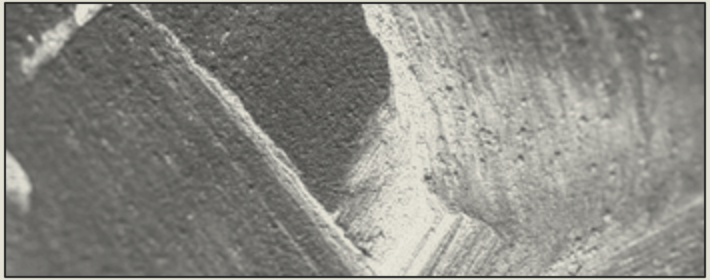
TRANSPORTATION:

ENHANCING ROAD SAFETY THROUGH A DRIVER FATIGUE DETECTION SYSTEM

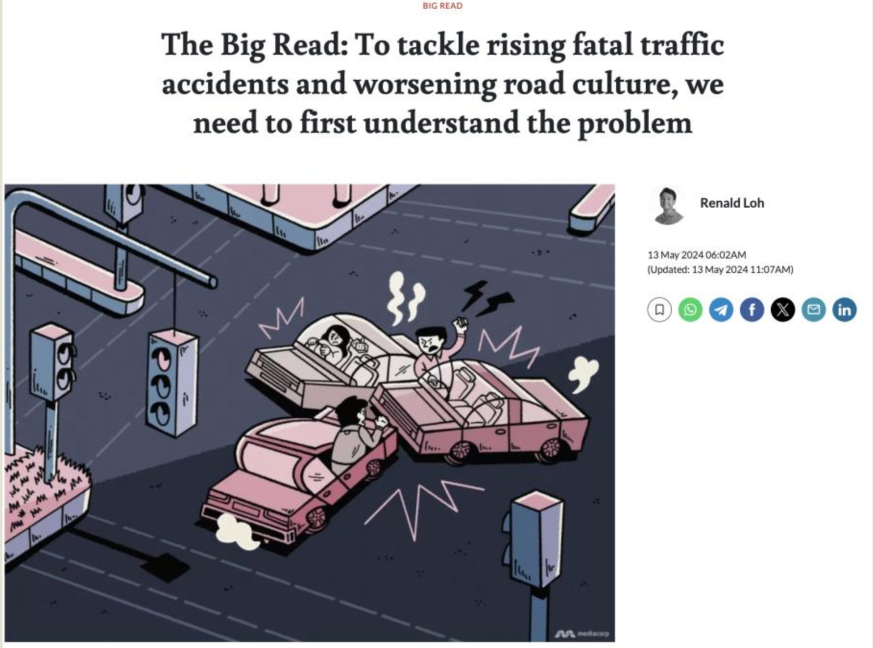
Track A. Design Track
Check Off 3

Team 6
Mubaraquali Muhammed Sufyanali (1006394)
Sean Phay Wei Xiang (1005969)
Stefanie Tan Hui Zhen (1006073)
Ong Zheng Han (1005867)



01 PROBLEM

Background Information



BIG READ

The Big Read: To tackle rising fatal traffic accidents and worsening road culture, we need to first understand the problem

Renald Loh

13 May 2024 06:02AM
(Updated: 13 May 2024 11:07AM)

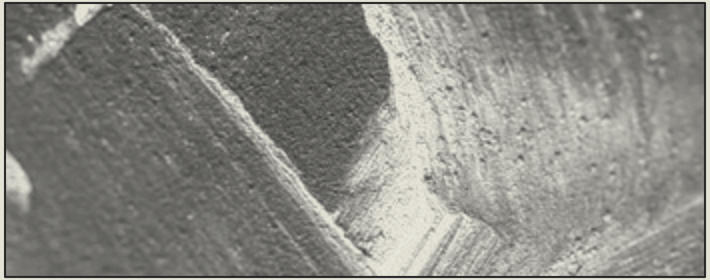
Driver fatigue impairs reaction time and decision-making, increasing accident risks.

Traffic fatalities **rose** from 104 (2022) to 131 (2023) in Singapore.

32.9% of Singaporean taxi drivers report fatigue while driving.

