YOLO
REAL VS FAKE PERSON IDENTIFICATION

MACHINE LEARNING (21CSC305P)

A Project by :
Vibhor Saxena (RA2211026030047)
Divyansh Bhardwaj (RA2211026030052)
Shreya Garg (RA2211026030022)

ABSTRACT

In this project, we developed a machine learning system that distinguishes between a live person and a photograph of a person in real-time using the YOLO (You Only Look Once) object detection framework. The primary objective is to enhance security measures in applications like biometric authentication by preventing spoofing attempts using static images.

OVERVIEW AND INTRO

What we did:

Developed a machine learning system using YOLO for real-time detection between real and fake persons

Usecase

Application in biometric security by preventing spoofing.

Tools used:

OpenCV, YOLO, custom dataset.

INTRODUCTION

- Facial recognition is widely used in security and authentication systems.
- However, spoofing attacks, where images or videos are used to deceive systems, pose a serious threat.
- Types of spoofing include printed photographs, images displayed on screens, and even 3D masks

- The goal of this project is to develop a system that can distinguish between real persons and fake images in real time.
- We use the YOLO (You Only Look Once) object detection framework, which is known for its speed and accuracy
- The approach includes collecting a dataset, preprocessing the data, training the YOLO model, and deploying it in a real-time application(ML Project Report).

EXISTING PROBLEM

Current facial recognition systems focus on static features and are easily fooled by spoofing attacks.

Spoofing methods such as printed photographs, screen-displayed images, or 3D masks undermine security systems.

This leads to unauthorized access, financial fraud, and privacy breaches.

PROPOSED SOLUTION

We propose a software-based solution using YOLO to perform real-time liveness detection.

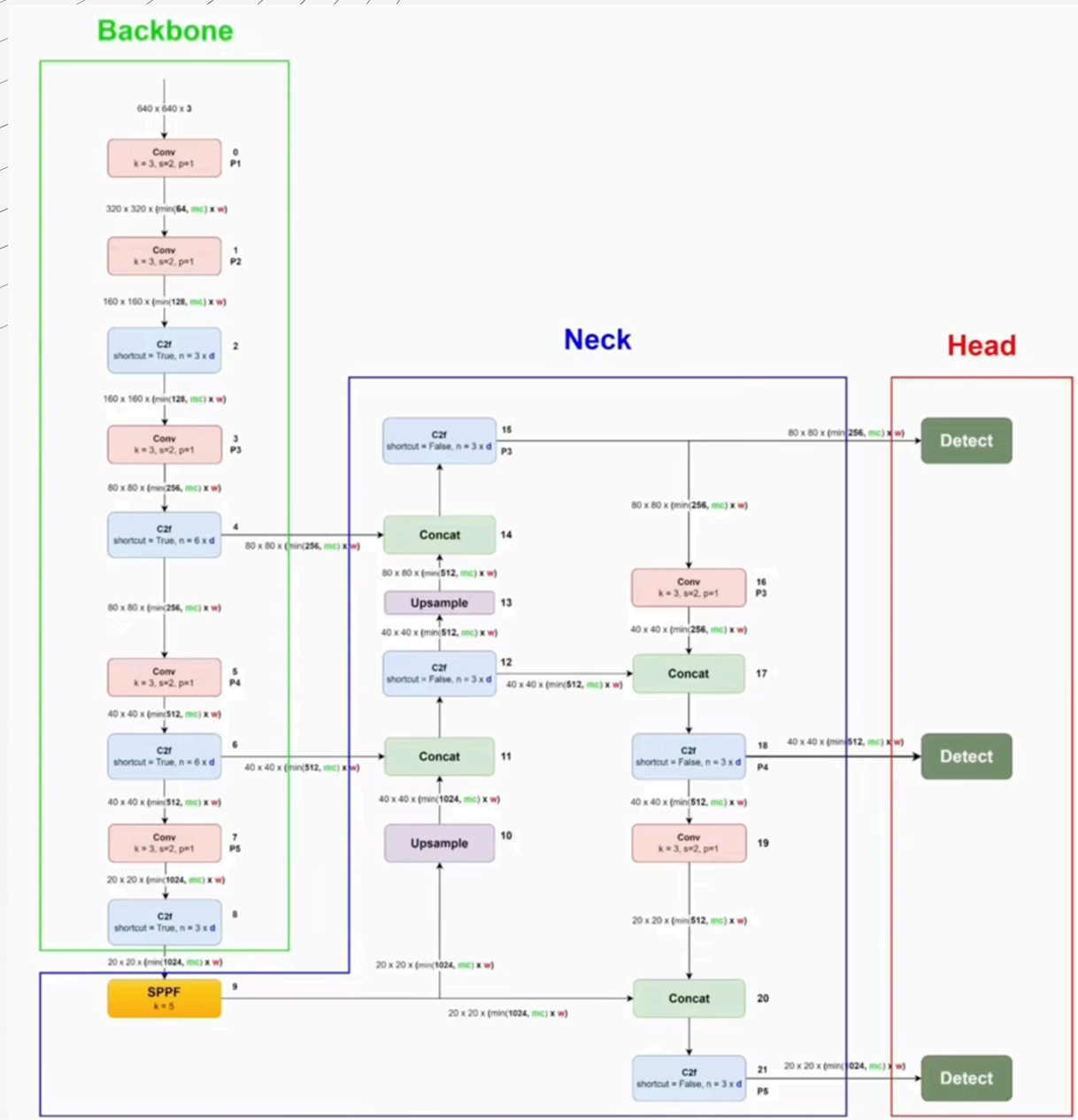
This system detects live persons vs fake images by analyzing webcam input, making it practical for real-world deployment.

