Journal of Information Technology and Digital World (ISSN: 2582-418X)
www.irojournals.com/itdw/

Comparative Analysis of Deep Learning Architectures for Video Surveillance in Smart Cities

Yuvarani Samypen

Senior Lecturer, School of AI Computing and Multimedia, Lincoln University College, Malaysia.

E-mail: yuvarani@lincoln.edu.my

Abstract

Video surveillance is a vital part of smart cities, providing continuous monitoring and evaluation of metropolitan areas to improve security, safety, and efficiency. It uses strategically positioned cameras and associated technology to gather and analyze video footage, allowing for preemptive reactions to emergencies and other crucial occurrences. In this study, we have used deep learning algorithms for a comparative analysis of video surveillance in smart cities. Deep learning revolutionizes video surveillance by enabling intelligent systems to evaluate video data in real time, recognizing abnormalities, objects, and behaviors, resulting in more precise and efficient security procedures. The best architecture for object recognition in deep learning is CNN. This review will utilize various algorithms for comparison, as provided by CNN. In this study, the analysis of video surveillance systems is compared using well-known algorithms. This review will propose one of the most effective algorithms for video surveillance in smart cities following the comparison of the algorithms.

Keywords: Deep Learning (DL), Video Surveillance, Security, Object Recognition, Convolutional Neural Networks (CNN).

1. Introduction

Video surveillance, also known as CCTV, involves the use of secure cameras to supervise and keep track of movements in a designated area for security, safety, or tracking reasons. These cameras capture live footage that can be observed in real-time or recorded for

DOI: https://doi.org/10.36548/jitdw.2025.4.001
Received: 09.09.2025, received in revised form: 08.10.2025, accepted: 22.10.2025, published: 07.11.2025

future examination [10]. It is widely utilized in public areas like streets, parking spaces, airports, and shopping malls, as well as in individual buildings and residences. Video surveillance footage can be examined to detect possible security challenges or criminal behavior, serve as evidence in court cases, and monitor employee productivity and adherence to safety rules.

During World War II (specifically in 1942), German military researchers created the first CCTV system for nighttime surveillance of bomber runways and damage detection. After World War II (1949), contractors in the United States began to create and market CCTV systems designed for manufacturing and business applications. In 1960, state security teams and police departments started employing CCTV as a surveillance strategy. The first police department to implement CCTV for monitoring public streets was in Olean, New York, in 1968 [11]. The basic elements and operations of video surveillances in smart cities are illustrated in figure 1. It states that networked cameras, sensors and communication devices collect and send data in real-time to central systems for tracking, identifying and responding to events or activities in cites.

