Advanced Sciences and Technology Journal

ASTJ vol. 1 (2024) 1003
10.21608/astj.2024.297016.1003
https://astj.journals.ekb.eg/

Intelligent Campus Community Monitoring and Tracking System

Nour Elhoda, Ezz-Eldin Sadaty, Mahmoud Kassem, Seif Ehab, Omer Elhanafy

Software Engineering Department, Faculty of Engineering and Technology, Egyptian Chinese University, Cairo, Egypt

ARTICLEINFO

Article history:
Received 11 June 2024
Revised 23 June 2024
Accepted 30 October 2024
Available online 5 December 2024

Handling Editor:
Prof. Dr. Dongran Song

Keywords:
Campus Safety
Facial Recognition
Real-time Monitoring
Data Security
Scalable Design
.

ABSTRACT

The Intelligent Campus Community Monitoring and Tracking System represents a significant leap forward in campus safety technology featuring a sophisticated integration of state-of-the-art facial recognition technology and a network of strategically placed cameras. This system leverages advanced algorithms, specifically Multi-task Cascaded Convolutional Networks (MTCNN) and Inception Resnet V1, for high-accuracy face detection and recognition. These algorithms identify individuals even in highly populated and dynamic campus environments, ensuring the system's efficacy in real-time security monitoring and threat detection. This robust system is meticulously designed to cater to the needs of its users—both administrative personnel and campus community members. The interface facilitates ease of navigation and operation, ensuring that all users, irrespective of their technical capabilities, can utilize the system efficiently. Privacy and data security are essential in designing and implementing this surveillance system. Given the use of sensitive biometric data, strict protocols are enforced to secure personal information against unauthorized access and breaches. The system's architecture is built to ensure that data integrity and confidentiality are always maintained, thus supporting the trust and confidence of the campus community. The system's architecture is not only robust but also highly scalable, designed to accommodate the growth and technological advancements of educational institutions. On a broader scale, the deployment of the Intelligent Campus Community Monitoring and Tracking System profoundly enhances the overall campus atmosphere by supporting safety and security.

1. Introduction:

The Intelligent Campus Community Monitoring and Tracking System marks a pivotal advancement in campus safety technology, integrating cutting-edge facial recognition technology with a strategically deployed network of surveillance cameras [1-6]. Powered by sophisticated algorithms like Multi-task Cascaded Convolutional Networks (MTCNN) and InceptionResnetV1, the system ensures highly accurate face detection and recognition even in dynamic campus environments [7-14]. This capability enables real-time security monitoring and threat detection, catering to the diverse needs of both administrative personnel and campus community members.

User experience lies at the heart of the system's design, with an intuitive interface facilitating seamless navigation and operation for users of all technical proficiencies. Whether for routine surveillance or addressing urgent security concerns, the system delivers swift and reliable functionality, empowering users to efficiently utilize its capabilities.

Central to the system's design is a steadfast commitment to privacy and data security. Stringent protocols safeguard sensitive biometric data against unauthorized access and breaches, upholding the highest ethical standards in surveillance activities [12-15]. Architecture always ensures the integrity and confidentiality of data, fostering trust and confidence within the campus community.

Furthermore, the system's scalability is paramount, designed to accommodate the evolving needs and technological advancements of educational institutions. Its seamless integration with existing security infrastructures minimizes disruption, while its adaptability allows for future expansion and incorporation of emerging security technologies without compromising performance.

On a broader scale, the deployment of the Intelligent Campus Community Monitoring and Tracking System revolutionizes campus security, fostering a safer and more conducive environment for academic and professional pursuits. Its real-time monitoring capabilities enable swift responses to emergencies, mitigating potential risks and enhancing overall safety [8]. This heightened security environment promotes productivity and alleviates stress among students and staff, cultivating a more enriching academic experience.

1.1. Motivation

The motivation behind the Intelligent Campus Community Monitoring and Tracking System is to enhance safety and security in campus environments [1]. Traditional security measures often fall short, so our system uses advanced surveillance technology, including facial recognition, to offer real-time tracking capabilities. This provides a reliable and efficient solution for monitoring individuals in public spaces like campuses, shopping malls, universities, and parks [2-12].

