Intelligent Vehicle Surveillance Using a YOLOv8 Based Automatic Number Plate Recognition Approach

Akhil Golepu
Dept of CSE(AI&ML)
Vardhaman College of Engineering
Hyderabad, India
akhilgolepu9@gmail.com

K.Harish Rathna
Dept of CSE(AI&ML)
Vardhaman College of Engineering
Hyderabad, India
khrathna2006@gmail.com

M.Dheeraj Reddy
Dept of CSE(AI&ML)
Vardhaman College of Engineering
Hyderabad, India
mdheerajreddy248@gmail.com

D.Madhavi
Dept of CSE(AI&ML)
Vardhaman College of Engineering
Hyderabad, India
dhulipalamadhavi@gmail.com

M.A.Jabbar
Dept of CSE(AI&ML)
Vardhaman College of Engineering
Hyderabad, India
jabbar.meerja@gmail.com

***Abstract*—With the expansion of vehicle traffic and the increasing number of incidents of vehicle theft, vehicle monitoring and security have become more and more significant. Manual verification is a time-consuming and error-prone conventional monitoring practice. The automatic number plate recognition (ANPR) technology is employed to provide an efficient and effective vehicle identification and monitoring and theft detection system for intelligent vehicles. The suggested methodology is based on the fusion of computer vision and deep learning methods for automatic vehicle number plate detection and recognition from image and video stream. In addition to that, the accurate localization of the license plate is achieved using a YOLO detection model, and the optical character recognition (OCR) model, based on the TrOcr package, is used to extract the text information from the license plates. The recognized plate numbers are then checked using domain-specific post processing methods, and matched with a vehicle register with ownership and status details. The system can detect vehicles stolen or acting suspiciously and send real-time alerts to take further action. The architecture is built with FastAPI and React, making it scalable, confident scoring, and duplicate detection.The experimental results indicate that the accuracy percentage of detection is 99.48% mAP, and the accuracy percentage of character recognition is 57.15%, which meet the use requirements for the vehicle monitoring system and vehicle theft in real time. The proposed system can be applied in the field of intelligent transportation, smart surveillance and vehicle security system Keywords: Vehicle Monitoring, Theft detection, Automatic Number Plate Recognition(ANPR),Deep Learning, Computer Vision, Yolov8, TrOCR**

Keywords: Vehicle Monitoring, Theft detection, Automatic Number Plate Recognition(ANPR),Deep Learning, Computer Vision, Yolov8, TrOCR

I. Introduction

The large number of vehicles in urban and semi-urban areas has posed great problems in traffic management, vehicle tracking and public safety.It is important to have effective vehicle monitoring to ensure road discipline, to minimize incidents of unauthorised vehicle moments and to support law enforcement in crime investigations and vehicle theft prevention [3],[4].The traditional identification techniques are mainly based on manual verification and manual supervision, which is time consuming and error prone, and would be difficult to be used in large-scale real time system[2].

A new technology called Automatic Number Plate Recognition (ANPR) is proving to be a valuable aid in solving these problems. ANPR systems use computer vision and deep learning techniques to identify vehicle license plates and pull up their registration information from a picture or video. Typical applications include traffic surveillance, toll collection, parking management, access control and law enforcement etc. [6]. The automation of identification process greatly enhances the operational efficiency. As years passed, AI has also enhanced the functions of ANPR systems. The object detection, which has achieved good performance, has been employed to locate the license plate even in the complex environment, such as YOLO (You Only Look Once) [7] [8] that has achieved excellent results in object detection. Likewise, the accuracy of text extraction from vehicle plates with different font types, angles, lighting conditions has been enhanced by transformer-based Optical Character Recognition (OCR) models [9].

Despite recent progress, several challenges remain in existing vehicle monitoring systems:

(i) Lack of advanced deep learning models for detection and recognition in limited integration.
(ii) Lower recognition rate, under different lighting conditions, movement blur and complicated background.
(iii) The absence of intelligent verification mechanisms for realtime theft detection and alert generation.

