BSCS-S25-010

03-134221-047 Zain ul Abdien

03-134221-042 Umer Ateeq

Automobile Inspector

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: **Muhammad Mudassar**

Department of Computer Sciences

Bahria University, Lahore Campus

January 2026

C e r t i f i c a t e

We accept the work contained in the report titled

"Automobile Inspector"

written by

Zain ul Abdien

Umer Ateeq

as a confirmation to the required standard for the partial fulfilment of the degree of

Bachelor of Science in Computer Science

Approved by:

Supervisor: Muhammad Mudassar

(Signature)

05 January, 2026

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	**Name**	**Signature**
03-134221-047	Zain ul Abdien	
03-134221-042	Umer Ateeq	

Date : 05 January, 2026

Dedication

Specially dedicated to

my beloved grandmother, mother and father

(Zain ul Abdien)

my beloved grandmother, mother and father

(Umer Ateeq)

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express our gratitude to my research supervisor, Mr. Muhammad Mudassar for his invaluable advice, guidance and his enormous patience throughout the development of the research.

In addition, we would also like to express my gratitude to our loving parent and friends who had helped and given me encouragement.

Zain ul Abdien
Umer Ateeq

Automobile Inspector

ABSTRACT

The used car market is expanding rapidly, yet fraud, misrepresented evaluations, and biased pricing erode trust between buyers and sellers. Traditional inspections are time-intensive, expensive, and subjective. To address these, the Automobile Inspector an AI-powered system automates vehicle condition assessment, fair pricing, and cost estimation, including a trip cost, fuel efficiency, and city-to-city calculator. It leverages Python, TensorFlow, PyTorch, OpenCV, and YOLO with a teacher-student framework (YOLOv8x as teacher, YOLOv8L as student) for detecting components like engines, interiors, and exteriors. The interior model uses DenseNet121 with transfer learning; sound analysis for early mechanical fault detection employs MFCC+CNN; text-based models incorporate XGBoost. Datasets are sourced from IAAI.com and Copart.com for engine sounds, interior/exterior images, and text features, plus additional engine sounds from YouTube and TikTok. Sophisticated image processing identifies dents, scratches, and structural defects. The frontend is built with Flutter and Dart, connected to Firebase for user authentication, while the backend uses FastAPI and MongoDB to store user-inputted images.

The system provides real-time market-based price forecasting, a RAG based chatbot for interactive queries, and a report generation feature that produces full-fledged PDF reports of car analysis. Overall accuracy we got in models is 81% ,it delivers objective, data-driven insights that empower buyers with transparency and confidence while boosting sellers' credibility.

Keywords: Car Inspection , Engine Inspection through voice , Car price prediction, YOLO , MFCC+CNN

TABLE OF CONTENTS

LIST OF TABLES

Table **TITLE** **PAGE**

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

Buying and selling used cars has been a new trend across the globe because of its cheap cost and demand in the market. The second-hand car market is, however, normally filled with fraudsters, latent flaws and unreliable evaluations. The conventional means of inspection are very manual and generally time and expense consuming methods that can be manipulated or affected by human error or prejudice as it is usually evaluated by a mechanic or a dealer. Such restrictions render the buyers and sellers unable to get access to transparent and credible information regarding the real state of a vehicle and the fair value of a vehicle in the market.

The key features of engine performance, accident history, concealed damages, and external wear-and-tear are impossible to define with the help of conventional checks. Buyers are easily victims of cosmetic changes whereas sellers find it hard to explain their prices without objective evidence. Such issues decrease the level of trust in the market and raise the level of financial risks.

Due to the recent progress in Artificial Intelligence (AI), Machine Learning (ML), and Computer Vision (CV), the process of car inspection will be possible to automate. An AI-based system can deliver true real-time assessments on engine conditions by sound analysis, identify dents and scratches by image processing, and forecast reasonable prices of cars by applying regression-based valuation algorithms. Also, the system can be further improved by incorporating real-time fuel prices and the trip cost calculators to make the system more practical to the ordinary person.

The proposed Automobile Inspector system will be based on Python, TensorFlow, OpenCV, YOLO , and database to provide transparent, unbiased, and reliable ideas. It is designed to make the used car business more efficient, cost effective, and create confidence between buyers, sellers and other parties interested in the business.

