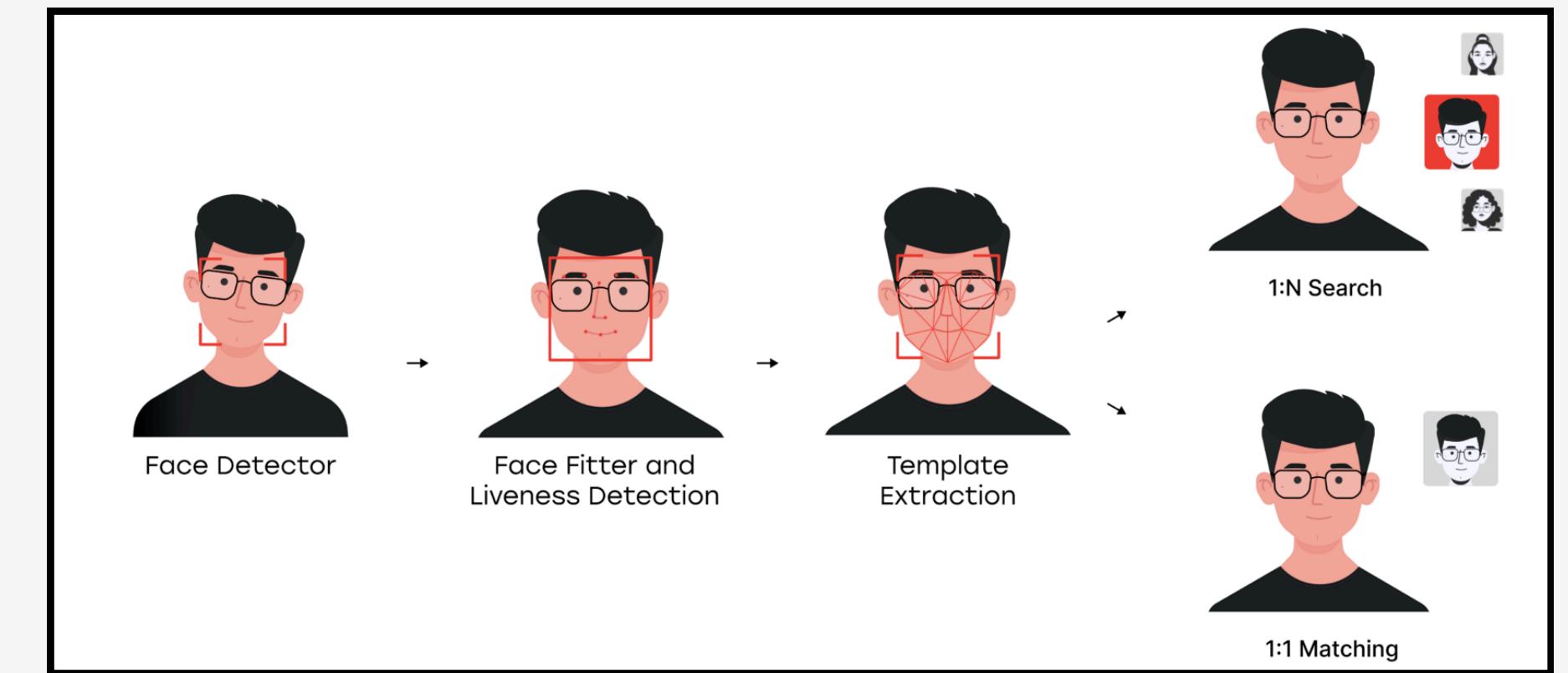
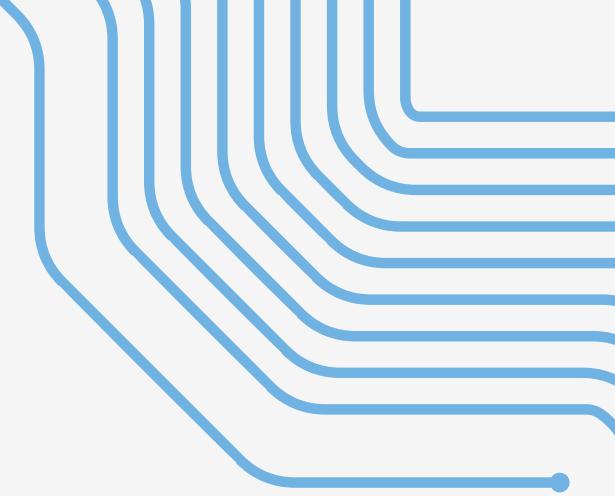


Face Detection

Using computer vision



By: Habiba yasser



Presentation Outline

- In this presentation, I will discuss
 - Introduction to Face Detection
 - Install Required Libraries (YOLO)
 - Accuracy improvement
 - Evaluation
 - Testing





Introduction

Face detection is a computer vision technique used to identify and locate human faces in images or videos and used in real-world apps such as security systems smartphones and surveillance cameras. I chose this project to understand how face detection works and to apply computer vision techniques using a YOLO-based model.

