Acta Univ. Sapientiae, Informatica, **16, 2** (2024) 286–306

DOI: 10.47745/ausi-2024-0015

Intelligent Video Recording Optimization using Activity Detection for Surveillance Systems

Youssef ELMIR
Laboratoire LITAN
École supérieure en Sciences et Technologies de l'Informatique et du Numérique
RN 75, Amizour, 06300, Béjaia, Algérie
✉ elmir@estin.dz
iD 0000-0003-3499-507X

Hayet TOUATI
École supérieure en Sciences et Technologies de l'Informatique et du Numérique
RN 75, Amizour 06300, Béjaia, Algérie
✉ h_touati@estin.dz

Ouassila MELIZOU
École supérieure en Sciences et Technologies de l'Informatique et du Numérique
RN 75, Amizour 06300, Béjaia, Algérie
✉ o_melizou@estin.dz

Abstract. Surveillance systems often struggle with managing vast amounts of footage, much of which is irrelevant, leading to inefficient storage and challenges in event retrieval. This paper addresses these issues by proposing an optimized video recording solution focused on activity detection. The proposed approach utilizes a hybrid method that combines motion detection via frame subtraction with object detection using YOLOv9. This strategy specifically targets the recording of scenes involving human or car activity, thereby reducing unnecessary footage and optimizing storage usage. The developed model demonstrates superior performance, achieving precision metrics of 0.855 for car detection and 0.884 for person detection, and reducing the storage requirements by two-thirds compared to traditional surveillance systems that rely solely on motion detection. This significant reduction in storage highlights the effectiveness of the proposed approach in enhancing surveillance system efficiency. Nonetheless, some limitations persist, particularly the occurrence of false positives and false negatives in adverse weather conditions, such as strong winds.

Key words and phrases: Activity detection, video surveillance, object detection, YOLOv9, motion detection, recording optimization, machine learning, deep learning, CNN, Faster R-CNN

1 Introduction

The widespread installation of surveillance cameras has led to a significant increase in visual data, with estimates predicting over 1.4 billion cameras worldwide by 2024, generating vast amounts of video data daily. This surge poses major challenges in terms of storage, requiring hundreds of petabytes to manage this critical information. Efficient methods to manage and facilitate the search of this data have become imperative.

Surveillance cameras, essential for security, create an information overload. A single HD camera can produce nearly 650 megabytes of data per minute, resulting in a substantial data management challenge when multiplied across thousands of cameras in urban areas. Much of this data is redundant or irrelevant, necessitating careful consideration of optimal storage methods to ensure the rapid retrieval of important information.

The key problem is optimizing the storage of surveillance camera data to prevent storage congestion while maintaining necessary recordings. Object and motion detection algorithms emerge as promising solutions, filtering sequences to record only significant activities. This approach addresses the challenge of information overload in video surveillance.

This study aims to optimize storage space through innovative methods using motion and object detection algorithms and to implement a solution capable of discriminating important scenes. This dual objective seeks to balance storage efficiency with the relevance of the recordings. The structure of the remain of this paper is as follows: Section 2 reviews related works in intelligent surveillance systems, comparing their strengths and weaknesses. In section 3, a proposed methodology of activity detection is introduced with a overall architecture of the proposed system. Implementation is presented in section 4, and experiment results and discussion in section 5. The paper concludes with a summary of findings and a discussion of future research directions.

2 Related Works

Previous studies have explored various methods for optimizing video recording. Background subtraction and frame differencing are commonly used for motion detection, while object detection methods like Faster R-CNN and YOLO have shown promising results in identifying specific objects in video streams.

- Arham et al [1] developed a comprehensive real-time object detection system integrating motion detection, face detection, and human activity recognition, effective in real-world applications but lacking comparison with existing systems and detailed dataset descriptions.

- Pal et al [2] proposed a composite block matching algorithm for efficient motion estimation in video sequences, enhancing accuracy and processing speed but introducing computational complexity that may limit real-time applicability.

- Sadoun et al [3] addressed challenges such as illumination changes and shadow detection, creating a background modeling and subtraction algorithm. The model detects mobile objects but struggles with scene changes and fixed cameras, needing more quantitative measures.

- Sreenu et al [4] reviewed deep learning techniques in intelligent video surveillance, highlighting advancements and challenges like computational complexity and the need for large datasets. They called for more robust models and multi-sensor data integration.

- Xia et al [5] presented a hybrid LSTM-CNN model for human activity recognition, showing superior accuracy but facing high computational costs and the need for extensive labeled data.

- Adarsh et al [6] proposed a model using YOLO and ResNet-34 for detecting suspicious behavior, achieving significant precision but requiring substantial computational resources.

- Lys et al [7] developed a motion detection and object recognition system using OpenCV but lacked detailed results on detection efficiency.

- Alajrami et al [8] proposed AI techniques to enhance human identification in surveillance, improving accuracy and speed but facing high computational demands and privacy concerns.

- Ullah et al [9] combined CNN and LSTM for human activity recognition, improving accuracy but facing computational challenges and the need for broader dataset evaluation.

- Boumediene et al [10] created a real-time object detection model using YOLO, achieving good accuracy but requiring a more diverse dataset for better generalization.

- Suradkar et al [11] proposed an automatic motion detection system improving surveillance efficiency but needing evaluation in varied environments.

