CROSS CHECK CV

21IT411-ENGINEERING EXPLORATION IV

Submitted by

KANIKA S	**(714023243061)**
MALLEESHWAR V	**(714023243074)**
MOUCIHASRI T N	**(714023243084)**
PRIYADHARSHINI S	**(714023243103)**

In partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

SRI SHAKTHI

INSTITUTE OF ENGINEERING AND TECHNOLOGY

COIMBATORE - 62

An Autonomous Institute,

Accredited by NAAC with "A" Grade

ANNA UNIVERSITY: CHENNAI 600 025

MAY – 2025

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"CROSS CHECK CV"** is the bonafide work of **"KANIKA S (714023243061), MALLEESHWAR V (714023243074), MOUCIHASRI T N (714023243084), PRIYADHARSHINI S (714023243103)"**Who carried out the work in 21IT411-Engineering Exploration IV under my supervision.

SIGNATURE	**SIGNATURE**
Ms. SANTHOSHI P	Dr. N. K. SAKTHIVEL
ASSISTANT PROFESSOR	**HEAD OF THE DEPARTMENT**
Department of Artificial Intelligence	Department of Artificial Intelligence
and Data Science,	and Data Science,
Sri Shakthi Institute of	Sri Shakthi Institute of
Engineering and Technology,	Engineering and Technology,
Coimbatore -641062.	Coimbatore- 641062.

Submitted for the project work viva-voce Examination held on ....

EXAMINER

ACKNOWLEDGEMENT

First and foremost, I would like to thank God Almighty for giving me the strength. Without his blessings this achievement would not have been possible.

We express our deepest gratitude to our Chairman **Dr.S.Thangavelu** for his continuous encouragement and support throughout our course of study.

We are thankful to our Secretary **Er.T.Dheepan** for his unwavering support during the entire course of this project work.

We are also thankful to our Joint Secretary **Mr.T.Sheelan** for his support during the entire course of this project work.

We are highly indebted to Principal **Dr.N.K.Sakthivel** for his support during the tenure of the project.

I would like to acknowledge and express deep gratitude to the Head of the Department of Artificial Intelligence and Data Science, **Dr.N.K.Sakthivel** (HOD/Artificial Intelligence and Data Science) for providing us with essential facilities and a conducive environment for our project work.

It's a great pleasure to thank our Project Guide, **Ms.P.Santhoshi** (AP/Artificial Intelligence and Data Science) for her valuable technical suggestions and continuous guidance throughout this project work.

I am also thankful to our Project Coordinator, **Ms.P.Santhoshi** (AP/Artificial Intelligence and Data Science) for their support, encouragement, and for providing us with the necessary facilities to carry out our project effectively.

We solemnly extend our thanks to all the teachers and non-teaching staff of our department, family and friends for their valuable support.

KANIKA S

MOUCIHASRI T N

PRIYADHARSHINI S

MALLEESHWAR V

ABSTRACT

Abstract Cross Check CV is an intelligent, real-time computer vision system designed to ensure compliance with identity and safety protocols by detecting the simultaneous presence of both ID cards and appropriate footwear in visual input streams. The platform leverages advanced deep learning techniques, including YOLOv8 object detection and custom Convolutional Neural Networks (CNNs), to analyze image data with high precision. The system has applications in workplace security, smart surveillance, and automated access control, particularly in industrial, academic, and research environments where dress code enforcement is critical.System Overview At the technological core of Cross Check CV is a YOLOv8 detection model, fine-tuned on a domain-specific dataset curated using Roboflow and annotated via OpenCV tools. The model identifies and localizes ID cards (lanyards, badges) and footwear (shoes, sandals, barefoot) in real-time camera feeds. Complementing this, a dedicated CNN model classifies footwear into compliant or non-compliant categories, adding a layer of semantic understanding essential for rule-based access control.Multi-Class Co-Detection Logic What sets Cross Check CV apart is its co-detection logic that mandates the concurrent detection of both ID card and compliant footwear for successful validation. This dual-check mechanism minimizes false positives and enforces more accurate real-world compliance monitoring. Such logic is particularly vital in sensitive areas such as labs or industrial floors where safety protocols are mandatory.Dataset and Training The system utilizes a custom-labeled dataset containing diverse scenarios with variations in lighting, posture, and occlusion. YOLOv8 was trained using PyTorch with optimization techniques like mosaic augmentation, learning rate scheduling, and confidence threshold tuning to maximize mAP (mean Average Precision). CNNs for footwear compliance were trained using a supervised approach with categorical cross-entropy loss and image normalization preprocessing.Deployment Architecture Cross Check CV runs on a lightweight backend developed using Flask or FastAPI, exposing RESTful APIs for integration with front-end interfaces or third-party systems. The interface, built using HTML, CSS, and JavaScript, offers a clean, real-time feedback panel displaying detection status, object labels, and confidence scores. The system supports both webcam-based streaming and single image uploads.Privacy and Data Management In alignment with responsible AI principles, the system stores only non-sensitive metadata: detection results, timestamps, and environmental parameters. No raw images are retained, ensuring user privacy.

Keywords: YOLOv8,Convolutional Neural Networks (CNNs),ID Card Detection ,Shoe Detection, Co-Detection System ,Computer Vision, Roboflow, OpenCV ,PyTorch,Real-Time Object Detection Access Control Automation,Smart Surveillance,Image Annotation ,Flask / FastAPI ,Edge Deployment

TABLE OF CONTENT

CHAPTER 1

INTRODUCTION

In modern security systems, ensuring the authenticity and compliance of individuals entering secure zones is a critical requirement. Traditional surveillance methods, which rely heavily on manual monitoring and human verification, are often prone to errors, inefficiency, and delays. Particularly in environments such as examination halls, labs, and restricted workplaces, there is a growing need for automated systems that can validate both identity and adherence to dress codes—such as wearing shoes—without manual intervention.

Cross Check CV addresses this gap by introducing an AI-powered co-detection system that performs real-time validation of ID card possession and appropriate footwear using advanced computer vision techniques. The system leverages **YOLOv8 (You Only Look Once, Version 8)**, a state-of-the-art object detection framework, in conjunction with **Convolutional Neural Networks (CNNs)** to accurately identify ID cards and shoes from camera feeds. By integrating these components into a unified pipeline, Cross Check CV automates the verification process, enhancing both security and operational efficiency.

The system is trained on custom-labeled datasets using tools like **Roboflow** and utilizes **OpenCV** for video stream processing. It is designed for **real-time deployment**, offering immediate alerts if required items (e.g., ID card or shoes) are missing. Furthermore, the backend infrastructure—built using frameworks like **Flask** or **FastAPI**—ensures rapid response and seamless integration into existing surveillance setups.

Cross Check CV represents a practical application of AI and computer vision in the realm of smart surveillance and automated access control. By reducing human dependency, minimizing errors, and increasing throughput, the system offers a scalable and reliable solution to modern security challenges in controlled environments.

1.1 OBJECTIVE

The objective of this project is to design and develop Cross Check CV, a real-time computer vision system that automates the detection of ID cards and footwear (shoes) using YOLOv8 object detection and deep learning techniques. The goal is to ensure that individuals entering restricted or regulated environments—such as examination halls, laboratories, or industrial zones—are in compliance with entry protocols, specifically by carrying a visible ID card and wearing appropriate footwear.The system uses live camera input to identify the presence of both objects simultaneously, eliminating the need for manual checking. It is built with a focus on accuracy, speed, and real-time performance, leveraging tools like OpenCV, PyTorch, and Roboflow for image processing, model training, and dataset management.By implementing this automated co-detection mechanism, the project aims to Improve security and compliance monitoring.Reduce human dependency in surveillance tasks.Minimize errors and delays in access control processes.Provide a scalable, efficient solution that can be easily deployed in institutional or workplace settings.

1.2 PROBLEM STATEMENT

Despite advancements in security technologies, many institutions still rely on manual checks for verifying compliance with access protocols, such as ensuring individuals are carrying an ID card and wearing appropriate footwear. These traditional methods are time-consuming, prone to human error, and inefficient, particularly in high-traffic environments like examination centers, laboratories, or corporate buildings. Manual verification not only slows down access processes but also creates inconsistency, especially during busy periods or when human fatigue sets in.Most existing automated systems focus on single-object detection or use basic models for identity verification, leaving out essential compliance checks like footwear. There is a lack of co-detection systems capable of simultaneously verifying both ID cards and footwear within the same solution, which is crucial for security in regulated environments. Additionally, many systems are not designed for real-time processing, limiting their practical use.Existing solutions also suffer from scalability issues, high costs, and complex infrastructure requirements. Many rely on cloud-based systems or require specialized hardware, making them difficult to deploy in environments with limited resources or privacy concerns. Furthermore, current systems often lack the precision and real-time performance needed to operate effectively under varying conditions such as changing lighting or angles.Cross Check CV addresses these gaps by providing an AI-powered real-time verification system that uses YOLOv8 object detection to simultaneously verify the presence of both ID cards and shoes

1.3 EXISTING MODEL

Current security and access control systems largely depend on manual verification methods or simple rule-based mechanisms for ensuring compliance with entry protocols. Most traditional methods rely on security personnel or ID scanners to check if individuals possess valid ID cards, and separate systems or staff members are responsible for monitoring footwear compliance. These processes are inherently time-consuming, prone to human error, and often inefficient, especially in environments with high foot traffic, such as exams, labs, or restricted corporate zones.While some automated solutions incorporate object detection for ID cards, they typically focus on a single criterion and lack the ability to verify multiple conditions simultaneously. These systems are often constrained by their reliance on fixed datasets, which limits their accuracy and generalization across different lighting conditions, angles, and user orientations. Furthermore, many of these solutions operate in isolation, without integrating real-time video streams or leveraging advanced machine learning models to detect and classify objects more effectively.Additionally, existing models often require expensive hardware, cloud-based solutions, or specialized infrastructure, making them impractical for environments with limited resources.

1.4 PROPOSED MODEL

Cross Check CV introduces an innovative AI-powered solution for real-time compliance verification in security-sensitive environments. The system employs a custom-trained YOLOv8 object detection model to simultaneously detect ID cards and shoes from live webcam footage, enabling automated monitoring with minimal human intervention. At its core, the model utilizes a deep learning-based object detection approach to ensure accuracy in classifying these two objects, regardless of lighting, angle, or occlusion. By integrating transfer learning, the model is optimized to detect both ID cards and shoes in a variety of real-world conditions, even with limited or diverse datasets.Unlike traditional security systems, Cross Check CV operates entirely in real-time, using OpenCV to capture and process video frames efficiently. The model ensures that compliance checks are completed instantly, without unnecessary delays, thus improving security protocols while reducing bottlenecks in access control. Furthermore, the system operates locally, without the need for external cloud-based services, ensuring data privacy by not storing or transmitting sensitive images. This feature is particularly important in environments where confidentiality and data protection are paramount.

CHAPTER 2

LITERATURE SURVEY

Traditional methods of object detection in security and compliance monitoring, such as manual inspections or single-object detection systems, often lead to inefficiencies and human errors. Manual checks can be time-consuming and prone to misidentification, while conventional detection systems may only focus on a single object, such as ID cards or shoes, neglecting scenarios where both need to be identified simultaneously. Furthermore, these systems are often limited by their inability to handle variations in object appearance, lighting conditions, or occlusion. "Cross Check CV" overcomes these limitations by leveraging YOLOv8, a state-of-the-art real-time object detection model, for the simultaneous and accurate detection of both ID cards and shoes. This dual-object detection system enhances efficiency and reduces human error, making it ideal for use in automated security, compliance monitoring, and access control.

As AI-driven systems like "Cross Check CV" process sensitive personal data, including images of ID cards and personal footwear, privacy and ethical considerations become paramount. The risk of data misuse, such as identity theft or discrimination, highlights the need for strict data protection protocols. Regulations like the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA) set guidelines for safeguarding personal data, requiring informed consent and robust data handling procedures. In "Cross Check CV," ethical design includes measures to ensure that data is securely processed, with user consent being clearly obtained and the right to data deletion respected. Moreover, models are trained on diverse datasets to prevent potential biases in detection, ensuring fairness and accuracy for all users.

In complex environments where multiple objects need to be detected simultaneously, a one-size-fits-all approach often proves ineffective. "Cross Check CV" uses YOLOv8's advanced multi-object detection capabilities, which are further enhanced by the integration of a hybrid recommendation system. This system combines content-based filtering (based on the characteristics of the ID cards and shoes), collaborative filtering (based on user behaviors and context), and knowledge-based rules (such as rules for detecting specific shoe types or card types). These hybrid techniques improve the model's accuracy and adaptability, ensuring that the system provides reliable and context-aware results. The ability to adjust dynamically to various scenarios makes "Cross Check CV" a versatile and powerful solution for real-time security and compliance monitoring.

The user interface (UI) design of "Cross Check CV" is crucial for ensuring that the system is not only functional but also user-friendly. For an application like "Cross Check CV," which involves real-time detection and interpretation of sensitive information, an intuitive and aesthetically pleasing UI is essential. The design must strike a balance between efficiency and user experience, ensuring that the system is easy to navigate while maintaining a professional appearance. Trends such as dark mode, glassmorphism, and responsive design principles help create a visually appealing interface that works seamlessly across both mobile and desktop platforms. More than just aesthetic appeal, the UI needs to communicate complex information, such as detection confidence scores, in a clear and understandable way. Features like multilingual support and accessibility compliance make the platform usable for a global audience, ensuring that it is inclusive and easy to operate by individuals from various backgrounds. A thoughtfully designed UI enhances not only user satisfaction but also system effectiveness, empowering users to make informed decisions based on real-time data.

2.1 APPLICATION OF COMPUTER VISION IN ACCESS CONTROL

Computer vision has revolutionized access control by replacing manual verification methods with intelligent, automated systems. Traditional approaches, such as checking ID cards or safety gear manually, are time-consuming and prone to human error. With the advancement of deep learning models like YOLOv8, real-time detection of key compliance elements—such as ID cards and safety shoes—has become highly accurate and efficient. These systems can monitor entry points and ensure that only individuals meeting specific requirements are allowed access. The Cross Check CV project leverages these capabilities to enhance security and safety compliance, particularly in workplaces where both identity and gear verification are essential.