Accuracy improvement

Using a YOLO-based pre-trained model significantly improved the accuracy of face detection. YOLO processes the entire image at once which allows it to detect faces more accurately and efficiently compared to traditional methods.

The model achieved an accuracy of approximately 81% based on evaluation metrics after training

```
model.train(  
    data="/content/face_dataset.yaml",  
    epochs=10,  
    imgsz=640,  
    batch=8,  
    project="face_detection",  
    name="yolov8_face"  
)
```

```
'metrics/mAP50-95(B)': 0.8109642827154587
```

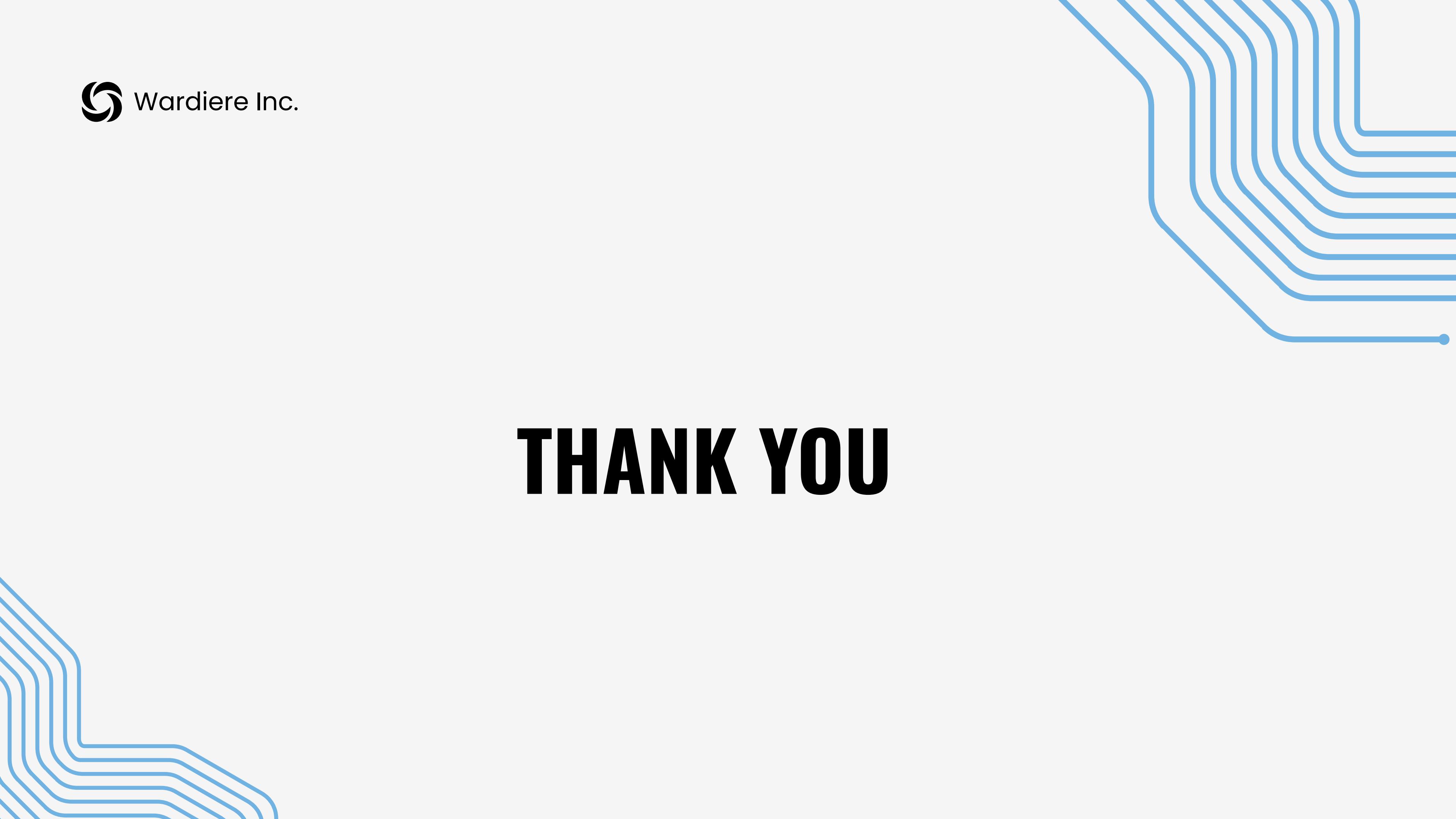
Testing Results

- The model was tested on unseen images after training and evaluating the model. The results show that the model successfully detected faces with good performance and accuracy.

```
results = model.predict(test_img, save=True)
results[0].show()
```

```
image 1/1 /content/facedetection..4.jpg: 352x640 14 Faces, 155.2ms
Speed: 9.9ms preprocess, 155.2ms inference, 1.3ms postprocess per image at shape (1, 3, 352, 640)
Results saved to /content/runs/detect/predict
```



A decorative graphic element consisting of several thin, light blue lines that curve and overlap, creating a wavy, dynamic feel. One line extends from the bottom left corner towards the center, while others form a curved arc from the top right corner towards the center.

THANK YOU