International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 04 | Apr 2024 www.irjet.net p-ISSN: 2395-0072

ENHANCING LIVE CCTV SYSTEMS: OBJECT DETECTION, TRACKING, AND EXPRESSIVE FOOTAGE STORAGE

Ankit Sharma[1], Arnav Zutshi[2], Mridul Sharma[3] ,Anshika Sharma[4], Prof. Pramila M. Chawan[5]

[1, 2, 3, 4]BTech Student, Dept. of Computer Engineering and IT, VJTI College, Mumbai, Maharashtra, India
[5]Associate Professor, Dept. of Computer Engineering and IT, VJTI College, Mumbai, Maharashtra, India

---

Abstract - In our rapidly evolving world, where the significance of data storage cannot be overstated, the efficient handling of such data has become paramount. Advanced storage technologies offer promising solutions for enterprises and organizations dealing with large-scale data collection. This project focuses on enhancing the efficiency of CCTV storage for traffic cameras by selectively recording essential data. Leveraging pre-trained weights from YOLOv3, the system employs frame-to-frame distance measurement to identify significant changes, thus avoiding unnecessary storage redundancy. Through integration with OpenCV's DNN module and utilizing ResNet18 for Feature Vector extraction, the system achieves swift loading of YOLOv3 weights, enabling real-time object detection and classification. This proposed model not only accurately pinpoints objects responsible for changes but also precisely outlines them with bounding boxes, providing an optimized solution for robust and efficient traffic surveillance systems.

Key Words: YOLOv3, object detection, OpenCV, ResNet

1.INTRODUCTION

In recent years, advancements in computer vision and deep learning have revolutionized the field of video surveillance. The ability to accurately detect and track objects in real-time has become crucial for various applications, including security monitoring, traffic management, and behavior analysis. One of the key challenges in video surveillance systems is the efficient handling and storage of vast amounts of data generated by continuous video streams.

This research paper focuses on addressing this challenge by leveraging state-of-the-art techniques in object detection and feature extraction. Specifically, we utilize YOLOv3 pretrained weights, a popular deep learning model known for its high-speed and accurate object detection capabilities. Additionally, we integrate the ResNet18 model for feature vector extraction, which enables us to capture rich representations of objects in video frames.

The novelty of our approach lies in the utilization of these feature vectors to calculate cosine similarities between consecutive frames in a video sequence. Cosine similarity is a metric commonly used to measure the similarity between two vectors by computing the cosine of the angle between them. By applying a threshold to the cosine similarity values, we can identify frames with low similarity, indicating significant changes or new objects entering the scene.

The main objective of our project is to develop a video surveillance system that intelligently selects and saves only those frames that contain meaningful changes or events. This selective recording approach not only optimizes storage utilization but also ensures that important information is preserved for subsequent analysis or retrieval.

Throughout this paper, we will describe in detail the methodology used for integrating YOLOv3 and ResNet18 models, the process of calculating cosine similarities, and the implementation of our selective frame-saving mechanism. We will also present experimental results and performance evaluations to demonstrate the effectiveness and efficiency of our proposed system compared to traditional continuous recording methods.

Overall, this research contributes to the advancement of video surveillance technologies by combining cutting-edge deep learning techniques with intelligent data selection strategies, paving the way for more resource-efficient and responsive surveillance systems in various domains.

2. LITERATURE REVIEW

The evolution of real-time object detection in computer vision has seen significant advancements with the introduction of the You Only Look Once (YOLO) framework. Redmon, Divvala, Girshick, and Farhadi (2016) introduced the YOLO framework, which revolutionized object detection by providing a unified approach for real-time processing. This framework streamlined object detection tasks, making them more efficient and accessible for various applications requiring immediate and accurate object identification.

Building upon the success of YOLO, Redmon and Farhadi (2018) proposed YOLOv3 as an incremental improvement, further enhancing the capabilities of real-time object detection. YOLOv3 represents a refinement of the original

YOLO architecture, incorporating additional features and optimizations to achieve higher accuracy and improved performance in object detection tasks.

In the realm of information retrieval and indexing, foundational works such as the Vector Space Model by Salton, Wong, and Yang (1975) have laid the groundwork for efficient automatic indexing and retrieval systems. This model, based on the representation of documents and queries as vectors in a multi-dimensional space, has been instrumental in developing search and retrieval algorithms used widely in information systems.

