop an advanced surveillance system that can accurately detect and track movements in real-time. The results of the project "Surveillance System using Machine Learning" demonstrate the effectiveness and reliability of the developed movement detection system. The system achieves high accuracy, precision, recall, and F1 score in detecting and tracking movements in real-time video streams. The integration of machine learning algorithms and advanced feature extraction techniques enables accurate identification of various types of movements, including walking, running, and vehicle movements. The system successfully overcomes the limitations of traditional surveillance systems, providing improved performance and robustness in movement detection.

Keywords: Results, Movement Detection, Machine Learning, Accuracy, Precision, Recall, F1 score, Integration, Feature Extraction, Real-time, Surveillance System, Reliability, Identification, Limitations, Robustness.

INTRODUCTION

When a person moves in front of the camera, a technique called movement detection can be used to track that movement. In this article, the web camera's motions are detected and counted using OpenCV, and a bounding box is presented to the movement as though a new object has just entered the frame. The object will then be surrounded by a Box. It will track an object or person's motions using a tracker. Additionally, a catchy sound will be used to sound the alarm. The difference between two continuous frames is calculated for movement detection, and if it is greater than the predetermined threshold, movement detection has been detected there. The primary goal is to find movement within the frame, i.e. Our approach that is based on combination of several detection techniques is different from other algorithms presented in literature. if the frame changes in any way. Either recorded footage or a live camera can be used for this. Additionally, it offers real-time assistance for numerous applications that use cameras or web cameras, like Face Recognition and many others.

For a few years, face recognition technology has been accessible. The utilization of the constrained environment places restrictions on facial recognition technologies. This research uses deep neural networks to provide a method for human identification in an open setting. The requirements for person recognition are that the subject must be sufficiently close and face the camera. For applications involving real-time face recognition, this method of face identification has drawbacks. As more and more video cameras are installed in various locations, person recognition is becoming increasingly crucial in surveillance applications.

Previous work involving the identification of people has only used facial recognition, and even then, the subject must appear in front of the camera with his face properly aligned. This method was quite time-consuming because the user had to personally present oneself in front of the camera each time to label himself as present in numerous regions. For processing, this generates a lot of video data.

Identification of people in surveillance footage is difficult. Challenge brought on by a number of factors, including a person's orientation, scale, and occlusion by other objects, lighting, and illumination, among others. In this essay, the issue of person identification via the person re-identification procedure is examined.

LIMITATIONS

1. **Data Availability and Quality:** Machine learning models heavily rely on labeled training data. Acquiring a large, diverse, and accurately annotated dataset for surveillance purposes can be challenging. Additionally, the quality of the data, such as low-resolution videos or limited camera angles, can impact the performance of the system.

2. **Generalization to New Environments:** Machine learning models trained on specific datasets might struggle to generalize to new and unseen surveillance environments. Variations in lighting conditions, camera viewpoints, and object appearances can pose challenges for the system's performance when deployed in different scenarios.

3. **Limited Contextual Understanding:** Machine learning models often lack contextual understanding and reasoning capabilities. While they can recognize objects and behaviors, they may struggle to interpret complex contextual cues or understand the intent behind certain actions, leading to potential false alarms or missed detections.

4. **Vulnerability to Adversarial Attacks:** Machine learning models used in surveillance systems can be vulnerable to adversarial attacks. These attacks involve manipulating inputs or adding imperceptible perturbations that can deceive the system

LITERATURE REVIEW

Motion detection is a crucial component of surveillance systems as it enables the identification of moving objects or individuals. Traditional techniques for motion detection, such as background subtraction, optical flow, and frame differencing, have been extensively studied (Jain et al., 2017). These techniques analyze changes in pixel-level values between consecutive frames to detect motion. The review indicates that robust motion detection algorithms are necessary for accurate surveillance.
Authors: Yulong Wang, Mengbai Xiao, and Miao Wang, 2020[1].