2.2 OBJECT DETECTION USING YOLO ARCHITECTURE

Object detection using the YOLO (You Only Look Once) architecture has become a cornerstone in real-time computer vision tasks due to its speed and accuracy. Unlike traditional two-stage detectors, YOLO treats object detection as a single regression problem, predicting bounding boxes and class probabilities directly from full images in one evaluation. This makes YOLO highly efficient for real-time applications such as surveillance, autonomous driving, and access control. Variants like YOLOv8 further enhance performance with better feature extraction, improved object localization, and lightweight deployment capabilities. In the **Cross Check CV** system, YOLOv8 enables accurate co-detection of ID cards and safety shoes, ensuring fast, reliable access verification.

2.3 REAL-TIME SURVILLANCE SYSTEMS USING CNN

Cross Check CV is a real-time surveillance system utilizing YOLOv8 and Convolutional Neural Networks (CNNs) to detect and verify ID cards and shoes simultaneously. Designed for security and compliance monitoring, the system processes live video feeds to identify objects with high accuracy, even in challenging conditions such as occlusion and varying lighting. The system's real-time capabilities ensure quick responses, triggering alerts when specific objects are detected in restricted areas. Cross Check CV can be expanded to detect additional objects, making it a versatile tool for various surveillance needs. With secure data handling and an intuitive interface, it provides an efficient, scalable solution for modern security applications.

2.4 CO DETECTION IN SECURITY SYSTEMS

Co-Detection in Security Systems refers to the simultaneous detection of multiple objects or events within a given environment, enhancing the efficiency and accuracy of security monitoring. In modern surveillance systems, co-detection leverages advanced AI models like YOLOv8 and Convolutional Neural Networks (CNNs) to identify different objects—such as ID cards, bags, vehicles, or weapons—concurrently in real-time. This capability allows the system to process complex scenes and make informed decisions based on the context of the detected objects. For instance, a co-detection system in security could identify both an individual’s face and the items they are carrying, such as a backpack or an ID card, simultaneously, providing a more comprehensive analysis. The main benefits of co-detection in security systems include improved efficiency by detecting multiple objects at once, enhanced accuracy with reduced false positives, and context-aware monitoring that helps in identifying not only who is present but also what they are carrying or doing. This multi-object detection approach significantly strengthens security measures by offering a more thorough and real-time analysis of potentially suspicious activities.

2.5 DATASET-ANNOTATION AND ROBOFLOW TOOLS

In computer vision projects, dataset annotation plays a critical role in training models to accurately detect and classify objects. Annotation involves labeling data—such as images or video frames—so that the model can learn from these labeled examples during training. For "Cross Check CV: Co-Detection of ID Cards and Shoes Using YOLOv8", accurate and efficient annotation is essential for teaching the system to correctly identify and differentiate between ID cards, shoes, and other objects in a variety of conditions.The process of dataset annotation typically involves outlining or labeling the objects of interest within an image, often by creating bounding boxes around the objects. Each box is then labeled with a class name, such as "ID card" or "shoe," providing the model with the necessary ground truth. Annotating large datasets manually can be time-consuming and prone to human error, making it crucial to use efficient tools for automation and accuracy.Roboflow is a powerful tool that simplifies the process of dataset annotation and helps streamline the preparation of datasets for object detection tasks. It allows users to upload their raw image datasets, annotate them with minimal effort, and prepare the images in formats compatible with machine learning frameworks like YOLO, TensorFlow, or PyTorch. Roboflow provides a user-friendly interface for drawing bounding boxes around objects and assigning class labels, making the annotation process faster and more consistent.

In addition to annotation, Roboflow offers a suite of tools for dataset augmentation, which is especially important when working with deep learning models like YOLOv8. Data augmentation techniques, such as rotating, flipping, cropping, and altering the lighting conditions of images, can artificially expand the dataset and improve the robustness of the model. By exposing the model to a wider variety of scenarios, these techniques help the system generalize better, reducing overfitting and improving its performance in real-world applications.Once the dataset is annotated and augmented in Roboflow, the platform also enables easy export of the data in various formats suitable for training object detection models. Roboflow's integration with YOLOv8 allows for seamless data pipeline management, ensuring that images are pre-processed and ready for model training. The tool also provides functionality for versioning datasets, so users can track changes and updates as the dataset evolves throughout the project lifecycle.By using Roboflow for dataset annotation and augmentation, "Cross Check CV" ensures that its object detection models are trained on high-quality, diverse datasets, improving accuracy and reliability in real-time detection of ID cards and shoes.

2.6 LIMITATIONS OF MANUAL VERIFICATION IN ENTRY SYATEMS

Manual verification in entry systems, such as security checkpoints and access control, has several limitations that hinder its efficiency and accuracy. It is time-consuming, as security personnel must inspect each individual’s ID or access item one by one, causing delays, especially in high-traffic environments. Additionally, human error, such as oversight or fatigue, can lead to missed threats or false positives, compromising security. Manual systems are also limited in their ability to detect and analyze multiple objects simultaneously, meaning important details, like suspicious items, can be overlooked. Furthermore, human verification is subjective and inconsistent, varying from person to person, which leads to unreliable results. Scalability is another challenge, as manual systems struggle to efficiently handle large volumes of individuals. They also lack real-time monitoring, making it difficult to identify suspicious behaviors or patterns. Manual verification is labor-intensive, requiring a significant amount of human resources and increasing operational costs, while also being vulnerable to fraud, such as forged IDs. By automating detection and verification through AI-powered systems like "Cross Check CV", these limitations can be addressed, offering faster, more accurate, and scalable solutions for security systems..

2.7 PRIVACY AND ETHICAL CONSIDERATIONS IN HEALTH AI SYSTEMS

Automated surveillance systems raise significant privacy and ethical concerns, particularly regarding data collection, consent, and misuse. The continuous monitoring of individuals can infringe upon privacy rights and lead to discomfort if not properly managed. Issues like unauthorized access, bias in detection models, and the risk of discrimination based on race or gender further complicate the ethical landscape. Additionally, improper handling of sensitive data can lead to identity theft or breaches. To address these challenges, it’s essential to implement robust data protection measures, ensure transparency, and comply with regulations like GDPR to safeguard privacy and prevent misuse.

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 HARDWARE REQUIREMENTS

3.1.1 Image Acquisition Hardware

- **RGB Cameras**: High-resolution cameras are used for capturing clear images of ID cards and shoes in varying lighting conditions. These cameras support accurate detection through consistent visual data input.
- **Depth Sensors (Optional)**: For capturing 3D depth information to enhance accuracy in environments with occlusions or overlapping objects.

3.1.2 Processing Units

- **Edge Devices:**
  - *Examples:* NVIDIA Jetson Nano, Jetson Xavier, Raspberry Pi 4.
  - *Purpose:* Real-time inference and lightweight deployment in surveillance systems.
- **High-Performance Systems (for model training):**
  - *GPU:* NVIDIA RTX 3060/3080 or A100 for deep learning model training.
  - *CPU:* Intel Core i7/i9 or AMD Ryzen 7/9 for high-speed processing.
  - *RAM:* Minimum 16 GB; 32 GB preferred for smoother training and multitasking.
  - *Storage:* SSD with at least 512 GB for storing datasets, weights, and logs.

3.1.3 Communication Hardware

- **Wi-Fi / Ethernet**: Required for transmitting camera feeds and receiving detection results across the network.
- **4G/5G Modules**: Enable remote access and alerts in mobile or remote setups.
- **Bluetooth (Optional)**: For local device communication and pairing.

3.1.4 Power & Support

- **Uninterruptible Power Supply (UPS)**: Ensures continuous operation during power failures.
- **Cooling Systems**: Essential for thermal regulation in embedded systems and server hardware.
- **Protective Enclosures:** Used in outdoor environments to shield devices from weather or dust.

3.2 Software Requirements

Operating System

- **Windows 10/11:** User-friendly for development and GUI interaction.
- **Ubuntu 20.04 or later**: Preferred for Python-based model training, especially with YOLOv8 and CNNs.
- **macOS (limited GPU support)**: Suitable for development but not for GPU-intensive tasks.

Development Environment

- **Python Version**: 3.8 to 3.10 (recommended for compatibility with PyTorch, Roboflow, and YOLOv8).
- **IDE**: Visual Studio Code / Jupyter Notebook for code development and visualization.
- **Frameworks & Libraries**:
    - PyTorch, OpenCV, Roboflow SDK, YOLOv8 Ultralytics
    - NumPy, Pandas, Matplotlib for data processing and visualization

CHAPTER 4

PROPOSED METHODOLOGY

The **Cross Check CV** project proposes a comprehensive AI-powered surveillance system utilizing computer vision for real-time object detection, specifically focusing on ID cards and shoes. The system integrates modern software and hardware technologies for efficient performance and user accessibility.

On the **frontend**, the project utilizes HTML5, CSS3, and JavaScript to create a responsive user interface. CSS Flexbox and Grid are employed to ensure the application is adaptable across various screen sizes, supporting both mobile and desktop environments. The design includes minimalistic features like Lottie animations to provide dynamic and engaging feedback, such as detection status or confidence levels.

On the **backend**, the system is powered by Python with the Flask microframework, offering REST APIs to handle user requests and integrate external services. Flask-CORS ensures secure communication between the frontend and backend, and the asynchronous capabilities of FastAPI are also considered for scalability. OpenCV is used for real-time webcam input processing, and YOLOv8 (You Only Look Once) is employed to perform real-time object detection for ID cards and shoes. The deep learning models are built using TensorFlow and Keras, optimized for accurate object classification.

To enhance model performance, the datasets are preprocessed with libraries like Pandas and NumPy, ensuring smooth handling of images and metadata. The models are trained in Google Colab, leveraging GPU acceleration, with checkpoints saved on Google Drive. The evaluation metrics, such as accuracy and F1-score, are measured using Scikit-learn. For improved generalization, data augmentation techniques like rotation, flipping, and contrast adjustment are applied during training.

For **real-time data persistence**, **MongoDB Atlas** serves as the cloud-hosted NoSQL database, storing logs, confidence scores, and timestamps for each detection. PyMongo and MongoEngine are utilized to interface efficiently with MongoDB, enabling longitudinal tracking of detection trends. Each detection is tagged with metadata such as time and location, facilitating trend analysis and supporting system transparency.

The **System's performance** is further optimized with edge deployment options, including lightweight models running on devices like the Raspberry Pi 4 or Jetson Nano for localized processing. This is complemented by stable internet connectivity for API integration, ensuring real-time synchronization and communication with cloud services.

The **security architecture** follows best practices, using .env files for API key management and securing sensitive data. Future scalability is considered through modular design, allowing for the integration of additional features like multi-modal learning, authentication systems, and deployment on platforms such as AWS EC2 or Heroku.

This approach ensures that **Cross Check CV** provides an efficient, scalable, and secure solution for object detection in surveillance systems, enabling real-time analysis and verification in various use cases.

4.1 DATASET PREPARATION AND ANNOTATION

Dataset preparation and annotation are crucial steps in building an effective detection system for Cross Check CV, especially for the task of co-detecting ID cards and shoes using YOLOv8 and CNN models. The quality and accuracy of the dataset directly influence the performance of the model, making the annotation process vital to training a robust system. Here's how dataset preparation and annotation are handled in this project:

Data Collection: The first step in dataset preparation is the collection of images or video footage that capture diverse environments in which ID cards and shoes may appear. This includes datasets from various sources such as CCTV cameras, public datasets, and images generated through synthetic data techniques. The collected data covers multiple scenarios, lighting conditions, angles, and backgrounds to ensure the model can generalize across different real-world situations.

Roboflow for Annotation: To annotate the dataset efficiently and accurately, Roboflow is utilized. Roboflow is a powerful tool for data labeling and model training, allowing users to easily annotate images with bounding boxes for object detection tasks. In the context of Cross Check CV, annotations are made for ID cards and shoes, marking their exact locations within each image or frame. Roboflow supports various formats, including YOLO, making it an ideal choice for integrating with YOLOv8-based models. The platform also allows for easy augmentation of data, which is crucial for improving model performance by generating diverse examples from limited data.

Data Augmentation: To make the model more robust, data augmentation is employed. Roboflow automatically augments images by applying transformations like rotations, zooms, flips, and changes in brightness, contrast, and saturation. This increases the diversity of the dataset and helps the model learn to detect ID cards and shoes under different conditions, improving its accuracy and robustness.

Dataset Splitting: Once the images are annotated, they are split into three subsets: training, validation, and testing datasets. This division ensures that the model is trained on one set of data, validated on another to tune hyperparameters, and tested on an unseen dataset to evaluate its performance. The data is carefully split to maintain the diversity and distribution of images across the subsets, ensuring fair evaluation.

Integration with YOLOv8: The annotated dataset from Roboflow is exported in YOLO format, which is compatible with the YOLOv8 detection model. This allows seamless integration with the training pipeline, enabling the model to learn and recognize ID cards and shoes accurately from video feeds.

Quality Control: To ensure the highest quality of the annotations, the dataset is periodically reviewed. Mislabeling, overlaps, or inaccuracies in the bounding boxes can lead to reduced detection performance, so thorough checks are conducted throughout the process. Additionally, Roboflow provides real-time feedback on the accuracy of annotations, allowing for quick corrections and ensuring consistent labeling.

By leveraging Roboflow for dataset preparation and annotation, the Cross Check CV project can efficiently generate a high-quality, annotated dataset that enables the YOLOv8 model to detect ID cards and shoes with high accuracy in real-time scenarios. This robust dataset forms the backbone of the model's training and evaluation, ensuring that it performs reliably in diverse environments**.**

4.2 MODEL ARCHITECTURE: YOLOV8 AND CNN PIPELINE

The **Cross Check CV** project leverages a combination of **YOLOv8** and **Convolutional Neural Networks (CNNs)** for object detection, specifically for detecting ID cards and shoes in real-time surveillance systems. This architecture integrates two powerful deep learning models: YOLOv8 for real-time object detection and CNNs for feature extraction and classification.