Shifting focus to action recognition and object tracking, Rao and Shah (2001) explored view-invariant representation and recognition of actions, contributing to the development of robust algorithms for recognizing actions across different perspectives and environments. Similarly, Yilmaz, Javed, and Shah (2006) conducted a comprehensive survey on object tracking techniques, providing insights into the challenges and advancements in this critical area of computer vision.

Collectively, these works highlight the dynamic landscape of computer vision research, showcasing advancements in real-time object detection, indexing models, action recognition, and object tracking. By leveraging insights from these studies, researchers continue to push the boundaries of computer vision applications, paving the way for innovative solutions in diverse domains such as surveillance, robotics, and human-computer interaction.

3. PROPOSED SYSTEM

3.1 Underlying Machine Learning Models

The project utilized pre-trained YOLOv3 weights on its configuration file as well as uses ResNet18 for feature vector extraction from each frame.

- YOLOv3: YOLOv3 is an advanced real-time object detection framework known for its high accuracy and improved performance in identifying objects swiftly and accurately.
- ResNet18: ResNet18 is a deep convolutional neural network architecture capable of extracting rich and complex feature vectors from input data, aiding in tasks such as image classification, object recognition, and feature representation learning.

3.2 Similarity Algorithms

Cosine similarity is applied by extracting features from video frames and calculating the cosine of angles between their feature vectors, useful for tasks like similarity search or video classification based on feature representations. It's essential for tasks like anomaly detection or similarity-based retrieval in video recognition systems.

3.3 Distance Algorithms

In video recognition, the Euclidean distance algorithm computes the distance between feature vectors extracted from frames, aiding tasks like clustering or nearest neighbor classification based on their spatial relationships in high-dimensional space.

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

p, q = two points in Euclidean n-space
q_i, p_i = Euclidean vectors, starting from the origin of the space (initial point)
n = n-space

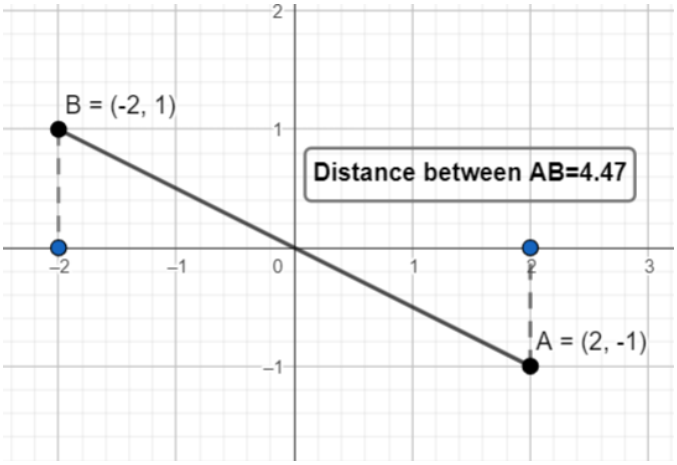

Fig 1: Euclidean distance method for distance calculation

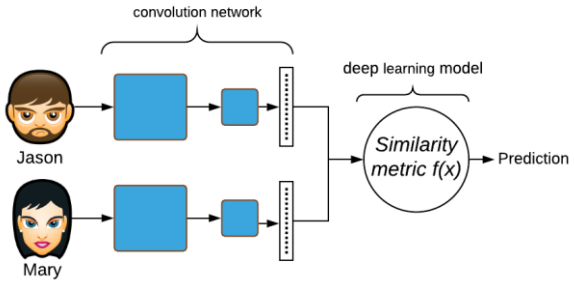

Fig 2: Cosine similarity for similarity search

3.4 Dataset Creation

The creation of high-quality datasets holds paramount importance in the realm of computer vision research, providing the foundation for robust and accurate algorithm training.

The Common Objects in Context (COCO) dataset, introduced by Lin et al. in 2014, stands as a pivotal and widely adopted resource for training object detection models. Renowned for its meticulous curation, COCO encompasses a diverse array of object categories within complex scenes, offering images with multi-object instances, varied backgrounds, and intricate contextual variations. With over 200,000 labeled images and 80+ object categories, COCO serves as a rich repository for training algorithms to navigate real-world complexities, enhancing their adaptability to diverse scenarios. Its widespread adoption as a benchmark dataset in the computer vision community underscores its pivotal role in advancing the field, fostering innovation, and pushing boundaries while remaining a cornerstone of research integrity.

