Frame Level Driver Drowsiness Prediction

Matjaž Zupančič Muc

I. Introduction

Driver drowsiness detection is a vital area of research in computer vision, aiming to address the pressing issue of drowsy driving and enhance road safety. The ability to accurately identify driver drowsiness in real-time has significant implications for accident prevention and driver alertness monitoring systems. With this in mind we implement a frame level driver drowsiness detection pipeline (i.e predictions about the driver's state are made based on a single frame). Predicting the driver's state at a single frame offers several benefits, including: real-time processing, efficient computation and cost-effectiveness. However the approach also has several downsides, such as limited temporal context (driver behavior and states can evolve over time, and a single-frame prediction may not provide a comprehensive understanding of the driver's ongoing state or intentions), false positives (the system trained at the frame level is more likely to make false positive predictions, e.g the system may mistake a blink for a microsleep session). In this study, our objectives are as follows: (1) Creating a custom dataset called FL3D, (2) Fine-tuning the ResMaskNet model, originally trained for Facial Expression Recognition, on the FL3D dataset, (3) Conducting a thorough evaluation of the model's performance and analyzing its predictions, (4) Implementing the complete prediction pipeline and evaluating its effectiveness.

II. Related Work

Several studies have been conducted to explore different approaches and techniques for driver drowsiness detection. The standard approaches rely on detecting the eye and the mouth region, and make predictions based on hand-crafted features (e.g the area of the mouth region, the Eye Aspect Ratio, etc). This idea is explored by the authors of [1]. The standard approaches have the following drawbacks: limited representation (hand-crafted features may not capture the full complexity and variability of facial expressions or driver states), sensitivity to variations (hand-crafted features may not generalize well across different individuals or conditions), engineering thresholds (optimal decision thresholds are often tailored to specific datasets), difficulty in capturing complex interactions (facial expressions and driver states are often influenced by complex interactions among different facial regions and features). The authors of [2] explore driver drowsiness detection by applying a recurrent and convolutional neural network to a sequence of frames.

III. Methodology

A. FL3D Dataset

We construct a custom dataset called the Frame Level Driver Drowsiness Detection (FL3D). FL3D was constructed using the NITYMED dataset (Night-Time Yawning-Microsleep-Eyeblink-driver Distraction), which was open-sourced by the authors of [3]. NITYMED consists of 130 videos displaying drivers in real cars, moving under nighttime conditions. The videos have been captured by an in-car camera mounted on the dash of the car. The participating drivers are: 11 males and 10 females with different features (hair color, beard, glasses, etc). The videos are split in 2 categories: Yawning, the drivers yawn 3 times in each video lasting approximately 15-25 seconds (107 videos). Microsleep, the drivers talk, look around and have microsleeps in videos lasting approximately 2 minutes (21 videos).

NITYMED dataset is intended for evaluation of drowsiness detection algorithms in addition to face, mouth and eye tracking applications. We should point out that the dataset doesn't have frame level ground truth labels, and thus can't be used for supervised learning at the frame level. This is where our dataset FL3D comes into play. We decided to label each frame in the NITYMED dataset, with one of the following three labels: alert, microsleep. yawning. We also deploy the retinaface ([4]) model to detect the drivers face, and the facial landmarks at each frame. Automatically detected bounding boxes are checked and corrected by a hand. When constructing the FL3D dataset we posed the following assumptions:

1) All frames corresponding to the microsleep session where the driver has closed eyes are labeled with microsleep. In theory one could experience a microsleep session with opened eyes, how ever such frames were not added to the FL3D dataset.
2) All frames corresponding to the yawning session where the drivers mouth is opened for more than rouhgly 2.5 centimeters were labeled with yawning.
3) Frames in which drivers blink are removed and not included in FL3D.

Final annotated and cleaned FL3D dataset consists of 53331 images. Since our goal is to build a model which can generalized across new drivers, the validation and test set is constructed with that in mind. The validation and test set include 3 females and 2 males, among which the 2 females and 1 male are completely new drivers, not present in the training set. Figure 1 shows a set of random frames from the training set, while Figure 2 shows a set of random frames from the validation set.