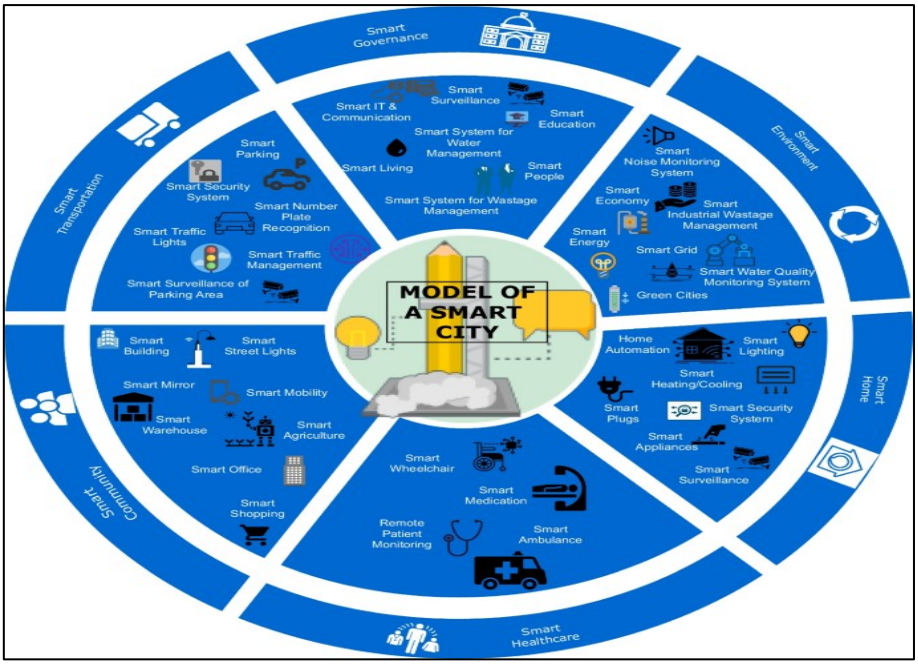

Figure 1. Basic Operations of Video Surveillance in Smart Cities [9]

Despite ongoing worries about privacy and application use, organizations can now adopt new technologies that facilitate more effective implementation and utilization of video surveillance. AI is being utilized more and more in video surveillance systems for the

identification and tracking of individuals and items, in addition to recognizing suspect behavior. This can assist organizations in enhancing their security and safety, as well as saving resources by decreasing the frequency of false detections. Deep learning, a category of AI, is tailored to examine extensive datasets. This characteristic makes it perfect for video surveillance systems, as it is capable of identifying and tracking individuals and objects, even in intricate and difficult surroundings. Network Video Recorders are gaining popularity due to their superior image quality, larger storage space, and quicker system implementation compared to traditional DVRs. A programming application software that examines footage from CCTV cameras is known as video analytics. Video analytics can serve to recognize and monitor individuals and items, in addition to uncovering suspicious behaviour [12].

The four core CNN architectures studied in this research—LeNet, AlexNet, VGG and YOLO—do not fully represent the range of advanced designs but are currently leading the fields of computer vision and smart surveillance. Better accuracy, feature representation and computational efficiency have been shown by recent designs like ResNet, EfficientNet and transformer-based models (such as Vision Transformer (ViT) and Swin Transformer). These models are able to learn complex visual correlations while remaining efficient on existing hardware by incorporating advanced methods such as compound scaling, self-attention and residual connections. The applicability of these comparisons to modern industry activities is restricted by the lack of these models. Future research should include these next-generation networks with the aim of providing an improved understanding of how traditional CNNs measure against hybrid and transformer-based models in terms of accuracy, real-time capabilities and adaptability to large-scale smart city environments.

1.1 Research Gap

Deep learning for video surveillance is commonly increasing, but there are still a number of unresolved issues in the research. The majority of earlier studies concentrates on conventional CNN designs like YOLO, SSD and faster R-CNN focusing on speed and accuracy of detection while neglecting scalability, dynamic environment adaptation and resource efficiency in edge deployments. The impact of real-world surveillance issues on long-term model dependability such as data imbalance, occlusion, changing lighting and noisy inputs is not widely researched. Furthermore, instead of using a variety of real-world data that accurately captures the conditions of smart cities, most evaluations are conducted using small or artificial datasets. Additionally, there are not many consistent performance indicators that combine

energy efficiency with precision, two factors that are essential for long-term, urban development. These drawbacks highlight the immediate need for deeper comparisons using recent architectures, more diverse datasets and multi-parameter optimization techniques that take implementation viability and model performance into consideration.

2. Literature Review

The proposed structure [1] incorporates a deep learning-based video surveillance approach that identifies prominent areas from a video frame without losing data and then encodes it in a smaller size. To test the framework's applicability, we used this technique on a variety of smart city case studies. The suggested approach yielded positive outcomes in the areas of bitrate (56.92%), peak signal to noise ratio (5.35 dB), and SR-based accuracy in segmentation (92% and 96% for two independent reference datasets). As a result, the development of low computational region-based video information makes it easier to enhance surveillance services in smartcities.