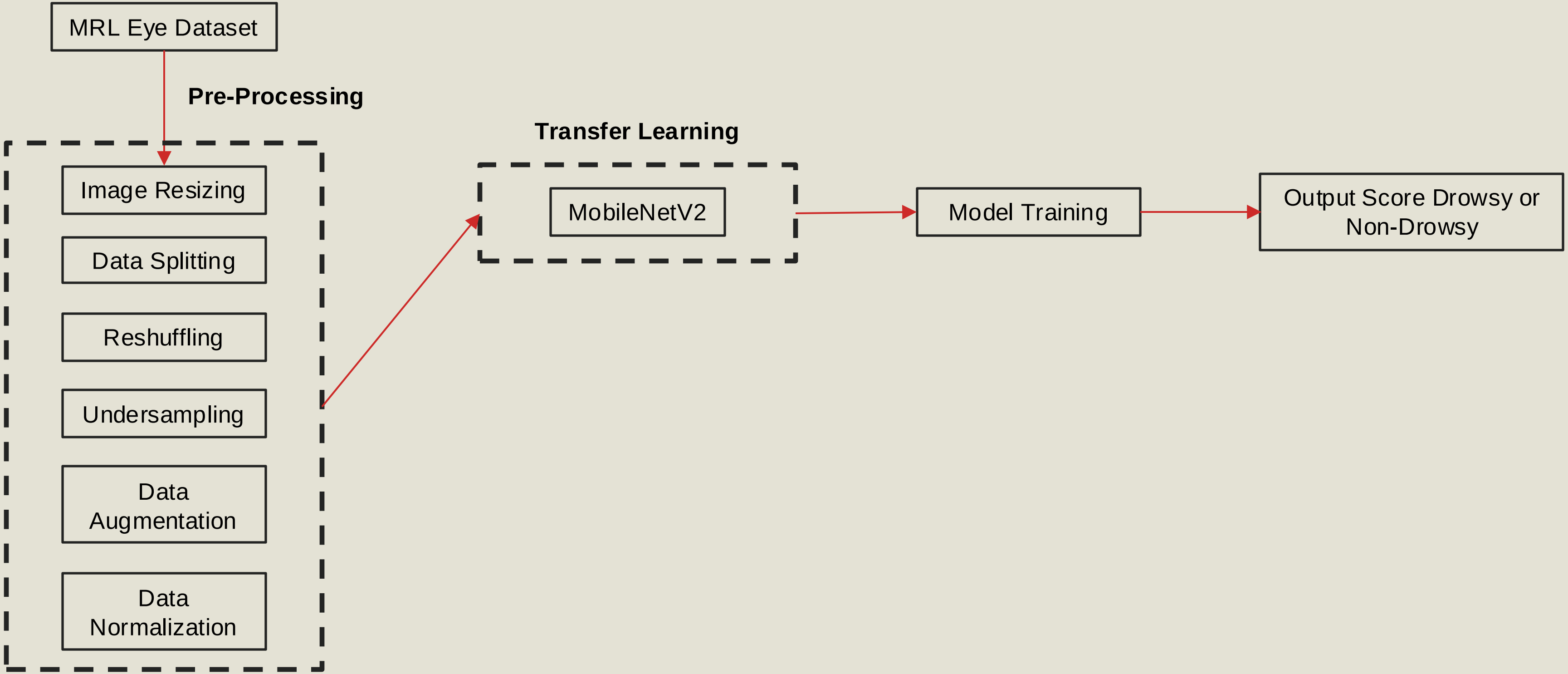
Problem Statement

Drivers often **underestimate their fatigue levels**, increasing accident risks. This project aims to create a real-time fatigue detection system to alert drivers, **preventing incidents** and **improving road safety**.



