ESP32-Based Distracted Truck Driver Monitoring System using FOMO

Odinachi Udemezuo Nwankwo, Dong Seong Kim and Jae-Min Lee
IT Convergence Engineering, ICT Convergence Research Center ,
Kumoh National Institute of Technology, Gumi, Korea.
odinachifoot@gmail.com, (dskim, ljmpaul)@kumoh.ac.kr

***Abstract*—Worldwide, road accidents are a significant issue, primarily affecting Middle and Lower-Middle-Class countries, where human errors are responsible for approximately 90% of these accidents. This research focuses on training a lightweight Mobilenet version 2 architecture variant known as Faster Object More Object (FOMO) for truck driver monitoring system that utilizes a camera to capture the driver unsafe driving behaviours while driving. Mobilenet V2 model variant (FOMO) has been trained and tested on a publicly available image driver dataset from Kaggle and so far has recorded an accuracy of 91.4% and has been embedded into ESP32 AI Thinker in form of Arduino code.**

***Index Terms*—Deep Learning, Distracted Driver, Computer Vision, Tiny-ML.**

I. Introduction

Distracted driving involves actions that take a driver's attention away from the road, increasing the risk of accidents. These distractions can be visual (e.g., using a mobile phone), manual (e.g., handling a phone), or cognitive (e.g., chatting), and they pose a heightened risk, including when conversing with passengers [1].

Motivation: Global Road Accident Crisis: Road accidents present a formidable worldwide challenge, with middle and lower-middle-class nations bearing the brunt. Human errors are the primary cause, contributing to roughly 90% of all road incidents.

Alarming Fatality Rates: The World Health Organization reports a staggering annual death toll of 1.35 million individuals due to road accidents, translating to an average of 64 daily fatalities worldwide.

Distracted Driving's Impact: A substantial portion of these accidents can be attributed to distracted driving, with activities like eating, mobile phone use, or conversing with passengers playing a significant role in these incidents.

Contribution: The primary contributions of this work include:

1) The primary achievement of this paper was using lightweight Mobilenet V2 variant (FOMO) developed by Edge Impulse to effectively learn distracted driving behaviors, ultimately achieving a classification accuracy that is in acceptable range (91.4%).
2) Another contribution of this work was conversion of the trained FOMO model into Arduino Code and deployment into ESP32 AI Thinker micro-controller using Edge Impulse.

II. Literature Review

Authors in paper [2] developed deep CNN models for driver distraction prediction.ResNet performed well, achieving 91% accuracy.

Another study [3] used CNNs to detect early signs of accidents due to driver stress, achieving 99% accuracy in just four epochs.

In a separate paper [1], a CNN model detected driver distractions. ResNet50 reached 94.50% accuracy, and MobileNetV2 achieved 98.12%, showing promise for real-time systems [4] [5] [6].

III. Methodology

The methodology used in this research is shown in Fig.1 whereby distracted truck driver dataset from Kaggle is used to train FOMO model.The ten classes to be predicted using 2100 input images (90% for training and 10% for validation) are as follows: c0: safe driving, c1: texting - right hand, c2: talking on the phone - right hand, c3: texting - left hand, c4: talking on the phone - left hand, c5: operating the radio, c6: drinking, c7: reaching behind, c8: hair and makeup, c9: talking to a passenger.

TABLE I
Comparison of FOMO Model with Other Models

Model	Accuracy
ResNet [2]	91%
ResNet50 [1]	94.5%
VGG-16 [3]	99.00%
MobileNet V2 [1]	98.12%
FOMO model	91.4%

IV. RESULTS AND DISCUSSION

From Table I, it can be seen that FOMO recorded classification accuracy of 91.4%. Here, there is trade-off between accuracy and portability of models. In terms of portability, it outperformed the other models since it was succesfully deployed into ESP 32 micro-controller.

In the same vein, Fig.3 depicts the confusion matrix result of the FOMO after testing the trained model with test dataset.