3.4 Object Detection

The system leverages the YOLOv3 algorithm for object detection on the COCO dataset, integrated with OpenCV for efficient processing. Prior to deployment, the pre-trained YOLOv3 model undergoes fine-tuning to enhance its accuracy in detecting diverse objects within complex scenes. The detection pipeline utilizes YOLOv3's robust feature extraction capabilities, coupled with efficient bounding box prediction mechanisms, enabling the system to identify multiple object instances simultaneously. This approach not only ensures high accuracy in object localization but also optimizes computational efficiency, making it suitable for real-time applications.

3.5 Feature Vector Extraction

The project utilizes ResNet18, a deep learning architecture known for its strong feature extraction capabilities, to extract feature vectors from individual frames of videos. These feature vectors capture meaningful information about the content of each frame, such as edges, textures, and object patterns. By leveraging the learned representations encoded in the ResNet18 model, the system transforms raw pixel data into compact and informative feature vectors. This process enables efficient comparison between consecutive frames using similarity metrics, allowing the system to identify frames with significant movement and selectively store them, thereby optimizing storage usage on CCTV devices while retaining important video content.

3.6 Video Storage

A threshold is set on the similarity measures using a hit and trial method to get the optimum results on the storage of expressive footage. The threshold for COsine similarity is set to around 0.75 and that of euclidean distance can be set to 0.5

3.7 Optical Flow for Object Tracking

Another addition to the project utilizes optical flow analysis on CCTV camera footage to determine the speed of cars on roads. By tracking the movement of pixels between consecutive frames, optical flow algorithms estimate the velocity of objects in the scene, enabling the system to calculate vehicle speeds. If overspeeding is detected based on predefined speed thresholds, the system triggers an alert mechanism to notify authorities and potentially fine the violators, contributing to road safety enforcement and traffic management.

4. SYSTEM ARCHITECTURE

The proposed Traffic Monitoring System, incorporating YOLOv3 for object detection and ResNet18 for feature extraction, along with cosine similarity analysis in Python, presents a transformative solution for efficient CCTV camera management on roads. This system offers numerous advantages across traffic surveillance domains, including accurate object detection, optimized storage usage, and enhanced security.

Utilizing YOLOv3, the system accurately detects and tracks vehicles, pedestrians, and other objects of interest in real time, providing critical data for traffic monitoring and management. Subsequently, ResNet18 extracts compact feature vectors from consecutive frames, while cosine similarity analysis helps identify frames with significant movement, thus saving storage space by selectively storing relevant frames.

SYSTEM DESIGN & ARCHITECTURE

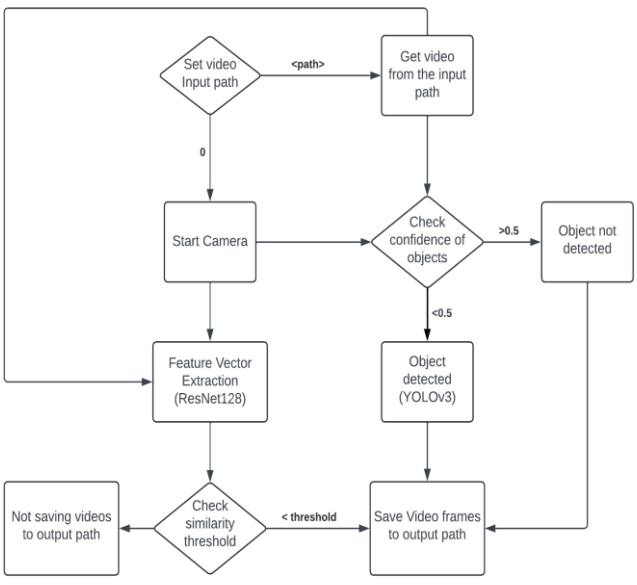

Fig 3: Live-CCTV Flow Diagram

5. FINDINGS AND ANALYSIS

The project's Graphical User Interface (GUI) is designed to offer users a seamless and engaging experience with three main interaction pathways. Users can either input a video path for object detection and expressive footage extraction or choose to utilize the system camera by providing '0' as an argument, ensuring flexibility and adaptability to different usage scenarios. Additionally, the GUI includes a designated field for specifying the output path, allowing users to determine where the extracted video with expressive footage will be stored. This feature adds a layer of customization and control over the system's output, enhancing user convenience.