02 SYSTEM DESIGN & METHODOLOGY

CNN Model Architecture



MRL Eye Dataset

Pre-Processing

Image Resizing

Data Splitting

Reshuffling

Undersampling

Data Augmentation

Data Normalization

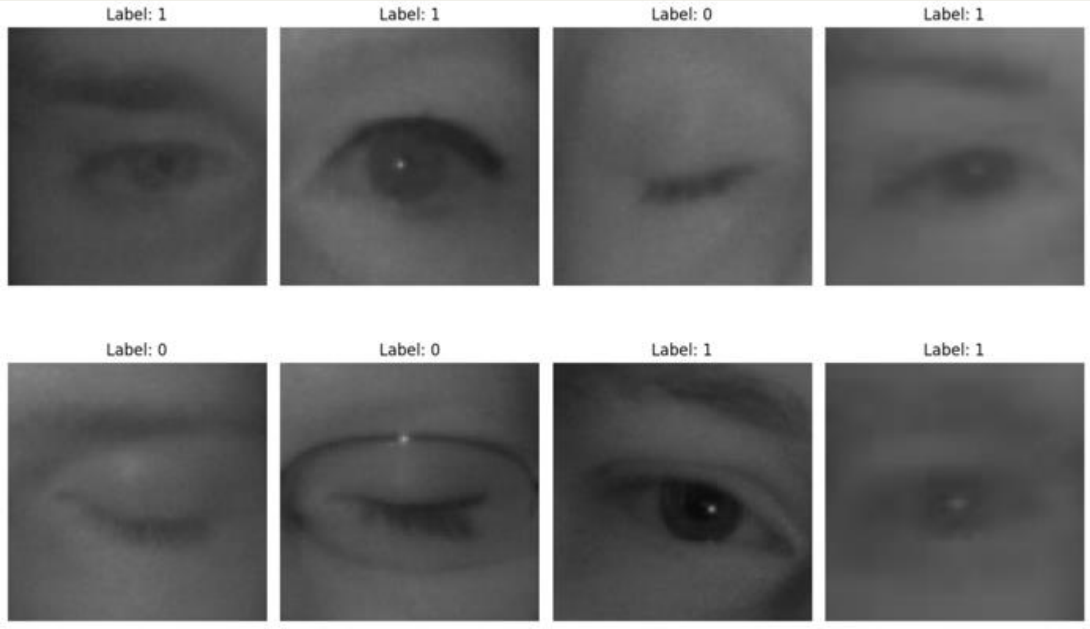
Transfer Learning

MobileNetV2

Model Training

Output Score Drowsy or Non-Drowsy

Dataset Pre-Processing



Sample of original images from Kaggle with their respective labels

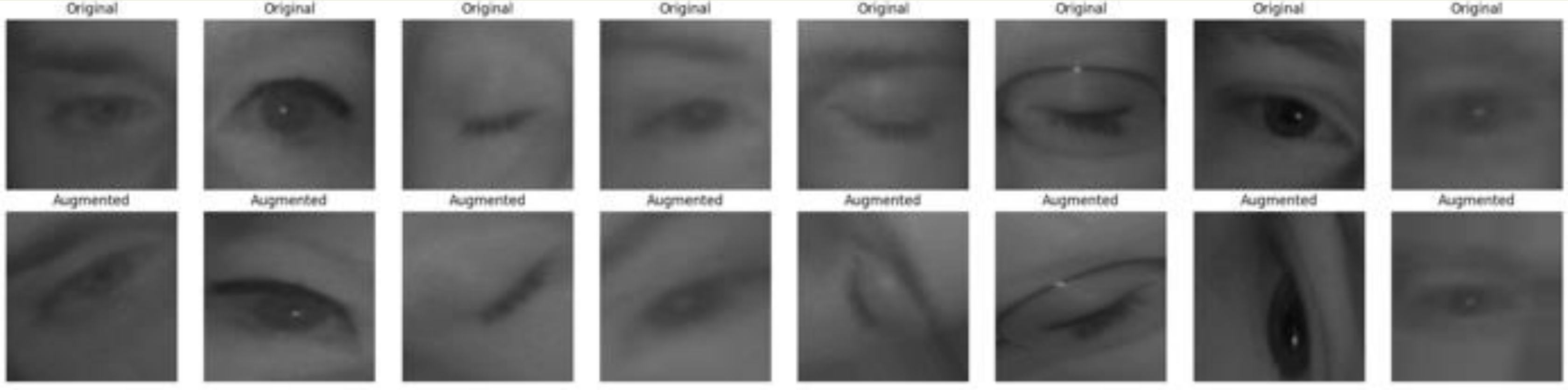
Source: Kaggle’s MRL Eye Dataset

Content: 84,00 images (grayscale, 8-bit)

- Categorized as **“Open-Eyes”** and **“Closed-Eyes”**
- Labels indicate **drowsy (1)** or **non-drowsy (0)** states

Challenges: Variations in lighting, eye positions, and individual differences

Dataset Pre-Processing



Sample of original images compared to augmented images

Resizing: Images standardized **to 224x224 pixels**

Dataset Split:

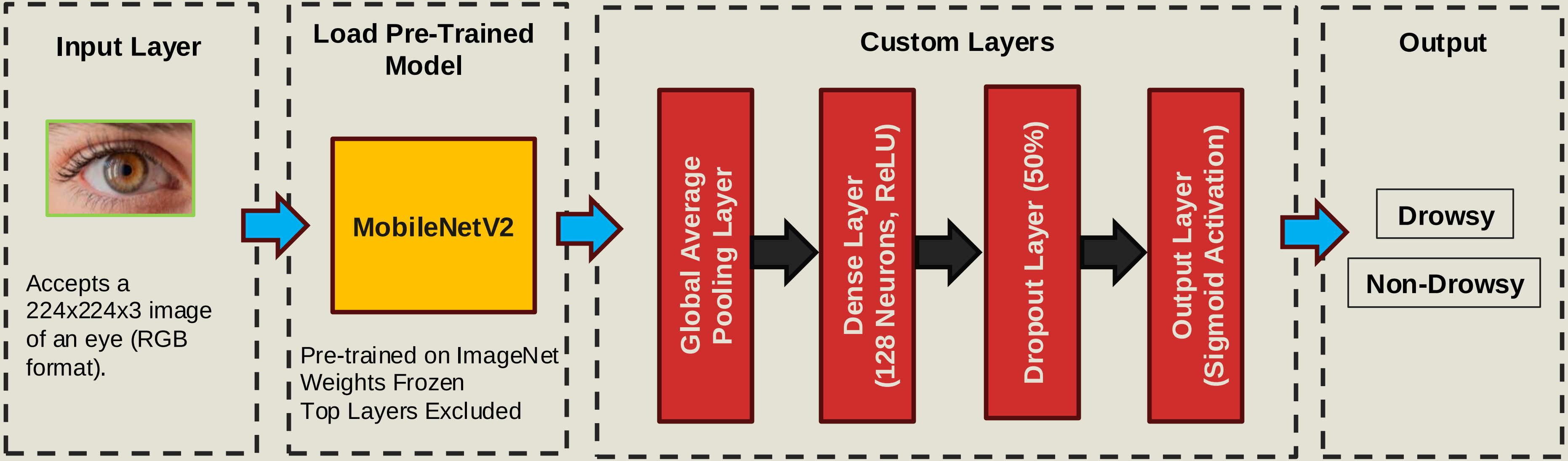
- 70% training
- 15% validation
- 15% testing

Data Augmentation:

- Rotations
- Flips
- Shifts
- Zooms
- Shearing

MobileNetV2 Model

Transfer Learning



Input Layer

Accepts a 224x224x3 image of an eye (RGB format).

Load Pre-Trained Model

MobileNetV2

Pre-trained on ImageNet
Weights Frozen
Top Layers Excluded

Custom Layers

Global Average Pooling Layer

Dense Layer (128 Neurons, ReLU)

Dropout Layer (50%)

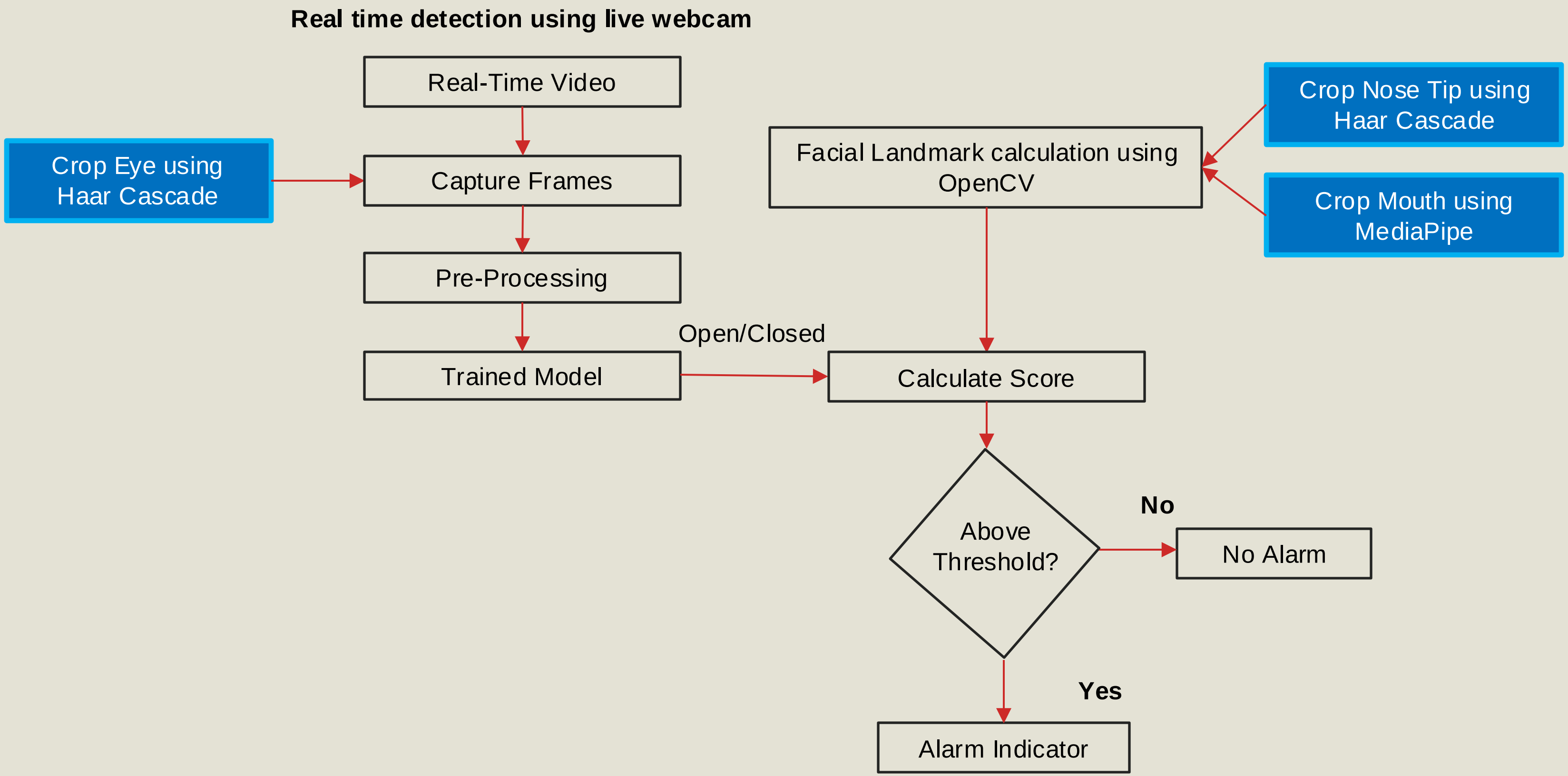
Output Layer (Sigmoid Activation)