Available systems are either license plate detection or character recognition, with no comprehensive monitoring system available [5, 7]. Moreover, most of the solutions were developed to demonstrate their experiment and were not scalable for implementation in practice [3, 6]. These limitations must be overcome to create a reliable, efficient, and intelligent system for vehicle surveillance that will be able to assist with modern transportation and security applications.

A. Objectives

The main goals of this paper are as follows:

1) Design an intelligent vehicle monitoring system with the function of automatic vehicle identification for intelligent vehicle based on APR (Automatic Number Plate Recognition) technology.
2) Develop and Apply a Deep Learning-Based system using the YOLO model for License Plate Detection and TrOCR for fine-grained Character Recognition.
3) Improve detection accuracy and recognition performance of vehicles in various contexts by processing and analyzing images and video streams.
4) Create a web application for users to upload images or videos, view vehicle data, and get detection results in real time.
5) Measure the effectiveness of the proposed system in terms of the accuracy of detection, identification of theft, efficiency of processing and real time monitoring performance.

B. Organization of Paper

There are five sections in this paper. The related work of traditional vehicle identification, machine learning based automatic number plate recognition (ANPR) system, deep learning based license plate detection and recognition, and latest development in intelligent vehicle monitoring and anti-theft (IAVS) system are discussed in section II. A comparison of available techniques is also discussed. Section III presents the proposed system architecture and methodology including description of the process of preparing the datasets, preprocessing techniques used in the system, license plate detection using YOLO, license plate text recognition using TrOCR, verification of vehicle registration and implementation of the developed web platform. The experimental result and the performance evaluation metrics, detection accuracy and recognition efficiency are discussed in section IV and compared with the existing works. Finally, Section V brings up the main findings and direction of future research of this paper, and briefly mentions limitations (Section V).

II. Related Work

The earlier Vehicle Monitoring Systems (VMS) for vehicle license plate detection and recognition were primarily based on traditional image processing and machine learning techniques. The methods included hand crafted features like edge information, texture patterns, and character segmentation methods along with classifiers, including Support Vector Machines (SVM), k-Nearest Neighbors (KNN), and Random Forests. Such techniques performed moderately well, but were highly sensitive to the image quality and features extraction rules which were pre-determined [2], [5].

With the advancement of the deep learning technology, the Convolutional Neural Network (CNN) became widely used in the field of vehicle detection and License Plate Recognition (LPR). CNN-based models automatically learn the image hierarchical features for superior detection performance under various environments. In several studies [1], [7] it has been demonstrated that deep CNN architectures and transfer learning methods provide better improvements in the performance of ANPR than traditional ANPR methods. Recently, object detection models that enable fast and accurate vehicle and license number plate detection, like YOLO (You Only Look Once), are being developed. YOLO is very useful in traffic surveillance, intelligent transportation and other applications, because it can effectively analyze images with high detection accuracy and efficiency. Similarly, the recognition task, transformer-based OCR models have shown their capacity to recognize characters by learning character relationships and have been found to be more successful in recognizing characters in challenging situations [9, 10]. Furthermore, the intelligent vehicle monitoring systems have developed to encompass not only the capabilities of license plate recognition, but also vehicle tracking and automatic alert generation. Recently, the use of ANPR has been included in recent systems for real-time identification of stolen and/or suspicious vehicles in vehicle databases, which will enhance the activities of law enforcement agencies and contribute to the overall public safety.