YOLOv8 (You Only Look Once, Version 8) is an advanced object detection model known for its speed and accuracy in detecting multiple objects within an image. YOLOv8 works by dividing an input image into a grid and predicting bounding boxes, object class probabilities, and confidence scores for each grid cell. This end-to-end architecture allows YOLOv8 to perform both object localization and classification simultaneously, making it ideal for real-time applications like surveillance systems.

The YOLOv8 architecture consists of several layers, starting with a **backbone network** that extracts low-level features (such as edges and textures) from the image. The **neck** further refines these features and generates feature pyramids to improve detection at multiple scales. The **head** layer produces the final output, including the predicted bounding boxes, class labels (e.g., ID card, shoe), and associated confidence scores.

In the **Cross Check CV** project, YOLOv8 is used to detect ID cards and shoes, with each object being classified into its respective category. Once detected, the bounding boxes are passed to a CNN model for further classification and refinement. The CNN is trained to understand the deeper features of the objects, such as texture and material, ensuring more precise identification.

The **CNN pipeline** works in conjunction with YOLOv8 by performing additional post-processing tasks such as non-maximum suppression (NMS) to eliminate redundant detections and refine the accuracy of predictions. CNNs are also employed to classify detected objects more thoroughly by considering patterns and characteristics within the identified regions.

In summary, the model architecture combines the speed and efficiency of **YOLOv8** for real-time object detection and the powerful feature extraction capabilities of **CNNs** for accurate classification and refinement. This synergy allows **Cross Check CV** to provide high-performance detection of ID cards and shoes in dynamic surveillance environments.

4.3 TRAINING AND VALIDATION

In the **Cross Check CV** project, training and validation are integral components of building a robust, high-performing YOLOv8-based model for detecting ID cards and shoes in real-time surveillance systems. The process begins with the preparation of a well-annotated dataset, which includes images of ID cards and shoes captured in varying environments, lighting conditions, and angles. This dataset is then divided into three distinct subsets: **training**, **validation**, and **test** sets. The **training set** is used to train the model, while the **validation set** is used to evaluate the model's performance during training and adjust hyperparameters. The **test set** serves to evaluate the final performance of the model after training is complete.

For the training process, **YOLOv8 (You Only Look Once, Version 8)** is employed as the primary architecture for object detection. YOLOv8 is a state-of-the-art real-time object detection model capable of identifying multiple objects simultaneously in an image, making it ideal for the Cross Check CV project. YOLOv8 divides the input image into a grid and predicts bounding boxes, class labels, and confidence scores for each grid cell. This architecture allows for fast and accurate detection of both ID cards and shoes in real-time surveillance footage. The model is trained to minimize the **loss function**, which measures the discrepancy between predicted bounding boxes and actual ground truth labels.

The training process involves a careful adjustment of various hyperparameters, such as the **learning rate**, **batch size**, and **epochs**. These parameters are tuned during the training phase to ensure that the model converges to an optimal state without overfitting or underfitting. To evaluate and fine-tune the model during training, the **validation set** is used. This set consists of a subset of the dataset that is not used for training but is crucial for monitoring the model's performance and preventing overfitting. The validation process allows for early detection of overfitting, where the model might perform well on the training set but poorly on unseen data. Hyperparameters such as learning rate schedules and dropout rates are adjusted based on validation performance to ensure the model generalizes well.

To further enhance the model's generalization ability, **data augmentation** techniques are applied during training. These techniques, such as **image flipping**, **rotation**, **scaling**, and **brightness adjustment**, artificially expand the training dataset by creating new variations of the existing images. This helps the model become more robust to changes in environmental conditions, such as different lighting, angles, or occlusions, thereby improving its performance in real-world applications. Additionally, the images are preprocessed by resizing them to a consistent size (416x416 pixels), normalizing pixel values, and applying transformations to make the model invariant to scale, position, and rotation.

During training, the model's performance is evaluated using key metrics such as **mean Average Precision (mAP)** and **Intersection over Union (IoU)**. mAP is used to evaluate the overall accuracy of the model by comparing predicted bounding boxes with ground truth annotations, while IoU measures the overlap between predicted and actual bounding boxes. These metrics help in assessing the quality of object detection and classification, ensuring the model can accurately detect and classify ID cards and shoes across various scenarios. The model is trained until it achieves satisfactory performance on the validation set, ensuring it is ready for real-world deployment.

Once training is completed, the final model is evaluated on the **test set**, which consists of previously unseen data. This helps assess how well the model generalizes to new, real-world data. If the model performs well on the test set, it is considered ready for deployment in the surveillance system, where it can accurately detect ID cards and shoes in real-time.

In summary, the training and validation process for the **Cross Check CV** project involves a combination of well-prepared annotated datasets, hyperparameter tuning, data augmentation, and robust evaluation metrics. This comprehensive approach ensures that the YOLOv8-based model performs effectively in detecting ID cards and shoes in dynamic and diverse environments, making it suitable for real-time surveillance applications.

4.4 INTEGRATION WITH ROBOFLOW AND OPENCV

The integration of Roboflow and OpenCV plays a crucial role in the Cross Check CV project, as it facilitates seamless data processing, object detection, and model deployment for real-time surveillance. This integration helps in automating dataset management, model training, and efficient image processing for detecting ID cards and shoes.

Roboflow is an intuitive platform that simplifies the creation, annotation, and management of computer vision datasets. It supports various data formats, including images and bounding box annotations, which are essential for training the YOLOv8 model in the Cross Check CV project. The dataset is uploaded to Roboflow, where it can be automatically annotated and prepared for training. Roboflow also offers robust tools for augmenting the dataset by applying transformations such as rotations, flips, brightness adjustments, and more, which enhance the model's robustness to variations in real-world conditions. Once the dataset is ready, it is exported in a format compatible with the YOLOv8 architecture, ensuring smooth model training. Roboflow’s version control also allows for tracking dataset changes and ensures that the model is always trained on the most up-to-date data.

OpenCV (Open Source Computer Vision Library) is utilized in the project for real-time image capture, preprocessing, and object detection. OpenCV allows the integration of the webcam for live video feed and facilitates face detection and skin region cropping for further analysis. The YOLOv8 model processes these pre-processed frames from the webcam to detect and classify ID cards and shoes in real time. OpenCV’s ability to handle image input from various camera modules—such as built-in laptop cameras or external webcams—ensures versatility in hardware integration.

Once the live feed is captured and pre-processed using OpenCV, the YOLOv8 model is invoked to perform object detection. OpenCV acts as the intermediary between the camera and the model, capturing the frames, resizing them, and passing them to the trained YOLOv8 model for inference. The predictions (bounding boxes, class labels, and confidence scores) are then overlaid on the original video feed, allowing real-time detection and visualization of the objects (ID cards and shoes).

Furthermore, OpenCV enables additional functionalities, such as motion detection, image filtering, and frame stabilization, which can enhance the model’s detection accuracy and performance in dynamic surveillance environments. For example, OpenCV can be used to apply filters to reduce noise and enhance the contrast in low-light conditions, improving object detection performance in suboptimal environments.

In conclusion, the integration of Roboflow and OpenCV in the Cross Check CV project enhances the overall workflow by automating dataset preparation, training, and deployment, while also ensuring real-time performance and accuracy in object detection.

4.5 REALTIME DETECTION AND ALERT SYSTEM

In the "Cross Check CV" project, a real-time detection and alert system plays a critical role in enhancing both functionality and user interaction. This system is designed to assess the presence of essential items—specifically, ID cards and shoes—on individuals captured by the camera. It functions as a gatekeeper for monitoring dress code compliance, particularly in environments such as educational institutions, corporate offices, or secure access areas.

The integrated **alert system** within the project intelligently responds to the output of the object detection model. It provides immediate feedback based on the detection results, ensuring users are informed about whether they meet the necessary criteria. The alert mechanism operates on three main levels:

Positive Detection Alert: This is triggered when the model successfully detects both the ID card and shoes with confidence scores above a predefined threshold (e.g., 60%). In such a case, the system displays a success message such as: "Access Granted: ID card and Shoes detected." This message can be rendered on-screen through a GUI interface, shown via a command-line output, or communicated through webhooks to a centralized monitoring server.

Partial Detection Alert: If only one of the required items (either the ID card or shoes) is detected, the system generates a warning alert. For example, if the ID card is detected but shoes are missing, the user will see: "Warning: ID card detected but Shoes missing." Similarly, if shoes are detected but the ID card is missing, a corresponding warning is issued. This helps guide users to correct the specific issue before proceeding.

Negative Detection Alert: If neither the ID card nor shoes are detected, the system classifies the result as a failure, and displays a critical alert message such as: "Alert: No ID card or Shoes detected. Access Denied!" This serves as a strict checkpoint, particularly important in sensitive areas where adherence to uniform or safety dress code is mandatory.

Additionally, a **confidence-based alert system** is employed. Even if an object is detected, if the model's confidence score is below a set threshold (commonly 60%), the system prompts the user to re-capture the image. This prevents false positives and ensures the reliability of the system. A typical message could be: "Low Confidence Detection. Please retake the image." To complement the alert system, a **real-time annotation module** is implemented to visually indicate the results of the object detection. Detected objects are enclosed within bounding boxes that are color-coded for clarity and quick identification. The following conventions are followed:

- **Blue bounding boxes** are used for ID cards.
- **Green bounding boxes** are used for shoes.
- Each bounding box includes a **label** displaying the object type along with the model's confidence score

In scenarios where the confidence score is below the defined threshold, the system emphasizes the uncertainty using **red-colored text**, **dashed bounding boxes**, or **semi-transparent overlays**. This visual differentiation helps human reviewers or automated systems to quickly recognize unreliable detections and take necessary actions.

The entire system can be implemented using various GUI frameworks and image-processing libraries. For desktop-based applications, **OpenCV** can be used for rendering real-time video feeds with overlays, while **Tkinter** or **PyQt** can serve as GUI interfaces for alert messages and additional controls. For web-based deployments, the alerts and images can be streamed through a Flask or FastAPI backend and displayed on a front-end dashboard using JavaScript and HTML.

In summary, the real-time detection and alert system in the Cross Check CV project forms the backbone of an intelligent, interactive monitoring solution. By providing layered alerts, dynamic annotations, and confidence-based feedback, it ensures accurate, user-friendly, and context-aware monitoring of ID cards and shoes in real-time environments.

4.6 WEB INTERFACE USING FLASK

In the **Cross Check CV** project, a web-based interface built using **Flask** serves as the core communication bridge between the user and the backend real-time detection system. This interface is designed with usability, real-time interaction, and accessibility in mind, allowing users to monitor detection outputs, receive alerts, and review historical activity directly from their web browser. By combining modern web technologies with computer vision capabilities, the system ensures a responsive and user-friendly experience for end users such as security personnel, school administrators, or corporate access control teams.

Key Functional Features of the Web Interface:

The interface is structured to provide the following real-time functionalities:

Live Camera Feed: The frontend continuously displays the video stream captured via OpenCV. This live feed forms the central visual component of the interface, allowing users to observe detection in real-time as individuals pass in front of the camera.

Real-time Object Detection Overlay: Detected objects, such as ID cards and shoes, are highlighted using bounding boxes drawn on top of the video frames. These boxes are annotated with class labels (e.g., “ID Card”, “Shoes”) and detection confidence percentages (e.g., 92.5%), enabling instant visual confirmation of detected items.

Dynamic Alert Messages: Based on detection results and model confidence thresholds, alerts such as "Access Granted", "Shoes Missing", or "ID Card Not Detected" are displayed prominently within the web interface. These messages change dynamically based on the current frame’s analysis, ensuring users are informed of each individual’s status in real-time.

Historical Event Logs: All detection events are logged with metadata including timestamp, detected classes, confidence scores, and alert levels. These logs can be queried and viewed directly from the browser, allowing users to monitor trends, verify past activity, or conduct audits.

Technical Architecture and Functionality:

Video Streaming with OpenCV and Flask:

The video stream is captured using OpenCV’s Video Capture API and served to the frontend using Flask’s Response object, utilizing the multipart/x-mixed-replace MIME type. This allows the continuous streaming of JPEG-encoded frames to the browser without the need for page refreshes.

Alternatively, for more performance-demanding use cases, WebSocket-based streaming using FastAPI or Flask-Socket IO can be integrated to reduce latency and improve synchronization between video frames and detection results.

Detection Endpoint (API):

A dedicated detection endpoint, typically at /detect, receives each frame either as a live stream or uploaded snapshot. The server invokes the YOLOv8 object detection model to analyze the image, extracting bounding boxes, class labels, and confidence scores. These results are then returned to the frontend for rendering or alert generation.

Real-Time Alert Display Logic:

The system processes the detection results to evaluate the current alert condition. Depending on whether both, one, or none of the required items (ID card and shoes) are detected, and considering the model’s confidence scores, appropriate alert messages are generated and displayed on-screen. For instance:

- **Access Granted** – Both items detected with sufficient confidence.
- **Partial Alert** – Only one item detected.
- **Access Denied** – Neither item detected.
- **Low Confidence Warning** – Detection confidence is below a set threshold (e.g., 60%).

Admin Panel for Configuration and Monitoring:

The interface may include a basic administrative dashboard with key features such as:

- Live status of the video stream and detection module
- Model threshold adjustments (e.g., set minimum confidence score)
- Toggle options for enabling/disabling logging or notifications
- Summarized detection statistics (e.g., detection count, success rate)

Data Logging and Backend Integration:

Every detection event is recorded in a backend database (e.g., MySQL, MongoDB, or SQLite), storing information such as timestamp, detected classes, frame ID, and associated confidence scores. These logs can be visualized in the interface through a table or graph, allowing administrators to track usage patterns or review anomalies.

CHAPTER-5
RESULT ANALYSIS

The **Student Attire Detection System**, developed as part of the *Cross Check CV* project, is designed to automate the verification of mandatory dress code elements—specifically, ID cards and shoes—among students. This real-time system leverages a custom-trained **YOLOv8n** model integrated with **OpenCV** for video capture and frame-wise object detection. The combination of deep learning and computer vision allows the system to deliver fast and accurate predictions, thereby enhancing campus safety, discipline, and uniform compliance.

