Current Journal of Engineering and Science Research
Volume 1, Issue 1, pp 39-46, November 2024
ISSN (online): 3049-1738
DOI: https://doi.org/10.47001/CJESR/2024.101005
Available Online at https://www.cjesr.com

Security Framework for Enhanced Surveillance Monitoring in Banking Systems with Behavioural Pattern Identification Using Machine Learning

[1]Abdulrahman A Ibrahim, [2]Noura Abdullah Kamal, [3]Yassir Ibrahem Rehiayan, [4]Reshied Solaiman Saleh

[1,2]Electrical and Computer Engineering, King Abdullah University of Science and Technology, Saudi Arabia
[3]Electrical Engineering, Alfaisal University, Riyadh, Saudi Arabia
[4]Computer and Information Sciences, Princess Nora bint Abdul Rahman University, Saudi Arabia

Abstract: Robberies, altercations, and various atypical incidents are increasingly prevalent in banking institutions. In response to these challenges, numerous video surveillance systems have been implemented; some relying on human oversight while others utilize artificial intelligence. Our objective is to create a robust surveillance system that leverages machine learning to identify anomalous behaviors and trigger alerts. Video surveillance entails the observation of specific behaviors that warrant attention, as well as the monitoring of scenes that deviate from the norm. This process involves pinpointing particular locations with a heightened likelihood of unusual activities, allowing for targeted monitoring by surveillance cameras.

Keywords: Cyber-physical system, Surveillance, ethics, regulation, computer vision, and video analytics, action modelling, CCTV.

I. INTRODUCTION

The bank employs for enhanced proactive monitoring across numerous branches, ATMs, and digital lobbies. By analyzing various parameters derived from these video recordings, the institution aims to address several operational challenges within its branches. The bank seeks to leverage video analytics to gain insights into customer sentiments, identify behavioral patterns and actions in specific locations for enhanced proactive monitoring, and ultimately improve service delivery to its clientele.

It is crucial for financial institutions and corporations to implement robust security systems to safeguard their valuable assets. This paper proposes a security framework that leverages image processing, touchscreen technology, and verification software, offering enhanced protection compared to existing systems. The proposed system is applicable in bank vaults, corporate environments, and personal secure locations. The object detection methodologies employed involve color processing, which utilizes primary filtering to remove irrelevant colors or objects from the images. Additionally, shape detection techniques are incorporated, utilizing edge detection methods. Through the application of object detection techniques in image processing, verification processes are effectively conducted. Video Surveillance is the process of identifying activities that are different from that normal one or the one which is anomalous or which is improper in behavior.

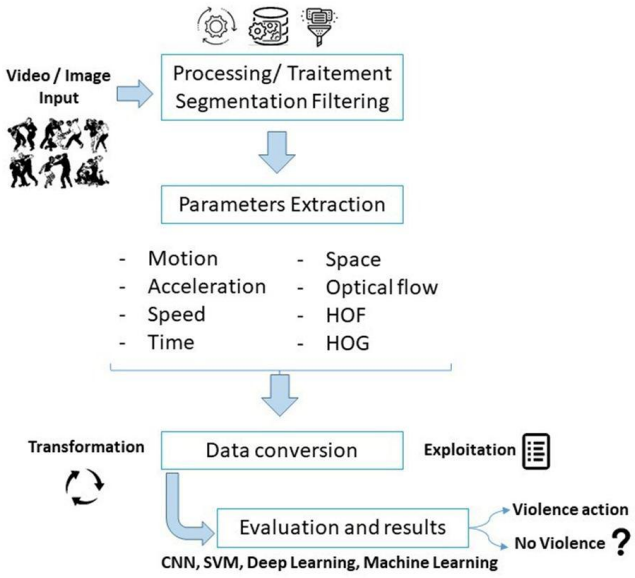

Figure 1: The violence detection systems

This is an automatic video anomaly detection process that reduces labor and waste time. Video Surveillance is very useful to identify abnormal events and maintain social control. In

Current Journal of Engineering and Science Research
Volume 1, Issue 1, pp 39-46, November 2024
ISSN (online): 3049-1738
DOI: https://doi.org/10.47001/CJESR/2024.101005
Available Online at https://www.cjesr.com

banks, Video Surveillance is used to provide a high level of security and solve financial problems banks will be under control and crimes will be minimized.