Output

Drowsy

Non-Drowsy

Prototype (OpenCV) Architecture


Real time detection using live webcam
Real-Time Video
Capture Frames
Pre-Processing
Trained Model
Crop Eye using Haar Cascade
Open/Closed
Facial Landmark calculation using OpenCV
Crop Nose Tip using Haar Cascade
Crop Mouth using MediaPipe
Calculate Score
Above Threshold?
No
No Alarm
Yes
Alarm Indicator

Fatigue Detection – Eye State Classification

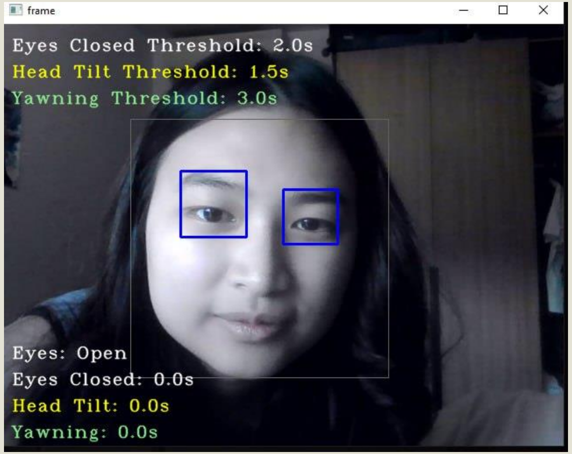
Model Overview:

- Utilizes the **pre-trained MobileNetV2** model.
- Fine-tuned on the **MRL Eye Dataset** for accurate eye state classification.

Classification Process:

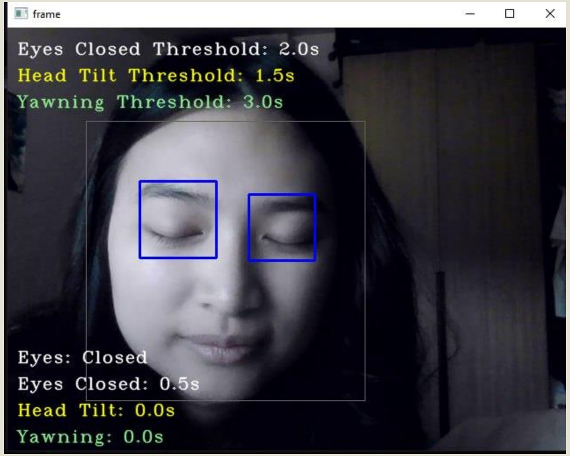
- Eye images are classified as **"open"** or **"closed"**.
- Analysis is performed **separately for each eye**.