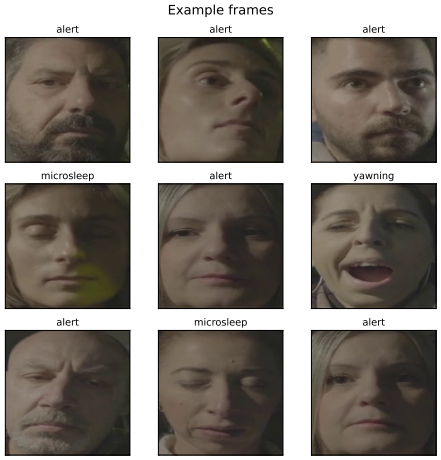


Figure 1: FL3D random subset of training images.

Figure 3 shows a histogram of labels in the FL3D training set (similar distribution is present in the validation and test set) We can see that most of the images belong to the alert class, as opposed to microsleep and yawning. Figure 4 shows

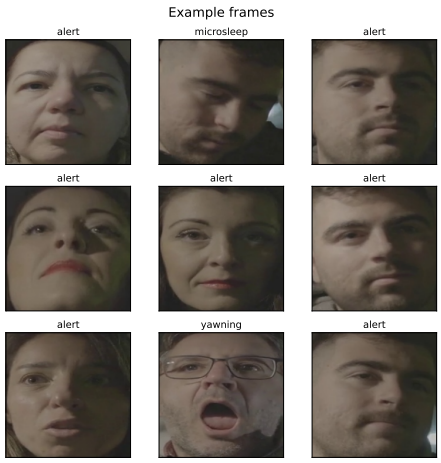


Figure 2: FL3D random subset of validation images.

the distribution of red, green and blue color channels in the training set. We can see that the distribution is skewed towards lower intensity values. FL3D dataset was open-sourced and is available for download:https://www.kaggle.com/datasets/matjazmuc/frame-level-driver-drowsiness-detection-fl3d.

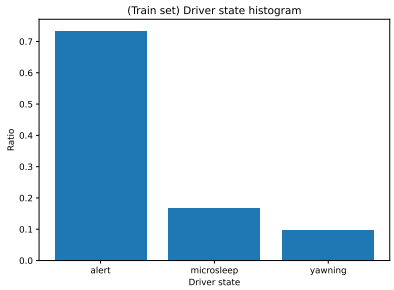


Figure 3: FL3D (Train set) label histogram

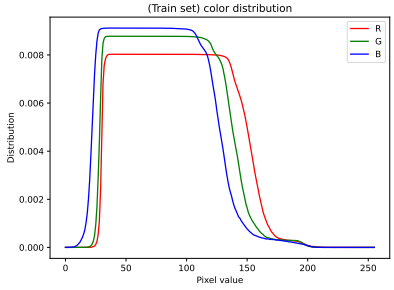


Figure 4: FL3D (Train set) color distribution.

B. Residual Masking Network

We chose to utilize the Residual Masking Network (ResMaskNet) proposed in [5], originally trained for Facial Expression Recognition (FER), to address the task of driver drowsiness prediction. FER involves identifying facial expressions from images, with key facial information located in the eye and mouth regions. The ResMaskNet incorporates a U-NET based localization network to enhance relevant features and assign higher weights to relevant locations in the feature maps, effectively highlighting important regions for accurate predictions. Using transfer learning we fine-tune the ResMaskNet (pretrained for FER) on the FL3D dataset.