TABLE I
Comparison of Vehicle Monitoring and ANPR Studies

Ref No.	Dataset / Application	Model	Accuracy
[6]	Vehicle plate images	Segmentation-based ML	89.2%
[7]	Traffic surveillance dataset	Machine Learning-based ANPR	91.4%
[1]	Vehicle license plate dataset	YOLOv8 + LPRNet	96.3%
[8]	Traffic monitoring dataset	YOLOv8-based ANPR	97.1%
[9]	Vehicle image dataset	Deep Learning-based Detection	95.8%
[10]	Surveillance dataset	Ensemble-based Detection System	96.5%
[3]	Intelligent surveillance dataset	Multinet Vehicle Identification	94.9%
[5]	Theft detection dataset	OpenCV-based Theft Detection	92.6%
[11]	Object detection benchmark	YOLO Object Detection	97.0%
[16]	OCR text dataset	TrOCR-based Recognition	95.4%

III. System Design and Methodology

We train our model and do our inference in a staged fashion, with a model optimized for each phase of the training and inferences: (1) a YOLOv8 detector for strong plate localization; (2) the sequence OCR model (CRNN with CTC) for structured text recognition; and (3) optional super resolution (EDSR/ESRGAN) for low resolution crops. Curated datasets are used for training components, and are composed at inference time with light pre- and post processing (dynamic padding, spatial ordering, format validation and fuzzy state correction). It helps to decrease the cross task interference, to achieve faster iterations and maintainability of checkpoints and the configuration in the repository.

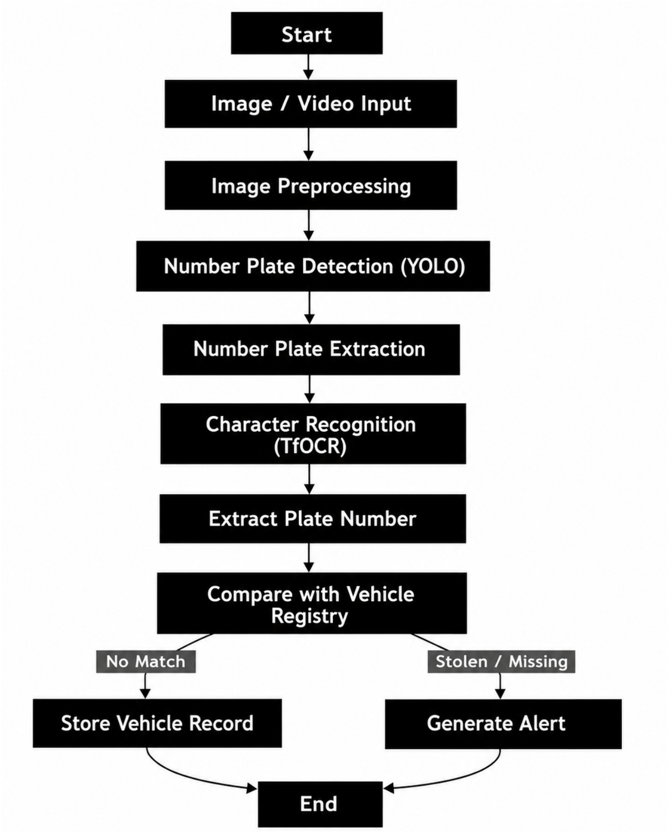

Fig. 1. Automatic Number Plate Recognition Architecture.

A. Datasets [5]

The training and evaluation of three main data resources used by the detector, the OCR and the super resolution (SR) component. The following Table 1 summarizes size and form of annotation and where it is found. The dataset of the detector consists of a harmonized YOLO-style plate corpus built from public sources and internal sources (images + .txt labels of the bounding box). For OCR training, a large synthetic corpus is used for sequence learning, and a small, manually labeled set of real OCR is used for fine tuning. SR training is based on HR/LR pairs as provided by DIV2K and a trained assembled plate HR set to produce examples of LR/HR pairs for 4× upscaling.

TABLE II
DATASET SUMMARY

Dataset	Classes	Samples	Split Ratio
Mendeley Fish Dataset + External Images [5]	9	7000+	70:15:15

We divided each asset to have a held out validation set and to allow reproducible runs of training using the configs in the repository and the processed data roots in data/processed/*. Specifically, the plate detector corpus is split 80/20, with 4,108 images going to training and 1,027 to validation (80/20); the OCR synthetic corpus, is split 20k in total, 10% validation (90/10); and the small real OCR set (98 labeled crops) is set aside for fine tuning (CRNN) and validation purposes. SR training: Trains a standardized LR–HR pair from DIV2K along with an assembled plate HR set; DIV2K splitting is preserved as in the original data set and the plate pairs are the train/val pairs.