Unsupervised decreases the manual work in anomaly detection. In an unsupervised model is input unlabeled data and the model learns from unlabeled one with the help of algorithms model analyzes and clusters the unlabeled dataset.

In the context of real-world anomaly detection, the primary objective is to identify and signal instances of abnormal behavior. Consequently, the detection of anomalies in banking can be viewed as a form of coarse-level video analysis, which distinguishes anomalous patterns from those that are considered normal.

II. PROBLEM STATEMENTS

This paper [1] presents learning framework. It uses representation errors to build sparse combinations and is reliable and efficient. The model is faithful to original sparse data and is verified by a large number of videos. The method is robust and distinguishes between abnormal patterns and normal patterns

This paper [2] proposes spatial anomaly detection that spans spatial scale, time, and space. It introduces challenges such as making anomalies dependent on scales, using different models of normalcy for different tasks, and using crowded scenes. Spatial anomaly is a broad term for any kind of extraordinary disruption to normal space-time.

This paper [3] proposes two methods to solve the problem of perceiving meaningful activities in a long video. The first involves handcrafted spatiotemporal local features and the feed-forward autoencoder for learning local features and classification.

In this paper [4] multiple local monitors generate alerts when an abnormal event is detected, but lack sequence monitoring and are not suitable for large-scale video surveillance projects.

In this paper [5] they have addressed the problem of learning spatiotemporal features using 3D ConvNets.and they are trained on large-sized video datasets.

[6] A high level of security and applications for classification of targets and analysis of behavior and detection of abnormal events.

Table 1: Different types of Machine Learning Algorithms

Machine Learning Algorithms			
Linear Regression	**Non-Linear Regression**	**Linear Classification**	**Non-Linear Classification**
Ordinary Least Squares Regression	Multivariate Adaptive Regression Spines (MARS)	Logistic Regression	Mixture Discriminant Analysis (MDA)
Stepwise Regression	Suport Vector Machine (SVM)	Linear Discriminant Analysis (LDA)	Quadratic Discriminant Analysis (QDA)
Principal Component Regression	K-Nearest Neighbor (kNN)	Partial Least Squares Discriminant Analysis	Regularized Discriminant Analysis (RDA)
Partial Least Squares Regression	Neural Network		Neural Network, Flexible Discriminant Analysis (FDA)
Ridge Regression	Classification and Regression Tress (CART)		Suport Vector Machine (SVM)
Least Absolute Shrinkage	Conditional Decision Trees		K-Nearest Neighbor (kNN)
Selection Opeartor (LASSO)			
ElasticNet	Modal Trees		Naive Bayes
	Rule systems		Classification and Regression Tress (CART)
	Bagging CART		C4.5
	Random Forest		PART
	Gradient Boosted Machines (GBM)		Bagging CART
	Cubist		Random Forest
			Gradient Boosted Machines (GBM)
			Boosted C5.0

In this paper [7], an anti-theft device detects theft using motion and generates an alarm, capturing images only when motion exceeds threshold value to save data space.

In this paper [8], this paper proposes an approach to improve the classification of normal and abnormal videos, but

Current Journal of Engineering and Science Research
Volume 1, Issue 1, pp 39-46, November 2024
ISSN (online): 3049-1738
DOI: https://doi.org/10.47001/CJESR/2024.101005
Available Online at https://www.cjesr.com

lacks important properties such as view- and scale invariance.

In this paper [9], LRP method propagates classifier decisions and finds voxels, providing unsupervised preprocessing with high accuracy rate.

III. CONVOLUTIONAL NEURAL NETWORKS

The current system employs deep Multiple Instance Learning (MIL), which is a variant of supervised learning. In this approach, a single class label is assigned to a collection of instances, which include both labeled positive and negative examples. The classifier is developed through a function-based learning process.