This study [2] analyzes the performance of three leading deep learning architectures: YOLOv5, SSD, and Faster R-CNN in smart surveillance situations. The algorithms are tested on conventional measures such as accuracy (mAP), speed (FPS), and computing efficiency using the COCO and personalized surveillance datasets. These results show that, whereas YOLOv5 excels in real-time performance, Faster R-CNN has superior accuracy but lags in speed. SSD strikes a balance between the two. This comparative analysis helps determine various architectures for surveillance systems depending on specific criteria.

A comprehensive review [3] of the current research on deep learning-based weapon detection was carried out to determine the methodologies employed, the main features of the existing datasets, and the significant issues in the field of automated weapon recognition. The Faster R-CNN and YOLO architectures were the most popular models. The combined use of real photos and artificial information led to enhanced performance. Numerous challenges in weapon recognition were observed, including insufficient lighting conditions and the difficulty of detecting small weapons, the latter being the most noticeable. Lastly, a few possibilities for the future are presented, with a particular focus on small weapon identification.

A comparison with three datasets [4]: University of California San Diego Pedestrian Dataset 2 (UCSD PED2), Chinese University of Hong Kong Avenue Dataset (CUHK Avenue),

and ShanghaiTech shows that the Proposed System (PS) technique regularly exceeds the traditional system technique. PS improves the Area Under the Curve (AUC) by 7.16% on UCSD PED2, 11.306% on CUHK Avenue, and 6.760% on ShanghaiTech. These outcomes demonstrate PS's exceptional performance in a variety of anomaly detection circumstances.

This research [5] conducted a systematic evaluation of the relevant research utilizing the Preferred Reporting Items for Systematic Reviews and Meta-Analyses Extension for Scoping Reviews (PRISMA-ScR) methodology. The evaluation included 266 of the 3622 eligible papers, with 47% presenting new techniques and 1% concentrating on technique implementation and comparison. Most research employed images instead of video as training or testing data, with 78% using clear data and only 7% employing both occluded and clear data. It was discovered that classic CNN architectures were primarily used. The study found a scarcity of research on the implementation and specification of CNN architectures, the creation of facial recognition models utilizing both clear and occluded videos and images, and the investigation of atypical CNN architectures. Factors such as occlusion, camera angle, and lighting conditions influenced facial recognition accuracy. This early study sheds light on the usage of CNNs in facial recognition and suggests that atypical CNN architectures should be studied further in future research.

The research [6] was conducted in Python utilizing the Keras package and TensorFlow as the backend. The goal was to create a network that performs at the cutting edge of video classification based on actions taken. Given the hardware limitations, there is a significant gap between the implementation options in this study and what is considered state-of-the-art. Throughout the work, various aspects in which this restriction affected development are described; however, it is demonstrated that this realization is viable and that producing expressive outcomes is attainable. The UCF101 dataset achieves 98.6% accuracy, compared to the best result ever reported, which was 98 percent. However, there are fewer resources available. Furthermore, the necessity of transfer learning in generating expressive outcomes is discussed, as well as the differences in performance between each architecture. Thus, this approach may pave the way for patentable outputs.

Effective and timely monitoring and calculation of building components are critical to controlling on-site building operations and assessing progress. Such activities are normally carried out by visual inspection, which makes them time-consuming and error-prone. This research proposes a video-based deep learning technique for automated building material

detection and counting. The proposed method was evaluated by identifying site workers and stacks of raised floor tiles in video footage from an exact indoor building site. The proposed YOLO v4 object identification system achieved higher average accuracy in less time than the standard YOLO v4 technique.