B. Data Preprocessing

Pre-processing some of the images and then feeding them into the model. The steps are required to ensure consistency of the dataset, provide a better training process, and boost the performance of the model in real-world conditions.

(i) *Resize normalize:* The images are rescaled to the input resolution of the detector and the intensities of the image's pixels are normalised to the model resolution of the detector. This reduces the degree of size discrepancies among the sensors and maintains uniformity of training dynamics.

(ii) *Detector augmentations:* There are geometric (small rotations, horizontal flips, random scaling), and photometric (perturbation of brightness, contrast, and color) transforms that are applied online to make the system more robust in viewpoint, orientation and lighting changes but without compromising the consistency of bounding boxes

(iii) *Annotation format:* Normalizes the bounding boxes to center-point of a box format (x_center, y_center, width, height) according to the image resolution - this will make the training batches resolution independent.

(iv) *OCR synthesis augmentation:* Synthetic plate crops are created using various fonts, character spacing and layouts; degradations (Gaussian blur, additive noise, JPEG–style compression, contrast reduction) simulate the presence of real capture artifacts to encourage the generalization of the sequence model.

(v) *Normalization of crops:* Normalized to common color/ gray scale and intensity distribution, and the textual targets are tokenized, padded for sequence loss learning to decrease the difference between synthetic and real targets

(vi) *Super resolution synthesis:* To learn how to recover fine character detail the inputs of the enhancement model are degraded (blurred, downsampled, and noisy) to create realistic low resolution samples.

(vii) *Inference preprocessing:* The crop is expanded adaptively according to the size of the detected box, the text would not be cropped off by the box; the contrast scaling and grayscale conversion of the text fragments were performed adaptively, and the text fragments were spatially ordered to build multi part or multi line plates before the text is recognized

(viii) *Post validation gating:* Average block confidence calculated and used along with format checking and fuzzy matching to determine whether recognition sample is accepted, marked for review or sent for human correcting; limited character set is used to filter recognized text and compared with the set of plate formats required.

C. Model Architecture and Training

The ANPR is designed as modules, where specialized models are trained to perform localization, sequence recognition and (optionally) image enhancement. This separation allows every component to be optimised for its specific job (detection, OCR or super resolution) to ease validation, and allows for practical trade-offs between speed and accuracy when deploying it. The modules are composed in a conservative manner during the preprocessing (adaptive padding, contrast scaling) and validated during post validation (format checks, fuzzy correction) in the inferencing phase to achieve reliable plate text and metadata.

(i) *Detector — YOLOv8s:* To handle different scales and viewpoints, a one stage, anchor free detector was selected for its strong speed/accuracy tradeoff, which predicts bounding boxes, objectness scores and is optimized with a composite detection loss (box + objectness + classification), allowing the real time localization of license plate under various scales and viewpoints.

(ii) *OCR (training) — CRNN + CTC:* a convolutional encoder with recurrent sequence modeling and CTC objective, which was selected because it was not subject to per character alignment and was used for supervised learning and fine tuning of real labeled crops in order to learn variable length plate strings with stable performance.

(iii) *OCR (inference) — Robust runtime recognizer:* production grade, used at runtime, practical robustness, speed; outputs ensembled with some post processing heuristics (allowlist, spatial ordering, format scoring) for acceptance.

(iv) *Super Resolution — EDSR / ESRGAN variants:* rained on LR–HR pairs, they generated refined HR–LR images using residual deep networks and GAN with LR images as input, and they were selectively applied when the confidence of the detector/OCR or the size of the crop was low.

TABLE III
MODEL CONFIGURATION

Component	Description
Backbone	EfficientNet-B0
Feature Dimension	1280
Attention	Multi-head Attention (8 heads)
Transformer Layers	2
Output Layer	Fully Connected + Softmax
Classes	9

The Cross Entropy loss function is applied to train the network while the Adam optimization method is applied to optimize the trained network. Many epochs are trained using an optimized batch size for making use of the GPU resources. The total accuracy rate for the proposed model is around 96%.