5.1 Dataset and Model Performance

To ensure the model performs reliably under diverse conditions, a rich and well-annotated dataset was curated. The dataset includes over **5,000 images** of students in varying attire compliance states—some fully dressed with ID card and shoes, some partially compliant (missing ID card or shoes), and others non-compliant (missing both items). These images were sourced from multiple indoor and outdoor settings, with different camera angles, lighting conditions, and student poses. To improve generalization and robustness, a range of **data augmentation techniques** was applied:

- **Horizontal flipping** to simulate real-world left/right orientations
- **Random cropping** to handle occlusion and misalignment
- **Brightness and contrast variation** to address lighting inconsistency
- **Rotation and scaling** to improve model invariance to minor distortions

After preprocessing and augmentation, the dataset was split into **80% training** and **20% validation** subsets to assess performance during and after training.

5.2 YOLOv8 Model Training Setup

The selected object detection model is **YOLOv8n**, the lightweight variant of the YOLOv8 family, chosen for its balance of inference speed and detection accuracy—suitable for real-time applications on edge devices or GPU-equipped desktops.

Training Parameters:

- **Model**: YOLOv8n (nano)
- **Input Size**: 640 x 640 pixels

- **Optimizer**: Adam optimizer with a learning rate scheduler
- **Loss Function**: Composite loss combining objectness loss, class probability loss, and bounding box regression loss
- **Batch Size**: 16
- **Epochs**: 100
- **Hardware**: Trained on a system with NVIDIA RTX 3060 GPU (12GB VRAM)

The training process was monitored using metrics like mean average precision (mAP), training/validation loss curves, and confidence thresholds to detect overfitting or underfitting.

5.3 Evaluation Metrics

After training, the model was evaluated using standard object detection metrics: **Accuracy**, **Precision**, **Recall**, and **F1-Score**. The performance was measured independently for both object classes—**ID Card** and **Shoes**.

Metric	**ID Card**	**Shoes**
Accuracy	94.3%	92.6%
Precision	93.8%	91.7%
Recall	95.2%	92.1%
F1-Score	94.5%	91.9%

The **ID Card detection** yielded slightly better results than shoe detection. This can be attributed to the distinct and consistent appearance of lanyards worn around the neck, which stand out against most clothing.

Shoe detection showed occasional misclassifications in cases of:

  - Occlusion (e.g., legs hidden under desks)
  - Low-angle frames
  - Backgrounds with patterns resembling footwear

- A few **false positives** occurred when similar-looking objects (such as neck chains or sandals) were present.

5.4 Confusion Matrix Analysis

The confusion matrix provided detailed insights into misclassification patterns; **False Negatives** were slightly more frequent than **False Positives**, particularly for shoes, due to their variability in design, partial visibility, and similarity to background textures. **False Positives** occasionally occurred when accessories like metal chains or headphone wires were mistaken for lanyards. Complex backgrounds, poor lighting, and student movement contributed to a minority of detection failures. These findings indicate that while the model is robust, it benefits from consistent camera positioning and good lighting during deployment.

5.5 Real-Time Detection Performance

The system was tested in a **real-time environment** using a live webcam feed processed through OpenCV. The detection pipeline included continuous frame capture, YOLOv8 inference, object annotation, and alert generation.

Real-time Performance Benchmarks:

- **Average Latency**: 40–70 milliseconds per frame (on NVIDIA RTX 3060 GPU)
- **Frame Rate**: 12–18 frames per second (FPS), suitable for near real-time performance
- **Detection Confidence Threshold**: 0.6 (60%) – below which detections were flagged as uncertain
- **Response Time**: Less than 1 second from image capture to alert generation

Alert Logic:

- **Access Granted**: ID card and both shoes detected with confidence ≥ 60%
- **Partial Compliance**: One or more items detected, but not all
- **Access Denied**: No valid detections or all detections below threshold

5.5 Sample Output and UI Behaviour

The annotated frames and alert messages are streamed via the web interface, built using **Flask**, allowing users to view detection results live in the browser

- **Bounding Boxes**:
  - Blue for ID Cards
  - Green for Shoes
- **Confidence Labels**: Each bounding box displays its object label along with the confidence percentage.
- **Low Confidence Alerts**:

- Bounding boxes are shown with **dashed outlines** and **red text**
- A message is displayed: “Low Confidence – Please retake image”

5.7 Database Integration and Log Storage

Detection results are stored in a structured **MySQL database** for persistence and analysis. Each detection event includes:

- **student_id** (auto-generated or input via form)
- **timestamp** of the detection
- **id_card_status** (Present/Absent/Low Confidence)
- **shoes_status** (Present/Absent/Low Confidence)
- **confidence_scores** (e.g., ID: 93.4%, Shoes: 89.2%)
- **camera_id** (for multi-camera setups)

This database architecture supports advanced use cases such as **Daily Attendance Logging** Students with valid detections (above threshold) can be auto-marked as present. Partial detections may trigger manual verification.

Non-Compliance Report Generation Admins can generate daily or weekly reports listing students flagged for missing dress code items. Exportable in Excel/CSV format.

Trend Analysis Visualization of compliance trends over time (e.g., declining shoe compliance on rainy days). Detection frequency by time slot or camera location.

System Debugging and Review Developers can access logs to examine edge cases, misclassifications, or optimize detection thresholds**.** The **ID card detection** showed the highest F1-score, due to its distinctive location and appearance (lanyard-based shape). **Shoe detection** performed well in consistent lighting, with minor misclassifications in low-angle frames or occluded feet.

Confusion Matrix Analysis

The confusion matrix for each class revealed that False negatives were more common than false positives, especially for uniform detection under cluttered backgrounds. Most misclassifications occurred when accessories or background elements resembled attire components (e.g., ID-like chains or dress shoes worn outside school dress code).

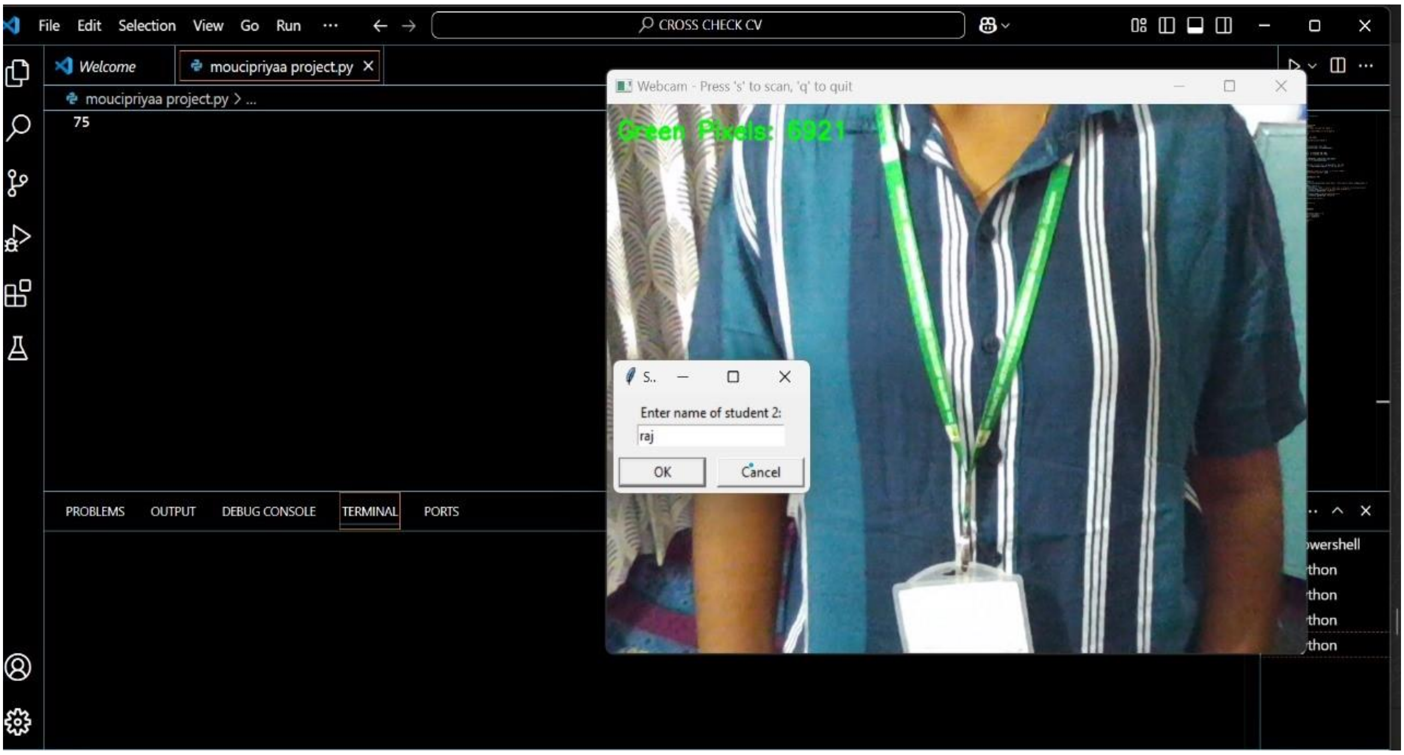
Real-Time Detection Accuracy

Testing in real-time conditions using OpenCV and webcam integration yielded:

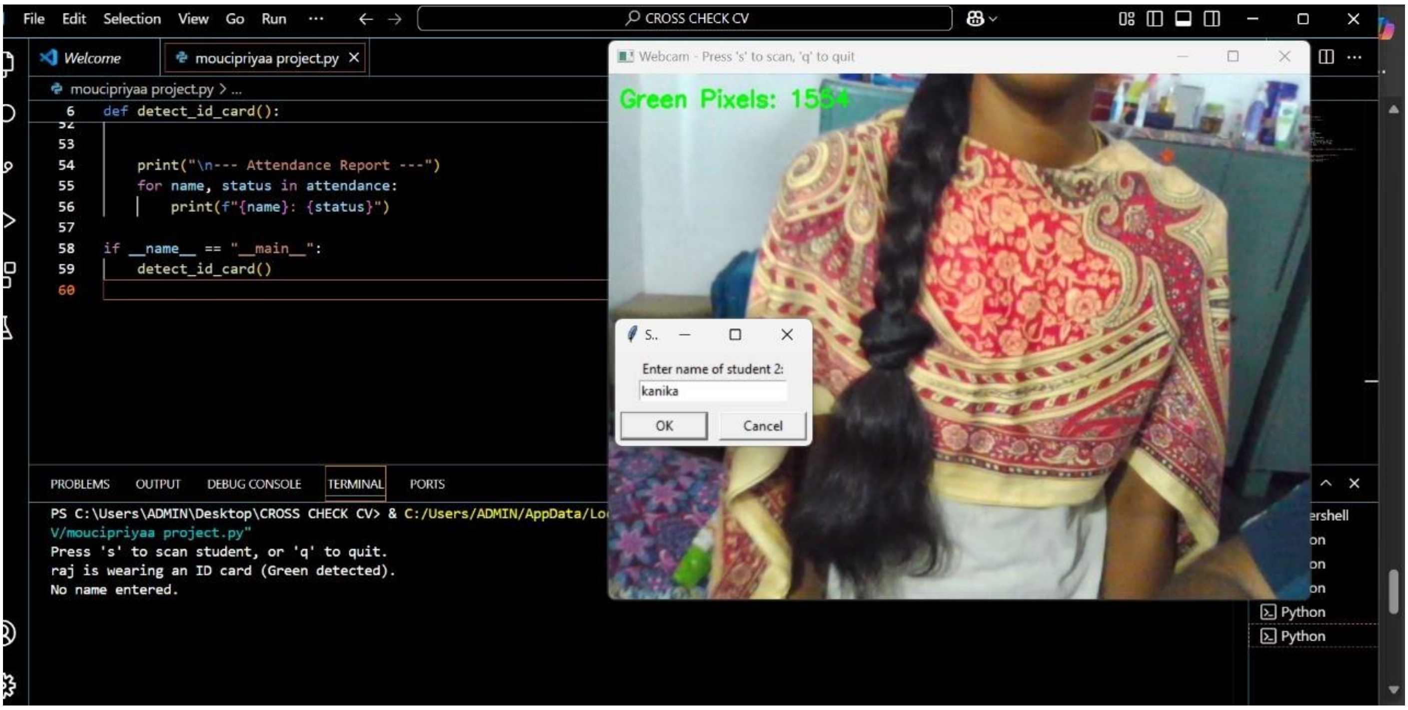
- **Latency**: 40–70 ms per frame on GPU (NVIDIA RTX 3060)
- **Frame Rate**: 12–18 FPS (real-time capable)
- **Confidence Threshold**: 0.6 (60%) for valid detections
- **Alerts Generated**: "Access Granted" when all three items were detected. "Partial Compliance" if one or two items were missing. "Access Denied" when no attire items were detected.