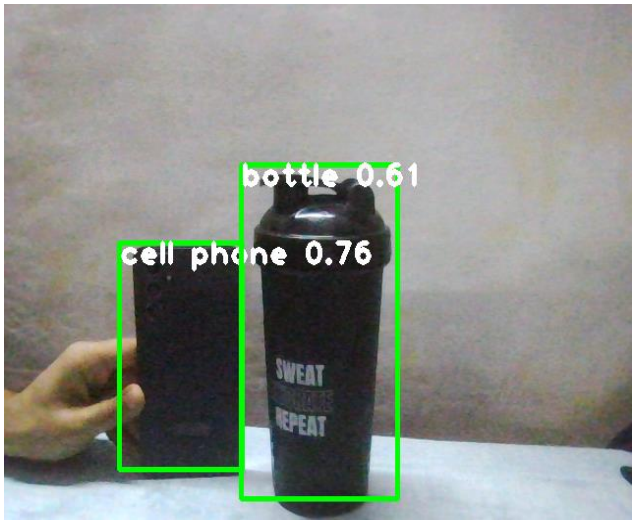

Fig 4: Object detection by the proposed system

Once the user initiates the recording process, the system seamlessly records from the camera until manually interrupted or completes the analysis of the provided video, generating an output file in .avi format. The use of a sophisticated ResNet model enables the extraction of feature vectors from each consecutive frame of the input video. These feature vectors undergo comparison using the cosine similarity function, calibrated with a threshold of 0.8 through iterative testing for optimal accuracy. This meticulous approach ensures that small details and expressive elements in the footage are preserved, mitigating the risk of information loss during the storage process.

While the system is operational, it opens another window that dynamically showcases either the live camera input or the video from the specified path. Concurrently, the system employs the YOLOv3 algorithm, trained on the COCO dataset, to detect objects within the input video. This real-time object detection capability enriches the user experience by providing visual feedback on detected objects, further enhancing the system's usability and interactivity.

Fig 5: Demonstration on a traffic camera

In essence, the project's GUI, coupled with advanced feature extraction techniques and intelligent object detection algorithms, empowers users to efficiently manage, analyze, and extract meaningful insights from video content. The seamless integration of these functionalities contributes to a comprehensive and user-centric solution for diverse video processing needs.

Table 1: Showcases why YOLOv3 is chosen over various other Object Detection Models (Based on Accuracy, Processing Speed and System Requirements that can be fulfilled)

Comparison of Object Detection Algorithms	Analysis & Conclusion
YOLOv2	Low Accuracy / Fast processing / Less System Requirements
YOLOv3	High Accuracy / Fast Processing / Less System Requirements
RetinaNet	High Accuracy / Slow Processing / More System Requirements
YOLOv4	High Accuracy / Fast Processing / More System Requirements

6. CONCLUSION

In conclusion, our CCTV detection project, utilizing YOLOv3 for object detection, ResNet18 for feature extraction, and cosine similarity for frame analysis, efficiently manages CCTV camera footage. It optimizes storage by identifying frames with significant movement, reducing overhead while preserving critical information. This approach is particularly beneficial for traffic surveillance, enabling accurate detection of violations, unusual activities, and specific vehicle types. The user-friendly GUI enhances interaction, allowing seamless input of video paths and initiation of processing tasks. Moving forward, this system promises to revolutionize traffic monitoring, enhancing safety, operational efficiency, and incident response capabilities in real-time scenarios.

REFERENCES

[1] Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767.

[2] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal Loss for Dense Object Detection. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.

[3] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016

[4] Redmon, J., & Divvala, S., & Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. arXiv:1506.02640.

[5] Salton, G., Wong, A., & Yang, C. S. (1975). A Vector Space Model for Automatic Indexing.

[6] Rao, D., & Shah, M. (2001). View Invariant Representation and Recognition of Actions.

[7] Yilmaz, A., Javed, O., & Shah, M. (2006). Object Tracking: A Survey.

[8] Chandan K & Shailendra S (Volume 83, pages 30113–30144) Security standards for real time video surveillance and moving object tracking challenges, limitations, and future: a case study

[9] L.N. Rani, Yuhandri & M Tajuddin (IJISAE - Vol. 11 No. 3 (2023))The Development of Residual Network (ResNet-18) Convolutional Neural Network (CNN) Architecture Combined with Content-Based Image Retrieval (CBIR) Method to Measure Logo Image Similarity Level