- Dhulekar et al [12] developed a two-step system for motion and object detection, suggesting deeper learning techniques for more complex object detection.

- Kapania et al [13] combined YOLOv3 and RetinaNet for robust object detection and tracking, demonstrating high performance but suggesting YOLOv3 for speed and efficiency.

- Dave et al [14] proposed a real-time action detection system addressing class imbalance and multi-label actions, achieving state-of-the-art performance but needing more representative datasets.

- Bosquet et al [15] developed STDnet-ST for small object detection, achieving state-of-the-art performance but proposing the use of GANs for generating synthetic small objects.

- Babiker et al [16] developed an automated system for recognizing human activities using neural networks, achieving high recognition rates but needing testing in complex settings.

- Deguerre et al [17] proposed a rapid object detection method in compressed videos, enhancing detection accuracy while reducing computational time but facing limitations with motion vector quality.

- Ren et al [18] introduced the Faster R-CNN algorithm for fast and accurate object detection, achieving high performance but facing computational intensity issues.

- Patil et al [19] proposed a method for crowd analysis using motion patterns and SVMs, achieving high recognition rates but relying on controlled datasets.

These studies collectively advance the field of video surveillance, addressing various challenges and proposing innovative solutions, though they often highlight the need for further research to overcome limitations in real-world applications as some of them are presented in Table 1 .

Table 1: Common Limitations in Reviewed Surveillance Systems

Limitation	**Description**
Fixed Camera Requirements	Systems like [3] and [11] work only with fixed cameras, limiting use in dynamic environments.
Lack of Real-time Processing	High computational complexity in systems like [2] and [4] hinders real-time performance.
Indoor/Outdoor Constraints	Systems such as [16] are designed for specific environments, limiting versatility.
Computational Requirements	High computational demands of algorithms like Faster R-CNN [18] and RetinaNet [13] challenge real-time deployment on limited devices.
Generalization Issues	Limited datasets affect generalization in systems like [10], [20], and [7].

To address limitations in human detection and surveillance, several research directions are proposed, including hybrid approaches that combine background subtraction with deep learning or optical flow with Support Vector Machines (SVM) for improved accuracy in dynamic environments. Enhancing generalization through diverse training datasets, synthetic data, data augmentation, and transfer learning can also improve model performance in real-world conditions. Developing versatile models for both indoor and outdoor settings using mixed datasets and context-aware mechanisms can enhance surveillance system adaptability.

This paper focuses on optimizing storage in surveillance systems by recording only significant actions using advanced activity detection techniques. Unlike traditional systems like Hikvision[1], which use continuous or basic motion detection, our approach selectively captures important activities, thus reducing storage needs. We evaluate our method against Hikvision to demonstrate its effectiveness in reducing storage without sacrificing surveillance quality.

The study explores intelligent video recording optimization through various techniques, including motion detection, background subtraction, optical flow, and advanced object detection methods like YOLOv3 and Faster R-CNN. A hybrid approach that combines initial motion detection with precise object tracking is identified as optimal for performance and resource efficiency. Future research should refine these models, expand datasets, and utilize advanced training devices. While integrating these techniques for real-time surveillance remains challenging, our hybrid approach aims to achieve more efficient and effective surveillance.

[1] https://www.hikvision.com

3 Methods

The Hikvision Surveillance System (HikvisionSS) was selected as the baseline for comparison in this study due to its existing deployment within our institution, École supérieure en Sciences et Technologies de l'Informatique et du Numérique (ESTIN[2]). The primary objective of this project is to optimize and reduce the storage space required for video recordings, which directly impacts the operational efficiency of the Hikvision system. As such, it was essential to evaluate the performance of our proposed approach in comparison to HikvisionSS, particularly in terms of the storage space required for recording, to ensure that our method provides tangible benefits within the context of its real-world application. We have chosen the Hikvision surveillance system as the baseline for comparison due to its widespread use in our institution and its standard recording modes, which include continuous recording and motion detection-based recording. Our proposed method focuses on an intelligent approach that records only significant actions, identified through advanced activity detection algorithms. This strategic selection allows us to directly assess the improvements in storage efficiency offered by our approach.

3.1 System Architecture

The proposed system integrates motion detection and object detection in a hybrid approach. As illustrated in Figure 1, the system begins with frame subtraction to identify regions of interest where motion is detected. These regions are then processed by the YOLO model, which detects and classifies objects, with a particular focus on human activity. This architecture ensures that only relevant scenes are recorded, significantly reducing unnecessary footage.

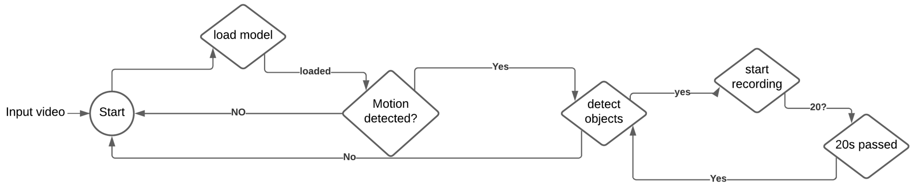


Figure 1: flowchart explaining the global architecture of the proposed system

[2] https://estin.dz/

Algorithm 1: Object Detection Driven Surveillance Video Recording

```
Require: Surveillance video stream
Ensure: Recorded video files with object activity
  Load the object detection Yolo model
  Initialize variables for motion detection and recording status
  while video stream exists do
    Read the current frame from the video stream
    Use the motion detection algorithm to detect motion in the current frame
    if  motion is detected then
      Use the object detection model to detect objects in the current frame
      if an object is detected  then
        if not currently recording then
          Start recording and note the time of last detection and currently
          recording == True
        end if
      end if
    end if
    if currently recording then
      if less than 20 seconds have passed since the last detection  then
        if objects are detected  then
          Update the time of last detection
        end if
        Write the current frame to the video file
      else
        if no objects are detected  then
          Stop recording and currently recording == False
        else
          Write the current frame to the video file and Update the time of last
          detection
        end if
      end if
    end if
  end while
  if Stop recording button = True  then
    Stop recording and currently recording == False
  end if
```