The solution does not require additional hardware, keeping costs low while improving security.

TECHNOLOGY USED

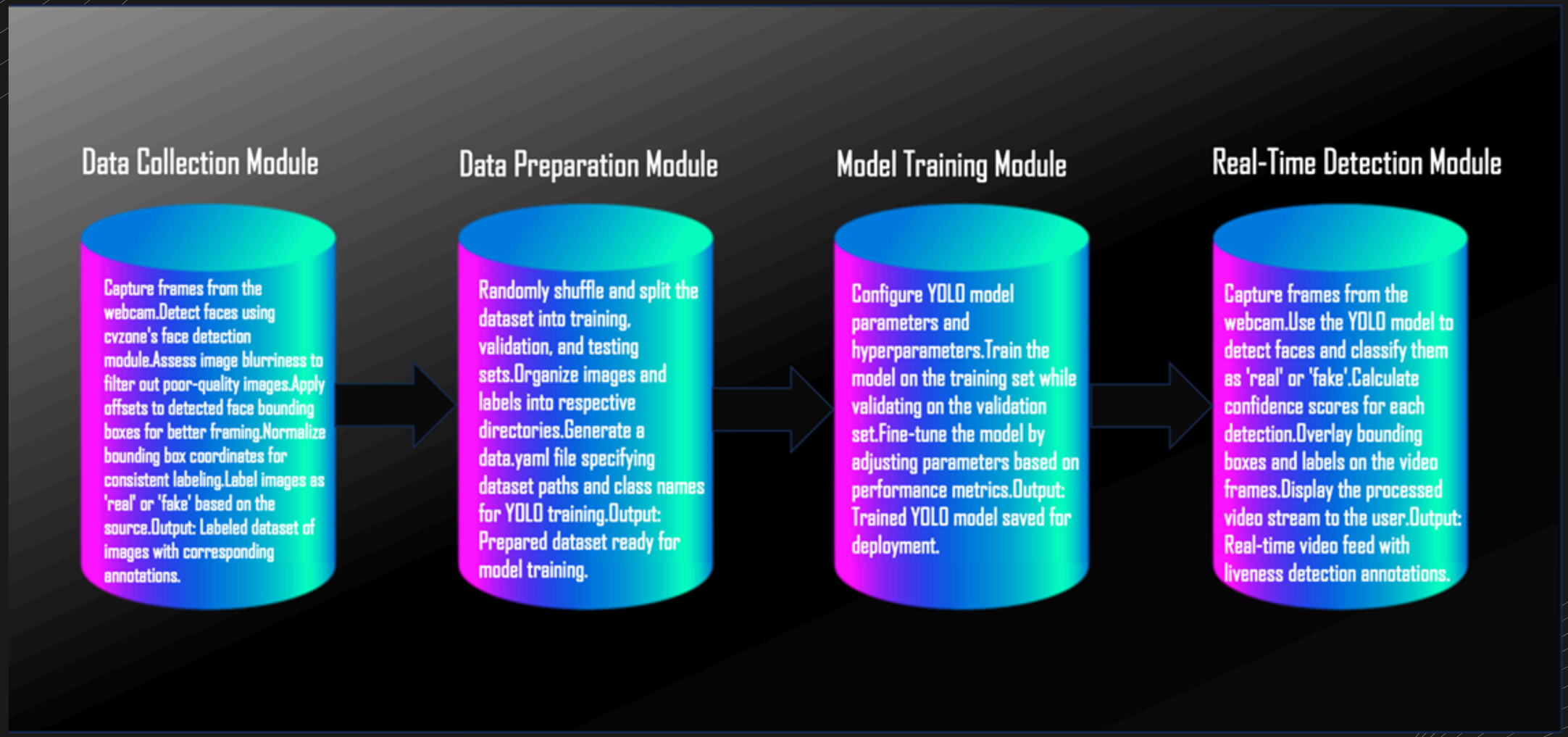
Programming Language: Python for ease of development and rich support for ML libraries.
Libraries: OpenCV for image processing, cvzone for face detection, and YOLO for object detection.
Hardware: A standard webcam is used for live video feed and real-time detection.