This study proposed three deep learning structures: OctDeepNet, OctDeepNet1, and OctDeepNet2, examining how different architectural elements like the number of layers, kernel sizes, and pooling measures affect classification accuracy. To evaluate the effectiveness of each architecture, evaluation metrics were utilized. The results show that the 50-layer deep learning architecture OctDeepNet2 is highly accurate compared to the 30-layer OctDeepNet and 17-layer OctDeepNet1 architectures. The outputs were examined for multiple batch sizes of 8, 16, and 32, as well as for different epochs of 25, 50, and 100. Utilizing a batch size of 32 for 100 epochs, OctDeepNet2 demonstrated improved accuracy of 98%, along with precision and recall values of 0.98 and 1.00, respectively, and an F score of 0.99. With its findings, the study offers important details for the choice of optimal deep learning architecture for categorizing retinal diseases based on OCT scans. The proposed framework aims to bolster ongoing initiatives in ophthalmology aimed at enhancing diagnostic precision and clinical procedures.

3. Methodology

A convolutional neural network (CNN), which is a kind of feedforward neural network, learns characteristics through the optimization of filters (or kernels). This category of deep learning networks has been utilized to analyze and generate predictions based on various forms of data, such as text, images, and audio. In the realms of image processing and computer vision, convolution-based networks are regarded as the standard for deep learning methods. They have been supplanted though not universally by more modern deep learning architectures like the transformer.

The development of a Convolutional Neural Network (CNN) utilized for video surveillance applications is depicted in Figure 2. It displays the steps involved in preprocessing, feature extraction and classification for identifying objects from surveillance video frames such as people and cars. Essentially, all inputs will undergo preprocessing before being transferred to a feature extraction method. This process will extract images to obtain qualitative representations for object detection. Then, the classification process will involve managing the

extracted images, where the objects will be classified as either a human or a vehicle. Consequently, the output image will feature the detected object with a specific representation.

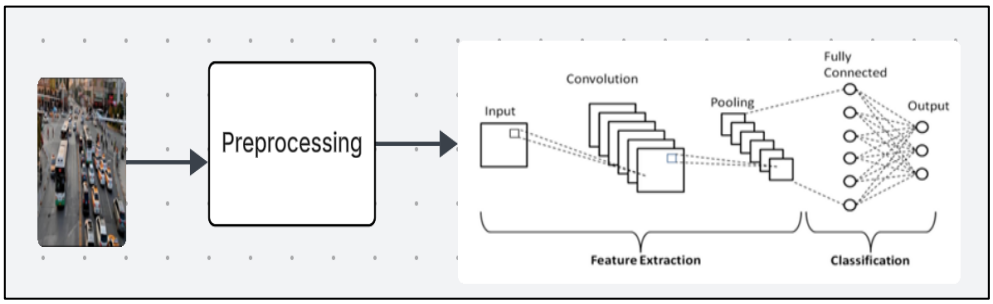

Figure 2. CNN Architecture Diagram [12]

Preprocessing is defined in this study as the preparation of all input images and video frames before transfer to the feature extraction layer. It is used to normalize the input data to make it easier for CNN architectures to efficiently learn relevant patterns for the process. This study illustrates that the preprocessing develops a qualitative visual representation suitable for object identification and classification. As a result, objects are identified as either a person or a vehicle; however, it does not identify the specific low-level procedures (such as normalizing or scaling). This preprocessing usually consists of converting frames to a fixed resolution, normalizing pixel values, reducing noise, splitting raw video into individual frames and may involve using data augmentation to enhance generalization based on standard CNN procedures in the context of video surveillance. Proper preprocessing allows each model to focus on significant features in surveillance data and provides consistent input quality across all studied architectures (LeNet, ALexNet, VGG and YOLO).

Based on the literature review, this study uses a range of common computer vision and public surveillance datasets and techniques. For example, the purpose of this study is to compare the accuracy, frames per second and computing efficiency in real-world smart city situations. YOLOv5, SSD and Faster R-CNN are assessed using the COCO dataset and customized surveillance datasets. The UCSD pedestrian Dataset 2 (PED2), CUHK Avenue and ShanghaiTech are the three well-known video anomaly detection datasets that are used in other cited research to compare models. The suggested work outperformed conventional techniques in identifying unusual behaviors in surveillance footage by achieving higher AUC scores. Furthermore, the evaluation highlights the limits of current datasets for face recognition in surveillance applications by finding that the majority of existing research uses images rather than complete videos, with only 7% including both incomplete and clean data. Overall, the

review work provides an effective framework for evaluating deep learning architectures in smart city video surveillance by offering a variety of datasets that include standard item detection tasks, real-world monitoring environments and anomaly recognition situations.