C. Addressing class imbalance and data-augmentation

In order to artificially increase the size of our training set, we used the following data transformations: Apply random rotation, with probability $p = 0.7$ in range $\Delta_\theta \in [-20, 20]$. Apply a horizontal flip, with $p = 0.3$. Apply a RGB Shift with $p = 0.75$, in range $\Delta_R \in [-5, 5], \Delta_G \in [-5, 5], \Delta_B \in [-5, 5]$. Induce a random brightness and contrast change with $p = 0.5$, in range $\Delta_b \in [-0.1, 0.1]$ and $\Delta_c \in [-0.1, 0.1]$. The values were chosen experimentally, i.e we observed the effect of each transformation and decided on reasonable thresholds which still produce data which could be found in the real world. We address the class imbalance problem by over-sampling the minority classes, i.e we sample each class proportional the inverse of its probability. Each time we sample data-augmentation is applied, to insure variability between samples, and prevent overfitting.

D. Prediction Pipeline

Additionally we implement an entire Prediction Pipeline (the entire implementation is available here: https://github.com/Matjaz12/Driver-Drowsiness-Prediction). The pipeline includes a face detection module, detection suppressor module and drowsiness prediction module. The pipeline is visualized on Figure 5. We can see that the pipeline takes in a single frame, detects the driver's face and predicts the current state of the driver. The face detection module implements the retinaface detector. The purpose of the detection suppressor module is to select the bounding box that corresponds to the driver's face from multiple bounding boxes detected in the input image (shown in Figure 5). Since there may be multiple individuals in the frame, including passengers and others, we need to eliminate bounding boxes that do not belong to the driver. To achieve this, we employ a straightforward heuristic. Given that the retinaface detector predicts K bounding boxes, we filter them by keeping only $k \leq K$ bounding boxes with a confidence level above $\tau = 0.5$. Among the remaining k bounding boxes, we designate the one with the largest area as the bounding box for the driver. We crop the image around the driver's bounding box and feed to our drowsiness prediction model. The final predictions could be filtered by the end application. Since the FL3D dataset doesn't include frames where the drivers are blinking, our trained model is likely to assign a microsleep label to a blinking frame. The predictions could be filtered by defining the width of the window, at which we make predictions, i.e we would predict the most common label in the window, and effectively remove the false predictions.

IV. Experiments

A. Model Evaluation

We fine-tune the ResMaskNet (pretrained for FER) on the FL3D dataset. We experimented with hyper-parameters and settled on the following: batch size $B = 128$, learning rate $\alpha = 3.5e-3$, weight decay: $1e-4$. The model is trained for $N = 15$ epochs by optimizing the Cross Entropy (CE) Loss using the Adam optimizer, and a learning rate scheduler, which multiplies the learning by $\gamma = 1/10$ every 5 epochs. During training we keep track of the macro F1-score and finally

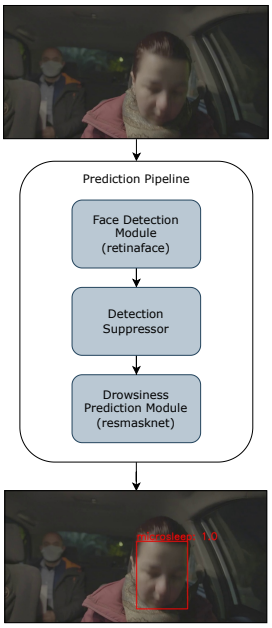
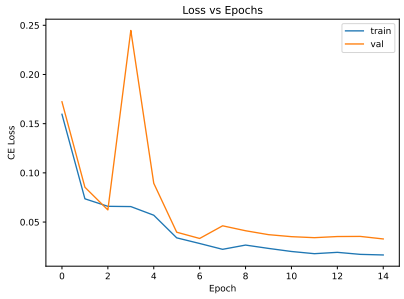
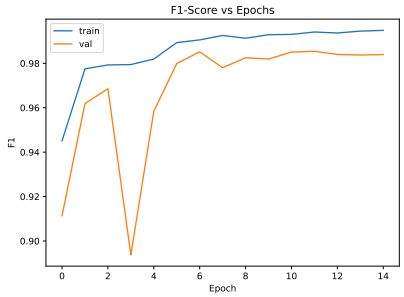


Figure 5: Prediction pipeline. Pipeline takes in a single frame, detects the driver's face and predicts the current state of the driver.

return the model with the highest F1-score on the validation set. Figure 6 shows training history (loss and Macro F1-score) on the train and validation set. We evaluate the model on the test set and obtain the following: macro precision $prec = 0.976$, macro recall $rec = 0.989$ and macro F1-score $F1 = 0.98$. Figure 7 shows the confusion matrix calculated on the test set.