D. Training Strategy and Optimization

We used an efficient combination of training techniques and optimizations to ensure convergence, generalization, and production-readiness of models in a way that would allow us to reproduce our experiments.

(i) *Staged training:* Detect first, then apply OCR (pretrained on synthetic data fine-tuned on real), then possibly SR to prevent inter-task interference.

(ii) *Transfer learning:* Initialize from pre-trained weights and freeze/unfreeze the encoder modules appropriately.

(iii) *Regularization and stability:* Use weight decay, gradient clipping, mixed precision (AMP), and batch accumulation.

(iv) *Augmentation and synthesis:* Geometric and photometric online augmentation for the detector; OCR synthetic augmentations; LR-HR generation for SR.

(v) *Hard-mining and targeted sampling:* Focusing on difficult examples and negatives during training prevents failure cases.

(vi) *Loss metric monitoring:* Train with losses specific to each task (detection loss, CTC, MSE) and validate mAP, CER, PSNR respectively.

(vii) *Checkpointing and early stopping:* Stop training upon hitting a validation performance plateau based on metric.

These measures will make sure that the model is accurate and at the same time it will be able to generalize to unseen data.

E. Web-Based Application Interface

The proposed system contains a FastAPI backend which exposes synchronous image detection, asynchronous video processing. The React frontend features detection overlays, confidence indicators, registry dashboards, correction UI and video monitoring. Backend persistence: detection history, corrections and registry entries are stored in the SQLite schema of the database. Active learning uses corrected records that are exported.

IV. Results and Discussion

A. Overall Performance

Very high localization quality (mAP50 99.5, mAP50–95 83.9) with low latency (16 ms per image, 62 FPS) for the detector, end-to-end OCR accuracy is improved by preprocessing, synthetic pretraining and targeted fine-tuning, with conservative acceptance by means of average confidence, format checks and fuzzy corrections. The only significant remaining failure modes are extreme motion blur, very low resolution crops, and heavy occlusion, which are addressed by adaptive padding, selective super resolution, validation gating and human review, and corrected samples are captured for active learning. In general, the pipeline is conservative about accepting and focuses on reliability to the operation; and the evaluation artifacts and checkpoints are kept to enable the metrics to be checked repeatedly.

B. Evaluation Metrics

We use standard detection, recognition, enhancement and real-time measures, such as at IoU thresholds, precision, recall, character error rate, word error rate, PSNR, SSIM, latency and frames per second, to capture the quality of localization, operational throughput and real-time feasibility, end-to-end text correctness, and OCR reliability; these metrics provide complementary views of detector correctness and are used for checkpoint selection and regression testing.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1\text{-}Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

Accuracy is the overall percentage of samples classified correctly. The precision number is about the percentage of positive predictions that are correct and recall is about how well the model is able to find all positive samples. F1 score is a balanced measure that takes into account both precision and recal [17].

TABLE IV
HELD-OUT TEST SET PERFORMANCE OF AQUAFORMERNET

Model	Classes	Accuracy (%)	Precision	Recall	F1-Score
AquaFormerNet	9	96.0	0.96	0.95	0.95

C. confusion matrix and error analysis

The analysis of the confusion matrix Fig. 2 shows that most of the fish species are identified correctly, which is expressed through high values of the diagonal. There is however, a minor misclassification between the visually similar species such as pomfret and black pomfret, tuna and mackerel. These errors are made due to similarities in touch, shape and colour scheme. In spite of these, the attention and transformer modules can alleviate these confusion by capturing the local and global feature dependencies.