1.2 Problem Statements

The traditional ways of inspecting cars have several issues which influence transparency, accuracy, and affordability of used car market. The following are some of the main challenges:

Manual and Time-Consuming Inspections: Conventional testing is done by specialists or mechanics, which takes much time and financial resources and may still be subject to human miss judgment and prejudice. The Secret Vehicle Problems: Customers do not always have the technical knowledge on how to identify the internal engine problems, or damages of the car because of an accident, and this leads to the financial loss and dissatisfaction.

False Pricing: There are no real time comparative tools which makes it difficult to sell items at fair and market related prices in the absence of this the sellers are likely to be confronted by disputes and mistrust.

Absence of Unified Solutions: The existing systems fail to offer users an integrated system that includes vehicle condition check, price forecast and cost forecast of a trip.

These shortcomings demonstrate the necessity of an automated and intelligent solution to provide correct diagnostics, reliable values and estimates about costs to build trust in the second-hand car market.

Aim and Objectives

The main goal of the current project will be to design and develop an AI-driven Automobile Inspector that will streamline vehicle inspections and valuations. AI Driven Evaluation System: Design AI and computer vision applications to test the engine, interiors, exterior and tires to appropriately evaluate their condition-Image Processing to detect External Damage: Using deep learning networks (YOLO, Faster R-CNN) detect dents, scratches, evaluate any external damage.

Engine Sound Analysis: Predict malfunctions of the engines that might be hidden without the use of RNN/CNN models and audio processing methods.

Automated Price Prediction: Develop a valuation model that uses the following factors; exhausted, make, model, engine well, and real-time market trends to estimate fair prices on vehicles.

Trip Cost Calculator: Combine prices of fuel and estimate miles to offer the user a cost forecast of travel, as well as ownership.

Increased Transparency: This will minimize biases and human mistake of inspections, making sure that buyers and sellers have reliable information

1.3 Scope of the Project

The Automobile Inspector project scope incorporates the following functions:

Core Functionalities:

- Automated checking of the health of the engine of vehicles by sound.
- Scratch, dent detection with visual inspection with computer vision.
- AI car valuation regression models that are dependent on the market.
- On the real-time fuel prices and trip cost estimation.

Beneficiaries:

Direct: Bought and sold by used car buyers, sellers and dealerships that are interested in clear reviews.

- Indirect: Insurance companies, mechanics and government regulators that control equitable market practices.

Optional Features (Future Enhancements):

- AI chatbot to help people with the frequently asked questions concerning vehicle inspection and maintenance.
- Repaint area detection of car body
- The Artificial Reality (AR) capabilities are used to identify live dents and scratches.
- Integration with e commerce car sites in the purchase and sale of cars

CHAPTER 2

LITERATURE REVIEW

2.1 Literature Review

The automobile marketplace has evoked a lot of research activities interest owing to the increased pressure on accountability and fairness in the assessment of vehicles. Manual expertise is a method of traditional inspection mechanisms that is not only time-consuming but also subjective and inaccurate. Along with the emergence of Artificial Intelligence (AI), Machine Learning (ML), and Computer Vision (CV), scholars have examined various methods of automating the process of diagnostics, damage recognition, and valuation.

Some of the studies have applied computer vision models like the YOLO to detect objects and identify defects in cars. Diwan et al. [1] provided a comparative analysis on YOLO and its descendants, in which they stressed out their strengths on real-time detection on dents and scratches. Likewise, Rath et al. [2] revealed the capabilities of regression-based models in predicting the outcomes through the analysis of real-world data which is directly applicable in predicting the price of vehicles in volatile markets.

In the sphere of engine sound diagnostics, Syed et al. [3] performed a comparative study of CNN and RNN models of voice recognition and demonstrated their effectiveness in audio signal processing. These results create a base on which to use the similar methods in the detection of any concealed engine failures by the application of sound

As noted by Garcia et al. [4], the ability of OpenCV to support both preprocessing and damage detection on images is yet another example of computer vision frameworks offering great flexibility in practical automotive applications. In addition, the recent studies have delved into the area of combining real-time data APIs, as well as predictive models, in dynamic decision-making. This integration enables the systems not only to test the constant conditions but also to introduce the variable ones like fuel prices and market values that vary.