3.2 Activity Detection

3.2.1 Motion Detection:

Frame subtraction is employed to detect changes between consecutive frames, indicating potential activity. This method is computationally efficient and suitable for real-time applications [21]. To examine the different algorithms used for motion detection, we reviewed various works such as [3] and [16] utilizing background subtraction, [8] employing optical flow, and [2] proposing a composite block matching algorithm.

3.2.2 Object Detection:

The YOLO model is used to detect and classify objects within the video frames. YOLOv9 is chosen for its balance between speed and accuracy, making it ideal for real-time surveillance.

An example of using the YOLO method is found in [6], where the authors detect suspicious individuals and hostile behavior. They use the YOLO model, pre-trained on the COCO dataset [3], for human detection in video frames. These frames are then processed by ResNet-34 to recognize activities, achieving a precision of 82%. Despite its effectiveness, the model's reliance on two deep learning models demands substantial computational resources

The workflow for recording surveillance videos driven by object detection is detailed in Algorithm 1. The algorithm outlines how the system processes video streams, detects motion, and records activity only when objects of interest are present in the frame, ensuring efficient storage usage and effective monitoring.

4 Implementation

4.1 Data Preparation and Analysis

The data preparation involved collecting, labeling, and processing images for training the model. The dataset was sourced from various public repositories, focusing on human and car detection. Key data preparation steps included labeling objects using the Roboflow platform, preprocessing images (auto-orientation, resizing, contrast adjustment), and applying data augmentation techniques like grayscale conversion.

The dataset was split into training, validation, and testing sets using a 70:20:10 ratio. The dataset comprised 300 images for human detection from Kaggle and 100 im-

[3] https://cocodataset.org/

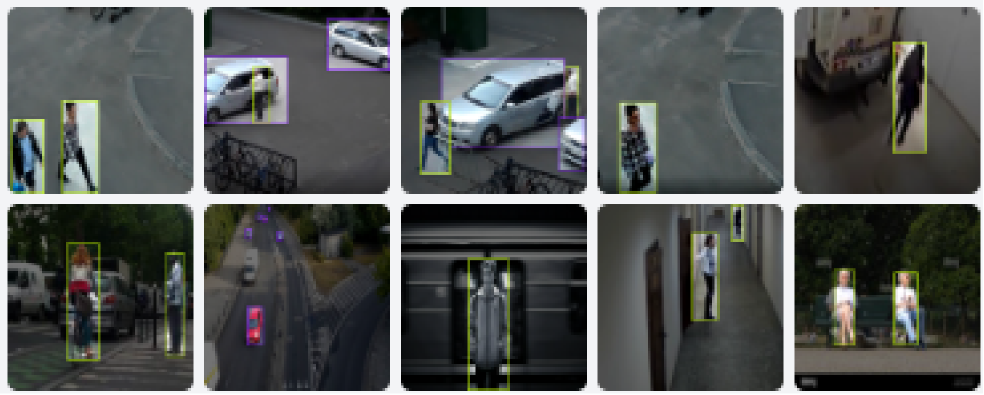


Figure 2: Samples from the dataset used to train YOLO model

ages for car detection from a GitHub repository, supplemented by additional images sourced from Google Images and other websites. The preparation process involved auto-orienting the images, resizing them to 646x640 pixels, and applying contrast adjustment and grayscale conversion to a portion of the dataset. The final dataset included 2,664 images labeled as 'Person' and 1,735 as 'Car', ensuring a well-balanced distribution across object classes.

The analysis highlighted the efficiency of the data preparation process, ensuring that the training model was robust and well-balanced across different scenarios and object classes. Visual tools like heatmaps and histograms were used to further analyze the distribution of annotations and objects within the dataset.

Figure 2 presents samples from the dataset, showcasing the diversity of scenarios and object classes included.

4.2 Model Training

4.2.1 Model Pre-training

The YOLO algorithm was implemented by cloning its repository from GitHub. Pre-trained weights from the MS COCO dataset were used as a starting point for fine-tuning on the custom dataset. The pre-training process involved running a script with key parameters, including 8 CPU workers for faster data loading, GPU utilization, a batch size of 16, and 500 training epochs. The model was trained from scratch using specific configurations and hyperparameters, with mosaic augmentation disabled after 15 epochs to enhance focus on original images. This pre-trained model was then fine-tuned on the custom dataset.

4.2.2 Model Fine-Tuning

The fine-tuning of the YOLO model on the custom dataset was performed using a modified script, which involved resizing input images to 640x640 pixels, setting a batch size of 16, and training for 25 epochs. The data configuration file was pointed to the custom dataset, while the model configuration file specified the YOLO architecture. Pre-trained weights from the initial training on the MS COCO dataset were used as the starting point.