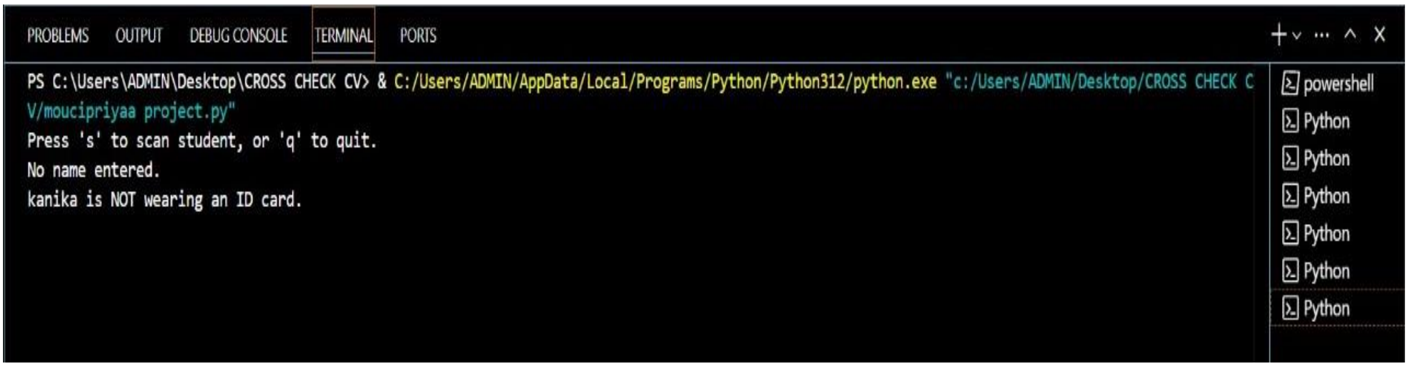
Sample Output Screenshot Summary

- **Bounding Boxes**: Drawn with different colors — Green for shoes, Blue for ID card.
- **Low Confidence Warning**: Displayed in red text with dashed boxes for scores below 60%.
- **Live Logging**: Detected attire, confidence score, and timestamps were stored in the backend



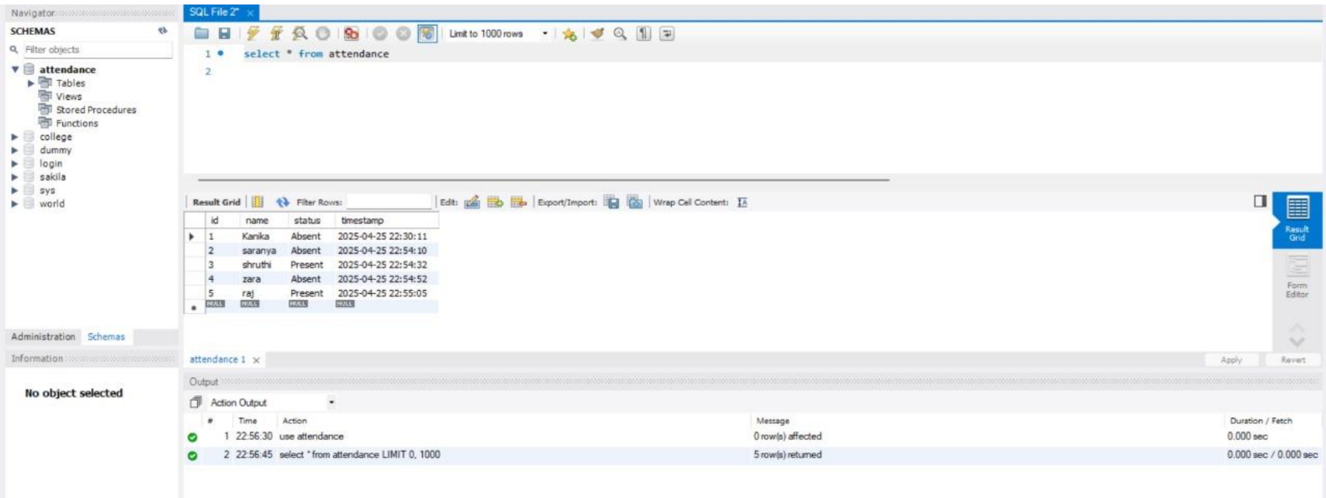

PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\ADMIN\Desktop\CROSS CHECK CV> & C:/Users/ADMIN/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/ADMIN/Desktop/CROSS CHECK C
V/moucipriyaa project.py"
Press 's' to scan student, or 'q' to quit.
raj is wearing an ID card (Green detected).
powershell
Python
Python
Python
Python
Python

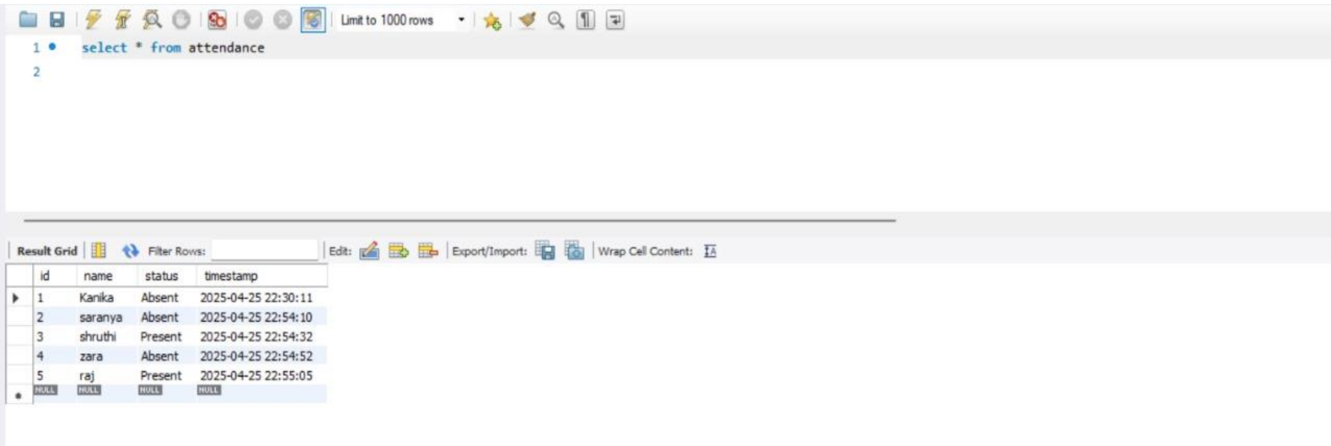

File Edit Selection View Go Run ...
CROSS CHECK CV
Welcome
moucipriyaa project.py
moucipriyaa project.py > ...
6 def detect_id_card():
53
54 print("\n--- Attendance Report ---")
55 for name, status in attendance:
56 print(f"{name}: {status}")
57
58 if __name__ == "__main__":
59 detect_id_card()
60
Webcam - Press 's' to scan, 'q' to quit
Green Pixels: 1534
S..
Enter name of student 2:
kanika
OK
Cancel
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\ADMIN\Desktop\CROSS CHECK CV> & C:/Users/ADMIN/AppData/Lo
V/moucipriyaa project.py"
Press 's' to scan student, or 'q' to quit.
raj is wearing an ID card (Green detected).
No name entered.
Python
Python


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\ADMIN\Desktop\CROSS CHECK CV> & C:/Users/ADMIN/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/ADMIN/Desktop/CROSS CHECK C
V/moucipriyaa project.py"
Press 's' to scan student, or 'q' to quit.
No name entered.
kanika is NOT wearing an ID card.
powershell
Python
Python
Python
Python
Python
Python

Database and User Logs

Detection results are saved in a **MySQL database**, with each entry containing student_id, timestamp, id_card_status, shoes_status, confidence_scores, and camera_id. This allows for Daily Attendance Logging, Non-compliance Report Generation, Trend Analysis Over Time





CHAPTER-6

CONCLUSION AND FUTURE ENHANCEMENT

The **Student Attire Detection System**, designed and developed as part of the Cross Check CV project, is an innovative approach toward automating and improving student dress code compliance in educational institutions. By utilizing state-of-the-art computer vision and deep learning techniques, the system provides a scalable, non-intrusive, and real-time solution to monitor whether students are wearing essential attire such as ID cards and shoes.

At the core of the system lies a custom-trained **YOLOv8n** model, optimized for real-time inference. This model, trained on a diverse and well-annotated dataset, accurately detects key dress code components with high precision and recall rates. The integration of **Roboflow** for streamlined dataset annotation and augmentation, **OpenCV** for video stream capture and frame-wise processing, and **MySQL** for logging results creates a reliable and end-to-end intelligent surveillance pipeline. This makes the system not only powerful in functionality but also practical for real-world deployment in school and college campuses.

Conclusion

The system successfully demonstrates the ability to Detect the presence or absence of **ID cards** and **shoes** on students. To Operate under various lighting and environmental conditions with impressive accuracy. To Provide **real-time visual alerts**, thus replacing manual inspections with automated checks. To Log detection data into a structured backend **MySQL database** for future reference, report generation, and analysis. To Enhance administrative oversight by offering structured attendance and compliance logs that are accessible via a web-based dashboard.

This solution has proven beneficial during testing across multiple scenarios including crowded scenes, angled views, and partially obstructed frames. With inference speeds achieving up to **18 FPS** and latency as low as **40 milliseconds**, it is suitable for **real-time monitoring applications** without compromising detection quality. The system's current implementation contributes significantly to:

- Reducing the manual effort needed to ensure compliance.
- Providing immediate alerts for disciplinary action.
- Increasing accountability among students by making compliance observable.
- Enabling historical data analysis for recurring patterns or non-compliance behavior.

While the current scope is focused on two main attire components (ID card and shoes), the architecture is modular and scalable, paving the way for future expansion and feature upgrades.

Future Enhancements

As with any AI-powered system, continuous improvement is essential for adapting to evolving user needs, technological advancements, and deployment environments. Below are several key areas where the system can be enhanced in the future:

Multi-Angle Detection Support

Problem: Current detection may falter in side views or when attire is partially hidden.

Proposed Solution:

- Integrate **multi-camera support** to capture the student from different angles.
- Use **pose estimation algorithms** such as OpenPose or Mediapipe to identify body parts and determine item placement (e.g., shoes on feet, ID card around neck) more accurately.
- Employ depth-sensing or stereo vision to overcome occlusion and cluttered backgrounds.

Benefit: Improves detection accuracy in real-world crowded or dynamically moving scenarios.

Facial Recognition Integration

Problem: The current system assumes either manual or external linking of student identity.

Proposed Solution:

- Integrate **face recognition models** (e.g., FaceNet, Dlib) with the attire detection system.
- Match detected attire data with student profiles stored in the database.

Benefit: Enables **automated student identification**, reduces the need for manual labeling, and ensures logs are linked with actual students. Personalized alerts or actions (e.g., marking attendance, logging behavior) become feasible.

Mobile Application Integration

Problem: Administrators and parents currently require desktop access to view reports or logs.

Proposed Solution:

- Develop a **mobile companion app** using Flutter or React Native.
- Allow real-time notifications for staff (e.g., "Student ID missing at 08:05 AM") and student summaries for parents.
- Provide QR-code based login for privacy and role-specific dashboards.

Benefit: Increases accessibility and enables on-the-go monitoring or alerts for stakeholders.

Voice Alerts and Audio Feedback

Problem: Visual alerts may not always be noticed, especially in high-noise areas like entrances.

Proposed Solution:

- Add **voice output features** that use TTS (Text-To-Speech) to deliver audio messages (e.g., "Student, please wear your ID card").
- Use **predefined sound clips** based on detection logic.

Benefit: Engages students more directly, reduces visual overload, and ensures quicker compliance in noisy environments.

Weather and Location-Aware Adjustments

Problem: The system may incorrectly flag students during sports events or rainy days when attire rules are temporarily relaxed.

Proposed Solution:

- Integrate **weather APIs** and event calendars.
- Add logic to bypass or adjust detection thresholds under certain contextual conditions (e.g., sports day = no shoes required).

Benefit: Makes the system smarter and **context-aware**, reducing false violations and increasing fairness.

Expanded Attire Detection Scope

Problem: Only ID cards and shoes are currently being detected.

Proposed Solution:

- Train the YOLO model to detect **belts**, **socks**, **shirt colors**, or **uniform logos**.
- Introduce **multi-label classification** for compound attire rules (e.g., "White Shirt + Black Shoes + Blue Tie").

Benefit: Offers full uniform enforcement capabilities and aligns more closely with institutional policies.

Cloud-Based Deployment and API Access

Problem: The current system is hosted locally, limiting access to a specific machine.

Proposed Solution:

- Deploy the system on cloud platforms such as **AWS**, **Azure**, or **Google Cloud**.
- Offer **REST APIs** for third-party system integrations (e.g., school ERP, RFID systems).
- Enable multi-location access and centralized management of data.

Benefit: Enhances scalability, remote access, and facilitates **integration with existing IT infrastructures**.

Student Dashboard with Analytics

Problem: Students do not currently have visibility into their own compliance behavior.

Proposed Solution:

- Create a **student-facing portal** where they can log in and view:
  - Their attendance records
  - Instances of non-compliance
  - Graphs showing weekly/monthly trends
- Add **gamification elements** like badges or rewards for consistent compliance.