1.2. Problem Definition

The core problem that the Intelligent Campus Community Monitoring and Tracking System aims to address is the difficulty of reliably tracking and ensuring the safety of individuals in large campus settings. These environments are often densely populated and expensive, making it challenging to keep an eye on everyone, especially in

situations where people can easily get lost. Traditional methods for monitoring such spaces are typically inadequate, as they cannot provide the real-time tracking and detailed oversight necessary to maintain consistent safety [8].

1.3. Objectives

1.3.1 Enhanced Safety for Campus Community:

Implement advanced real-time monitoring technology to ensure the safety and security of all campus community members, including students, staff, and their relatives.

1.3.2 User-Centric System Design:

Create a user-friendly interface for easy tracking and monitoring by both administrators and individuals, focusing on intuitive use and accessibility for all.

1.3.3 streamlined and Automated Surveillance:

Combine automation with comprehensive data management to reduce manual effort, increase efficiency, and support informed decision-making.

1.3.4 Set New Standards in Technological Innovation:

Lead in advancing campus monitoring systems, balancing technological sophistication with the needs and well-being of the campus community.

1.3.5 Efficient and Reliable Searching for Relatives:

The system stands out with its efficient and reliable approach to locating relatives in crowded, populated areas [13].

2. Case Study

2.1 For Admin

2.1.1 Logging In:

A security person arrives at his workstation and logs into the security system using his credentials [6].

2.1.2 Accessing the System Dashboard:

Once logged in, the system moves to the System, where security person views the main dashboard. This dashboard displays an overview of all security aspects he needs to monitor.

2.1.3 Surveillance Monitoring:

Security person clicks on the "Surveillance Monitoring" module to view live feeds from various security cameras installed throughout the facility. This allows him to keep an eye on real-time activities and ensure everything is under control.

2.1.4 Historical Movement Review:

After monitoring the live feeds, security person reviews the "Historical Movement" logs to analyze any unusual activities or patterns from the past 24 hours. This feature is crucial for investigating incidents after they have been flagged by the system or reported by the staff.

2.1.5 Accessing and Generating Reports:

Next, security person uses the "Access Report" function to generate security reports. These reports provide detailed information about incidents, staff interventions, and system alerts. He reviews the reports to ensure all incidents have been properly addressed and documented [10].

2.1.6 Searching Specific Data:

Security person uses the "Search" feature to look up specific events or access control logs related to a recent security alert. This function allows him to quickly find precise information without manually sifting through extensive logs.

2.1.7 Profile Management:

Before ending his monitoring session, the security person updates his profile and checks the settings in the "Profile Management" module. This includes updating his contact information and changing his password, ensuring his access credentials are secure and up to date [11].

3. Implementation

3.1. Face Detection and Recognition

This section focuses on implementing the initial phase of our project, which involves face detection and recognition. Using computer vision and machine learning techniques, we aim to capture and analyze video frames in real-time [14-15].

3.2. Camera Feed

facial recognition system that processes video streams in real-time to identify individuals. It uses the MTCNN model for detecting faces within video frames and the InceptionResnetV1, a neural network pre-trained on the 'vggface2' dataset, to extract facial embeddings. These embeddings represent unique facial features and are stored along with corresponding labels from the dataset.

The system is designed to handle multiple video feeds simultaneously by employing threading, where each thread is responsible for processing a separate video feed. As each frame is captured from the video, the script detects faces and generates embeddings. It then compares these embeddings against previously stored ones to

recognize known individuals. If a face's embedding closely matches one in the database, the system recognizes the individual; otherwise, it marks them as "Unknown."

Additionally, the script keeps track of the times everyone is detected in the video feed, recording their entry and exit times. This data is periodically sent to a backend server. To ensure robust performance across multiple video feeds, the script manages multiple threads efficiently, ensuring resources are properly allocated. This setup not only enhances security measures but also enables real-time monitoring and data management in various applications [4].