Altogether, the literature confirms the possibility of AI based vehicle inspection systems that combine image analysis, audio analysis and real-time data to overcome the inefficiency in used car industry. Although the research conducted in the past has covered these techniques individually, there is a need to bridge the gap of developing a single platform through which engine diagnostics, damage detection, price prediction, and cost estimation can be implemented with great accuracy, and that is the role of the proposed Automobile Inspector.

2.2 Literature Table

Table 2. 1: A review of the relevant literature

Study	**Year**	**Model**	**Findings**	**Results**	**Dataset**
Diwan et al. [1]	2022	YOLO and successors	Comparative evaluation of YOLO architectures for object detection in automotive contexts. Identified strengths in detecting dents, scratches, and anomalies in real time.	High accuracy in object detection	Private
Rath et al. [2]	2020	Multiple Linear Regressi on	Applied regression for predictive modelling of dynamic outcomes. Relevant for predicting vehicle prices based on multiple parameters.	$R^2 = 0.94$	Private
Syed et al. [3]	2021	CNN, RNN	Comparative analysis for audio-based classification (voice detection). Demonstrated strong	Accuracy improve	Private

			potential for analysing sound anomalies.	ment over baseline	
Garcia et al. [4]	2015	OpenCV (Image Processin g)	Demonstrated effective preprocessing, feature extraction, and object detection using OpenCV.	N\A	N\A
Diwan et al. [5]	2022	YOLO + datasets	Comprehensive study of challenges and datasets in object detection using YOLO.	Accuracy improve ment	Public

2.3 Conclusion

The literature outlines the disruptive nature of the AI and ML in automating vehicle diagnostics and appraisals. Computer vision studies show that it is capable of detecting surface-level changes (e.g. dents, scratches) with a high degree of reliability and that audio classification models (CNNs and RNNs) can be useful in detecting engine malfunctions. Predictive models based on regression have application to market valuation and provide greater transparency in pricing.

Nevertheless, the research that has been done usually concentrates on single issues either engine diagnostics, damage detection, or pricing without offering an integrated platform. Automobile Inspector is proposed to fill this gap and bring all the engine sound analysis, visual inspection, automated price prediction and real time trip cost estimation into one intelligent system. This integrated method is consistent with the existing research results and leans the boundary to the development of an integrated system of AI-driven car inspection

CHAPTER 3

DESIGN AND METHODOLOGY

3.1 Introduction

The suggested Automobile Inspector system can be described as the intelligent platform that will be automated to examine used cars based on various factors including engine status, surface damages, estimated prices, and the estimated costs of the trip. The algorithm combines Computer Vision (CV), Machine Learning (ML), and Deep Learning (DL) models to make precise estimations, and real-time database management and user communication capabilities.

The development of the system is a planned procedure that comprises four significant stages:

First, data Collection and Preprocessing car images, engine sounds, and market prices data are collected and cleansed. Second, Model Development and Training model that is generated to be used in inspecting and detecting anomalies are image processing models (YOLO) and audio-based models (hybrid CNN).Then, system Integration a car information, user accounts, calculators, and chatbot modules are linked to provide a smooth workflow.

In the end deployment and User Interaction the final solution is deployed with web/mobile interface in real-time appraisals, estimation of costs and chatbot assistance.

This approach uses the lowest number of manual interventions, lowers the expenses of inspection, and offers clear, transparent, and real-time information about the state of vehicles and their price

3.2 ERD Diagram

The Automobile Inspector system (Figure 3.1) is an Entity Relationship Diagram (ERD), which depicts the most important entities, their attributes, as well as the relationships with each other that allow the platform to operate smoothly. The central participants and their functions shall be as follows:

• USER

Are the registered persons (buyers, sellers, mechanics or dealers) who make use of the system. Attributes such as user-id, name, email and password-hash are maintained by each user. The car records belong to the users and they have the capability of carrying out inspections, queries and calculations.