YOLOv8 Architecture Diagram



- YOLOv8 architecture consists of three main components: the Backbone, Neck, and Detection Head. The Backbone is responsible for feature extraction and uses CSPDarknet53, which improves information flow and gradient propagation during training. The Neck incorporates PANet, which enhances multi-scale feature detection, while the Detection Head predicts bounding boxes, class probabilities, and objectness scores, making YOLOv8 highly efficient and accurate for real-time object detection tasks.

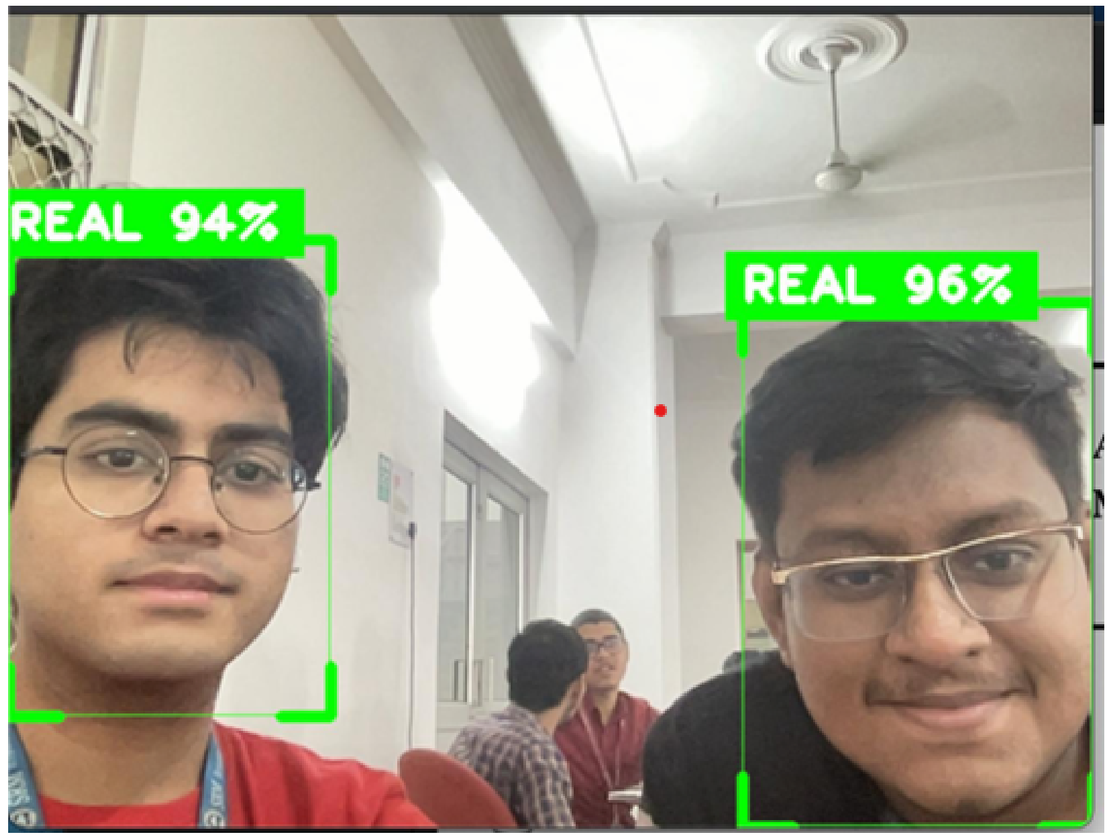
Methodology and Modules

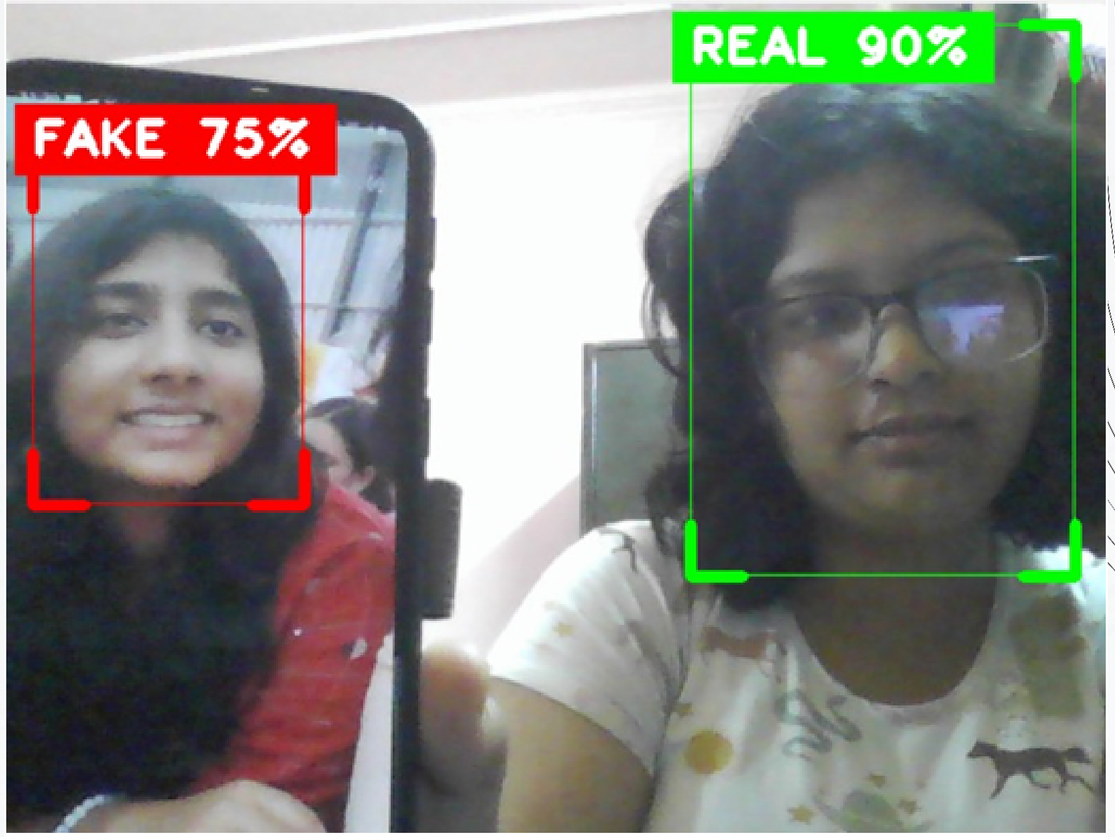


- The Data Collection Module captures frames from a live webcam feed, detects faces using cvzone, and filters out blurry images. Detected faces are labeled as 'real' or 'fake' based on their source, with bounding boxes adjusted and normalized for consistent labeling. The output is a labeled dataset ready for training.
- The Data Preparation and Model Training Modules organize the collected dataset into training, validation, and testing sets. The YOLO model is trained and fine-tuned based on performance metrics, resulting in a trained model ready for real-time deployment.
- In the Real-Time Detection Module, the trained YOLO model processes live video feeds from the webcam, detecting and classifying faces as 'real' or 'fake' with confidence scores. The system displays these results with annotated frames, providing immediate feedback to users.

RESULTS

The system successfully distinguishes between live persons and fake images in real-time.
The YOLO model maintained high accuracy during testing, with real-time performance using a webcam.





CONCLUSIONS