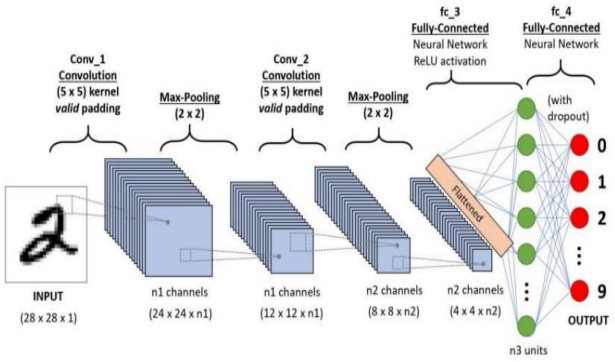

Figure 2: CNN Architecture

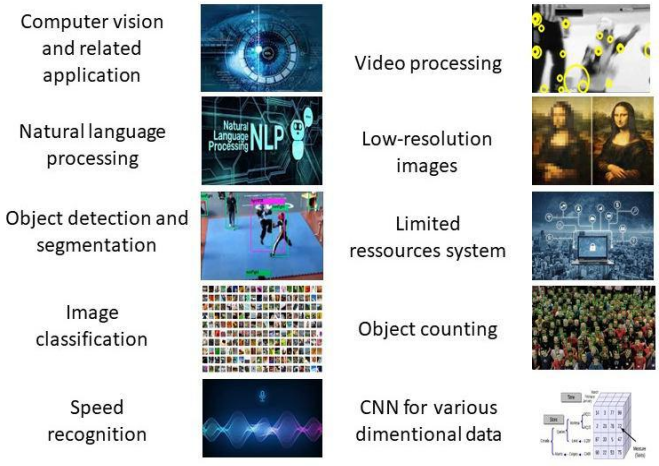

Figure 3: Applications of CNN

In contrast, Convolutional Neural Networks (CNNs) are primarily utilized for image classification. CNNs are specifically structured to learn spatial hierarchies of features from images through the use of multiple layers.

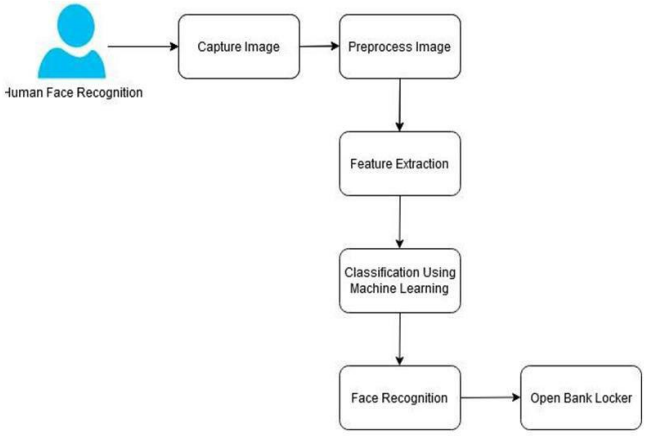

Figure 4: System Diagram

This technique falls under supervised learning, as the model is trained with labeled data. The main difference is CNN & Deep MIL.CNN processes individual images while Deep MIL is designed to process bags of images. Both algorithms have specific characteristics of anomaly and data available for training.

If abnormal activities are well-defined and have distinct features then the CNN approach is appropriate. CNN is effective in learning visual patterns and identifying abnormal events based on visual characteristics. Hence for video surveillance in a bank,the CNN algorithm is best when compared to Deep MIL.

IV. IMPORTING LIBRARIES AND LOADING DATASETS

Firstly imported several libraries including OpenCV, TensorFlow, and MoviePy, and defines several functions to work with image and video data. Then importing some basic Python libraries such as NumPy and datetime and sets up inline plotting with matplotlib.

Then importing several functions from TensorFlow and sci-kit-learn libraries. It also imports the to_categorical function for converting labels into one-hot encoded vectors and the plot_model function for visualizing the architecture of a model.

The random seed has been established for the NumPy, random, and TensorFlow libraries. The variable constantseed_constant serves as the seed value, and by configuring this seed, the random number generation within

Current Journal of Engineering and Science Research
Volume 1, Issue 1, pp 39-46, November 2024
ISSN (online): 3049-1738
DOI: https://doi.org/10.47001/CJESR/2024.101005
Available Online at https://www.cjesr.com

these libraries will be reproducible. Such reproducibility is advantageous for debugging purposes and for maintaining consistent outcomes.