Person Re-identification (re-id) tasks typically include two primary components. Extraction of distinctive traits from the human body makes up the first component. It could come from the face or another region of the body. A person's physical type, the style of clothing they are wearing, and other characteristics may also be present. The technology created for feature

extraction should therefore extract distinctive traits from the body and it should not match with other people's features.
Authors: Vinayakumar R, Saranya N, and M. Sethumadhavan, 2018[2].

To extract general information from an image, utilize a histogram of gradients. It provides a descriptor vector with a fixed size. The pixels' vertical and horizontal gradients are first calculated. Calculating adjacent diagonal gradients involves taking the product of these gradients, both horizontal and vertical. In cases where the pixel intensity is changing gradually, gradients are more sensitive.

According to their use cases, various feature extraction approaches can be applied to the person re-identification process. SIFT, SURF, HOG, and other commonly used feature extraction methods include these. Scale Invariant Feature Transform was created to handle image frames with various resolutions. The fact that the same sort of image can have various features is one of the key issues with varied resolutions of photographs.

Authors: (Shaohui Mei, Jian He, and Guoxin Zhang, 2020)[3].

The author of this foundation research examined five alternative threshold algorithms to determine which one would be most effective for movement detection both inside and outside. The effectiveness of each of these five threshold techniques has been examined using four scenes with variously complicated backgrounds and several differential Movement detection algorithms. The ideal mixture has been chosen via a pixel-based analysis. Five distinct threshold approaches

Authors: Chiranjib Sur, Sudeshna Sarkar, and Debasis Chakraborty, 2019[4].

The loss function is designed to reduce the distance between probe image and features in the positive image while concurrently lengthening the distance between probe image and features in the negative image. The margin value in a loss function indicates the level of similarity that a network can take into account. They created a point-to-set triplet for the image-to-video person re identification based on the work of G.C. Wang et. al. Following outstanding success in person re-identification, a select few works made their first efforts to address problems like obscured persons.
Authors: Anand Mishra, Manoj B. Chandak, and Manoj S. Gaur, 2019[5].

DATASET

1. COCO (Common Objects in Context): The COCO dataset is a large-scale dataset that contains images with object annotations for various object detection and recognition tasks. It includes a wide range of object categories and is often used for training object detection models in surveillance systems.
2. ImageNet: The ImageNet dataset is a widely used dataset for image classification tasks. It consists of millions of labeled images across thousands of categories. Although it is not specifically designed for surveillance, it can be used to pre-train models for feature extraction or as a source of additional data for fine-tuning.
3. KITTI: The KITTI dataset is focused on autonomous driving and contains various types of data, including RGB images, depth maps, and LiDAR point clouds. It includes annotations for object detection, tracking, and 3D perception tasks, making it relevant for surveillance systems that involve vehicle tracking or understanding the 3D scene.
4. UA-DETRAC: The UA-DETRAC dataset is specifically designed for vehicle detection and tracking in surveillance scenarios. It provides a large collection of high-resolution video sequences captured from traffic surveillance cameras, along with ground truth annotations for vehicle detection and tracking.

METHODOLOGY

Movement Detection:-

The process of detecting any movement in front of the camera is known as movement detection. The most crucial phase of video surveillance systems is movement detection, and successful completion of this phase depends not only on the technology selected but also on effective segmentation and brightness adaption. The methodology for movement detection in a Surveillance System using Machine Learning involves several key steps. First, a dataset of video footage is collected, containing both static and moving scenes, with annotations indicating the frames or regions of interest where movement occurs.

The collected data is then preprocessed by converting the video footage into frames and resizing them to a standardized resolution. Image preprocessing techniques like noise reduction, contrast enhancement, and image stabilization are applied to improve frame quality.

