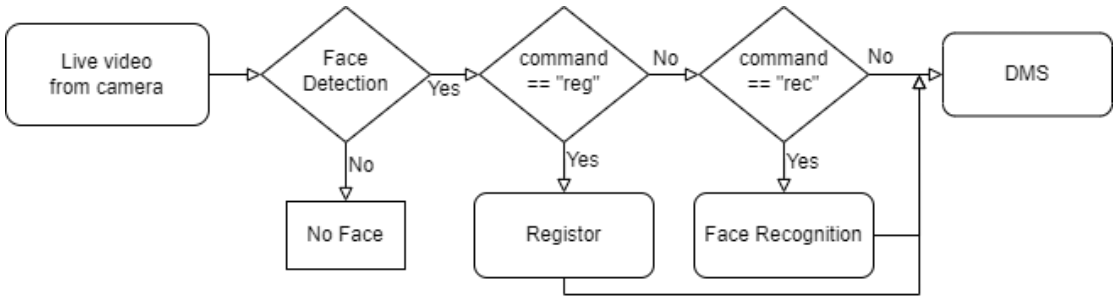
DMS

1. Workflow Diagram



2. Driver Actions to be Detected

(1). Close Eye
(2). Yawn
(3). Lower head
(4). Distraction

3. Model Preparation

3.1 shape_predictor_68_face_landmarks.dat

This is trained on the ibug 300-W dataset. The license for this dataset excludes commercial use and Stefanos Zafeiriou, one of the creators of the dataset, asked me to include a note here saying that the trained model therefore can't be used in a commerical product. So you should contact a lawyer or talk to Imperial College London to find out if it's OK for you to use this model in a commercial product.

Also note that this model file is designed for use with dlib's HOG face detector. That is, it expects the bounding boxes from the face detector to be aligned a certain way, the way dlib's HOG face detector does it. It won't work as well when used with a face detector that produces differently aligned boxes, such as the CNN based mmod_human_face_detector.dat face detector.

3.2 dlib_face_recognition_resnet_model

This model is a ResNet network with 29 conv layers. It's essentially a version of the ResNet-34 network from the paper Deep Residual Learning for Image Recognition by He, Zhang, Ren, and Sun with a few layers removed and the number of filters per layer reduced by half.

The network was trained from scratch on a dataset of about 3 million faces. This dataset is derived from a number of datasets. The face scrub dataset, the VGG dataset, and then a large number of images I scraped from the internet. I tried as best I could to clean up the dataset by removing labeling errors, which meant filtering out a lot of stuff from VGG. I did this by repeatedly training a face recognition CNN and then using graph clustering methods and a lot of manual review to clean up the dataset. In the end about half the images are from VGG and face scrub. Also, the total number of individual identities in the dataset is 7485. I made sure to avoid overlap with identities in LFW.

The network training started with randomly initialized weights and used a structured metric loss that tries to project all the identities into non-overlapping balls of radius 0.6. The loss is basically a type of pair-wise hinge loss that runs over all pairs in a mini-batch and includes hard-negative mining at the mini-batch level.

The resulting model obtains a mean error of 0.993833 with a standard deviation of 0.00272732 on the LFW benchmark.

4. Customization Parameters

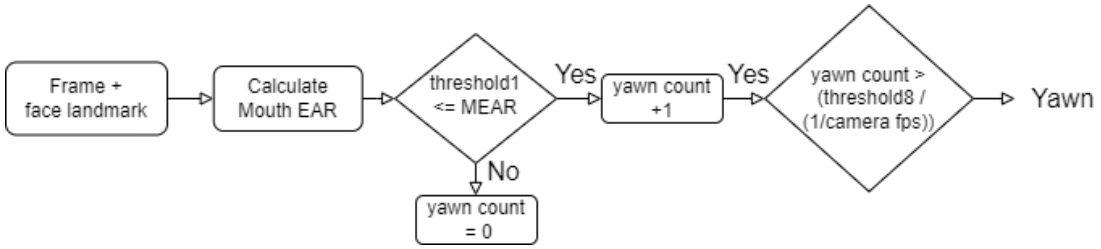
Parameter	Definition	Default
Image frame width	INPUT_COL	640
Image frame height	INPUT_ROW	360
Camera FPS	fps_lim	12
Threshold 1	mear_tresh	
Threshold 2	ear_tresh	
Threshold 3	yaw_tresh	
Threshold 4	head_basic	
Threshold 5	head_moveY	45
Threshold 6	avg_pitch	
Threshold 7	pitch_tresh	27
Threshold 8	mear_time_tresh	2
Threshold 9	ear_time_tresh	3
Threshold 10	yaw_time_tresh	1
Threshold 11	pitch_time_tresh	2
Record result	record	0
Select ROI (0:left, 1:middle, 2:right)	selectROI	1
Get frames to calculate normal state thresholds *	LAG	30

* We will get LAG frames to calculate Threshold 1、2、3、4、6 in normal face state.