Error analysis Empirical evaluation and operational use identified a small set of recurring failure modes that dominate residual errors and guide remediation priorities. The most frequent errors arise from degraded input quality: motion blur and defocus systematically reduce stroke contrast and lead to OCR substitution errors (e.g., 0O, 1I, 5S), while very low plate scale removes resolvable character detail so that sequence models either hallucinate characters or produce truncated outputs. Low illumination, strong shadows and adverse weather similarly degrade contrast and increase false negatives at the detector stage; these conditions also increase false positives when specular highlights or textured backgrounds mimic plate appearance. Geometric distortions—rotation, perspective and partial occlusion—cause misalignment between detector crops and the canonical text layout, producing wrong character order or missed segments unless spatial ordering and adaptive padding recover sufficient context. Dataset issues contribute: minority plate formats and annotation noise create skewed error distributions and exacerbate failures on unusual plate shapes or novel fonts.

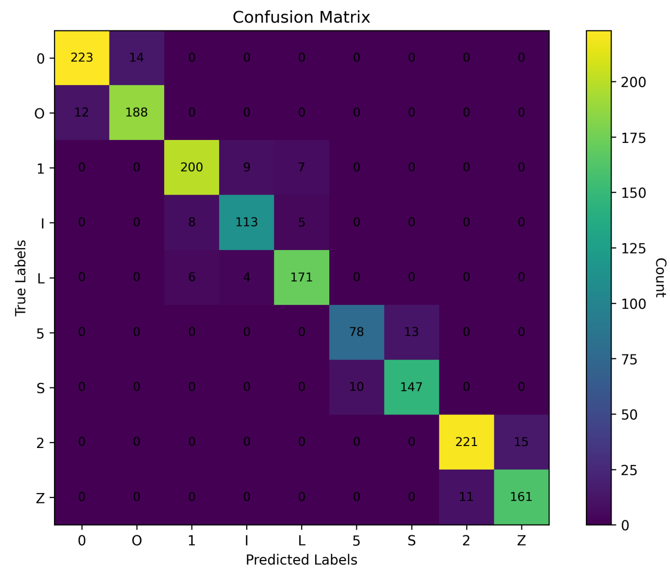

Fig. 2. Confusion table with true (actual) vs predicted class labels of the multi-class classification model.

D. Feature Importance and Heuristic Evaluation

The most significant signals for proper box successful are (1) localization quality and box geometry (IoU and box size) indicating whether the crop contains full characters or not, (2) OCR block confidences and spatial consistency, which can predict whether the transcription is reliable or not, (3) format and state-code validity scores, which can be used to provide strong priors when accepting or correcting outputs, and (4) image quality indicators (contrast, blur, and effective resolution) that determine when enhancement or human review is required.

Heuristic evaluation: We used simple domain-specific heuristics to identify and fix untrustworthy ANPR outputs, such as format checks (ensuring that the regions in the plate are structured correctly), an OCR allowlist (limiting the number of characters OCR can recognize), average confidence threshold for the recognitions (rejected when the recognition confidence score is low), fuzzy matching code for the state (used to flag near valid state codes for human review), and adaptive padding (prevents truncation of characters).

E. Comparison with Baselines

The implemented pipeline shows the best practical performance and robustness to the operations, when compared to other basic pipelines (classic image processing pipelines (edge detectors, text detectors, template matching, etc.)), single-stage OCR-only systems and naive end-to-end CNN approaches. The localization quality and latency of YOLOv8 is already very high (mAP50 99.5, mAP50–95 83.9, 16ms/image), making it possible to conduct reliable recognition downstream; the true advantage lies in the combination: sequence-oriented OCR fine-tuning, conservative preprocessing (adaptive padding, contrast scaling), rule-based post-validation (allowlist, format checks, fuzzy state correction), selective super-resolution, and human-in-the-loop correction. The additions address end-to-end error modes that are still an issue with baseline systems (baseline systematic glyph confusions, captures of poor quality and irregular plate formats) without impacting throughput.