Open



```
Eyes Closed Threshold: 2.0s
Head Tilt Threshold: 1.5s
Yawning Threshold: 3.0s

Eyes: Open
Eyes Closed: 0.0s
Head Tilt: 0.0s
Yawning: 0.0s
```



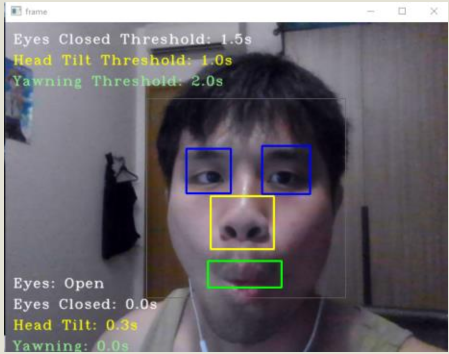
```
Eyes Closed Threshold: 2.0s
Head Tilt Threshold: 1.5s
Yawning Threshold: 3.0s

Eyes: Closed
Eyes Closed: 0.5s
Head Tilt: 0.0s
Yawning: 0.0s
```

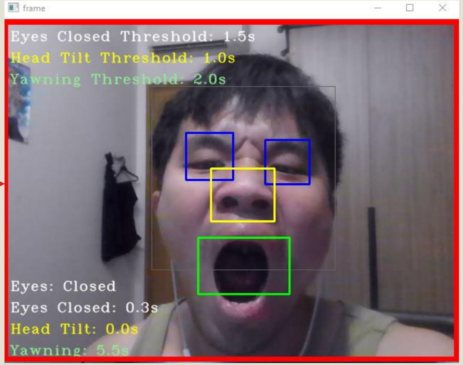
Closed

Fatigue Detection – Yawning

Mouth Closed



```
Eyes Closed Threshold: 1.5s
Head Tilt Threshold: 1.0s
Yawning Threshold: 2.0s
Eyes: Open
Eyes Closed: 0.0s
Head Tilt: 0.3s
Yawning: 0.0s
```



```
Eyes Closed Threshold: 1.5s
Head Tilt Threshold: 1.0s
Yawning Threshold: 2.0s
Eyes: Closed
Eyes Closed: 0.3s
Head Tilt: 0.0s
Yawning: 5.5s
```

Mouth Open

Model Overview:

- Uses Google’s Mediapipe FaceMesh model to detect facial landmarks, focusing on points around the mouth.
- Extracts **Mouth Aspect Ratio (MAR)** to measure mouth openness.

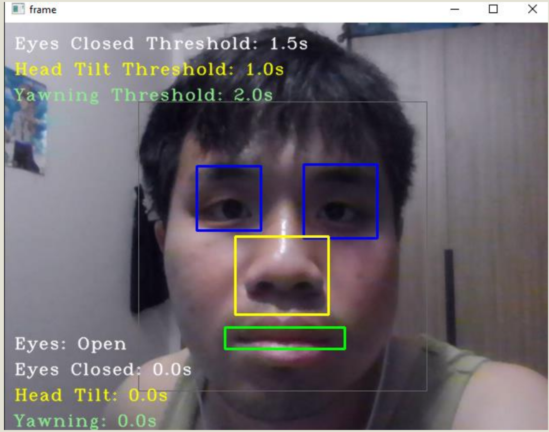
$$\mathrm{MAR} = \frac{(verticle1 + verticle2)}{2 \times horizontal}$$

Classification Process:

- **MAR** is calculated using distances between vertical and horizontal mouth landmarks
- **High** MAR Value -> **Yawning** -> **Fatigue**

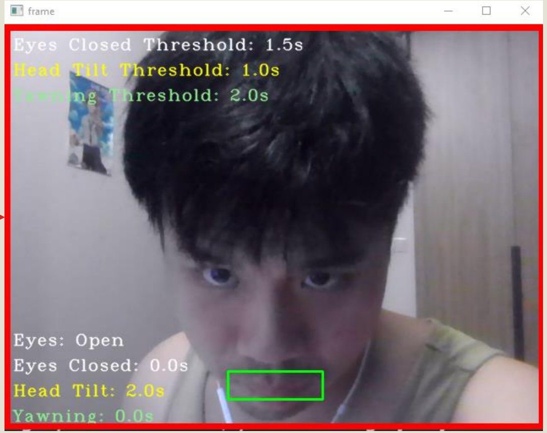
Fatigue Detection – Head Tilt

No Head tilt



```
Eyes Closed Threshold: 1.5s
Head Tilt Threshold: 1.0s
Yawning Threshold: 2.0s

Eyes: Open
Eyes Closed: 0.0s
Head Tilt: 0.0s
Yawning: 0.0s
```



```
Eyes Closed Threshold: 1.5s
Head Tilt Threshold: 1.0s
Yawning Threshold: 2.0s

Eyes: Open
Eyes Closed: 0.0s
Head Tilt: 2.0s
Yawning: 0.0s
```