4.2.3 Evaluation of Training Results

After completing the YOLO model training and fine-tuning, a thorough evaluation was performed using both qualitative and quantitative metrics to assess the model's performance. The validation process utilized a separate dataset to ensure the model's ability to generalize to unseen data. Key parameters for validation included using the prepared dataset, the best model weights from training, a batch size of 16, images resized to 640x640 pixels, a confidence threshold of 0.001, an IoU threshold of 0.5, and a maximum of 300 detections per image.

4.3 Comparative Analysis of Trained Models

The comparative analysis of the trained models focuses on identifying the most suitable model for deployment by evaluating various performance metrics, such as precision, recall, and mean Average Precision (mAP), in addition to considering computational resources and deployment constraints. This analysis not only aids in selecting the optimal model but also informs iterative improvement strategies by highlighting areas for refinement. The 7th Model, with the highest precision (87.0%), recall (82.0%), and mAP (90.1%), was selected for the proposed system, as it offers the best balance between precision and recall, ensuring accurate object detection and classification in video feeds.

4.4 Model Deployment

The deployment process involves preparing the chosen YOLOv9 based model by converting it into a deployable format and configuring it for integration. Once deployed, the model is accessible via an API, allowing seamless integration into the application environment. This setup ensures the model performs accurate inference tasks and provides reliable object detection capabilities, making it ready for real-world application scenarios.

Table 2: Comparative Table of Trained Models

Model	Version	Dataset Size	Epochs	Precision	Recall	mAP (0.5)
1st Model	9	664 Images	50	77.9%	73.7%	76.8%
2nd Model	8	664 Images	50	70.0%	71.7%	74.3%
3rd Model	9	1177 Images	30	81.6%	73.3%	80.9%
4th Model	9	1469 Images	30	85.3%	75.2%	82.4%
5th Model	9	1294 Images	30	83.6%	75.1%	82.2%
6th Model	9	1622 Images	25	86.5%	76.7%	85.5%
7th Model	**9**	**1920 Images**	**25**	**87.0%**	**82.0%**	**90.1%**
8th Model	9	1927 Images	100	85.1%	82.6%	88.7%

4.5 Model Evaluation and Training Visualization

The model's performance was evaluated using key metrics such as Precision (P), Recall (R), mean Average Precision at IoU threshold 0.5 (mAP50), and mean Average Precision across IoU thresholds from 0.5 to 0.95 (mAP50-95). Precision, which indicates the accuracy of positive predictions, was 0.869 overall (Car: 0.855, Person: 0.884). Recall, reflecting the model's ability to detect all relevant instances, was 0.824 overall (Car: 0.83, Person: 0.819). The mAP50 was 0.891 (Car: 0.899, Person: 0.883), showing high precision and recall at an IoU threshold of 0.5, while the mAP50-95 was 0.558 (Car: 0.638, Person: 0.478), demonstrating robustness across different IoU settings. These metrics, detailed in Table 3, highlight the model's strong performance in detecting cars and persons with high precision and recall, and reasonable generalization across IoU thresholds.

Table 3: Model Performance Metrics for Precision, Recall, and mAP.

Category	Precision	Recall	mAP50	mAP50-95
Car	0.855	0.83	0.899	0.638
Person	0.884	0.819	0.883	0.478
Overall	0.869	0.824	0.891	0.558

The confusion matrix, which evaluates the model's performance by comparing predicted labels to true labels, includes metrics such as True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN). The model demonstrates efficient performance, with a pre-processing time of 0.5 milliseconds, an inference time of 25.1 milliseconds, and Non-Maximum Suppression (NMS) time of 5.2 milliseconds per image. The inference time is particularly significant, as it reflects the model's capability for real-time image processing, which is crucial for applications requiring rapid decisions.

Table 4: Performance Speed Metrics.

Performance Metric	**Time (milliseconds per image)**
Pre-processing Time	0.5
Inference Time	25.1
Non-Maximum Suppression (NMS) Time	5.2

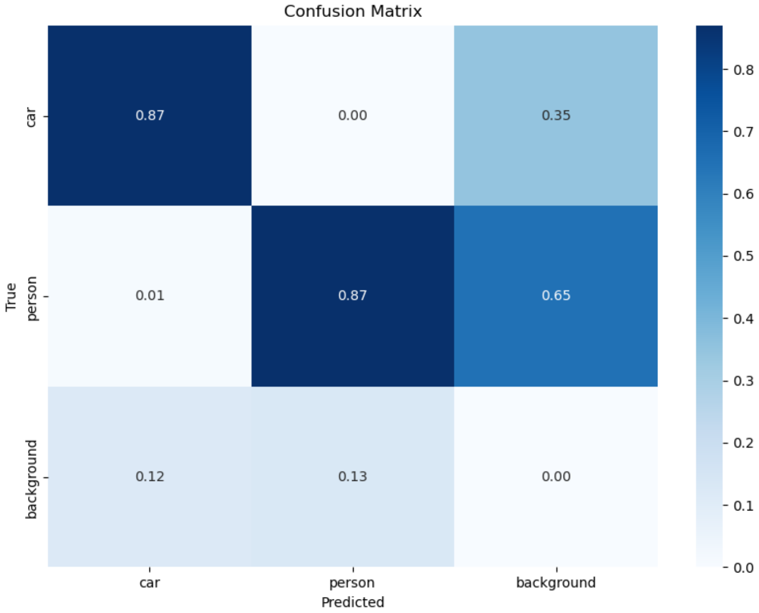


Figure 3: Confusion matrix for the proposed video recording optimization method applied to surveillance footage

.