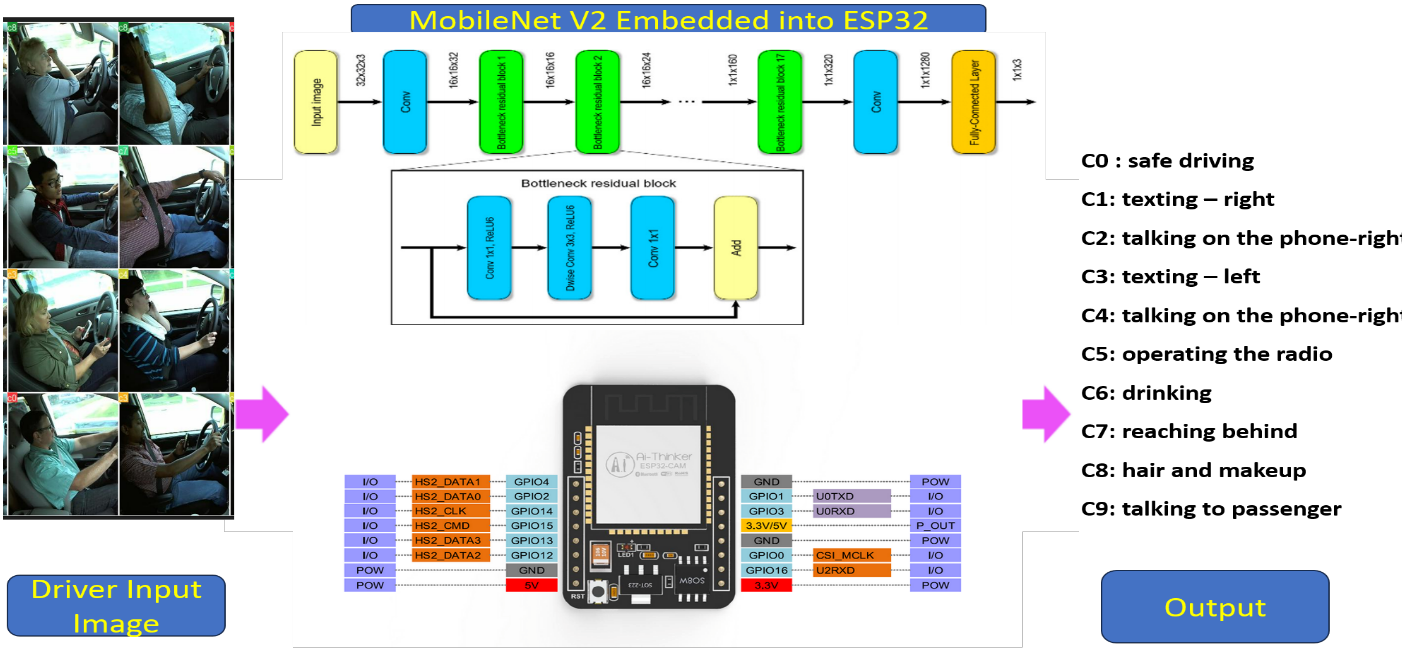
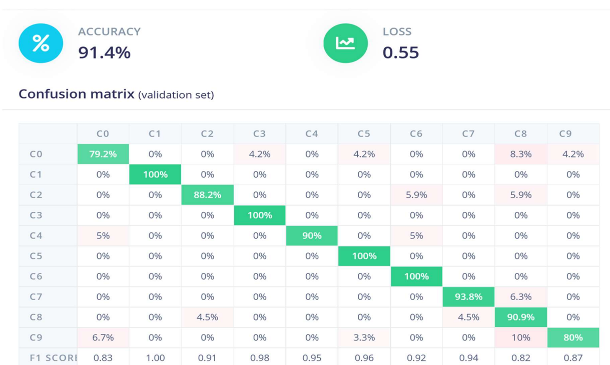


Fig. 1. System Architecture Showing MobileNet V2 Variant, FOMO Embedded into ESP32 AI Thinker



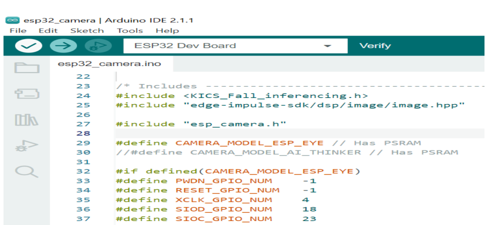
ACCURACY 91.4%

LOSS 0.55

Confusion matrix (validation set)

