# YOLO-Based Deep Learning Design for In-Cabin Monitoring System with Fisheye-Lens Camera

# Poon, Y.S., Lin, C.C., Liu, Y.H., Fan, C.P.: YOLO-based deep learning design for in-cabin monitoring system with fisheye-lens camera. In: Proceedings of the IEEE International Conference on Consumer Electronics (ICCE-2022), pp. 1–4 (2022). <https://doi.org/10.1109/ICCE53296.2022.9730235>

#### Basic Idea

This paper presents an image-based in-cabin monitoring system using YOLO-based deep learning models with a fisheye-lens camera to detect driving behavior and in-vehicle occupants, improving driving safety. The system is designed to recognize normal and distracted driving conditions, as well as identify back-seat passengers and pet dogs.

#### Key Contributions

The key contributions include teh use of various YOLO models(YOLOv3-tiny, YOLOv3-tiny-3l, YOLO-fastest, YOLO-fastest-xl, and YOLO-fastest-three scales) for in-cabin monitoring. The paper demonstrates that the YOLO-fastest-three-scales model offers the best performance in terms of F1-score1 and mAP2. In contrast, the YOLO-fastest-xl model excels in minimizing the False Negative Rate3 (FNR). The proposed system can process up to 30 frames per second on a GPU-based embedded device.

#### Technical Approach

The authors employed YOLO-based deep learning models, adapting them to process RGB-format images captured by a fisheye-lens camera placed at the in-car roof center. The models were trained using a multi-scale input training scheme, and different YOLO architectures were explored to find the best-performing model for real-time in-cabin monitoring. The design also incorporates data augmentation techniques to enhance model robustness.

#### Data and Model Approach

The data was collected using a fisheye-lens camera, providing images with a resolution of 3264x2448 pixels and a 360-degree field of view.

The LabelImg tool was used for labeling, with the data being divided into training, validation, and testing sets.

Data augmentation was employed to increase the dataset size and prevent overfitting.

The YOLO-based models were configured using the Darknet framework, with parameters tuned for optimal performance.

The YOLO models used include YOLOv3-tiny, YOLOv3-tiny-3l, YOLO-fastest, and YOLO-fastest-xl, with unified input sizes of 416x416 pixels and an iteration count of 40,200 for training.

\*\*In comparison to our current research,

#### Key Results

The YOLO-fastest-three scales model achieved the best F1-Score (95.89%) and mAP (97.16%), while the YOLO-fastest-xl model had the lowest False Negative Rate (2.63%). The system could process up to 30 FPS on an NVIDIA Xavier platform, demonstrating its suitability for real-time applications.

#### Key Limitations

The authors mention that future work will explore next-generation YOLO models or other CNN architectures to improve performance further. They also plan to expand the training datasets and add more categories and behaviors to the in-cabin monitoring functions.

#### Other Notable Aspects

A notable aspect is the use of a fisheye-lens camera with a wide field of view (360 degrees), which allows comprehensive monitoring of the entire cabin. The integration of the YOLO-fastest-three scales model, optimized for **real-time performance**, is also a significant feature that stands out.

#### Definitions I needed to search up:

1. **F1-Score** is a measure of a test's accuracy, specifically the harmonic mean of precision and recall. It is useful for imbalanced datasets where one class is more frequent than others. The F1-Score ranges from 0 to 1, where 1 indicates perfect precision and recall.
   1. **Precision**: The ratio of correctly predicted positive observations to the total predicted positives.
   2. **Recall**: The ratio of correctly predicted positive observations to all observations in the actual class.



1. **Mean Average Precision (mAP)** is a metric used to evaluate the accuracy of object detection models. It represents the mean of the Average Precision (AP) values across all classes in a dataset.
   1. **Precision-Recall Curve**: A plot showing the trade-off between precision and recall for different thresholds.
   2. **Average Precision (AP)**: The area under the Precision-Recall curve for a single class.



1. *(I realized what this was immediately after searching it up but it was a good review)* **False Negative Rate (FNR)** is a metric used to measure the proportion of actual positives that are incorrectly identified as negatives by a model. It represents the likelihood that a test fails to detect a positive case when it is present.
   1. **False Negatives (FN)**: The number of positive instances incorrectly predicted as negative.
   2. **True Positives (TP)**: The number of positive instances correctly predicted as positive.