Figure 3 shows the confusion matrix for activity classification (car, person, and background) in surveillance footage. The diagonal elements represent correctly classified instances, while the off-diagonal elements correspond to misclassifications. The matrix indicates an accuracy of 87% for detecting cars and persons, with slightly lower accuracy for the background class. Misclassifications mainly occurred between 'person' and 'background,' highlighting areas for potential model improvement.

The training progress is illustrated through plots showing the evolution of loss components and performance metrics over epochs. The bounding box regression loss (train/box_loss) decreased from 1.5 to 1.1, indicating improved accuracy in predicting bounding box coordinates. The classification loss (train/cls_loss) fell from 1.6 to about 0.6, reflecting enhanced classification performance. The Distribution Focal Loss (train/dfl_loss) also decreased from 1.5 to 1.25, demonstrating increased precision. Precision (metrics/precision) and recall (metrics/recall) metrics improved from 0.65 to approximately 0.85 and 0.82, respectively. Validation losses (val/box_loss, val/cls_loss, val/dfl_loss) followed similar trends, indicating good generalization. The mean Average Precision at an IoU threshold of 0.5 (metrics/mAP_0.5) increased from 0.45 to 0.90, and the mean Average Precision across IoU thresholds from 0.5 to 0.95 (metrics/mAP_0.5:0.95) rose from 0.40 to 0.75, highlighting the model's robust performance.

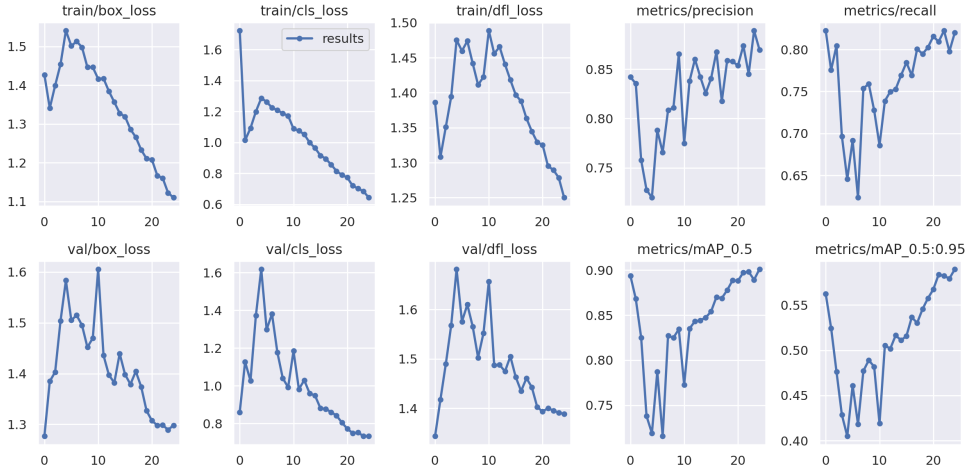


Figure 4: Training and Validation Performance Metrics of the 7th YOLOv9 based Model.

Figure 4 displays training and validation losses, precision, recall, and mean Average Precision (mAP) metrics over 25 epochs. The training and validation losses decrease consistently, indicating effective learning. Precision and recall improve steadily, with recall nearing 0.90 by the end of training. The mAP metrics also show positive trends, with mAP_0.5 surpassing 0.8. These results suggest the model continues to improve, with further training potentially enhancing performance. The visualizations in Figure 4 provide insights into the model's learning progress, crucial for assessing overfitting or underfitting. Overall, the best YOLOv9 model, fine-tuned with the custom dataset, exhibits high accuracy and efficiency, making it suitable for deployment in the proposed intelligent video recording optimization solution.

5 Deployment and Results

5.1 Storage Optimization

. This setup involves two different scenarios: one using the proposed intelligent recording approach and the other employing the Hikvision surveillance system's traditional continuous recording and motion detection features. The primary objective is to quantify the reduction in storage space achieved by our method, without sacrificing the accuracy and reliability of the surveillance footage. To validate the performance of the proposed application and the study objective, which focuses on the Optimization of Recording Storage for Surveillance Systems, a comparative study was conducted using Hikvision monitoring software[4] at ESTIN. Both systems were tested simultaneously under identical conditions to measure their effectiveness in optimizing storage by comparing the length of recorded videos. All recordings were made using a unified format (MP4, codec H.264, 1280x720, 16:9, 30 fps, bitrate 7 703 kbps).

5.1.1 Equipment and Configuration

Two types of cameras were used: a fixed camera and a flexible dome camera, positioned strategically in front of ESTIN's main gate and the building of Labs' building. The Hikvision system was configured with its specific hardware and software setup, while the proposed system was deployed on a local machine, specifically an i7 Core ThinkPad. This setup created a controlled environment to accurately measure and compare the performance of both systems in terms of storage optimization.