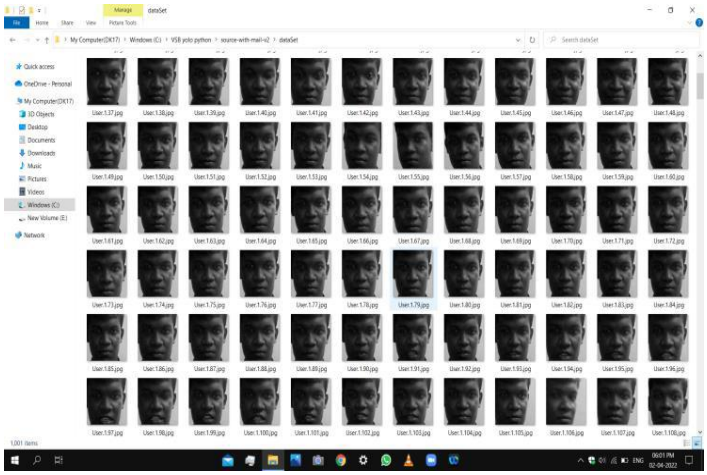

Figure 5: Dataset Creation

Additionally, a Matplotlib figure has been created to showcase a random selection of two videos from the UCF50 dataset. This dataset comprises video clips depicting human actions, with each clip categorized into one of 50 distinct classes.

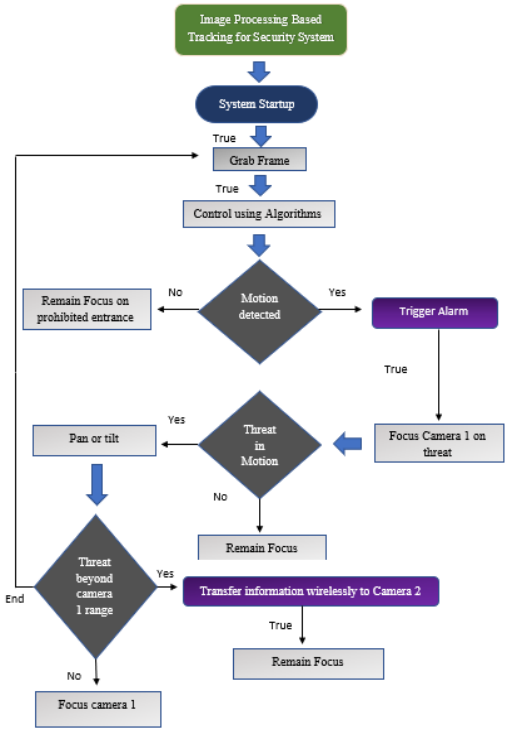

Figure 6: System flow

The code initiates by compiling a list of all class names present in the UCF50 dataset. Subsequently, it generates a random sample of two class names and selects a random video file from each of these classes. For each chosen video file, the code utilizes OpenCV's VideoCapture function to read the first frame of the video, converts the BGR frame to an RGB format, overlays the class name text onto the frame using OpenCV's putText function, and finally displays the frame within a subplot of the Matplotlib figure.

The code iterates through both selected videos and displays them side by side in the figure.The resulting figure has a total of 8 subplots, with each subplot showing a different frame from one of the two selected videos.