• CAR_INFO

Focal node which contains the data on vehicles such as make, model, year and other details. The cars are all associated with their owner (user_id) and are the fundamental record of the associated data like pictures, sounds, checks, and projections.

• CAR_IMAGES

Vehicle pictures are posted in stores (file path image type), which are subjected to computer vision models to identify dents, scratches. Every picture is associated to a certain car.

• CAR_ENGINE_SOUND

Has uploaded engine sounds (file_path). These are processed with the help of audio processing and ML/DL models (CNN) to detect possible engine malfunctions.

• INSPECTION

The inspection findings of car, which will include engine status, paint status and the date the car was inspected. It connects prediction and image/sound analysis with the car to have a general condition report.

• PRICE_PREDICTION

Leaves records with the predicted fair market value of a car in terms of its condition, miles and market data. The attributes are predicted price and model used. This allows sellers to rationalize prices and buyers to measure fairness.

• CALCULATORS

Provides other practical features like cost of the trip estimation and analysis of fuel consumption. Every calculation (calc_type, input_data, result) is identified to a particular user.

• CHATBOT_QUERY

Helping to interact with the chatbot which is powered by AI. Users are able to make queries (query_text) about vehicle inspection, price or how the system can be used and can get an answer (response_text) automatically.

The complete ERD is shown in **Error! Reference source not found.**.

Figure 3. 1: Entity Relationship Diagram (ERD)

3.3 Flow of Project

The framework of the Automobile Inspector system begins with the User who interacts with the application by either logging in or signing up. Upon successful registration, the user information is stored in the database for further processing. From the Main Dashboard, the system branches into four major functional modules: Car Price Prediction, Calculators, Car Inspection, and Chatbot Assistance.

- Car Price Prediction Module
  In this section, users can upload exterior and interior car images, upload engine sound files, and enter car details. The system then applies AI models to process this data and generates a predicted car price. The prediction result is displayed in real-time, helping buyers and sellers understand the fair value of the vehicle.
- Calculators Module
  This section provides additional functionalities such as the Mileage Calculator and Trip Cost Calculator. By entering relevant data, users receive instant results for fuel consumption and trip cost estimations, which assist in financial planning and operational decisions.

- Car Inspection Module

  This module focuses on vehicle condition analysis. The uploaded engine sound is analyzed to detect potential mechanical issues. Based on the sound analysis, the system makes an Engine Trouble Decision. If no trouble is detected.

- Chat Module
  The AI powered Chatbot enables users to interact with the system by submitting queries. The chatbot searches the queries database for possible answers. If a relevant answer is found, the chatbot provides an automated response. If not, a fallback message is displayed, ensuring that the user always receives some form of guidance or redirection.

This structured design ensures smooth navigation between modules and effective integration of AI functionalities for accurate predictions, condition assessments, and user support. The flow diagram (Figure 3.2) illustrates the entire process, highlighting the seamless interaction between users, data inputs, AI models, and system outputs.