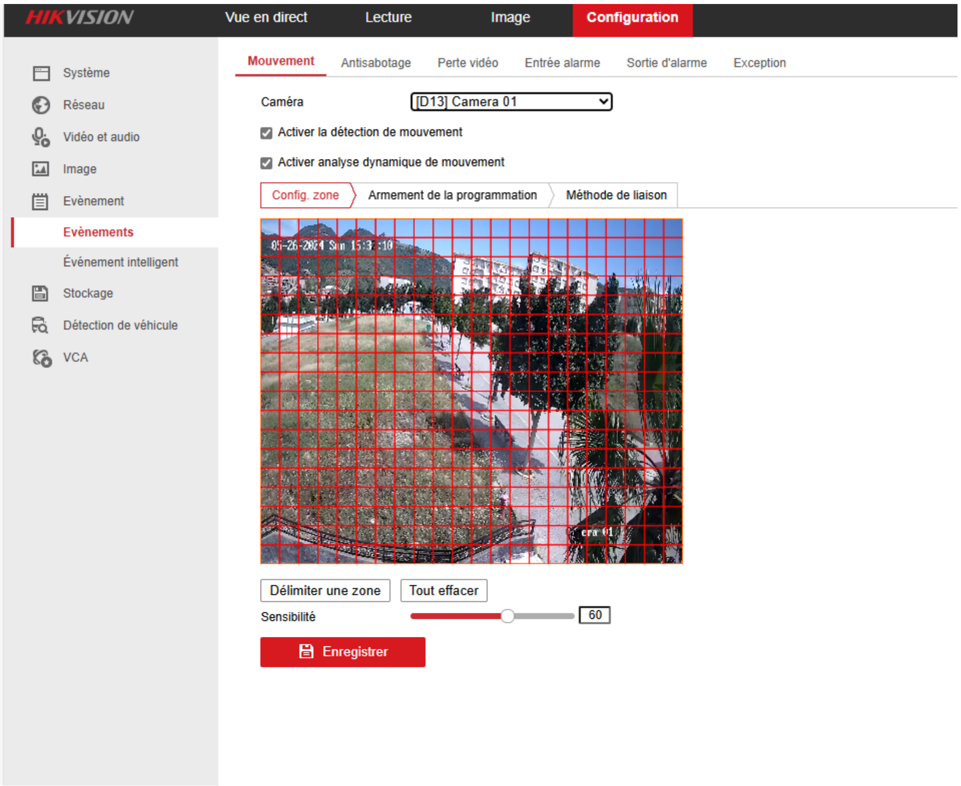
For the experimental evaluation, two distinct video records were used. The first video was a real-time test conducted live with the first camera This period of time in a workday, particularly from 2 PM to 3 PM, is likely one of the most active for students, teachers, and workers at ESTIN. The first video captures the scene in front of the main gate of ESTIN during this busy period, from 2 PM to 3 PM. The second video was recorded offline, after the real-time scene was captured. This second recording was done using a flexible camera positioned in front of the Labs' building from 2:30 PM to 2:45 PM. This dual approach allowed us to test the proposed

[4] https://www.hikvision.com/en/support/download/software/

method under both live and post-recording conditions, providing a comprehensive evaluation of its performance.

5.1.2 Real-Time Test

- Objective: Compare the storage optimization between the proposed system and the Hikvision software over a one-hour period of live surveillance with motion detection enabled.
- Methodology: Both systems were launched simultaneously and ran for one hour with motion detection enabled on both.
- Metrics Collected: Length of the recorded videos from both systems.



Caméra [D13] Camera 01

☑ Activer la détection de mouvement

☑ Activer analyse dynamique de mouvement

Sensibilité 60

Figure 5: Screenshot of the Hikvision System with Motion Detection Enabled

5.1.3 Video Upload Test

- Objective: Compare the performance of the proposed system and the Hikvision software using a pre-recorded video.
- Methodology: A 15-minute video was processed by both systems with motion detection enabled.
- Metrics Collected: Length of the resulting videos after processing.

Due to practical constraints, the number of video records used in this study was limited. However, the selected videos were chosen for their representativeness in assessing the effectiveness of the proposed system in optimizing storage space compared to the HikvisionSS baseline. Despite these limitations, the focus of our analysis remains on demonstrating the efficiency and applicability of our approach within the context of real-world surveillance scenarios.

5.1.4 Analysis of Storage Efficiency

The performance of the two systems was evaluated based on the following criteria:

- Recorded Video Length: The total duration of the videos recorded by each system under the same conditions. A shorter recorded length indicates better storage optimization.
- Detection Accuracy: The ability to correctly identify and record relevant activities (human and vehicular) without missing any significant events.

Table 5: Comparison of Recorded Lengths and Storage for Different Test Types

Test Type	**System**	**Video Length (minutes & seconds)**	**Storage Size (MegaBytes)**
1-Hour Real-Time Test (live)	Hikvision SS	60m & 00s	1 917
	Proposed	13m & 45s	769
15-Minute Video Test (recorded)	Hikvision SS	6m & 53s	379
	Proposed	6m & 20s	355

To evaluate the efficiency of the proposed method, a comparative analysis was conducted using two test types: a 1-hour real-time test and a 15-minute video test. The results in Table 5 compare the recorded lengths and storage sizes between the traditional Hikvision surveillance system and the proposed activity detection system. In the 1-hour real-time test, the Hikvision system recorded 60 minutes, consuming 1 917 MB, while the proposed system recorded only 13 minutes and 45 seconds, using 769 MB. This discrepancy is largely due to Hikvision's motion detection being highly sensitive to minimal motion, such as the movement of trees and palms caused by the wind, which triggered recording frequently. In contrast, the proposed system only initiated recording after detecting significant motion, using object detection to ensure that the recorded scenes contained relevant objects like cars or persons. In the second 15-minute video test, there was no wind, and both systems recorded only significant motions when detected. Therefore, the Hikvision system recorded 6 minutes and 53 seconds, consuming 379 MB, whereas the proposed system recorded 6 minutes and 20 seconds, using 355 MB.