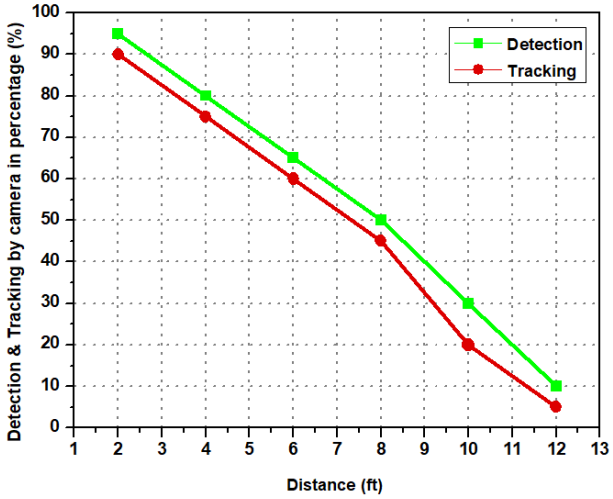

Figure 7: comparison of camera vs. distance

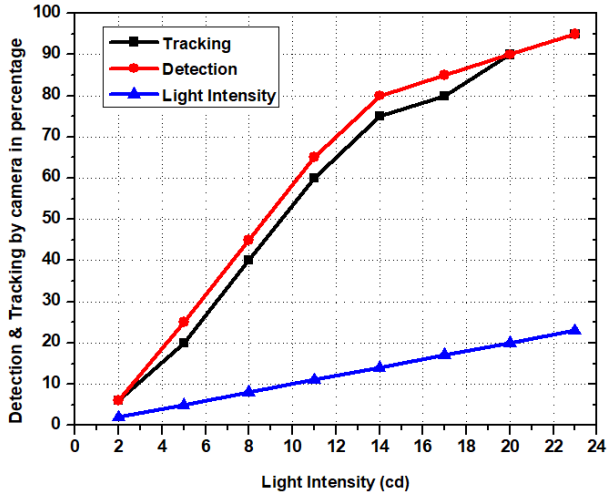

Figure 8: Comparison detection of tracking vs. light intensity

Current Journal of Engineering and Science Research
Volume 1, Issue 1, pp 39-46, November 2024
ISSN (online): 3049-1738
DOI: https://doi.org/10.47001/CJESR/2024.101005
Available Online at https://www.cjesr.com

We have then set up some variables for later use in the code.

image_height and image_widthare set to 64, which is the size of the input images that will be used to train a deep-learning model.

max_images_per_class is set to 400, which is the maximum number of images that will be used per class for training the model.

dataset_directory is set to "UCF50", which is the directory containing the UCF50 dataset.

classes_list is a list containing the two class names ("Abnormal" and "Normal") that will be used for training the model.

model_output_size is set to 2, which is the number of output classes for the model. This is equal to the length of classes_list

Feature Extraction

A Python function has been developed to extract frames from a video file and return them as a list. This begins by creating an empty list named frames_list, designated to hold the extracted video frames.

Subsequently, the function employs OpenCV's VideoCapture method to sequentially read frames from the video file. Each frame is resized to a predetermined dimension of image_height and image_width, both set to 64 in the preceding code block.

Following the resizing process, the frame undergoes normalization by dividing its normalized frame is then added to the frames_list. This procedure continues as the function reads frames from the video file, repeating the aforementioned steps until all frames have been processed. Once all frames have been read, the function releases all resources and returns the frames_list.

V. ACTUAL WORKING

We have created a function called create_dataset() which is used to extract features and labels from the videos of the UCF50 dataset. Finally, it converts the features and label lists to numpy arrays and returns them.

The create_dataset() function takes a while to execute since it extracts frames from all the videos in the UCF50 dataset, and then creates a balanced dataset with a fixed number of frames from each class. Depending on the number of videos in the dataset, this process can take several minutes. Once the function has been completed, it returns the features and label arrays, which contain the extracted frames and corresponding class labels, respectively.

We used Keras' to_categorical method to convert the labels into vectors using one-hot encoding. In this case, the to_categorical method is applied to the labels array that contains the class labels for each image in the dataset. The resulting variable one_hot_encoded_labels contains the one-hot encoded vectors for each class label. Finally, we stored the training history in the model_training_history variable.

VI. RESULTS

The model.save() function facilitates the preservation of the model in a Hierarchical Data Format (HDF5) file, designated with the .h5 extension. This functionality enables subsequent loading of the model for making predictions on novel datasets. Upon execution, a graphical representation is generated, illustrating the training and validation loss on the y-axis against the number of epochs on the x-axis. The training loss is depicted by a blue line, whereas the validation loss is represented by a red line.