	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
C0	79.2%	0%	0%	4.2%	0%	4.2%	0%	0%	8.3%	4.2%
C1	0%	100%	0%	0%	0%	0%	0%	0%	0%	0%
C2	0%	0%	88.2%	0%	0%	0%	5.9%	0%	5.9%	0%
C3	0%	0%	0%	100%	0%	0%	0%	0%	0%	0%
C4	5%	0%	0%	0%	90%	0%	5%	0%	0%	0%
C5	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%
C6	0%	0%	0%	0%	0%	0%	100%	0%	0%	0%
C7	0%	0%	0%	0%	0%	0%	0%	93.8%	6.3%	0%
C8	0%	0%	4.5%	0%	0%	0%	0%	4.5%	90.9%	0%
C9	6.7%	0%	0%	0%	0%	3.3%	0%	0%	10%	80%
F1 SCORE	0.83	1.00	0.91	0.98	0.95	0.96	0.92	0.94	0.82	0.87

Fig. 2. Trained FOMO Confusion Matrix



esp32_camera | Arduino IDE 2.1.1

File Edit Sketch Tools Help

ESP32 Dev Board Verify

esp32_camera.ino

```
/* Includes ---------------------------------------
#include <KICS_Fall_inferencing.h>
#include "edge-impulse-sdk/dsp/image/image.hpp"

#include "esp_camera.h"

#define CAMERA_MODEL_ESP_EYE // Has PSRAM
//#define CAMERA_MODEL_AI_THINKER // Has PSRAM

#if defined(CAMERA_MODEL_ESP_EYE)
#define PWDN_GPIO_NUM     -1
#define RESET_GPIO_NUM    -1
#define XCLK_GPIO_NUM     4
#define SIOD_GPIO_NUM     18
#define SIOC_GPIO_NUM     23
```

Fig. 3. Trained FOMO Model Converted to Arduino Code

V. Conclusion

The study successfully used FOMO to classify distracted driver events, adapted the model to Arduino code, and deployed it on an ESP32 AI Thinker with an onboard camera. Future plans involve integrating IoT technology for enhanced data visualization and analysis.

Acknowledgments

This work was partly supported by Innovative Human Resource Development for Local Intellectualization program through the Institute of IITP grant funded by the Korea government(MSIT) (IITP-2024-2020-0-01612, 33.3%) and by Priority Research Centers Program through the NRF funded by the MEST(2018R1A6A1A03024003, 33.3%) and this work was supported by (NRF-2022R1I1A3071844, 33.3%).

References

[1] M. U. Hossain, M. A. Rahman, M. M. Islam, A. Akhter, M. A. Uddin, and B. K. Paul, "Automatic driver distraction detection using deep convolutional neural networks," *Intelligent Systems with Applications*, vol. 14, p. 200075, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667305322000163

[2] K. Srinivasan, L. Garg, D. Datta, A. A. Alaboudi, N. Z. Jhanjhi, R. Agarwal, and A. G. Thomas, "Performance comparison of deep CNN models for detecting driver's distraction," *Computers, Materials & Continua*, vol. 68, no. 3, pp. 4109–4124, 2021. [Online]. Available: http://www.techscience.com/cmc/v68n3/42484

[3] A. Christy, P. Shyry, G. Meeragandhi, and M. Praveena, "Driver distraction detection and early prediction and avoidance of accidents using convolutional neural networks," *Journal of Physics: Conference Series*, vol. 1770, p. 012007, 03 2021.

[4] O. U. Nwankwo, C. I. Nwakanma, D. S. Kim, and J.-M. Lee, "Iot-assisted intelligent vehicle tracking system using cloud computing," in *2022 13th International Conference on Information and Communication Technology Convergence (ICTC)*, 2022, pp. 1677–1679.

[5] P. C. Hoa Tran-Dang, Nicolas Krommenacker and D.-S. Kim, "The internet of things for logistics: Perspectives, application review, and challenges," *IETE Technical Review*, vol. 39, no. 1, pp. 93–121, 2022. [Online]. Available: https://doi.org/10.1080/02564602.2020.1827308

[6] G. B. Tunze, T. Huynh-The, J.-M. Lee, and D.-S. Kim, "Sparsely connected cnn for efficient automatic modulation recognition," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 15 557–15 568, 2020.