The review work's table and figures evaluate the efficiency of several CNN-based architectures (LeNet, AlexNet, VGG and YOLO) for video surveillance in smart cities. A systematic evaluation of several features including accuracy, speed (FPS), scalability, object identification capabilities, edge deployment viability and real-time performance is shown in table 1. The table indicates that LeNet is fast and lightweight but it is not good at detecting objects, which makes it inadequate for modern surveillance systems. The AlexNet model is not fully developed for real-time object identification, and it provides moderate speed and accuracy. VGG's advanced design allows it to achieve high accuracy, but in real-time its size and slow processing make it ineffective. YOLO is the most efficient and useful solution for smart city surveillance as it combines object detection, better speed, high accuracy and real-time capabilities.

Accuracy, speed, scalability and object identification performance were the standards used to evaluate the deep learning architectures. These features were selected because they are directly connected to the operational needs of smart city surveillance systems. The model's accuracy measures can be used to identify and categorize objects and are important in reducing missed detections and false alarms in critical security situations. Real-time video analysis is directly impacted by FPS and it is a measure of processing speed. The delays may lead to safety risks. An architecture's scalability is determined by its ability to implement across extensive surveillance camera networks without experiencing performance losses or adding to computational work. Its' ability to accurately identity and locate multiple items under various situations is measured by object detection performance and is important in spotting anomalies, automobiles and human activity in dynamic urban environments.

The study's comparative research is predicted on well-known video surveillance datasets including ShandhaiTech, USCD Pedestrian (PED2), COCO and CUHK Avenue databases. These datasets provide a variety of real-world scenarios such as crowded environments, changing lighting and complicated settings when trying to assess the resilience of deep learning architectures. Noise, occlusion and data imbalance have an enormous effect on model performance.

Table 1. Comparison Table for Algorithms of CNN Architecture for Video Surveillance in Smart Cities

Factors	LeNet	AlexNet	VGG (VGG16/ VGGG19)	YOLO
Introduced	1998	2012	2014	2016 & ongoing improvements
Architecture	Shallow CNN	Deep CNN	Very Deep CNN	Object Detection CNN
Use case	Digit/ Character recognition	Image-classification	Image-classification	Real-time object Detection
Accuracy	Low	Moderate	High	Very High
Speed (FPS)	Very High	Medium	Low	Very High
Scalability	Poor	Moderate	Poor	Excellent
Object Detection	Not designed for object detection	Not directly	Not directly	Native object detection
Edge Development	Lightweight but outdated	Heavy for edge use	Too large	Many lightweight variants
Real-Time capability	Yes	Somewhat	No	Yes

3.1 Efficient Constraints in Deep Learning

The size of the dataset, model depth and available processing power have a significant impact on the amount of time that deep learning architectures for video surveillance require to train. Deep networks like LeNet are suitable for low-resource environments and rapid testing as they have few parameters and require a short time to train. Advanced designs like VGG16 and VGG19 need hours or even days to be trained on high-performance GPUs due to their complex layer configurations and large parameter numbers. They continue to require sufficient GPU support to perform at their peak, and AlexNet achieves an effective balance between accuracy and training time. YOLO provides relatively fast convergence with minimal

processing costs when trained on GPUs with significant memory bandwidth to optimize object detection in real time.