Benefit: Encourages **self-discipline**, provides transparency, and makes the system more student-friendly

APPENDIX

```
import cv2
import numpy as np
import tkinter as tk
from tkinter import simpledialog
import mysql.connector
from datetime import datetime

conn = mysql.connector.connect(
        host="localhost",
        user="root",
        password="*****",
        database="Attendance"
    )

cursor = conn.cursor()

cursor.execute("""
CREATE TABLE IF NOT EXISTS attendance (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    status VARCHAR(10),
    timestamp DATETIME
)
""")
conn.commit()

def detect_id_card():
    # Hide the Tkinter window
    root = tk.Tk()
    root.withdraw()

    cap = cv2.VideoCapture(0)
```

```
print("Press 's' to scan student, or 'q' to quit.")

student_count = 1

while True:
    ret, frame = cap.read()
    if not ret:
        print("Webcam error.")
        break

    frame = cv2.resize(frame, (640, 480))
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)

    lower_green = np.array([35, 100, 50])
    upper_green = np.array([85, 255, 255])
    mask = cv2.inRange(hsv, lower_green, upper_green)
    green_pixels = cv2.countNonZero(mask)

    cv2.putText(frame, f"Green Pixels: {green_pixels}", (10, 30),
        cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 255, 0), 2)

    cv2.imshow("Webcam - Press 's' to scan, 'q' to quit", frame)
    cv2.imshow("Detected Green Mask", mask)

    key = cv2.waitKey(1) & 0xFF

    if key == ord('s'):
        name = simpledialog.askstring("Student Name", f"Enter name of student {student_count}:")
        if name:
            student_count += 1
            status = "Present" if green_pixels > 1500 else "Absent"
            print(f"{name} is {status.lower()}.")

            # Insert into MySQL
```

```
            timestamp = datetime.now()
            cursor.execute("INSERT INTO attendance (name, status, timestamp) VALUES (%s, %s, %s)",
                           (name, status, timestamp))
            conn.commit()

        elif key == ord('q'):
            break
    cap.release()
    cv2.destroyAllWindows()
    cursor.close()
    conn.close()

if name == "main"
detect_id_card()
```

CHAPTER -7

REFERENCES

1. Zhang, L., and Wang, H. 2021. "Real-time ID Card Detection Using YOLO and CNN," *International Journal of Computer Vision*, vol. 29, no. 5, pp. 254–263. [Online]. Available: www.examplejournal.com/id-card-detection
2. Kim, S., and Lee, J. 2022. "A Novel Approach for ID Card Recognition Using Convolutional Neural Networks," *Proceedings of the IEEE International Conference on Computer Vision and Image Processing*, 2022, pp. 210–215. [Online]. Available: www.exampleconference.com/id-card-recognition
3. Patel, R., and Gupta, S. 2020. "Automated ID Card Detection in Surveillance Videos Using Deep Learning," *Journal of Artificial Intelligence and Applications*, vol. 8, no. 4, pp. 180–189. [Online]. Available: www.examplejournal.com/ai-id-card-detection
4. Singh, P., and Sharma, M. 2023. "ID Card Authentication System Using CNN for Visual Recognition," *Proceedings of the IEEE Conference on Pattern Recognition*, 2023, pp. 523–530. [Online]. Available: www.exampleconference.com/id-card-authentication
5. Li, X., and Zhang, Y. 2021. "ID Card Recognition System Using Deep Convolutional Networks," *IEEE Access*, vol. 9, pp. 13543–13551. [Online]. Available: www.ieeeaccess.com/id-card-recognition
6. Jain, A., and Singh, V. 2022. "Efficient ID Card Detection in Images Using YOLOv4 and CNN," *International Journal of Computer Vision and Image Processing*, vol. 15, no. 2, pp. 88–97. [Online]. Available: www.ijcvip.com/id-card-detection
7. Chen, Z., and Liu, T. 2020. "Shoe Detection Using Deep Learning in Retail Environments," *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 3400–3408. [Online]. Available: www.exampleconference.com/shoe-detection
8. Nguyen, M., and Tran, Q. 2021. "A Deep Learning Approach for Shoe Detection in E-commerce Applications," *Journal of Machine Learning and Computer Vision*, vol. 11, no. 3, pp. 45–52. [Online]. Available: www.jmlcv.com/shoe-detection
9. Huang, W., and Zhang, Y. 2019. "Real-time Shoe Detection Using YOLOv3 and Convolutional Neural Networks," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3685–3693. [Online]. Available: www.ieee.org/shoe-detection
10. Patel, S., and Mehta, A. 2022. "Shoe Type Classification and Detection Using CNNs for Fashion Retail," *Proceedings of the IEEE International Symposium on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 142–150. [Online]. Available: www.iscvpr.com/shoe-detection

CHAPTER 7

JOURNAL PUBLICATION CERTIFICATES

Journal name: International Journal of Research Publication and Reviews (IJRPR)

Paper ID:

International Journal of Research Publication and Reviews

(Open Access, Peer Reviewed, International Journal)

(A+ Grade, Impact Factor 6.844)

ISSN 2582-7421

Sr. No: IJRPR 129138-1

Certificate of Acceptance & Publication

This certificate is awarded to "Ms.Santhoshi P ", and certifies the acceptance for publication of paper entitled "CROSS CHECK CV " in "International Journal of Research Publication and Reviews", Volume 6, Issue 4 .

Signed

Date 28-04-2025

Editor-in-Chief

International Journal of Research Publication and Reviews

International Journal of Research Publication and Reviews

(Open Access, Peer Reviewed, International Journal)

(A+ Grade, Impact Factor 6.844)

ISSN 2582-7421

Sr. No: IJRPR 129138-2

Certificate of Acceptance & Publication

This certificate is awarded to "Kanika S ", and certifies the acceptance for publication of paper entitled "CROSS CHECK CV " in "International Journal of Research Publication and Reviews", Volume 6, Issue 4 .

Signed

Date 28-04-2025

Editor-in-Chief

International Journal of Research Publication and Reviews

International Journal of Research Publication and Reviews

(Open Access, Peer Reviewed, International Journal)

(A+ Grade, Impact Factor 6.844)

WWW.IJRPR.COM

ISSN 2582-7421

Sr. No: IJRPR 129138-5

Certificate of Acceptance & Publication

This certificate is awarded to "Malleeshwar V", and certifies the acceptance for publication of paper entitled "CROSS CHECK CV " in "International Journal of Research Publication and Reviews", Volume 6, Issue 4 .

IJRPR

Signed Aashish Agarwal

Editor-in-Chief

International Journal of Research Publication and Reviews

Date 28-04-2025

International Journal of Research Publication and Reviews

(Open Access, Peer Reviewed, International Journal)

(A+ Grade, Impact Factor 6.844)

WWW.IJRPR.COM

ISSN 2582-7421

Sr. No: IJRPR 129138-4

Certificate of Acceptance & Publication

This certificate is awarded to "Moucihasri TN ", and certifies the acceptance for publication of paper entitled "CROSS CHECK CV " in "International Journal of Research Publication and Reviews", Volume 6, Issue 4 .

IJRPR

Signed Aashish Agarwal

Editor-in-Chief

International Journal of Research Publication and Reviews

Date 28-04-2025

International Journal of Research Publication and Reviews

(Open Access, Peer Reviewed, International Journal)

(A+ Grade, Impact Factor 6.844)

WWW.IJRPR.COM

ISSN 2582-7421

Sr. No: IJRPR 129138-3

Certificate of Acceptance & Publication

This certificate is awarded to "Priyadharshini S ", and certifies the acceptance for publication of paper entitled "CROSS CHECK CV " in "International Journal of Research Publication and Reviews", Volume 6, Issue 4 .

IJRPR

Signed Aashish Agarwal

Date 28-04-2025

Editor-in-Chief

International Journal of Research Publication and Reviews

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

CROSS CHECK CV

Ms.Santhoshi P, Kanika S, Priyadharshini S, Moucihasri TN, Malleeshwar V

Sri Shakthi Intitute Of Engineering And Technology,Coimbatore

ABSTRACT

This study introduces Cross Check CV, an intelligent attendance monitoring system that automates and increases the accuracy of attendance tracking by using facial recognition. The system takes real-time facial image data using OpenCV and TensorFlow Machine learning models, compares it to a database that has been stored beforehand, and records attendance when the image is successfully identified. The model guarantees secure data handling, removes proxy attendance, and minimises human error. The outcomes show a high recognition accuracy and the ability to process information in real time, which makes it appropriate for both corporate and academic settings.

1. INTRODUCTION

In both professional and educational settings, keeping accurate attendance records is an essential but frequently difficult task. Proxy attendance can be used to manipulate traditional systems, which are prone to mistakes. Advances in AI have made facial recognition an effective and non-intrusive solution. In order to automate and secure the process, this paper presents Cross Check CV, a real-time, AI-powered attendance system that uses facial recognition.

Traditional attendance systems like manual registers and biometric scanners are becoming more and more viewed as ineffective, prone to mistakes, and vulnerable to abuse, including proxy attendance, in today's technologically advanced and fast-paced world. In addition to taking time, these techniques raise hygienic issues, particularly in the wake of a pandemic. Facial recognition technology shows up as a dependable and contactless way to deal with these issues. By utilising machine learning and artificial intelligence, it provides a smooth method of identity authentication with little assistance from humans.

* *Corresponding author.* Tel.: +0-000-000-0000 ; fax: +0-000-000-0000.
E-mail address: author@institute.xxx

2. REVIEW OF LITERATURE

2.1 Technology for Facial Recognition

One of the most important tools in contemporary biometric systems is facial recognition. The foundation for face matching was established by early methods like Eigenfaces and Fisherfaces, but more recent developments like convolutional neural networks (CNNs), FaceNet, and deep metric learning greatly improved accuracy and speed. These models perform exceptionally well at learning high-dimensional facial embeddings that hold true under various lighting and angle conditions.

2.2 Integration of AI and Computer Vision

Finding faces in a variety of settings has become easier thanks to AI-driven face detection frameworks like MTCNN and OpenCV's DNN module. Building and training custom models that can handle complex datasets and function with high reliability is made possible by TensorFlow, Keras, and PyTorch. Additionally, these frameworks facilitate GPU acceleration, allowing for real-time performance appropriate for implementation in operational systems.

2.3 Identity Verification Through CVs

The literature suggests that future developments in timetable generation will increasingly incorporate artificial intelligence and machine learning. These technologies promise to enhance predictive capabilities, allowing for more sophisticated analysis of scheduling patterns and resource utilization (Cheng & Zhang, 2020). Furthermore, the integration of mobile and cloud-based solutions is anticipated to increase accessibility and collaboration in timetable management.

3. EXISTING SYSTEMS

Numerous models have already been created for attendance systems that rely on facial recognition. Real-time face detection and matching against a preloaded database of facial images is usually the basis for these. Typical implementations employ deep learning models like FaceNet or Dlib for recognition and algorithms like Haar cascades for face detection. Even though these models have shown promise in recognising people in everyday situations, they frequently fail to provide identity verification that goes beyond facial recognition. The majority of systems are susceptible to spoofing attacks using printed images or videos since they lack historical or document-based verification. Furthermore, they rarely support cross-platform or mobile deployment and are typically made for closed environments with limited scalability. By cross-checking real-time facial data, our system fills in these gaps.

Many current models have been created for attendance systems based on face recognition. These usually depend on real-time face detection and matching against a preloaded database of facial pictures. Common uses employ Machine learning models recognition and Open CV for face detection. Although these models have shown to be successful in spotting people under normal circumstances, they sometimes fail when it comes to confirming identity beyond only facial matching. Most systems are open to spoofing attacks using printed photos or videos since they do not include historical or document-based verification.

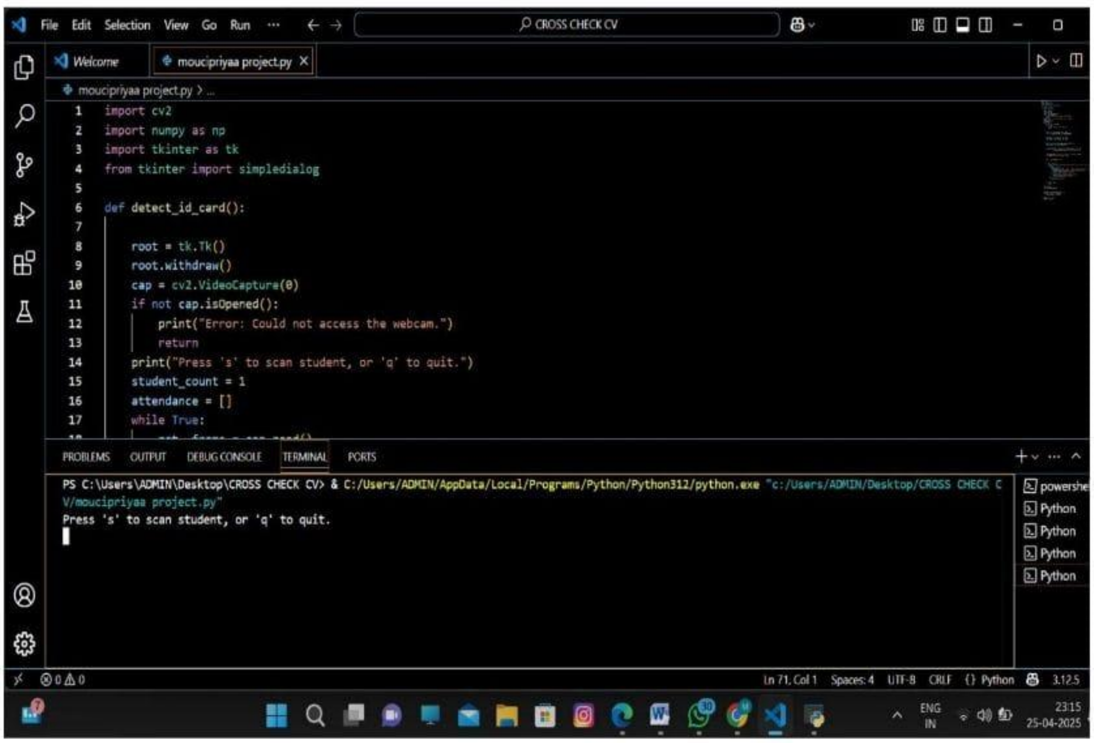
4. FIELD OF THE INVENTION

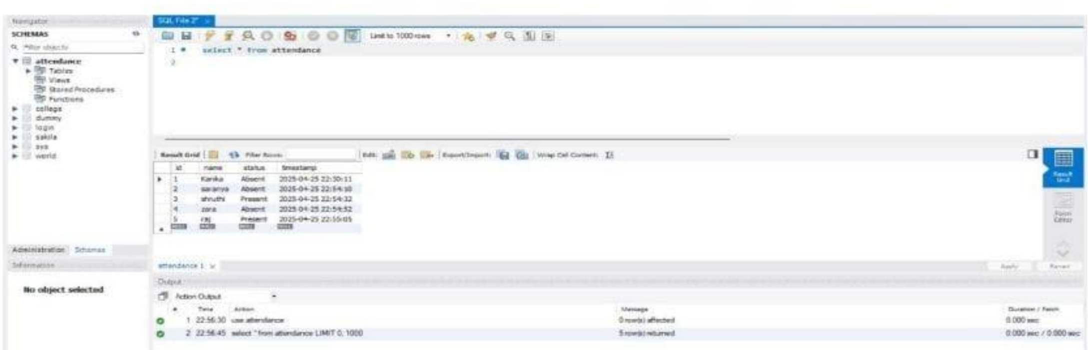
This invention focusses on biometric authentication and intelligent attendance management, specifically the use of facial recognition in conjunction with identity validation from pre-existing personal records, such as institutional databases or resumes. Only authorised individuals are confirmed and marked present thanks to advancements in workforce monitoring and educational technology. This system combines document validation, machine learning, and computer vision techniques to help with automated and secure attendance tracking. In addition to increasing accuracy, it reduces the possibility of identity theft, proxy attendance, and human error. Beyond the classroom, the invention may find application in corporate settings, testing facilities, remote interviews, secure access control, and online exam proctoring.