In this project, we successfully developed a real-time liveness detection system that differentiates between live individuals and spoofing attempts using photographs. By leveraging the YOLOv8 object detection framework, we created a machine learning model capable of operating efficiently without the need for specialized hardware, making it both practical and cost-effective for widespread deployment.

Our comprehensive methodology encompassed data collection, preprocessing, model training, and real-time application development. The system demonstrated high accuracy in classifying faces as 'real' or 'fake' in live video streams, effectively addressing security vulnerabilities in facial recognition systems related to spoofing attacks. The use of YOLOv8 ensured that the system maintained real-time performance without compromising detection capabilities.

REFERNCES

1. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767. Retrieved from https://arxiv.org/abs/1804.02767
2. Jocher, G. (2023). Ultralytics YOLOv8 Documentation. Ultralytics. Retrieved from https://docs.ultralytics.com/
3. OpenCV Development Team. (2023). OpenCV: Open Source Computer Vision Library. Retrieved from https://opencv.org/
4. Suleyman, E. (2021). cvzone: Computer Vision Zone Library. GitHub Repository. Retrieved from https://github.com/cvzone/cvzone
5. Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. International Conference on Learning Representations. Retrieved from https://arxiv.org/abs/1412.6980
6. Chingovska, I., Anjos, A., & Marcel, S. (2012). On the Effectiveness of Local Binary Patterns in Face Anti-spoofing. In 2012 BIOSIG - Proceedings of the International Conference of Biometrics Special Interest Group (pp. 1-7). IEEE