(a) Train and validation loss vs epochs.



(b) Train and validation Macro F1-score vs epochs.

Figure 6: Training history.

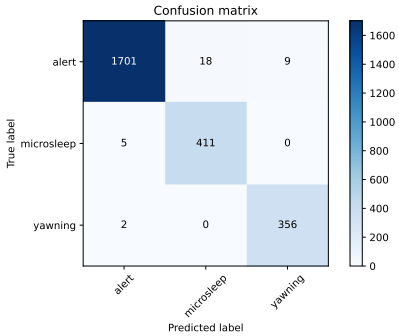


Figure 7: Confusion Matrix on the test set.

B. Prediction Analysis

Figure 8 shows a set of false predictions. We can see that the model struggles in cases where the driver has slightly shut eyes and or maintains a downward gaze. For example in the image in the first row and first column the driver is looking at the speedometer and thus appears to be sleeping, to the model. In the first row, second column the model predicts that the person is yawning, which is not incorrect, as this was indeed a part of a yawning session. This indicates that there are some ambiguities in the labels. In the first row, third column we can see that model mistakes the a microsleep session for an alert state, this is likely due to the fact the people with such pose were usually in a alert state (i.e they were looking out the window). In the second row and first column the driver is clearly laughing in an odd pose, while our model mistakes this for sleeping. In order to improve the performance of the model we would need additional images which would cover more variation of the pose and driver state appearance, e.g images of people laughing, looking at the speedometer, etc.

In order to gain additional insights into the workings of our algorithm, we utilize the guided GradCam method (proposed by the authors of [6]). Guided Grad-CAM is a technique used to visualize and interpret the decision-making process of a deep learning model. It combines the concepts of Grad-CAM and guided backpropagation to highlight the regions of an input image that are important for the model's prediction. By leveraging the gradient information from both the class activation maps and the guided backpropagation, Guided Grad-CAM provides detailed visual explanations of the model's attention and helps identify the salient features driving its decision-making. Figure 9 shows a set of Guided GradCam maps. In the first row the model predicts the alert state, we can see that the most important region corresponds to the opened eye. In the second row, first column the model predicts the a microsleep state, we can clearly see that it bases it decision based on region around closed eyes. In the last row, last column the model predict a yawning label, and deems pixels around opened mouth the most important.

C. Pipeline Evaluation

We evaluate the entire Prediction Pipeline on a out-of-distribution (OOD) dataset, retrieved from:https://universe.roboflow.com/augmented-startups/drowsiness-detection-cntmz/dataset/1. We should point out that the images are taken with a different camera, with a different resolution, at a different angle. Images are also taken during daylight, as opposed to images taken at night in the

Figure 8: Test set false predictions.

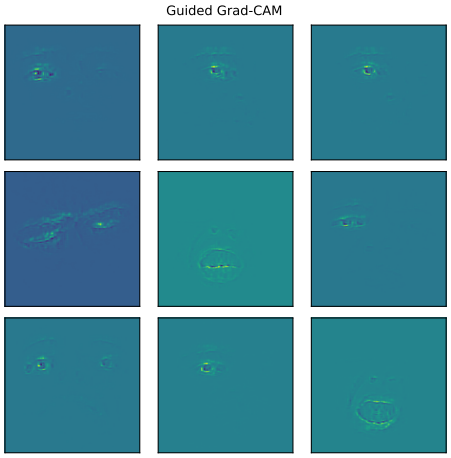


Figure 9: Guided GradCam maps of a random test subset.