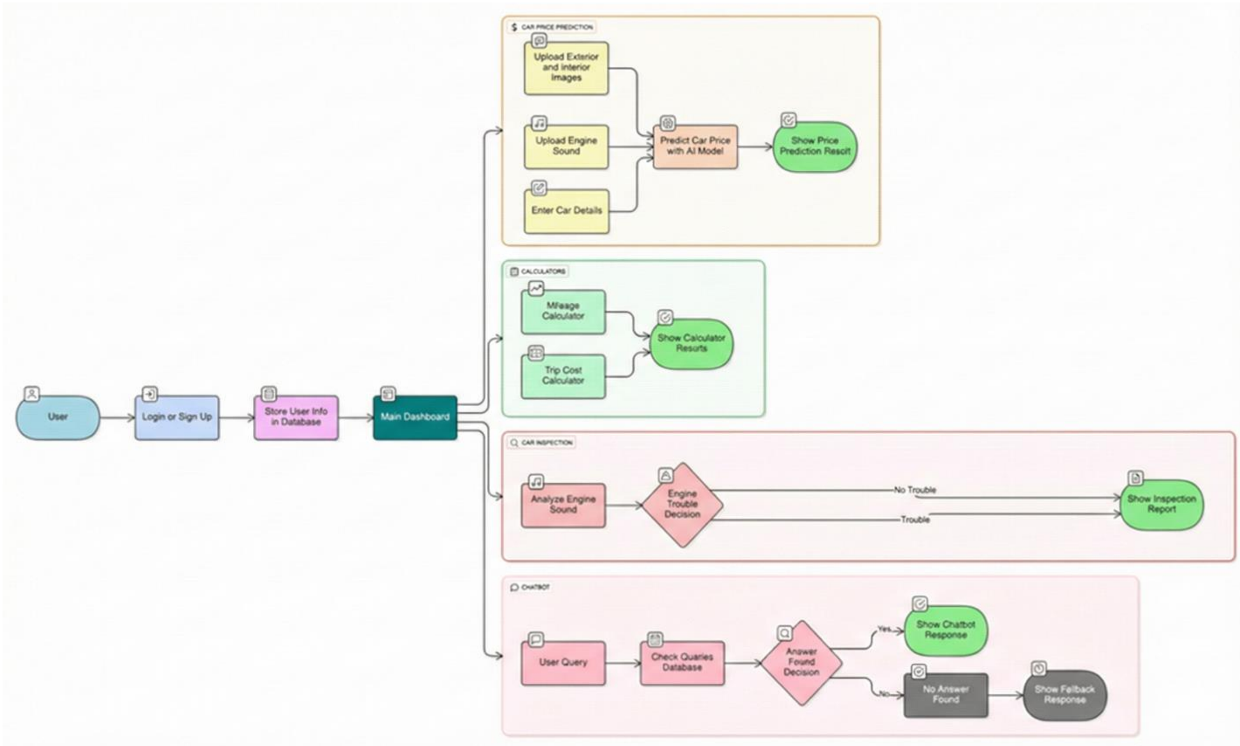


Figure 3. 2: Automobile Inspector flow Diagram

3.4 Data Collection and Pre processing

In the first phase, the system undertakes the data collection in creating training and testing data to the Automobile Inspector application. The data in this project is collected manually as opposed to the traditional systems that base the data on the available datasets to make them relevant and authentic. The IAAI.com gathers vehicle specifications, car images, and engine sounds, whereas YouTube provides the datasets on trouble engine sounds. This methodology will make sure that the data is reflective of real-life situations with differences in car models, sound conditions.

The gathered information is then in pre-processing so that it is regular and is ready to train the AI models. Pre-processing includes:

• Null Value Elimination Removal of incomplete or corrupted information records in car specifications, sound clips or images.

• Noise Reduction in Audio Cleaning engine sound recordings will eliminate the background noise in order to better detect the fault.

Image Normalization and Augmentation Image scaling, resizing and augmenting with car images to control the dataset and enhance the strength of the inspection model.

Labeling: When using supervised model training, data was categorized into some form of labels like original paint, engine trouble and no trouble.

These measures would ensure the quality of the input data is high and contributes to the reliability and accuracy of AI models in predicting the condition of vehicles, and engine diagnosis.

3.5 Automobile Inspector Model Training

Automobile Inspector system has several AI methods used in various tasks:

- Car Image (Dent Detection) Analysis

Computer vision systems like YOLO (You Only Look Once) are trained using labeled exterior images as a sign of scratches, dents. YOLO has the advantage of being real-time, which is why it can be utilized in mobile inspection systems such as the Automobile Inspector. Its model is trained in an iterative manner with several epochs optimizing the accuracy of this bounding box and the accuracy of the prediction of classes via the feedback of validation.

The trained YOLO model generalizes across car models, colors, lighting conditions as well as angles of a camera. In the course of inspection, after the user uploads exterior images, the YOLO model processes them and points out areas of damages. Such predictions are built into the price-estimation engine which automatically optimizes the valuation by the severity and nature of damage detected.

- Engine Sound Analysis

  To analyse audio, CNN (Convolutional Neural Networks) is applied to categorize car trouble sounds into 23 distinct mechanical problems. MFCC (Mel-Frequency Cepstral Coefficients) [6] features are processed before being inputted into the CNN model to be trained so that they can properly detect particular component malfunctions such as engine, transmission, brakes, and suspension issues.