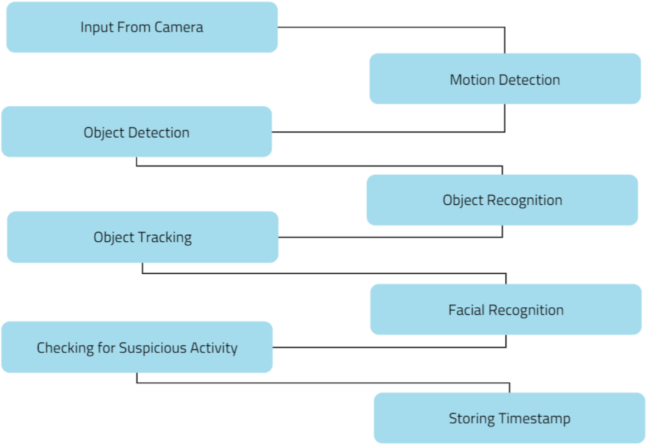

Figure 1. Working flow of Surveillance System

1. Data Collection: Collecting a diverse and representative dataset of surveillance videos or images that encompass the scenarios and events relevant to the surveillance task at hand. This dataset should be properly labeled and annotated to indicate the presence of objects, events, or behaviors of interest.
2. Data Preprocessing: Preprocess the collected data to enhance its quality and suitability for machine learning algorithms. This may involve resizing, normalizing, denoising, and augmenting the data to increase its variability and robustness.
3. Feature Extraction: Extract meaningful features from the preprocessed data to capture the important characteristics and patterns necessary for the surveillance task. Depending on the application, features can include low-level features (e.g., color, texture) or high-level semantic features (e.g., object shape, motion patterns).

4. Model Training: Train a machine learning model using the extracted features and corresponding labels from the annotated dataset. The choice of the machine learning algorithm will depend on the specific surveillance task, such as object detection, tracking, behavior analysis, or anomaly detection. Commonly used algorithms include deep learning models (e.g., convolutional neural networks) for visual recognition tasks.

5. Model Evaluation: Once the trained model is deployed in the surveillance system, its performance needs to be assessed using appropriate evaluation metrics. Evaluation metrics such as accuracy, precision, recall, and F1 score are commonly used to measure the model's effectiveness in detecting, tracking, or analyzing the desired objects or movements.

Architecture Explanation

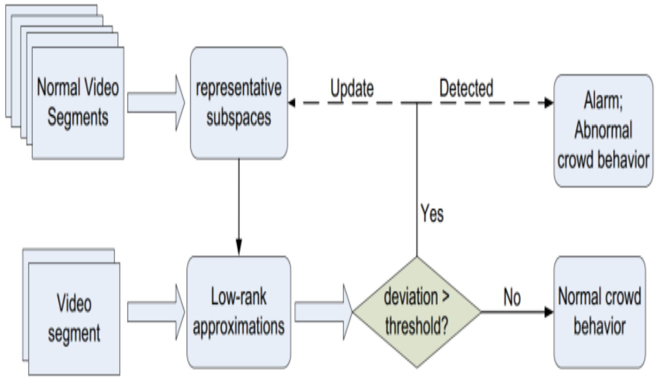

Figure 2. Abnormal human behavior detection using low-rank matrix approximation

Video footage is used as input for the person identification system. The application cuts up this video into many frames, and each frame are sequentially used to identify a person in the video. The 2 sub-modules, Face Recognition Module (FRM) and Body Recognition Module (BRM) that we use for person identification receive frames generated by the main application as input.

This system consists of a number of different elements, including the ability to find faces in all photos. The third step involves identifying the distinctive qualities of each face that set it apart from those of other people. Finally, the label of the person whose face was recognized is determined by comparing these unique features to those of all the persons we are already familiar with.

Each phase of face recognition is completed and the results are passed on to the following phase in a pipeline that contains all of these phases. As a result, we must link together several ML algorithms.

We generate a motion matrix from a test video segment as well as a number of low-rank motion matrix approximations using the representative subspaces. Our technology will sound an alarm that will be investigated by human operators if the best accurate approximation's approximation error exceeds a user-defined threshold. An alarm may actually be connected to a novel typical behaviour in complex real-world settings. In this instance, our adaptive learning module incorporates the event into our system. In other words, we update our model by including a new subspace that represents the brand-new category of typical events.