The initial argument of this method pertains to the test features, referred to as "features_test," while the concluding argument corresponds to the test labels, identified as "labels_test." It is crucial to acknowledge that the accuracy obtained from the test dataset serves as a significant metric for evaluating the model's efficacy on previously unseen data, thereby providing assurance that the model is not exhibiting overfitting tendencies with respect to the training dataset.

The trained model was stored in a file whose name encapsulates the current date and time, along with the model's evaluation loss and accuracy metrics. The model.save() function effectively saves the model in HDF5 format, utilizing the .h5 file extension. This capability allows for future loading of the model to facilitate predictions on new data. Furthermore, a graph is produced that delineates the training and validation loss on the y-axis in relation to the number of epochs on the x-axis, with the blue line signifying training loss and the red line denoting validation loss.

Current Journal of Engineering and Science Research
Volume 1, Issue 1, pp 39-46, November 2024
ISSN (online): 3049-1738
DOI: https://doi.org/10.47001/CJESR/2024.101005
Available Online at https://www.cjesr.com

The graph shows how the loss decreases over time as the model trains. The goal is to have both lines decrease, but not to overfit the training data by having the validation loss increase while the training loss decreases.

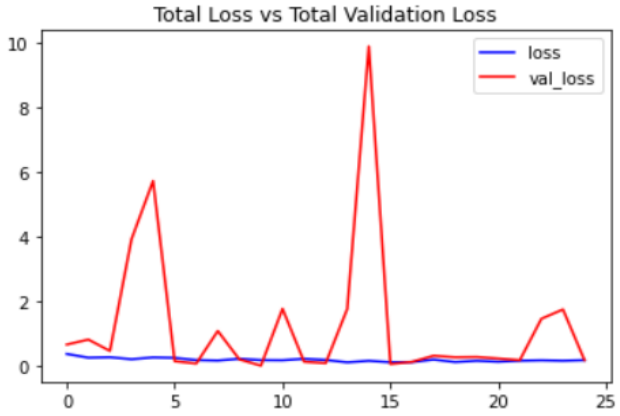

Figure 9: The loss measurement of total validation

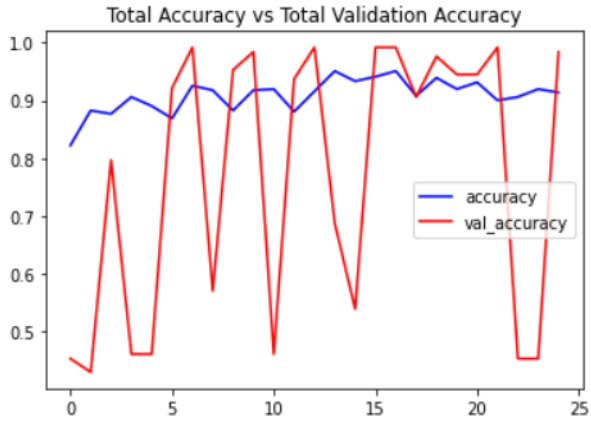

Figure 10: Comparison between Total Accuracy vs Total Validation Accuracy

In the above graph, the blue line indicates the accuracy, and the red line indicates the validation accuracy. A function named predict_on_live_video() has been developed, and a Deque object with a predetermined size has been initialized to facilitate the implementation of a moving or rolling average. The video file is accessed through a video capture object. The overlayed video files are generated using a video writer object, where the frames are sequentially read and written. These frames are resized to specific dimensions, and normalization is performed by dividing the resized frame. The normalized image frame is then passed to the model, which outputs predicted probabilities. These predicted label probabilities are appended to the Deque object. It is ensured that the Deque is populated prior to commencing the averaging process, after which the predicted label probabilities stored in the Deque are converted into a NumPy array. The average of the predicted label probabilities is calculated column-wise, and the predicted probabilities are transformed into labels by identifying the index of the maximum value. The class name corresponding to the predicted label is accessed, and the class name text is overlaid on the frame. Subsequently, the input video, which is anomalous in nature, is loaded, and the predict_on_live_video() function is invoked to initiate the prediction process.