5. Detailed DMS Flow

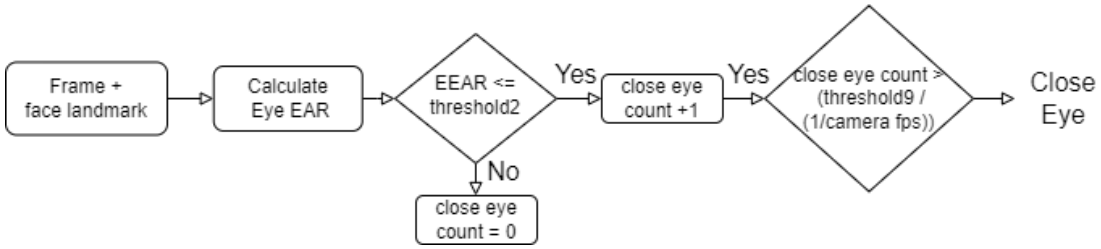
5.1 Yawn

程式開始時，會先利用 LAG 個 frame 計算 threshold1，即嘴巴的平均 EAR。當 Mouth EAR > threshold1，則 Yawn count +1。若 Yawn count >= threshold8 / (1/camera fps) ，則判斷駕駛正在打哈欠。



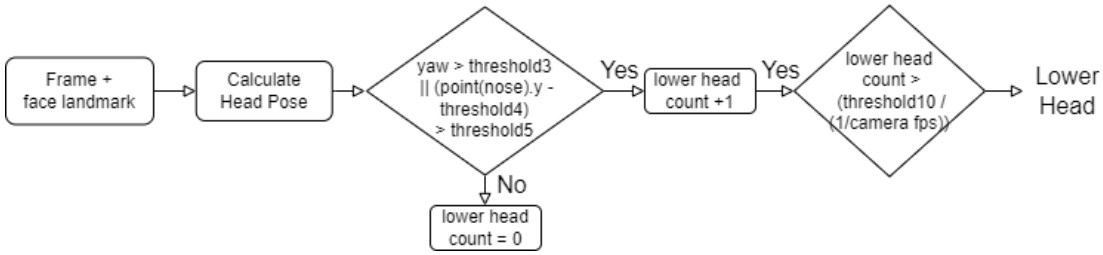
5.2 Close Eye

程式開始時，會先利用 LAG 個 frame 計算 threshold2，即雙眼的平均 EAR。當 Eye EAR > threshold2，則 Close Eye count +1。若 Close Eye count >= threshold9 / (1/camera fps) ，則判斷駕駛正在閉眼。



5.3 Lower Head

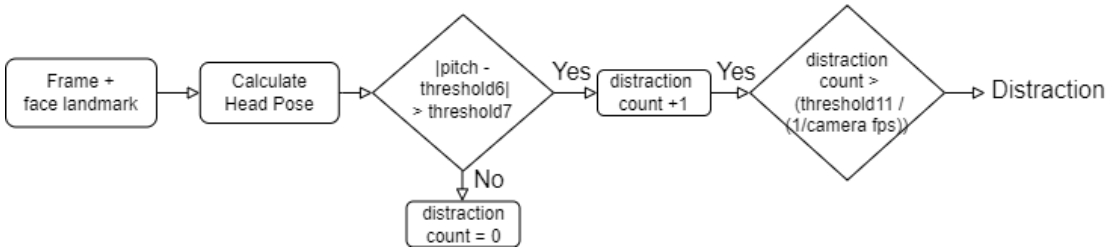
程式開始時，會先利用 LAG 個 frame 計算 threshold3、4，即頭的垂直平均角度和鼻子的點的平均高度；threshold5 為頭的 y 方向位移的距離。當頭的垂直角度 yaw > threshold3 或鼻子的點的高度- 平均高度 threshold4 > 位移距離 threshold5; Lower Head count +1。若 Lower Head count >= threshold10 / (1/camera fps) ，則判斷駕駛正在低頭。



5.4 Distraction

程式開始時，會先利用 LAG 個 frame 的計算 threshold6，即頭的平均水平角度；threshold7 為

水平旋轉角度。當頭的水平角度 Pitch – 平均水平角度 threshold6 > 水平旋轉角度 threshold7，則 Distraction count +1。若 Distraction count >= threshold11 / (1/camera fps) ，則判斷駕駛正在分心。



6. Evaluation

Testing of the application will be evaluated on below scenarios.

(1). Evaluation with live video from webcam on PC.

(2). Evaluation with live video from camera on V3H2.

7. Target Performance

Performance in R-Car V3H2：13 frames per second.

8. References

(1). Tutorial 1：Link

(2). Tutorial 2：Link

(3). Tutorial 3：Link

(4). Tutorial 4：Link

(5). Yawn Detection：Link

(6). Drowsiness detection：Link

(7). Real-Time Eye Blink Detection using Facial Landmarks：Link

(8). Head Pose Estimattion：Link

(9). OpenCV Head Pose computation overview：Link