  What we actually used:

  CNN only (no RNN/LSTM)

  23 different types of troubles (not only normal or troubled)

  MFCC characteristics (not spectrograms)

  Multi class (not binary) classification.

  Our model architecture:

  The model architecture employs the use of 2D convolutional layers with the batch normalization to obtain powerful feature extraction, which is succeeded by max pooling layers that minimize dimensionality at the expense of significant acoustic patterns. These features are reduced to a global average before turning them into dense layers with dropout regularization to reduce overfitting in the classification process. The last output node is a 23-neuron unit of particular feed-backs of parts of the car that can fail, allowing the system to discern specific mechanical issues such as: AC compressor failure, brake wear, engine rod knock, and so on, which allows the car valuation system to make accurate deductions on prices.

- Price Prediction Model

  A machine learning model (regression based) is trained to estimate the fair market value of a car based on car specifications (make, model, year, mileage) as well as inspection results. The model will give buyers and sellers fair price estimates.

The evaluation of the models is based on conventional performance measures:

Accuracy = (TP + TN) / (TP + TN + FP + FN)

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

F1-score = 2 × (Precision × Recall)/(Precision + Recall).

Where:

TP = True Positives (identified faults)

TN= True Negatives (correctly recognized cases of no issue)

FP = False Positives (marked issues false)

False Negatives (missed issues): FN.

In case the models do not match target accuracy levels, retraining is conducted using better pre-processing and augmentation of data.

- Chatbot Training and Implementation

  Alongside the inspection models, the system integrates an AI powered chatbot to provide user support. The chatbot is trained using FAQs and predefined responses related to car inspections, price predictions, and trip calculations. It uses RAG techniques to understand user queries and fetch relevant answers from its database. If no direct answer is found, a fallback response is generated to maintain interaction flow.

- System Testing and Accuracy Evaluation

  The models are subsequently tested on unseen data of IAAI and YouTube sources after training. Accuracy, precision, recall, and F1-score are used to measure the performance of image detection, audio classification, and price prediction. In case there is less than expected results, models are fine-tuned and re-trained and then integrated.

- Mobile Application Integration

The trained models are incorporated in Flutter/Dart mobile application that is the main user interface. The system is API connected so that the external images and sounds uploaded by the users could be processed by the car inspection models to provide real time inspection reports.

Price prediction model offers real time unbiased market prices.

Caculator module is used to calculate the cost of miles and the cost of the trip to help the user with ease.

Chatbot provides interactive support and instructions to the users.

This integration will guarantee the seamless user experience, the transparency in the evaluation of the vehicle condition, and the enhancement of buyer and seller decision-making.