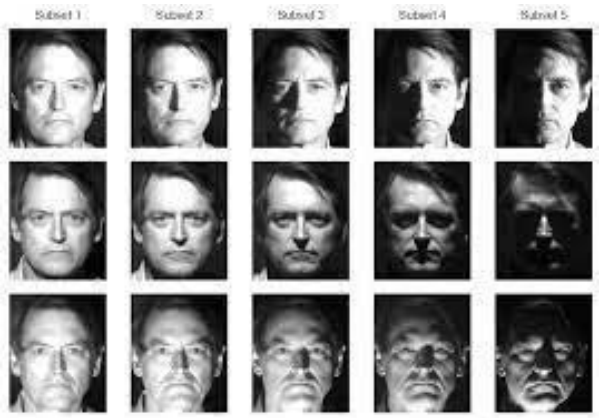

Figure 11: Output that shows the detection of Facial Recognition

Loading video DSA gfg intro actual video getting sub clip from it and showing a final clip. Here is the output; these are the frames of an abnormal video that detect abnormal activities by displaying abnormal on top with red color.

VII. CONCLUSION

The research endeavor will concentrate on the design and development of a video surveillance system aimed at addressing security challenges, particularly in the banking sector, thereby mitigating the occurrence of anomalous events. This system is engineered to detect unusual activities by analyzing video frames provided as input to the model, subsequently triggering an alarm. Upon successful implementation, the system could be deployed in various settings, including banks, residential security systems, museums, and public areas during nighttime. When an unknown or unrecognized person is identified, motion is monitored within the live video stream, facilitated by dual-axis pan-tilt servos that allow the camera to pursue the individual. This methodology illustrates the efficacy of adaptive automatic facial recognition in addressing security challenges. In the future, the system is anticipated to integrate smartphone alerts and coordinate video

Current Journal of Engineering and Science Research
Volume 1, Issue 1, pp 39-46, November 2024
ISSN (online): 3049-1738
DOI: https://doi.org/10.47001/CJESR/2024.101005
Available Online at https://www.cjesr.com

documentation of suspicious behaviors from cloud storage.

REFERENCES

[1] J. Hong Yoon, C.-R. Lee, M.H. Yang, and K.-J. Yoon. "Online multi-object tracking via structural constraint event aggregation", in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, 2016, pp. 1392–1400.

[2] W. Zhang, H. Zhou, S. Sun, Z. Wang, J. Shi, and C.C. Loy, "Robust multi-modality multi-object tracking, in: Proc. IEEE Int. Conf. Comput. Vis., Seoul, Korea, 2019, pp. 2365–2374.

[3] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs", arXiv preprint arXiv:1606.00915, 2016.

[4] Y. Guo, Y. Liu, T. Georgiou and M. S. Lew. "A review of semantic segmentation using deep neural networks. International Journal of Multimedia Information Retrieval", Vol. 7 pp. 87-93, 2018.

[5] B. Mabrouk and E. Zagrouba, "Abnormal behaviour recognition for intelligent video surveillance systems: A review", Expert Systems with Applications, vol. 91, pp. 480-491, 2018.

[6] M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, "Deep learning approach combining sparse autoencoder with SVM for network intrusion detection". IEEE Access, vol. 6, pp. 52843-52856, 2018.

[7] N. Khan, A. Ullah, I. U. Haq, V. G. Menon, and S. W. Baik, "SD-Net: Understanding overcrowded scenes in realtime via an efficient dilated convolutional neural network", Journal of Real-Time Image Processing, vol. 18(5), pp. 1729-1743, 2021.

[8] M. T. Awad, S. M. Aldaw, S. M. Aldaw and B. A. r. Osman. "Video Security System for Intrusion Detection," 2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), Khartoum, Sudan, 2019, pp. 01-04.

[9] R. Socha and B. Kogut. "Urban Video Surveillance as a Tool to Improve Security in Public Spaces," Sustainability, vol. 12, pp. 1-12, 2020.

[10] K. N. Pentaiah and P. J. Ker. "Development of a Microcontroller-based Portable Surveillance System with User Alert Notification." Journal of Environmental Science and Technology, vol. 10, pp. 80-87, 2017.