Model viability is significantly affected by hardware limitations and simple designs like LeNet can operate on CPUs or edge devices but deeper models like VGG require strong GPUs with lots of VRAM and tensor cores. YOLO maintains efficient O(n2) O(n^2) O(n2) complexity by utilizing common convolutional filters and grid-based detection and VGG shows O(n3) O(n^3) O(n3) increase in operations due to its depth and input resolution. As a result, YOLO balances accuracy, deployment efficiency and computing cost across a variety of hardware platforms to provide the highest real-time video surveillance in smart cities.

4. Discussion

The CNN-based deep learning architectures such as LeNet, AlexNet, VGG, and YOLO were compared using specific parameters designed to satisfy the real-life needs for smart city video surveillance. Accuracy was included to evaluate each model's dependability, as accurately identifying objects is an important component in reducing false alarms in security systems. Speed (FPS) and real-time capabilities are considered because of the surveillance operations required for immediate processing and quick decision-making. The network depth and structure are evaluated to have a direct impact on memory consumption, computational effort, and the capacity to learn high-level features. Scalability was investigated to discover the efficiency of each model design that may be implemented over large monitoring networks with many cameras and various setting options. Object detection capabilities are highlighted, as recognizing people, cars, and unusual behavior is the main purpose of the surveillance system. As a result, models designed for detection, like YOLO, have significant benefits. Finally, the viability of edge implementation is considered in light of the growing demands for operating models on limited resources for edge devices in smart city infrastructure. These specifications ensure an accurate and application-based evaluation process that achieves an optimal balance between technical performance and practical distribution factors. It is important to reduce false positives and false negatives in real-time detection for dependable video surveillance, as inaccurate warnings have the potential to distract operators or fail to identify risks. Effective methods are used to improve data quality by using a detailed dataset with a range of lighting conditions, camera angles, occlusions, and difficult issues.

Figure 3 effectively follows the findings of Table 1 by displaying the performance patterns of different designs across significant parameters like accuracy and real-time processing. The figure shows that YOLO consistently performs better than other models, particularly in tasks involving real-time object detection when accuracy and speed are important. This figure demonstrates that LeNet and AlexNet fail to meet current surveillance requirements, while VGG, despite its accuracy, fails in terms of speed. This study concludes that YOLO is the best deep learning architecture for video surveillance in smart cities as it provides the best balance between performance and usability.

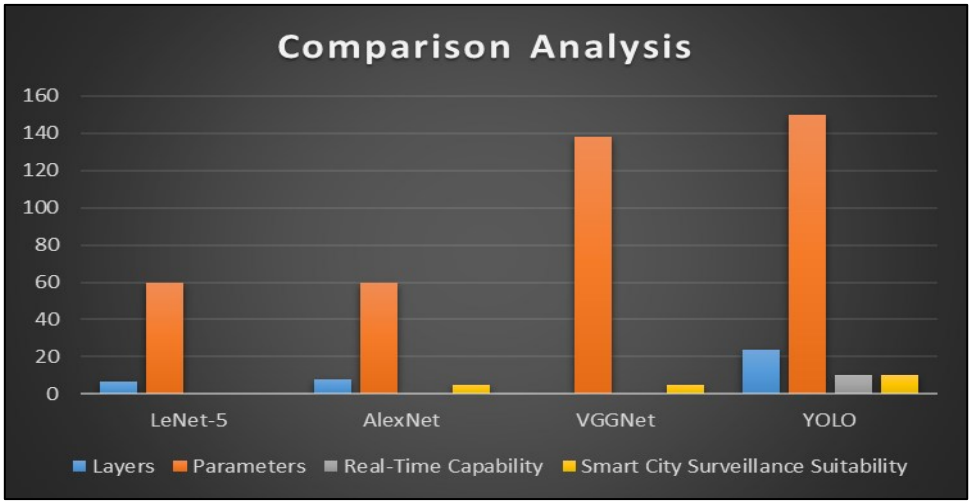

Figure 3. Comparison Analysis of the Algorithms

Figure 3 illustrates that YOLO outperforms other algorithms in video surveillance within smart cities, particularly for real-time object detection. This involves primarily a comparison of the capability and parameters used for video accessibility and surveillance suitability. The real-time image processing capability of YOLO makes it an ideal choice for surveillance tasks in smart cities, like monitoring traffic and detecting pedestrians, where both speed and precision are essential. By examining both Table 1 and Figure 3, it becomes evident that YOLO can be readily applied to video surveillance, as it is fundamentally intended for object detection, which aligns with the primary function of video surveillance: to detect objects and identify any suspicious activities in order to guarantee the security and safety of the people.