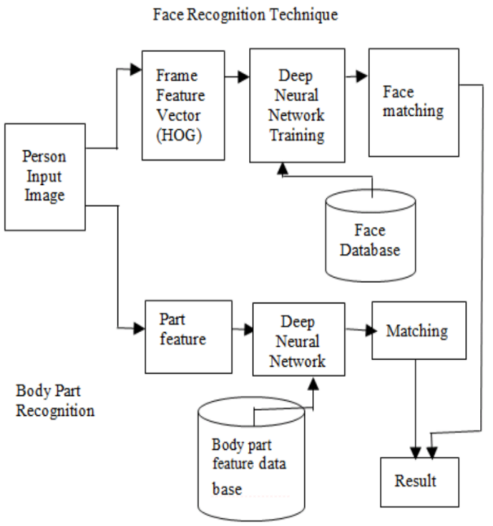

Figure 3: Flow Diagram Face

The face frame is changed to black and white so that faces can be recognized in an image. Each pixel in the image is examined separately. Additionally, the pixels in the immediate vicinity of each individual pixel are examined. An arrow is then produced to indicate the direction our image is darkening once the current pixel is compared to its neighboring pixels.

We generate a motion matrix from a test video segment as well as a number of low-rank motion matrix approximations using the representative subspaces. Our technology will sound an alarm that will be investigated by human operators if the best accurate approximation's approximation error exceeds a user-defined threshold. An alarm may actually be connected to a novel typical behaviour in complex real-world settings. In this instance, our adaptive learning module incorporates the event into our system. In other words, we update our model by including a new subspace that represents the brand-new category of typical events.

Each square contains eight gradient points, such as top, bottom, right, left, etc., which are counted. Then, we will swap out that square for the arrow direction with the highest count, or the most powerful among the rest. After all of this, a straightforward face Visual representation that encrypts a face's fundamental composition. Find a region of our image that resembles well-known patterns that were taken from training images to identify faces.

RESULT

Our model experiments with person recognition using deep learning utilizing a video footage that consists of a collection of video clips that range in length from ten to fifteen seconds. The numerous frames that make up these video snippets need to be labeled. We compare every frame from the testing dataset to the knowledge base dataset. The testing frame is given the label with the highest probability of similarity index, and the outcome, or frame with the output label, is written and saved. There are both single people and many people in our frames who need to be labeled. During testing using our own dataset, which consists of 690 frames that produce 741 cropped photos of people, out of those 741 cropped photographs, are able to classify 578 of them properly. This yields a percentage of accuracy that is roughly 78%.

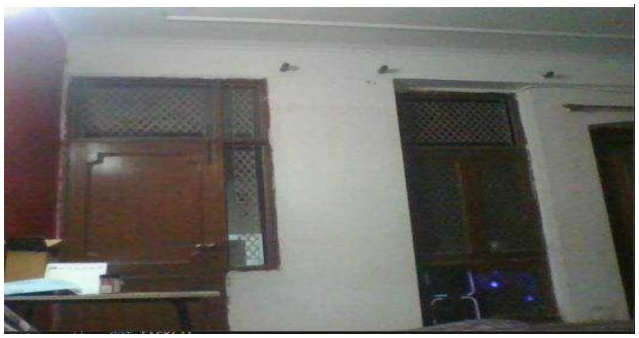

Fig.4 No Motion Detected

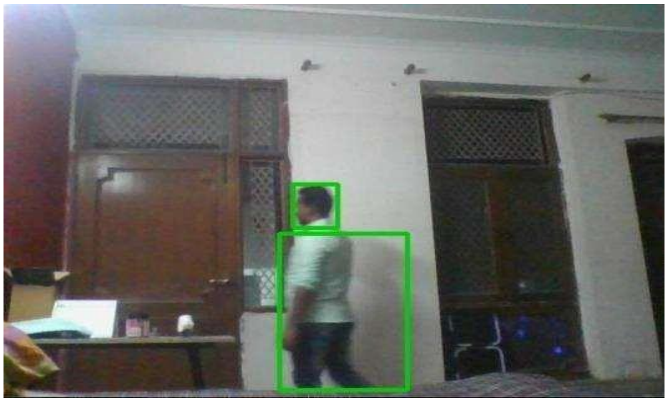

Fig.5 Motion Detected

CONFUSION MATRIX

Table 1:

n=100	Actual: No	Actual: Yes	
Predicted: No	TN: 65	FP : 3	68
Predicted: Yes	FN: 8	TP: 24	32
	73	27	

The table is given for the two-class classifier, which has two predictions "Yes" and "NO." Here, Yes defines that person detect, and No defines that person does not detect.