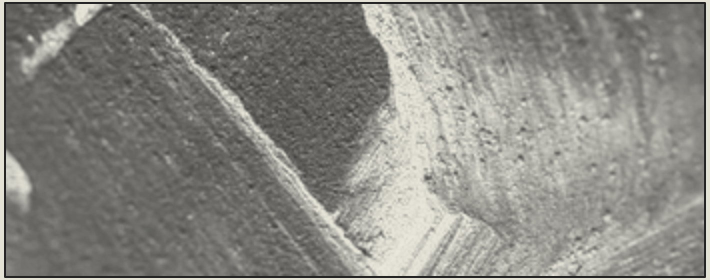
Head tilt

Model Overview:

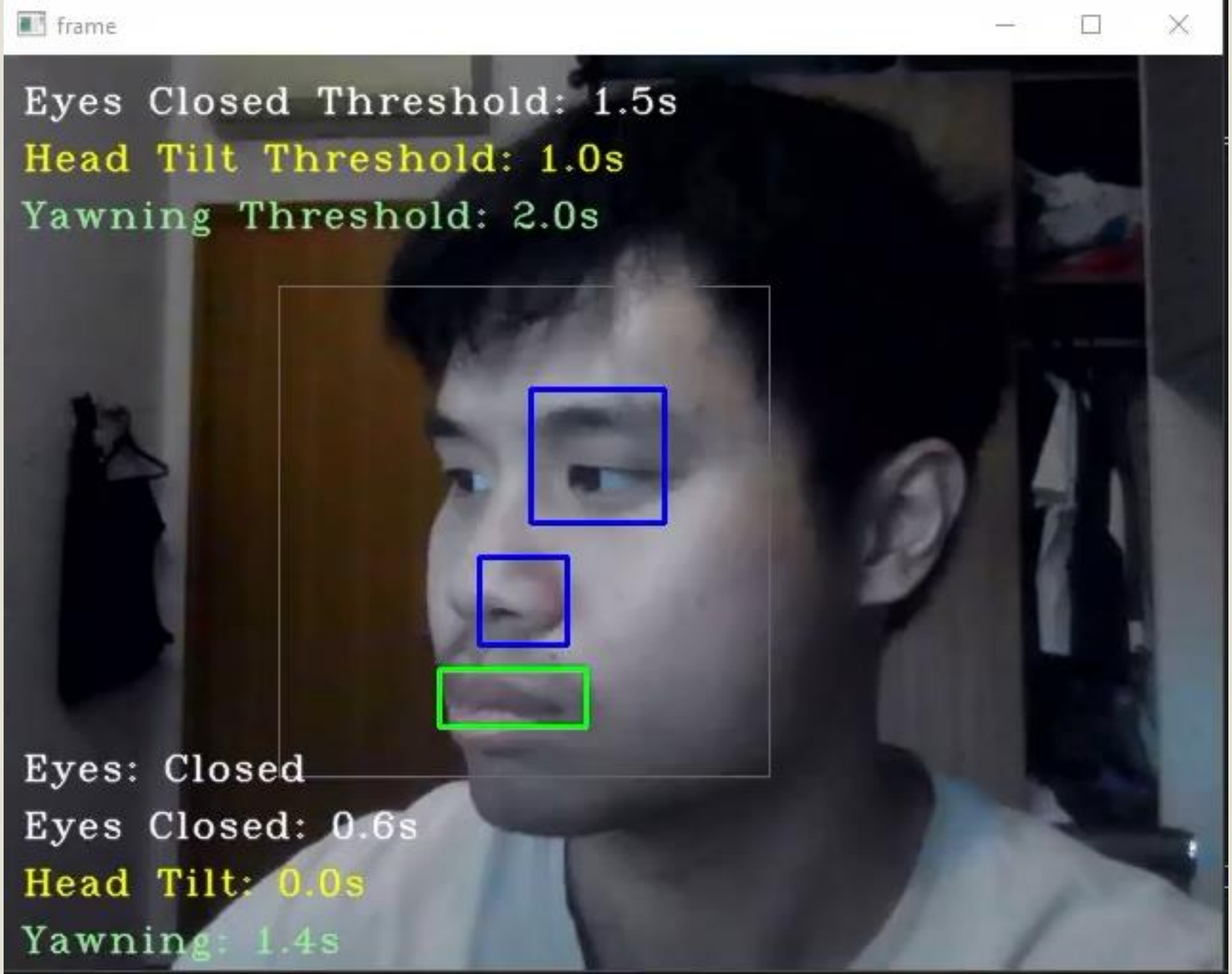
- Uses Haar Cascade for detecting the face and nose in the grayscale frame
- Calculates **y-coordinate** of the nose tip to determine head tilt angle