5. Conclusion

In this study, deep learning techniques are utilized to compare video surveillance approaches for smart cities. CNN algorithms such as LeNet, AlexNet, YOLO, and VGG are

selected for comparison among deep learning architectures. In the context of smart cities, the architecture that is most useful and frequently utilized for real-time, scalable, and accurate surveillance is YOLO. While VGGNet is robust, it requires a lot of resources, and LeNet and AlexNet are no longer viable options. Architectures such as YOLO, which strike a balance between speed and accuracy, are of greatest benefit to surveillance systems. In the future, real-time AI algorithms will be integrated at the edge to achieve the necessary speed and responsiveness. By focusing on privacy-preserving analytics in video surveillance systems, the aim is to ethically develop these systems. Models with low power consumption will be utilized to provide ubiquitous surveillance applications.

Reference

[1] Zahra, Asma, Mubeen Ghafoor, Kamran Munir, Ata Ullah, and Zain Ul Abideen. "Application of region-based video surveillance in smart cities using deep learning." Multimedia Tools and Applications 83, no. 5 (2024): 15313-15338.

[2] Deka, Brajen Kumar. "A Comparative Study of Deep Learning Architectures for Real-Time Object Detection in Smart Surveillance Systems." International Journal of Machine Learning, AI & Data Science Evolution E-ISSN: 3067-5073 1, no. 01 (2025): 32-42.

[3] Santos, Tomás, Hélder Oliveira, and António Cunha. "Systematic review on weapon detection in surveillance footage through deep learning." Computer science review 51 (2024): 100612.

[4] Suma, S. "A Deep Learning based Integrated Memory Aware Twin AutoEncoder Network for Anomaly Detection in Video Surveillance on Edge Devices." International Journal of Intelligent Engineering & Systems 18, no. 1 (2025).

[5] Nemavhola, Andisani, Serestina Viriri, and Colin Chibaya. "A Scoping Review of Literature on Deep Learning Techniques for Face Recognition." Human Behavior and Emerging Technologies 2025, no. 1 (2025): 5979728.

[6] Hemamalini, V., D. Jayasutha, V. R. Vinothini, R. Manjula Devi, A. Kumar, and E. Anitha. "Innovative Video Classification Method based on Deep Learning Approach." Recent Patents on Engineering 19, no. 2 (2025): E271023222880.

[7] Wong, Johnny Kwok Wai, Fateme Bameri, Alireza Ahmadian Fard Fini, and Mojtaba Maghrebi. "Tracking indoor construction progress by deep-learning-based analysis of site surveillance video." Construction Innovation 25, no. 2 (2025): 461-489.

[8] Rajan, Ranjitha, and S. N. Kumar. "Deep Learning Architectures for OCT Images Retinal Disease Classification." SN Computer Science 6, no. 2 (2025): 1-16.

[9] Sharma, Himani, and Navdeep Kanwal. "Video surveillance in smart cities: current status, challenges & future directions." Multimedia Tools and Applications (2024): 1-46.

[10] https://www.isarsoft.com/knowledge-hub/video-surveillance#:~:text=Video%20surveillance%20is%20the%20use,also%20referred%20to%20as%20CCTV.

[11] https://www.3sixtyintegrated.com/blog/2023/07/26/history-video-surveillance/

[12] https://medium.com/@sasirekharameshkumar/deep-learning-basics-part-8-convolutional-neural-network-cnn-4cff567bad46