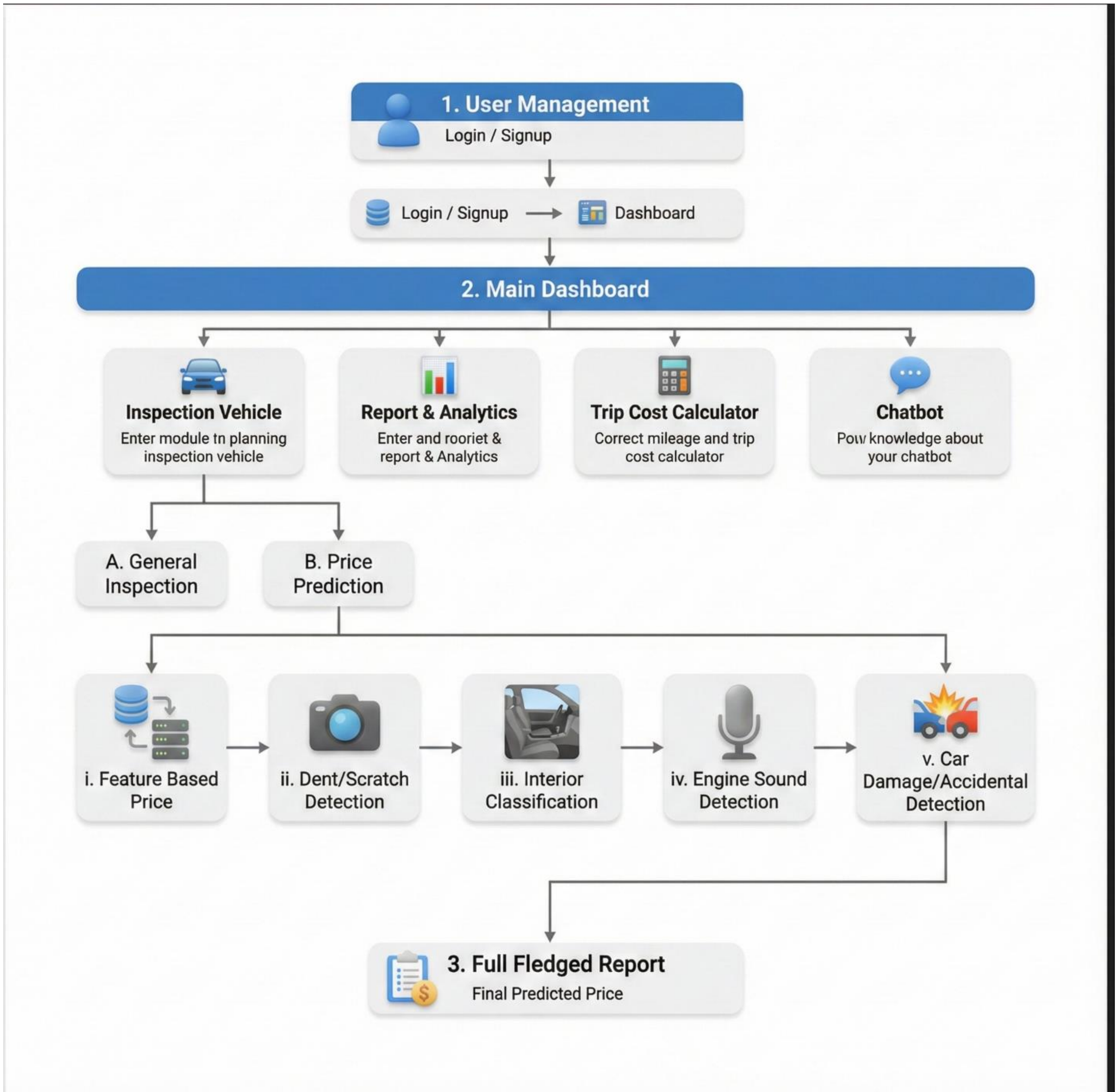


Figure 3. 3: Automobile Inspector Flow Diagram

CHAPTER 4

DATA AND EXPERIMENTS

4.1 Overview of Dataset

The data sets utilized in the project of the Automobile Inspector are obtained in various locations, as there are a number of independent inspections within the system which consist of price estimation with feature inputs, engine sound classification, interior condition examination, exterior harm discovery, accidental detection, and body paint verification. Since each of the inspections is a distinct machine learning model, the datasets also vary in the structure, volume, and type.

4.1.1 Price Prediction (Feature Based Dataset)

The price prediction model uses a structured dataset which has a number of thousands of vehicles records and each record has their respective fields which include make, model, manufacturing year, mileage, engine capacity, transmission type, fuel type, and the respective market value. The data is presented in a table format and it has both numerical and categorical variables. Being directly descriptive of the specifications and signs of condition of a vehicle, these fields can be employed successfully in a supervised regression model to approximate a base price of a car. The data will give a sound basis on which to study the trends in prices of various kinds of vehicles, their segments, and utilization of the vehicles.

4.1.2 Engine Sound Dataset

The engine sound analysis module is conditioned on a set of audio data made of 305 audio samples of car trouble sounds of 23 different mechanical faults. All audio clips are standardized into one-three seconds to provide equal processing and comparison. There are recorded markings of a vast array of mechanical parts including the engine block, valves, belts, transmission, brakes, suspension, and electrical systems. Such labels enable the model to relate particular acoustic signatures with particular mechanical problems. The audio samples are converted to MFCC (Mel-Frequency Cepstral Coefficient) features [6], which allows the system to learn the unique frequency and time-related characteristics that reflect mechanical issues related to the overall condition of a vehicle and price determination.

