CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

CAPSTONE PROJECT - 1

CAMPUS SURVEILLANCE AND ANALYSIS SYSTEM

SYNOPSIS

By

Anuja Alice Thomas, Dennis P James, Dijin Dominic

(2348013, 2348025, 2348027)

A capstone project synopsis submitted in partial fulfilment of the requirements for the award of degree of Master of Science (Data Science) of CHRIST (Deemed to be University)

May – 2024

4MDS

CAPSTONE PROJECT – 1

SYNOPSIS

Project Title:

Campus Surveillance and Analysis System

Introduction:

The proposed web project aims to enhance campus security and operational efficiency by analysing live video feeds to monitor the entry and exit of individuals (students). The system will identify whether individuals are wearing ID cards, categorize them based on tag colors as student, faculty or hosteler, and potentially enforce dress codes with the latest Machine Learning techniques. This technology-driven approach ensures a safer and more organized campus environment.

Objectives:

The following are the key objectives of the project:

1. **Real-Time Surveillance:** Monitor live video feeds to track individuals entering and leaving the campus.
2. **ID Card Detection:** Identify individuals who are not wearing their ID cards.
3. **Categorization by Tag Color:** Differentiate between students, faculty, and hostelers based on the color of their ID tags.
4. **Entry/Exit Analysis:** Provide detailed reports on the number of individuals entering and exiting the campus.
5. **Hostel Intrusion Detection:** Detect and alert unauthorized intrusions in hostel premises based on tag color.
6. **Gender Identification:** Optionally identify individuals as male or female to enhance security measures.

Components of the Project:

1. **Video Surveillance Hardware:** Cameras strategically placed at entry and exit points of the campus and hostels.
2. **Video Processing Software:** Real-time video analysis using machine learning and computer vision techniques to detect and identify individuals.
3. **Database:** Storage system for recording data related to entries, exits, ID card status, and other relevant metrics.
4. **Web Interface:** User-friendly web portal for accessing reports, live feeds, and alerts.

5. **Machine Learning Models:** Models trained to recognize ID cards, tag colors, dress codes, and gender.
6. **Notification System:** Mechanism for sending alerts and notifications to relevant authorities in case of policy violations or security breaches.

Technology Stack:

1. **Frontend Framework:** React.js for building a dynamic and responsive user interface.
2. **Backend Server:** Node.js with Express.js for server-side logic and API development, or Flask for a Python-based backend.
3. **Machine Learning:** TensorFlow, Keras, or PyTorch for developing and deploying machine learning models.
4. **Computer Vision:** OpenCV for image processing and real-time video analysis along with YOLO (You Only Look Once).
5. **Generative AI techniques:** To build a potential generated dataset for the id card detection.

Agile Tools and Methodology:

To efficiently manage the project within the tight academic schedule, we will utilize Agile methodology, specifically Scrum. This approach promotes iterative development, flexibility, and active collaboration. Below are the key Agile tools and methodologies we will use:

1. **Google Meet:** For tracking tasks, sprints, project progress, team communication, and quick updates.
2. **Git and GitHub:** For version control and collaborative development.
3. **Google Drive or Confluence:** For documentation and knowledge sharing.

These tools will help streamline the development process, ensure transparency, and facilitate effective communication among team members.

Agile Model Timeline

Sprint 0: Initial Planning (Week 1: June 24 - June 30)

1. Requirements Gathering and Feasibility Studies: Understand the project requirements and evaluate its feasibility.
2. High-Level System Design and Planning: Design the overall architecture and plan the development phases.
3. Setting Up Development Environment and Agile Tools: Prepare the development environment and select Agile tools for project management.

Sprint 1: Setup and Basic Structure (Week 2: July 1 - July 7)

1. Frontend Setup: Initialize the React.js environment and create basic UI components.
2. Backend Setup: Set up the Node.js/Flask server and develop initial API endpoints.
3. Database Setup: Design the database schema and set up MongoDB/PostgreSQL.

Sprint 2: Initial Features Development (Week 3: July 8 - July 14)

1. Video Processing Algorithms: Develop algorithms for ID card detection and gender classification using OpenCV, Yolo, and machine learning models.
2. Basic API Integration: Integrate video processing features with the backend API.

Sprint 3: Advanced Features Development (Week 4: July 15 - July 21)

1. Role Classification: Implement classification of individuals based on tag colors to differentiate students, faculty, and hostelers.
2. Vehicle Detection: Develop and integrate the vehicle detection module.

Sprint 4: Frontend Enhancements (Week 5: July 22 - July 28)

1. Dashboard Development: Build the main dashboard for displaying live camera feeds and data visualizations.

Sprint 5: Testing and Integration (Week 6: July 29 - August 4)

2. Unit and Integration Testing: Conduct thorough testing of individual components and their integration.
3. System Validation: Validate the system for accuracy, reliability, and performance.

Sprint 6: Final Adjustments and Preparation (Week 7: August 5 - August 11)

1. Pilot Testing: Deploy the system locally or on a cloud platform for pilot testing on campus.
2. Feedback Gathering: Collect feedback from pilot testing to identify any issues or improvements.
3. Final Tweaks: Make necessary adjustments based on the feedback.
4. Presentation Preparation: Prepare presentation materials, including slides, demo scripts, and documentation.

Sprint 7: Final Presentation and Optional Deployment (Week 8: August 12 - August 18)

1. Project Presentation: Present the project to stakeholders or academic supervisors, demonstrating its features and functionality.
2. Optional Deployment: If time permits, deploy the system on a cloud platform for full-scale use; otherwise, demonstrate using local servers.
3. Documentation and Final Report: Compile all project documentation and prepare the final report for submission.

Implementation Plan:

1. **Requirement Analysis:** Gather detailed requirements and perform feasibility studies.
2. **System Design:** Design the architecture of the system, including hardware setup and software modules.

3. **Development:**
    - Develop video processing algorithms for ID card detection and categorization.
    - Create a web interface for data visualization and reporting.
    - Set up databases and integrate with video processing software.
4. **Testing:** Rigorously test the system for accuracy, reliability, and performance.
5. **Deployment:** Deploy the system on the campus, starting with pilot testing and then full-scale implementation.
6. **Maintenance and Updates:** Regularly update the system to incorporate new features and address any issues.

Conclusion:

The Campus Surveillance and Analysis System is designed to leverage advanced technology to improve campus security and operational efficiency. By monitoring and analysing live video feeds, the system will ensure that only authorized individuals are present on campus and that they adhere to campus policies. The future scope of the project promises even greater capabilities, making it a robust solution for campus management and security.

Keywords:

Real-Time Surveillance | ID Card Detection | Tag Color Categorization | Campus Security | Machine Learning | Video Processing | Dress Code Enforcement | Attendance Tracking

Future Scope:

The project is aimed to later be extended in order to incorporate the following objectives:

1. **Dress Code Enforcement:** Analyse if individuals are adhering to a predefined dress code (e.g., formal wear).
2. **Advanced Categorization:** Further distinguish between different groups within the campus, such as faculty and students, using specific tag colors.
3. **Behavioural Analysis:** Monitor and analyse the behaviour of individuals for unusual activities or potential threats.
4. **Attendance Tracking:** Integrate with the campus attendance system to automatically record the presence of students and faculty.
5. **Automated Alerts:** Set up automatic alerts for unauthorized access, missing ID cards, or dress code violations.
6. **Data Integration:** Integrate with other campus management systems for a comprehensive view of campus activities and security.