[11] S. Tanwar, P. Patel, K. Patel, S. Tyagi, N. Kumar, and M. S. Obaidat. "An advanced Internet of Thing based Security Alert System for Smart Home." 2017 International Conference on Computer, Information and Telecommunication Systems (CITS), Dalian, 2017, pp. 25-29.

[12] Umadevi V Navalgund, Priyadharshini. K "Crime Intention Detection System Using Deep Learning" KLE Technological University Hubballi, India IEEE 2018.

[13] Ya Wang, Tianlong Bao, Chunhui Ding, Ming Zhu "Face Recognition in Real-world Surveillance Videos with Deep Learning Method" Department of Information and Technology University of Science and Technology of China IEEE 2017.

[14] H. Kim, R. Sakamoto, I. Kitahara, T. Toriyama and K. Kogure, "Robust Foreground Extraction Technique Using Background Subtraction with Multiple Thresholds", Opt. Eng., vol. 46, no. 9, (2007), pp. 097 004-1– 097 004-12.

[15] Adam, Amit, Ehud Rivlin, Ilan Shimshoni, and Daviv Reinitz. "Robust real-time unusual event detection using multiple fixed-location monitors." IEEE Transactions on pattern analysis and machine intelligence 30, no. 3 (2008): 555-560.

[16] Tran, Du, LubomirBourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. "Learning spatiotemporal features with 3d convolutional networks." In Proceedings of the IEEE international conference on computer vision, pp. 4489-4497. 2015.

[17] Genshan, Zhang, Wu Qiuhong, Geng Kai, and Han Guodong. "Video analysis system of intelligent surveillance based on Bayesian." In Proceedings of 2011 International Conference on Computer Science and Network Technology, vol. 2, pp. 1008-1011. IEEE, 2011.

[18] Yun-Xia Liu, Yang Yang, Aijun Shi, Peng Jigang, Liu Haowei. "Intelligent monitoring of indoor surveillance video based on deep learning" *Shandong province's electronic information products quality supervision and inspection Institute, Jinan, China IEEE 2019.

[19] Amit Adam, Ehud Rivlin, Ilan Shimshoni and David Reinitz, Robust Real-Time Unusual Event Detection Using Multiple Fixed-Location Monitors, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 30, NO. 3, MARCH 2008.

Current Journal of Engineering and Science Research
Volume 1, Issue 1, pp 39-46, November 2024
ISSN (online): 3049-1738
DOI: https://doi.org/10.47001/CJESR/2024.101005
Available Online at https://www.cjesr.com

[20] O. R. Vincent, O. Folorunso "A Descriptive Algorithm for Sobel Image Edge Detection" ,Proceedings of Informing Science & IT Education Conference (InSITE) 2009.

[21] Y. Chen, S. Yu, W. Sun and H. Li, "Objects Detecting Based on Adaptive Background Models and Multiple Cues", ISECS International Colloquium on Computing, Communication, Control, and Management, vol. 1, Issue 3-4, (2008), pp. 285 – 289.

[22] Srinivasan, Vignesh, et al. "Interpretable human action recognition in the compressed domain." 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017.

[23] Abhishek Dutta, Andrew Zisserman, "The VIA Annotation Software for Images, Audio and Video" Dept. of Engineering Science, University of Oxford.

[24] J. Dai, Y. Li, K. He, & J. Sun, "R-fcn: Object detection via region-based fully convolutional networks". In Advances in neural information processing systems, pp. 379–387, 2016.

[25] W. Luo, J. Xing, A. Milan, X. Zhang W. Liu, and T. K. Kim. "Multiple object tracking: A literature review". Artificial Intelligence vol. 293, pp. 1-32, 2021.

Citation of this Article:

Abdulrahman A Ibrahim, Noura Abdullah Kamal, Yassir Ibrahem Rehiayan, & Reshied Solaiman Saleh. (2024). Security Framework for Enhanced Surveillance Monitoring in Banking Systems with Behavioural Pattern Identification Using Machine Learning. *Current Journal of Engineering and Science Research*. 1(1), 39-46. Article DOI: https://doi.org/10.47001/CJESR/2024.101005

***** End of the Article *****