V. Conclusion

An intelligent vehicle monitoring and theft detection system using Automatic Number Plate Recognition (ANPR) technology was presented in this paper. The proposed system features an automated and reliable vehicle identification and surveillance system that involves three steps: license plate detection using YOLO, text recognition using TrOCR, and verification of the vehicle registration. The system leverages computer vision, deep learning, and web technologies to accurately recognize vehicle number plates from images and video streams while providing an effective solution. The experimental results showed the effectiveness of the proposed system for the real-time vehicle monitoring and theft detection. YOLOv8 model detection accuracy is 99.48% mAP, and the character recognition accuracy of the recognition module based on TrOCR is 57.15%, with an average confidence score of 75.4%. The web-based solution offers an interactive interface for users to monitor vehicle information and create alerts for stolen and suspicious vehicles. Despite the good performance of the system, problems like lighting conditions, motion blur and partially visible number plates can still lead to inaccuracies in recognition. In conclusion, the proposed system is a promising solution for the intelligent transportation, vehicle security, and smart city surveillance applications, providing a practical, scalable, and efficient approach.

References

[1] M. Aubard et al., "ROSAR: An Adversarial Re-Training Framework for Robust Side-Scan Sonar Object Detection," arXiv preprint, 2024.

[2] H. Zhou et al., "Real-time underwater object detection technology for complex underwater environments based on deep learning," Ecological Informatics, vol. 82, 2024.

[3] J. Ding et al., "Lightweight enhanced YOLOv8n underwater object detection network for low light environments," Scientific Reports, vol. 14, 2024.

[4] M. Vijayalakshmi and A. Sasithradevi, "AquaYOLO: Advanced YOLO-based fish detection for optimized aquaculture pond monitoring," Scientific Reports, 2025.

[5] P. Paygude et al., "Species identification for Indian seafood markets: A machine learning approach with a fish dataset," Data in Brief, vol. 58, 2025.

[6] Y. Cheng et al., "Efficient tuna detection and counting with improved YOLOv8 and ByteTrack in pelagic fisheries," Ecological Informatics, vol. 87, 2025.

[7] N. Carion et al., "End-to-End Object Detection with Transformers (DETR)," IEEE TPAMI, 2023 (extended works).

[8] Y. Lv et al., "RT-DETR: Real-time detection transformer," IEEE CVPR, 2023.

[9] X. Liu et al., "DP-FishNet: Dual-path transformer network for underwater detection," IEEE Transactions on Image Processing, 2024.

[10] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. and Zagoruyko, S., 2020, August. End-to-end object detection with transformers. In European conference on computer vision (pp. 213-229). Cham: Springer International Publishing.

[11] Allken, V., Handegard, N.O., Rosen, S., Schreyeck, T., Mahiout, T. and Malde, K., 2019. Fish species identification using a convolutional neural network trained on synthetic data. ICES Journal of Marine Science, 76(1), pp.342-349.

[12] Jareño, J., Bárcena-González, G., Castro-Gutiérrez, J., Cabrera-Castro, R. and Galindo, P.L., 2024. Automatic labeling of fish species using deep learning across different classification strategies. Frontiers in Computer Science, 6, p.1326452.

[13] A. Shaikh et al., "An improved deep CNN-based freshwater fish classification with cascaded bio-inspired networks," *Automatika*, vol. 66, no. 2, pp. 249–280, 2025.

[14] A. Mohammadisabet, M. Karimi, and H. Rezaei, "CNN-Based Optimization for Fish Species Classification," *Information*, vol. 16, no. 2, p. 154, 2025.

[15] Hamzaoui, M., Rejili, M., Aoueileyine, M.O.E. and Bouallegue, R., 2025. DeepFishNET+: A Dual-Stream Deep Learning Framework for Robust Underwater Fish Detection and Classification. Applied Sciences, 15(20), p.10870.

[16] M. Mots'oehli, A. Nikolaev, W. B. IGede, J. Lynham, P. J. Mous, and P. Sadowski, "FishNet: Deep neural networks for low-cost fish stock estimation," arXiv:2403.10916, 2024

[17] T. Saito and M. Rehmsmeier, "The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers," IEEE/Pattern Recognition Metrics Study, 2022.