3.3. Flask application:

Provides a user-friendly interface for accessing and viewing live video feeds from multiple cameras with facial recognition capabilities, making it useful for security, surveillance, or monitoring applications [3].

3.3.1 Facial Recognition Camera Setup:

The application initializes and manages connections to multiple cameras for capturing video feeds. Each camera is associated with its own thread for processing, allowing for simultaneous video processing from multiple cameras.

3.3.2 Video Streaming and Facial Recognition:

The application captures frames from each camera and processes them using a facial recognition algorithm. Detected faces are recognized and annotated with their names in real-time on the video feed.

The processed video frames are streamed to a web interface, enabling users to view live video feeds from different cameras.

3.3.3 Web Interface Features:

The web interface provides a main page that lists all available camera feeds.

Users can access individual camera feeds by selecting a camera from the list, allowing them to view the live video stream with facial recognition overlays.

3.3.4 API Endpoint for Camera Listing:

The application also provides an API endpoint that lists all active cameras, allowing other applications or services to discover and access the camera feeds programmatically.

3.3.5 Clean Resource Management:

The application ensures proper resource management by releasing camera connections and stopping processing threads.

4. Architecture

The architecture diagram describes a layered structure for a campus monitoring and tracking system. Here's a summary of its components:

Hardware Layer: Contains cameras essential for capturing video feeds.

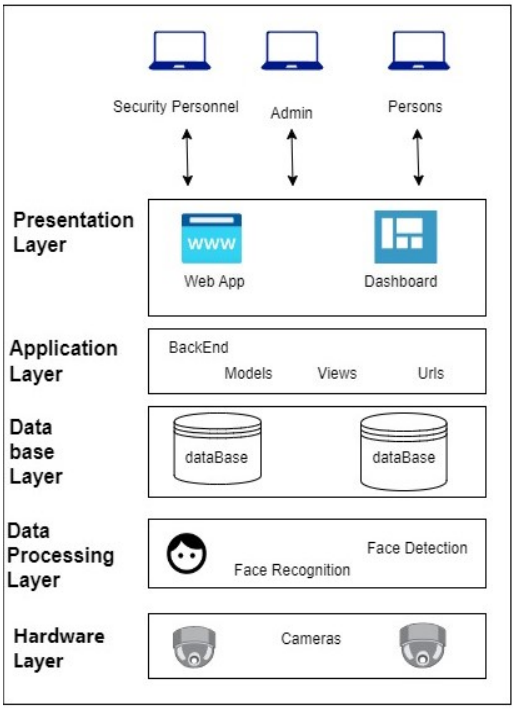
Data Processing Layer: Processes video inputs for face detection and recognition.

Database Layer: Manage database to store critical data like person and movement logs.

Application Layer: Comprises back-end components (Models, Views, URLs) that handle the system's business logic and data routing.

Presentation Layer: Features a Web App for general monitoring and a Dashboard for administrative tasks.

User Interaction: Involves three user types—Security Personnel, Admin, and Persons—each interacting with the system through different interfaces for surveillance, management, and tracking.



System Architecture

5. Future improvements

Enhanced AI Algorithms: Invest in research and development to improve the accuracy and efficiency of facial recognition algorithms. By refining these algorithms, the system can more accurately identify individuals even in crowded and dynamic environments, reducing false positives and enhancing overall tracking capabilities.

Predictive Analytics: Implement predictive analytics algorithms to anticipate and prevent security incidents before they occur. By analyzing patterns in movement and behavior, the system can identify potential risks and alert security personnel proactively, thereby minimizing response time and enhancing overall campus safety.

Accessing and Generating Reports: Admin uses the "Access Report" function to generate security reports. These reports provide detailed information about incidents, staff interventions, and system alerts. He reviews the reports to ensure all incidents have been properly addressed and documented.

Community Feedback Mechanisms: Implement feedback mechanisms within the system to gather input from the campus community regarding their experiences and suggestions for improvement. By incorporating user feedback, the system can continuously evolve to better meet the needs and preferences of its users while fostering a sense of community engagement and ownership.