Classification Process:

- A significant change in the y-coordinate of the nose tip indicates a **Head Tilt** -> **Fatigue**

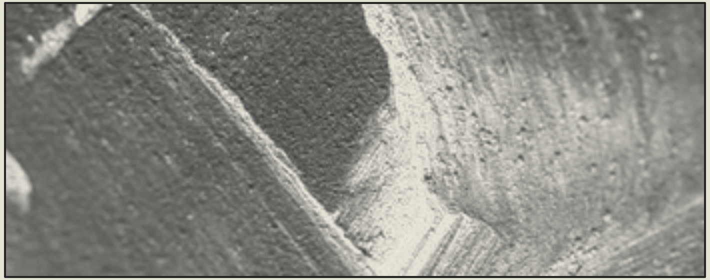


03 PROTOTYPE DEMOSTRATION



```
Eyes Closed Threshold: 1.5s
Head Tilt Threshold: 1.0s
Yawning Threshold: 2.0s

Eyes: Closed
Eyes Closed: 0.6s
Head Tilt: 0.0s
Yawning: 1.4s
```



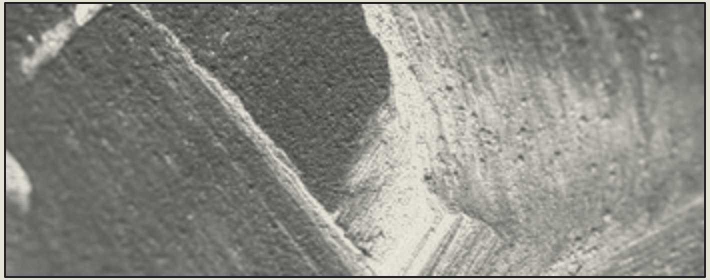
04 EXPERIMENT RESULTS

Training Results

Epoch 1	Training Accuracy: **87.24%** Validation Accuracy: **91.52%** Validation Loss: **0.2021**
Epochs 2–7	Minimal improvements on the validation dataset; **early stopping** triggered at epoch 7.
Final Evaluation	**Test Accuracy**: **91.24%**, shows robust generalization. **Benefits**: Balanced performance and computational efficiency while minimizing overfitting.

Conclusion

This systematic approach ensures reliable performance on unseen data with optimized training efficiency.



05 FUTURE WORKS

Future Work: Enhancing the Drowsiness Detection System

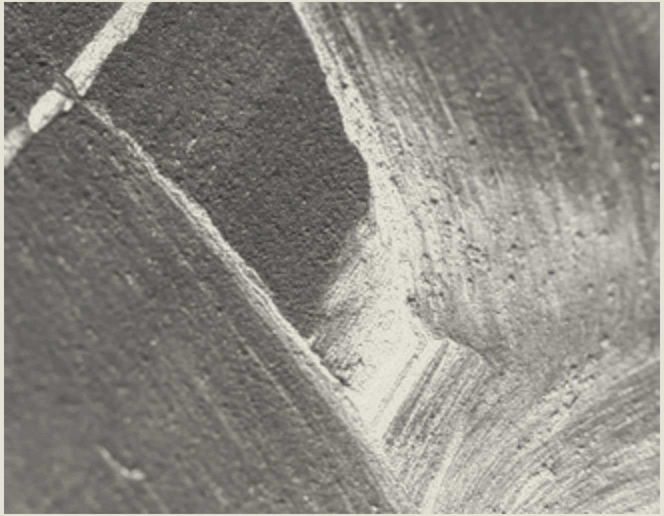
Mobile and Embedded Device Deployment

- Optimize the system for portable platforms (e.g., **Raspberry Pi**) to enable in-vehicle integration.
- Focus on reducing computational overhead while maintaining accuracy for real-world use.

Real-Time Alert System Enhancements

- Develop a more adaptive alert system:
- **Yawning Timeframe**: If the user yawns for a repeated number of times withing the timeframe → Activates the alarm
- **Verbal warnings**: Provide prompts like “Please pull over and rest.”
- **Driver feedback**: Recommend breaks at regular intervals.

Goal: Improve system functionality, portability, and real-world impact for safer and more reliable driver monitoring.



THANK YOU!