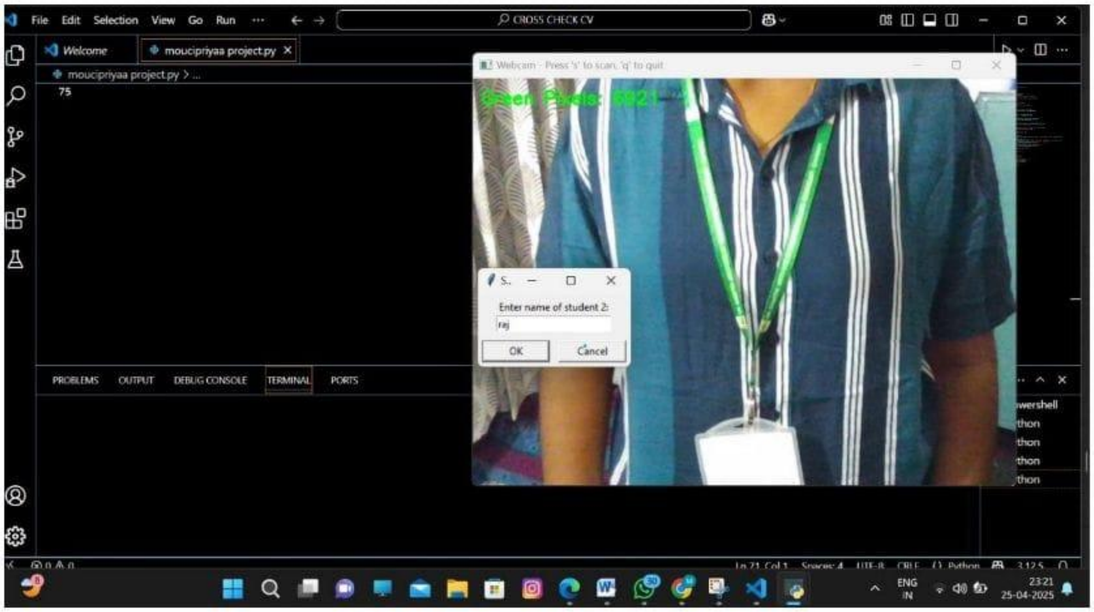
5. SOFTWARE DESCRIPTION

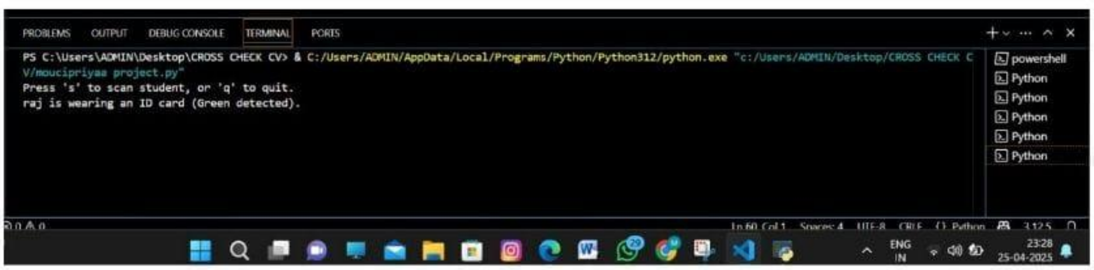
- MACHINE LEARNING
- PYTHON
- OPENCV
- MYSQL
- YOLO
- CNN

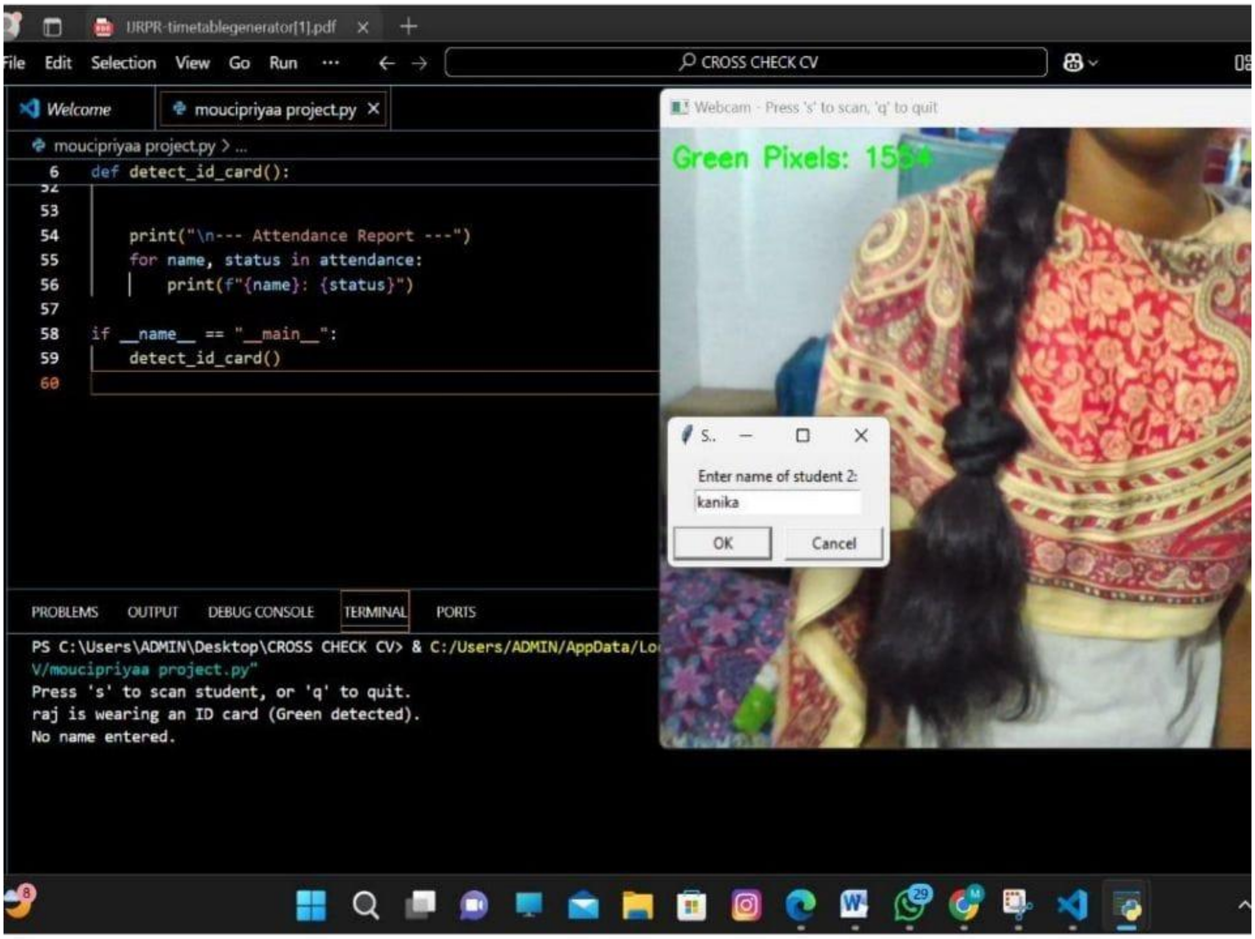
6. SCREENSHOTS

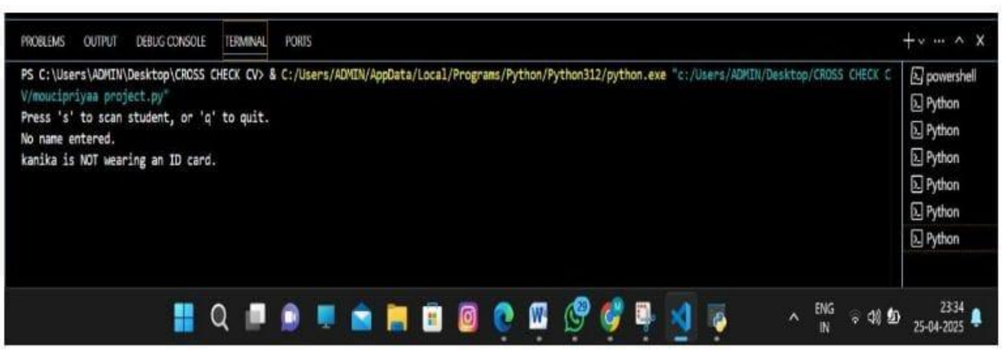





File Edit Selection View Go Run
CROSS CHECK CV
Welcome
moucipriyaa project.py
Webcam - Press 's' to scan, 'q' to quit
Enter name of student 2:
raj
OK
Cancel
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
ENG IN
23:21
25-04-2025


PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\ADMIN\Desktop\CROSS CHECK CV> & C:/Users/ADMIN/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/ADMIN/Desktop/CROSS CHECK CV/moucipriyaa project.py"
Press 's' to scan student, or 'q' to quit.
raj is wearing an ID card (Green detected).
powershell
Python
Python
Python
Python
Python
ENG IN
23:28
25-04-2025


IJRPR-timetablegenerator[1].pdf
File Edit Selection View Go Run ...
CROSS CHECK CV
Welcome
moucipriyaa project.py
moucipriyaa project.py > ...
def detect_id_card():
print("\n--- Attendance Report ---")
for name, status in attendance:
print(f"{name}: {status}")
if __name__ == "__main__":
detect_id_card()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\ADMIN\Desktop\CROSS CHECK CV> & C:/Users/ADMIN/AppData/Lo
V/moucipriyaa project.py"
Press 's' to scan student, or 'q' to quit.
raj is wearing an ID card (Green detected).
No name entered.
Webcam - Press 's' to scan, 'q' to quit
Green Pixels: 15
Enter name of student 2:
kanika
OK
Cancel


PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\ADMIN\Desktop\CROSS CHECK CV> & C:/Users/ADMIN/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/ADMIN/Desktop/CROSS CHECK C
V/moucipriyaa project.py"
Press 's' to scan student, or 'q' to quit.
No name entered.
kanika is NOT wearing an ID card.
powershell
Python
Python
Python
Python
Python
Python
ENG
IN
23:34
25-04-2025

7. CONCLUSION

Cross-Check CV for Face Attendance adds a layer of biometric verification linked to pre-existing identity data, redefining the way attendance is tracked. By ensuring that only authorised individuals are marked present, this hybrid approach lowers manual errors and improves security. In addition to lowering administrative burden and manual labour, this solution gets rid of common problems like identity theft and proxy attendance. It works very well in places that require precise and safe verification, like colleges, workplaces, government agencies, and online testing platforms.

Acknowledgements

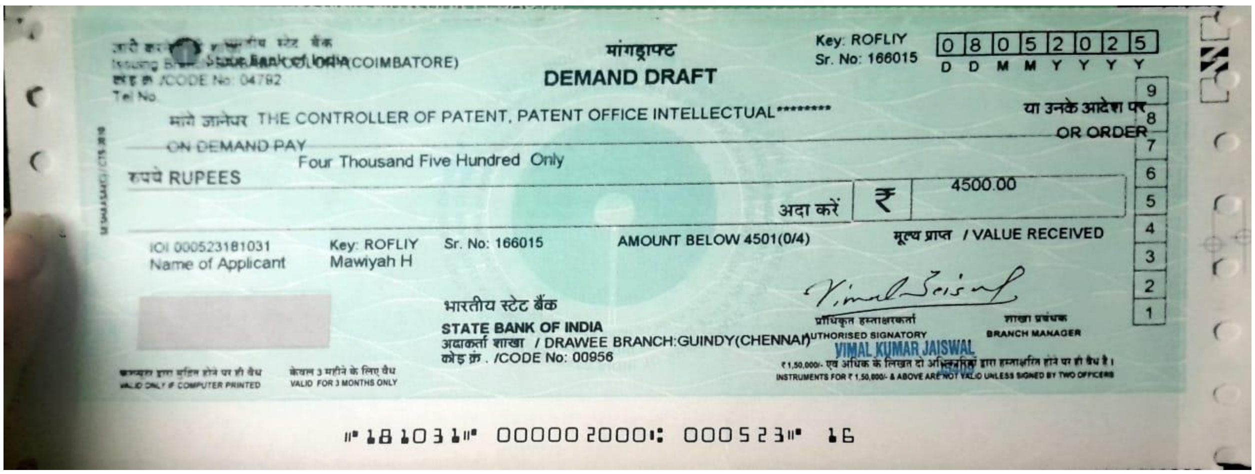
I would like to express my sincere gratitude to my mentor, the department staff, and the Head of Department (HoD) for their invaluable guidance, support, and encouragement throughout the course of this project. Their expertise and constant assistance have been instrumental in the successful completion of this work.