Scalable Camera Deployment: Conduct a thorough site survey to identify critical areas that currently lack sufficient surveillance coverage. Gradually increase the number of cameras, focusing first on high-traffic areas, entry/exit points, and secluded zones that are potential risk spots. Use a mix of fixed and pan-tilt-zoom (PTZ) cameras to maximize coverage and functionality.

Advanced Camera Technologies: Incorporate cameras with higher resolutions and better low-light performance to enhance image quality, which is crucial for accurate facial recognition. Consider the use of thermal imaging cameras for improved detection in various environmental conditions and to add another layer of data for anomaly detection.

Maintenance and Continuous Improvement: Plan for regular maintenance checks and updates to the camera systems to ensure they remain in optimal condition and incorporate the latest security technologies. Gather feedback from system users and security staff to continually assess and improve the camera network and its integration with the overall monitoring system.

Video Stream Recording: By enabling the system to record video streams, continuous monitoring of campus activities over time becomes possible. This feature aids in conducting trend analysis and pattern detection, providing valuable insights into long-term campus dynamics and security trends.

Mobile Application Development: Creating a mobile application companion for the system can enable users to access real-time tracking of their loved ones on their smartphones, providing convenience and peace of mind even when they are on the move

6. Conclusion

In essence, this system not only transforms campus security measures but also sets a new standard for how educational institutions approach safety in densely populated environments. Through its advanced technological framework, commitment to user accessibility, stringent data protection measures, and scalable design, the Intelligent Campus Community Monitoring and Tracking System promises a secure, efficient, and adaptable safety solution, heralding a new era of campus security management

References

[1] M. Abadi, et.al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org

[2] Prof. P Y Kumbhar1 , Mohammad Attaullah2 , Shubham Dhere3, Shivkumar Hipparagi4 ,"Real Time Face Detection and Tracking Using OpenCV, 2017

[3] Y. Kim, W. Park, and J. Shin, “Broadface: Looking at tens of thousands of people at once for face recognition,” in ECCV, 2020

[4] Y. Shi, X. Yu, K. Sohn, M. Chandraker, and A. K. Jain, “Towards universal representation learning for deep face recognition,” in CVPR, 2020

[5] Peng Lu, Baoye Song , Lin Xu (2021) Human face recognition based on convolutional neural network and augmented dataset

[6] Prof. P Y Kumbhar1 , Mohammad Attaullah2 , Shubham Dhere3, Shivkumar Hipparagi4 ,"Real Time Face Detection and Tracking Using OpenCV, 2017

[7] Jose Edwin; M. Greeshma ; Mithun T. P. Haridas ; M.H. Supriya, “Face Recognition based Surveillance System Using FaceNet and MTCNN on Jetson TX2” 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), 06 June 2019

[8] Zhiyi Cheng, Xiatian Zhu, Shaogang Gong ,"Surveillance Face Recognition" ,2018

[9] Jun Wang, Yinglu Liu, Yibo Hu, Hailin Shi,"FaceX-Zoo: A PyTorch Toolbox for Face Recognition",2021

[10] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments.In Workshop on Faces in ’Real-Life’ Images: Detection, Alignment, and Recognition, 2008.

[11] Yi-Tsung Chien; Yea-Shuan Huang ; Sheng-Wen Jeng ; Yao-Hong Tasi ; Hung-Xin Zhao “A real-time security surveillance system for personal authentication” IEEE 37th Annual 2003 International Carnahan Conference onSecurity Technology, 2003. Proceedings

[12] B. Kranthikiran and P. Pulicherla, "Face Detection and Recognition for use in Campus Surveillance", International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 3, 2020.

[13] Zhang Jian and Song Wan-Juan, "Face detection for security surveillance system", 2010 5th International Conference on Computer Science & Education, pp. 1735-1738, 2010.

[14] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, Yu Qiao ,"Joint Face Detection and Alignment using MTCNN",2016

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. arXiv preprint arXiv:1512.00567, 2015