These results demonstrate significant storage savings by recording only scenes with detected activity, reducing storage requirements without compromising important data. The storage savings chart underscores the efficiency of the proposed method compared to traditional continuous recording.

However, the proposed system has some limitations, such as false positives and false negatives in adverse weather conditions like strong winds, which can cause motion artifacts that affect the accuracy of detection as it is illustrated in Figure 3 and in some amples of Figure 6.

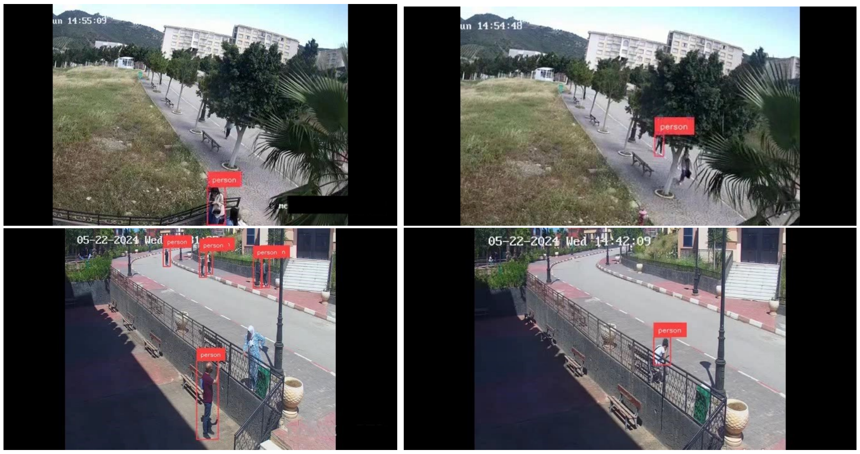


Figure 6: Samples of real-time validation

The validation results demonstrate that the proposed system significantly outperforms the Hikvision software in optimizing storage. In the one-hour real-time test, the proposed system recorded only 13 minutes and 45 seconds of video, compared to Hikvision's 60 minutes, even with Hikvision's intelligent motion detection option enabled. This highlights the proposed system's superior ability to filter out irrelevant footage and focus on significant activities.

In a 15-minute video upload test, the proposed system recorded 6 minutes and 20 seconds, while Hikvision recorded 6 minutes and 53 seconds. Although Hikvision captured the necessary scenes, it encountered some bugs. The proposed system proved more efficient by accurately detecting and recording pertinent activities, thereby minimizing storage usage.

These results confirm the effectiveness of the proposed model in optimizing recording storage for surveillance systems. Compared to Hikvision, which relies on continuous or basic motion-based recording, our method significantly reduces the storage space required by recording only meaningful activities. This optimization does not compromise the ability to capture critical events, as evidenced by the comparative analysis, making our approach a valuable tool for enhancing the efficiency and cost-effectiveness of surveillance operations.

6 Conclusion

In conclusion, this work effectively demonstrates that intelligent activity detection can optimize storage space in surveillance systems. By selectively recording only meaningful actions, our approach significantly reduces storage requirements compared to traditional systems like Hikvision, which rely on continuous or motion-based recording.

This study developed a hybrid system that combines motion detection via successive frame subtraction with the YOLOv9 model for detecting significant activities. YOLOv9 was selected for its superior detection capabilities and efficiency, achieving an accuracy of 87%. The combination of motion detection for initial screening with advanced activity detection optimizes both performance and resource usage, particularly benefiting the recording system at ESTIN by addressing storage and resource management challenges. Validation results showed substantial optimization in both time and storage space.

Future work should focus on refining model structures to further enhance detection precision and efficiency, leveraging more powerful training devices and larger, high-quality datasets. While our project made significant strides, further advancements are possible with additional resources and improved data. The methodologies developed hold potential applications beyond security, benefiting any field requiring real-time object detection and intelligent video recording.

Although the scope of our experiments was limited due to practical constraints, the results highlight the potential of the proposed method for optimizing video recording in surveillance systems. Future studies should consider expanding the dataset and exploring additional comparative methods to further validate the system's performance.

Author Contributions: *"Conceptualization and implementation, T. H. and O. M.; methodology and writing, T. H., O. M., and Y. E.; supervision, Y.E. All authors have read and agreed to the manuscript's published version."*

Funding: This research received no external funding

Data Availability: The data generated and analyzed in this study are not publicly available due to privacy and proprietary restrictions. Requests for access to the data can be directed to [h_touati@estin.dz, o_melizou@estin.dz], subject to approval and compliance with [any relevant conditions or restrictions].

Acknowledgments

We express our deepest gratitude to Mr. Nabil Aoughlis, Ms. Lylia Aberkane and Mr. Ridha Chekroune for their guidance and support.