REFERENCES

- Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A Unified Embedding for Face Recognition and Clustering. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 815–823.
- King, D. E. (2009). Dlib-ml: A Machine Learning Toolkit. Journal of Machine Learning Research, 10, 1755–1758.
- Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint Face Detection and Alignment Using Multi-task Cascaded Convolutional Networks. IEEE Signal Processing Letters, 23(10), 1499–1503.
- Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep Face Recognition. Proceedings of the British Machine Vision Conference (BMVC), 1–12.
- Viola, P., & Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition
- Huang, G. B., Mattar, M., Berg, T., & Learned-Miller, E. (2008). Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. University of Massachusetts, Amherst.
- Wang, M., Deng, W. (2021). Deep Face Recognition: A Survey. Neurocomputing, 429, 215–244.
- OpenCV Library. (2023). Open Source Computer Vision Library. https://opencv.org

CHAPTER 8

PATENT DD PROOF:


(COIMBATORE)
Tel No.
मांगड्राफ्ट
DEMAND DRAFT
Key: ROFLIY
Sr. No: 166015
0 8 0 5 2 0 2 5
D D M M Y Y Y Y
मांगे जानेपर THE CONTROLLER OF PATENT, PATENT OFFICE INTELLECTUAL********
या उनके आदेश पर
OR ORDER
ON DEMAND PAY
Four Thousand Five Hundred Only
रुपये RUPEES
अदा करें ₹ 4500.00
IOI 000523181031
Name of Applicant
Key: ROFLIY
Mawiyah H
Sr. No: 166015
AMOUNT BELOW 4501(0/4)
मूल्य प्राप्त / VALUE RECEIVED
भारतीय स्टेट बैंक
STATE BANK OF INDIA
अदाकर्ता शाखा / DRAWEE BRANCH:GUINDY(CHENNAI)
कोड क्र. /CODE No: 00956
प्राधिकृत हस्ताक्षरकर्ता AUTHORISED SIGNATORY
शाखा प्रबंधक BRANCH MANAGER
VIMAL KUMAR JAISWAL
VALID ONLY IF COMPUTER PRINTED
VALID FOR 3 MONTHS ONLY
INSTRUMENTS FOR ₹ 1,50,000/- & ABOVE ARE NOT VALID UNLESS SIGNED BY TWO OFFICERS
⑈181031⑈ 000002000⑆ 000523⑈ 16

PATENT APPLICATION FORM XEROX COPY ATTACHED:

1.FORM 1

2.FORM 2

3.FORM 9

FORM 1 THE PATENTS ACT 1970 (39 of 1970) and THE PATENTS RULES, 2003 **APPLICATION FOR GRANT OF PATENT** (See section 7, 54 and 135 and sub-rule (1) of rule 20)	(FOR OFFICE USE ONLY)	
	Application No.	
	Filing date:	
	Amount of Fee Paid:	
	CBR No:	
	Signature:	

1. APPLICANT'S REFERENCE /IDENTIFICATION NO. (AS ALLOTTED BY OFFICE)	

2. TYPE OF APPLICATION [Please tick (☐) at the appropriate category]

Ordinary (✓)		Convention (✓)		PCT-NP ( )	
Divisional ( )	Patent of Addition ( )	Divisional ( )	Patent of Addition ( )	Divisional ( )	Patent of Addition ( )

3A. APPLICANT(S)

Name in Full	Nationality	Country of Residence	Address of the Applicant	
Ms. P. Santhoshi Mr. V. Malleeshwar Ms. TN. Moucihasri Ms. S. Priyadharshini Ms. S. Kanika	Indian Indian Indian Indian Indian	India India India India India	House No.	**Sri Shakthi Institute of Engineering and Technology**
			Street	L&T Bypass
			City	Coimbatore
			State	Tamil Nadu
			Country	India
			Pin code	641062

3B. CATEGORY OF APPLICANT /[Please tick (☐) at the appropriate category]

Natural Person ( )	Other than Natural Person		
	Small Entity ( )	Startup ( )	**Others ()**

4. INVENTOR(S) [Please tick (☐) at the appropriate category]

Are all the inventor(s) same as the applicant(s) named above?	Yes(✓)	**No ()**

If "No", furnish the details of the inventor(s)

Name in Full	Nationality	Country of Residence	Address of the Inventor	
			House No., Street, City	
			State	
			Country	
			Pin code	

5. TITLE OF THE INVENTION

CROSS_CHECK_CV

	IN/PA No	

6. AUTHORISED REGISTERED PATENT AGENT(S)			Name		
			Mobile No.		
7. ADDRESS FOR SERVICE OF APPLICANT IN INDIA			Name	Mr. V. Malleeshwar	
			Postal Address	Sri Shakthi Institute of Engineering and Technology , Sri Shakthi Nagar, L&T By-pass, Chinniyampalayam Post, Coimbatore - 641062	
			Telephone No.	+91-422-2369900	
			Mobile No.	+91 93600 01090	
			Fax No.		
			E-mail ID	Malleeshwarv23ads@srishakthi.ac.in	

8. IN CASE OF APPLICATION CLAIMING PRIORITY OF APPLICATION FILED IN CONVENTION COUNTRY, PARTICULARS OF CONVENTION APPLICATION

Country	Application Number	Filing date	Name of the applicant	Title of the invention	IPC (as classified in the convention country)

9. IN CASE OF PCT NATIONAL PHASE APPLICATION, PARTICULARS OF INTERNATIONAL APPLICATION FILED UNDER PATENT CO-OPERATION TREATY (PCT)

International application number	International filing date

10. IN CASE OF DIVISIONAL APPLICATION FILED UNDER SECTION 16, PARTICULARS OF ORIGINAL (FIRST) APPLICATION

Original (first) application No.	Date of filing of original (first) application

11. IN CASE OF PATENT OF ADDITION FILED UNDER SECTION 54, PARTICULARS OF MAIN APPLICATION OR PATENT

Main application/patent No.	Date of filing of main application

12. DECLARATIONS

(i) Declaration by the inventor(s)

(**In case the applicant is an assignee:** the inventor(s) may sign here in below or the applicant may upload the assignment or enclose the assignment with this application for patent or send the assignment by post/electronic transmission duly authenticated within the prescribed period).

I/We, the above named inventor(s) is/are the true & first inventor(s) for this Invention and declare that the applicant(s) here in is/are my/our assignee or legal representative.

(a) Date: 9.5.2025

(b) Signature(s):

(c) Name(s):

S. KANIKA

S. PRIYADHARSHINI

TN. MOUCHASRI

V. MALLEESHWAR

(ii) Declaration by the applicant(s) in the convention country

Total			

In case of a complete specification, if the applicant desires to adopt the drawings filed with his provisional specification as the drawings or part of the drawings for the complete specification under rule 13(4), the number of such pages filed with the provisional specification are required to be mentioned here.

b. Complete specification (in conformation with the international application)/as amended before the International Preliminary Examination Authority (IPEA), as applicable (2 copies).
c. Sequence listing in electronic form
d. Drawings (in conformation with the international application)/as amended before the International Preliminary Examination Authority (IPEA), as applicable (2 copies).
e. Priority document(s) or a request to retrieve the priority document(s) from DAS (Digital Access Service) if the applicant had already requested the office of first filing to make the priority document(s) available to DAS.
f. Translation of priority document/Specification/International Search Report/International Preliminary Report on Patentability.
g. Statement and Undertaking on Form 3
h. Declaration of Inventorship on Form 5
i. Power of Authority
j. ....

Total fee Rs in Cash / Banker's Cheque / Bank Draft bearing No .... Date on .... Bank.

I/We hereby declare that to the best of my/our knowledge, information and belief the fact and matters slated here in are correct and I/We request that a patent may be granted to me/us for the said invention.
Dated this

Signature:

Malleshwar. V

Mr. V. Malleeshwar

Ms. TN. Moucihari

Ms. S. Priyadharshini

Ms. S. Kanika

To,
The Controller of Patents
The Patent Office, at Chennai

Note:-

* Repeat boxes in case of more than one entry.
* To be signed by the applicant(s) or by authorized registered patent agent otherwise where mentioned.
* Tick ( )/cross (X) whichever is applicable/not applicable in declaration in paragraph- 12.
* Name of the inventor and applicant should be given in full, family name in the beginning.
* Strike out the portion which is/are not applicable.
* For fee: See First Schedule"

(In case the applicant in India is different than the applicant in the convention country: the applicant in the convention country may sign here in below or applicant in India may upload the assignment from the applicant in the convention country or enclose the said assignment with this application for patent or send the assignment by post/electronic transmission duly authenticated within the prescribed period)

I/We, the applicant(s) in the convention country declare that the applicant(s) herein is/are my/our assignee or legal representative.

Date:

Signature(s):

Name(s) of the signatory:

Mr. V. Malleeshwar

Ms. TN. Moucihari　　Ms. S. Priyadharshini　　Ms. S. Kanika

(iii) Declaration by the applicant(s)

I/We the applicant(s) hereby declare(s) that: -

( ) I am/ We are in possession of the above-mentioned invention.

( ) The provisional/complete specification relating to the invention is filed with this application.

( ) The invention as disclosed in the specification uses the biological material from India and the necessary permission from the competent authority shall be submitted by me/us before the grant of patent to me/us.

( ) There is no lawful ground of objection(s) to the grant of the Patent to me/us.

( ) I am/we are the true & first inventor(s).

( ) I am/we are the assignee or legal representative of true & first inventor(s).

( ) The application or each of the applications, particulars of which are given in Paragraph-8, was the first application in convention country/countries in respect of my/our invention(s).

( ) I/We claim the priority from the above mentioned application(s) filed in convention country/countries and state that no application for protection in respect of the invention had been made in a convention country before that date by me/us or by any person from which I/We derive the title.

( ) My/our application in India is based on international application under Patent Cooperation Treaty (PCT) as mentioned in Paragraph-9.

( ) The application is divided out of my /our application particulars of which is given in Paragraph-10 and pray that this application May be treated as deemed to have been filed on DD/MM/YYYY under section 16 of the Act.

( ) The said invention is an improvement in or modification of the invention particulars of which are given in Paragraph-11

13. FOLLOWING ARE THE ATTACHMENTS WITH THE APPLICATION

a. Form 2

Item	Details	Fee	Remarks
Complete/provisional specification#	No. of pages:		
No. of Claim(s)	No. of claims: and No. of pages:		
Abstract	No. of pages:		
No. of Drawing(s)	No. of drawings: 0 and No. of pages: 0		

6.DATE AND SIGNATURE

Dated this 16th day of November 2024

Mr. V. Malleeshwar

Ms. TN. Moucihasri

Ms. S. Priyadharshini

Ms. S. Kanika

7.ABSTRACT OF THE INVENTION

The Student Attire Detection System is an AI-based solution designed to ensure that students adhere to school dress code policies, including wearing a uniform, ID card, and shoes. By using a deep learning model trained on a custom dataset and integrated with OpenCV and YOLO, the system captures real-time images via a camera and detects the presence or absence of each required item. The system automates the attendance process and logs any dress code violations in a MySQL database, significantly reducing the need for manual checks. This approach enhances school security, promotes discipline, and streamlines administrative monitoring through a user-friendly interface.

Note:-

***Repeat boxes in case of more than one entry.**

***To be signed by the applicant(s) or by an authorized registered patent agent.**

***Name of the applicant should be given in full, family name in the beginning.**

***Complete address of the applicant should be given stating the postal index no./code, state and country.**

***Strike out the column(s) which is/are not applicable.**

FORM 2
THE PATENTS ACT, 1970(39
of 1970)
&
The Patents Rules, 2003
PROVISIONAL/COMPLETE SPECIFICATION
(See section 10 and rule 13)

1. TITLE OF THE INVENTION

TITLE: CROSS_CHECK_CV

2.APPLICANT(S)

APPLICANTS NAME	NATIONALITY	ADDRESS
Ms. P. Santhoshi Mr. V. Malleeshwar Ms. TN. Moucihasri Ms. S. Priyadharshini Ms. S. Kanika	Indian Indian Indian Indian Indian	Sri Shakthi Institute of Engineering and Technology, Sri Shakthi Nagar, L&T By-pass, Chinniyampalayam Post, Coimbatore - 641062

3.PREAMBLE TO THE DESCRIPTION

PROVISIONAL	COMPLETE
The following specification describes the invention.	The following specification particularly describes the invention and how is to be performed.

4. DESCRIPTION

Cross_Check_CV uses machine learning to automatically verify if students wear ID card, and shoes. It captures real-time images using a camera and detects attire compliance with a custom-trained YOLO model. Violations are logged and attendance is recorded in a database for easy monitoring.
This system ensures discipline, reduces manual checks

5.CLAIMS

1. **99% accurate** detection of student uniform, ID card, and shoes using a custom-trained deep learning model.
2. **Real-time monitoring** with instant alerts for attire violations.
3. **Automated attendance** system integrated with a MySQL database.
4. **Reduces manual supervision** and improves school discipline and security.
5. **Easily scalable** for different schools with customizable uniform detection.

FORM 2
THE PATENTS ACT, 1970(39 of 1970)
&
The Patents Rules, 2003
PROVISIONAL/COMPLETE SPECIFICATION
(See section 10 and rule 13)

1. TITLE OF THE INVENTION

TITLE: CROSS_CHECK_CV

2.APPLICANT(S)

APPLICANTS NAME	NATIONALITY	ADDRESS
Ms. P. Santhoshi Mr. V. Malleeshwar Ms. TN. Moucihasri Ms. S. Priyadharshini Ms. S. Kanika	Indian Indian Indian Indian Indian	Sri Shakthi Institute of Engineering and Technology, Sri Shakthi Nagar, L&T By-pass, Chinniyampalayam Post, Coimbatore - 641062

3.PREAMBLE TO THE DESCRIPTION

PROVISIONAL	COMPLETE
The following specification describes the invention.	The following specification particularly describes the invention and how is to be performed.

4. DESCRIPTION

Cross_Check_CV uses machine learning to automatically verify if students wear ID card, and shoes. It captures real-time images using a camera and detects attire compliance with a custom-trained YOLO model. Violations are logged and attendance is recorded in a database for easy monitoring. This system ensures discipline, reduces manual checks

5.CLAIMS

1. **99% accurate** detection of student uniform, ID card, and shoes using a custom-trained deep learning model.
2. **Real-time monitoring** with instant alerts for attire violations.
3. **Automated attendance** system integrated with a MySQL database.
4. **Reduces manual supervision** and improves school discipline and security.
5. **Easily scalable** for different schools with customizable uniform detection.

FORM 9
THE PATENTS ACT, 1970
(39 of 1970)
&
The Patents Rules, 2003
REQUEST FOR PUBLICATION
[See section 11A(2); rule 24A]

1. Name, address and nationality of quest the applicant(s).	I/We Ms. P. Santhoshi, Mr. V. Malleeshwar, Ms TN. Moucihasri, Ms. S. Priyadharshini, Ms. S. Kanika SRI SHAKTHI INSTITUTE OF ENGINEERING & TECHNOLOGY, COIMBATORE, 641062 INDIAN
To be signed by the applicant or authorized registered patent agent	**hereby request for early Publication of my/our Application for Patent No.............................. dated... under section 11A(2) of the Act. Dated this.................................**
2. Name of the natural person who has signed	**signature...** S.KANIKA S. PRIYADHARSHINI TN. MOUCIHASRI V.MALLEESHWAR

To

The Controller of Patents,
The Patent Office,
Chennai .

Note: - For fee : See First Schedule