FL3D dataset. We compute the mean Intersection over Union (mIOU) of the face detection and detection suppressor module, and compute the macro precision, recall and F1 score of the drowsiness prediction module. We obtain $mIOU = 0.52$ with 2 failures (i.e the face detection module failed to detect the driver twice), and macro $prec = 0.76$, $rec = 0.72$ and $F1 = 0.71$. We should point out the the $mIOU$ is small because the ground truth annotations cover a much greater area (i.e they cover the face plus some background) compared to the retinaface predictions. Figure 10 shows example predictions, the first column shows correct predictions, while the second column shows incorrect predictions. We can see that the performance on OOD datase is worse than performance on the hold out set. To address this we could define the parameters of the data collecting process (camera type, angle, etc) and keep them constant. Or even better we could train the model on a larger dataset, covering additional variation.

V. Conclusion

In this work we created a custom FL3D dataset constructed specifically for driver drowsiness detection. The ResMaskNet model, originally designed for Facial Expression Recognition, was adapted and fine-tuned on the FL3D dataset. This enabled the model to learn and extract features relevant to driver drowsiness prediction. The complete prediction

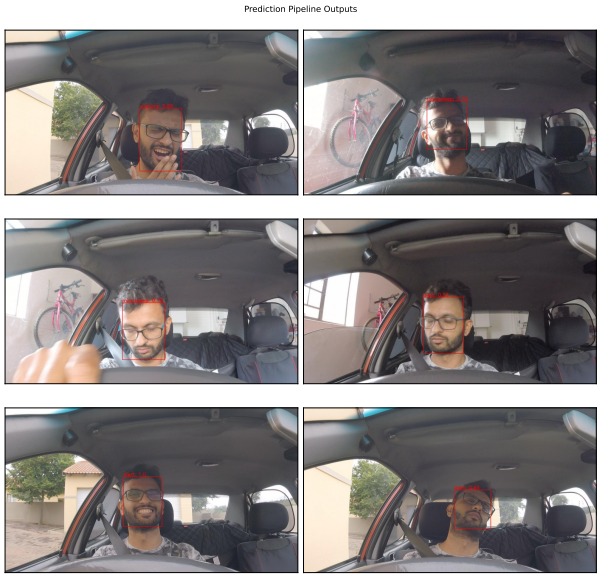


Figure 10: Prediction Pipeline outputs. The first column shows correct predictions, while the second column shows incorrect predictions.

pipeline, encompassing data preprocessing, feature extraction, and decision-making, was implemented. We achieved great performance on the holdout set (which includes drivers not present in the training set) and significantly worse performance on the OOD data. In future work, it is recommended to explore additional techniques for feature extraction and representation learning, such as attention mechanisms or recurrent neural networks, to capture temporal dependencies. Further refinements to the prediction pipeline can be made by incorporating contextual information or integrating multi-modal data sources. Additionally, the deployment of the developed system in real-world scenarios and the evaluation of its effectiveness in varying conditions are important areas for future investigation.

References

[1] S. Titare, S. Chinchghare, and K. Hande, "Driver drowsiness detection and alert system," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, pp. 583–588, 06 2021.

[2] E. Magán, M. P. Sesmero, J. M. Alonso-Weber, and A. Sanchis, "Driver drowsiness detection by applying deep learning techniques to sequences of images," *Applied Sciences*, vol. 12, no. 3, p. 1145, 2022.

[3] N. Petrellis, S. Zogas, P. Christakos, G. Keramidas, P. Mousouliotis, N. Voros, and C. Antonopoulos, "High speed implementation of the deformable shape tracking face alignment algorithm," in *2021 24th Euromicro Conference on Digital System Design (DSD)*. IEEE, 2021, pp. 174–177.

[4] S. I. Serengil and A. Ozpinar, "Lightface: A hybrid deep face recognition framework," in *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*. IEEE, 2020, pp. 23–27. [Online]. Available: https://doi.org/10.1109/ASYU50717.2020.9259802

[5] L. Pham, T. H. Vu, and T. A. Tran, "Facial expression recognition using residual masking network," in *2020 25Th international conference on pattern recognition (ICPR)*. IEEE, 2021, pp. 4513–4519.

[6] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.