Conflicts of Interest: The authors declare no conflicts of interest.

References

[1] M. Arham, A. Srivastava, A. G, and R. A. B, "Motion detection and human activity recognition for security," *International Journal of Engineering Research & Technology (IJERT)*, vol. 11, no. 05, 2023 (⇒ 288).

[2] A. K. Pal, B. Biswas, M. D. Jichkar, A. N. Jena, and M. Kumar, "Object detection driven composite block motionestimation algorithm for high-fidelitysurveillance video coding," 2023 (⇒ 288, 290, 293).

[3] S. Abdelbaki and O. Yacine, "Détection et suivi des objets mobiles: Application dans un environnement de foule," M.S. thesis, Université ECHAHID HAMMA LAKHDAR D'EL OUED, Algérie, 2016 (⇒ 288, 290, 293).

[4] G. Sreenu and S. Durai, “Intelligent video surveillance: A review through deep learning techniques for crowd analysis,” *Journal of Big Data*, vol. 6, no. 1, pp. 1–27, 2019 (⇒ 288, 290).

[5] K. Xia, J. Huang, and H. Wang, “Lstm-cnn architecture for human activity recognition,” *IEEE Access*, vol. 8, pp. 56 855–56 866, 2020 (⇒ 288).

[6] S. Adarsh, S. P. Giridhar Kannan, B. Vidhyasagar, and J. Arunnehru, “Suspicious activity detection and tracking in surveillance videos,” *Int J Emerg Technol Innovative Res*, vol. 7, no. 5, pp. 75–79, 2020 (⇒ 288, 293).

[7] R. Lys and Y. Opotyak, “Development of a video surveillance system for motion detection and object recognition,” *Advances in Cyber-Physical Systems*, 2023 (⇒ 288, 290).

[8] E. Alajrami, H. Tabash, Y. Singer, and M.-T. E. Astal, “On using ai-based human identification in improving surveillance system efficiency,” *2019 International Conference on Promising Electronic Technologies (ICPET)*, pp. 91–95, 2019 (⇒ 288, 293).

[9] A. Ullah, K. Muhammad, J. Del Ser, S. W. Baik, and V. H. C. de Albuquerque, “Activity recognition using temporal optical flow convolutional features and multilayer lstm,” *IEEE Access*, vol. 7, pp. 51 177–51 188, 2019 (⇒ 288).

[10] N. BOUMEDIENE *et al.*, “Détection d’objet en temps réel en utilisant une approche basée sur l’apprentissage profond,” Ph.D. dissertation, Université Ibn Khaldoun-Tiaret-, 2022 (⇒ 289, 290).

[11] H. Suradkar, A. Kolte, S. Jamdade, and S. Gokhale, “Automatic surveillance using motion detection,” *International Journal of Engineering Research and General Science*, vol. 3, no. 2, p. 525, 2015, ISSN: 2091-2730 (⇒ 289, 290).

[12] P. Dhulekar, S. Gandhe, A. Shewale, S. Sonawane, and V. Yelmame, “Motion estimation for human activity surveillance,” in *2017 International Conference on Emerging Trends & Innovation in ICT (ICEI)*, IEEE, 2017, pp. 82–85 (⇒ 289).

[13] S. Kapania, D. Saini, S. Goyal, N. Thakur, R. Jain, and P. Nagrath, “Multi object tracking with uavs using deep sort and yolov3 retinanet detection framework,” in *Proceedings of the 1st ACM Workshop on Autonomous and Intelligent Mobile Systems*, 2020, pp. 1–6 (⇒ 289, 290).

[14] I. R. Dave *et al.*, “Gabriellav2: Towards better generalization in surveillance videos for action detection,” in *2022 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, IEEE, 2022 (⇒ 289).

[15] B. Bosquet, M. Mucientes, and V. M. Brea, “Stdnet-st: Spatio-temporal convnet for small object detection,” *Pattern Recognition*, vol. 116, p. 107 929, 2021 (⇒ 289).

[16] M. Babiker, O. O. Khalifa, K. K. Htike, A. Hassan, and M. Zaharadeen, “Automated daily human activity recognition for video surveillance using neural network,” *2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)*, pp. 1–5, 2017 (⇒ 289, 290, 293).

[17] B. Deguerre, C. Chatelain, and G. Gasso, “Fast object detection in compressed jpeg images,” in *2019 ieee intelligent transportation systems conference (itsc)*, IEEE, 2019, pp. 333–338 (⇒ 289).

[18] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016 (⇒ 289, 290).

[19] S. Patil and K. S. Prabhushetty, “An efficient motion based group level activity recognition for intelligent video surveillance,” in *2021 Asian Conference on Innovation in Technology (ASIANCON)*, IEEE, 2021, pp. 1–8 (⇒ 289).

[20] H. R. Aradhya *et al.*, “Object detection and tracking using deep learning and artificial intelligence for video surveillance applications,” *international journal of advanced computer science and applications*, vol. 10, no. 12, 2019 (⇒ 290).

[21] S. Manchanda and S. Sharma, “Analysis of computer vision based techniques for motion detection,” in *2016 6th international conference-cloud system and big data engineering (confluence)*, IEEE, 2016, pp. 445–450 (⇒ 293).

Received: 27.06.2024; *Revised: 07.11.2024*; *Accepted: 26.12.2024*