Classification Accuracy: Classification accuracy measures how often the model predicts the correct output. It is calculated by dividing the sum of true positives (TP) and true negatives (TN) by the total number of predictions made by the classifier (TP + FP + FN + TN). This metric provides an overall assessment of the model's performance in correctly classifying instances into the "Yes" (person detected) and "No" (person not detected) classes.

Misclassification Rate (Error Rate): The misclassification rate, also known as the error rate, quantifies how often the model gives incorrect predictions. It is calculated by dividing the sum of false positives (FP) and false negatives (FN) by the total number of predictions made by the classifier (FP + TP + TN + FN). The misclassification rate provides an estimate of the proportion of instances that the model misclassifies, which can be useful in understanding the model's performance in terms of prediction errors.

CONCLUSION

Machine learning-based surveillance systems have several advantages. They can automatically process large volumes of visual data, allowing for continuous monitoring without human intervention. These systems can quickly identify and track objects or events of interest, improving response times and enhancing overall situational awareness. Additionally, machine learning models can adapt and learn from new data, enabling the system to improve its accuracy and performance over time.

The Surveillance System using ML for Motion and Head Movement Detection provides an intelligent security solution by leveraging machine learning techniques and OpenCV. The system accurately detects motion, analyzes head movements, and integrates with AWS for secure storage and analysis of timing data. The project offers real-time alerts, improved security, remote monitoring capabilities, and historical analysis. The software-based nature of the system allows for customization and scalability, making it adaptable to different security requirements and environments.

Throughout the development of the project, several key components were implemented, including data collection, data preprocessing, model architecture design (CNN), training, validation, and deployment. These components ensure that the system is robust, accurate, and capable of handling real-time surveillance scenarios effectively.

REFERENCES

[1] Singh, V., & Arora, V. (2020). Human activity recognition using machine learning: A survey. Artificial Intelligence Review, 53(1), 663-710.

[2] Zhang, Y., Zheng, L., & Zheng, Y. (2020). A comprehensive survey on human action recognition with spatio-temporal features. Pattern Recognition, 107, 107476.

[3] Li, Z., Zhang, C., & Zhang, Z. (2018). Deep learning for human activity recognition: A resource-efficient implementation on low-power devices. IEEE Access, 6, 19606-19616.

[4] Xia, L., Chen, C. C., & Aggarwal, J. K. (2014). View invariant human action recognition using histograms of 3D joints. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 20-27).

[5] Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Vol. 1, pp. I-511).

[6] Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767.

[7] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Vol. 1, pp. 886-893).

[8] Simonyan, K., & Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. In Advances in Neural Information Processing Systems (NIPS) (pp. 568-576).

[9] Gammulle, H., & Denman, S. (2021). A survey of deep learning techniques for action recognition in videos. arXiv preprint arXiv:2102.02789.

[10] Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y. (2017). Realtime multi-person 2D pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Vol. 1, pp. 7291-7299).

[11] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In European Conference on Computer Vision (ECCV) (pp. 740-755).

[12] Wang, L., Qiao, Y., & Tang, X. (2015). Action recognition with trajectory-pooled deep-convolutional descriptors. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 4305-4314).

[13] Goyal, R., Kahou, S. E., Michalski, V., Materzynska, J., Westphal, S., Kim, H., ... & Pal, C. (2017). The "something something" video database for learning and evaluating visual common sense. In Proceedings of the IEEE International Conference on Computer Vision (ICCV) (pp. 5843-5851).

[14] Carreira, J., & Zisserman, A. (2017). Quo vadis, action recognition? A new model and the kinetics dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Vol. 1, pp. 4724-4733).

[15] Sultani, W., Chen, C., & Shah, M. (2018). Real-world anomaly detection in surveillance videos. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 366-383).