4.1.3 Interior Condition Dataset

The data that was used to assess the interior of vehicle are car cabin images that are gathered by using diverse web databases, open-source data sets, as well as manually selected samples. Images of the state of the car seats, dashboard, steering wheel, flooring, and the cabin area around are provided in the records of the dataset. Such samples consist of clean interiors and interiors that have visible imperfections like stains, damaged upholstery, dust, and sunken parts. The dataset has also been prepared with the cases taken under the variation of light, quality of cameras and angles to ensure that the model is strong in the real world.

Each record of the data set will have a picture and a corresponding label as to the condition of the interior. These labels are used to classify the state of "Luxurious, Good, Average or Poor state. The exertion of different interior categories enables the model to make generalization at various degrees of cleanliness and wear and tear. The dataset has a fairly balanced representation of all the categories with a slightly larger number of images depicting good-condition interiors than severely worn interiors. This is in line with reality on the ground because users will be more inclined to post the image of a vehicle that is in proper condition and maintained.

The variety of the dataset renders it useful in the creation of computer vision models aimed at the evaluation of automobiles. The pictures are taken in diverse types and models of the vehicle, inside planning, seat materials and environmental differences, which assists the model to acquire significant and generalized trends. This data thus provides a real world of training on the interior quality analysis and plays a role towards a better and tighter process of inspection at the Automobile Inspector system.

4.1.4 Exterior Damage (Dent/Scratch) Dataset

The exterior damage detection are trained on an image dataset, which is obtained on the basis of real-vehicle accident listed vehicles offered on the automobile auction sites, including IAAI and Copart, among other samples obtained online through sources of car-inspectors. The data has front, back, left, and right side images that are taken under varying lighting conditions, camera angle and surface reflection so that the model can be trained with real life situations. Every car adds several high-resolution images, which are usually four or eight. The dataset is comprised of balanced elements of undamage surfaces, dented panels of different depths, light and deep scratches, and pictures of mixed types of damage. There are also some challenging cases in some of the samples like a strong reflection, shadows and metallic paints finish which assist the model to deal with complex backgrounds [7]. The labels on all images were created manually by placing bounding boxes on the visible damage areas that enable the detection model to learn to localize the damage with precision instead of just learning to view an image as damaged or clean. The data was then split into training, validation and test data sets. The variety of car models, settings, and image quality enables the YOLO-based detector to be generalized in a large-scale in processing user-posted pictures of vehicles in an inspection.

4.1.5 Accidental/Structural Damage Dataset

These images specifically contain vehicles with frame damage, broken parts, or signs of major impact [7].

Labels include Non Accidental, Minor Accident, and Major Accident.

This dataset helps the model determine whether the car has undergone structural or safety related damage.

Across all inspection categories, each dataset supports a different ML pipeline. Together, they create a complete set of real world training resources for building an AI based automobile evaluation system. These datasets enable analysis of mechanical health, physical condition, and overall market value foundations for automated car inspection and price prediction.

4.2 Feature Extraction and Preprocessing

Since each inspection category uses a different type of data text, audio, and images the feature extraction pipeline varies according to the input format.

4.2.1 Feature Extraction from Structured Vehicle Inputs

The feature based price estimation module processes structured vehicle information such as make, model, year, mileage, fuel type and transmission. Before these inputs can be used by the regression model, the data is preprocessed to ensure consistency. Missing values are handled, and categorical attributes such as brand, model and fuel type are encoded using One Hot Encoding so the model can interpret them numerically. Numerical fields, including mileage and engine capacity, are normalized or standardized to maintain uniform scale across all features. After preprocessing, the cleaned and encoded feature set serves as input to the machine learning regressor, which computes the vehicle's base price before adjustments from other inspection modules.

4.2.2 Audio Feature Extraction for Engine Sound Analysis

The engine sound analysis module relies on a structured audio preprocessing pipeline designed to extract meaningful acoustic features from raw engine recordings. Before feature extraction, every audio sample is normalized and converted to a fixed duration of three seconds to ensure uniformity across the dataset. This standardization removes variations in loudness and eliminates the effect of different recording lengths. After normalization, Mel-Frequency Cepstral Coefficients (MFCCs) are extracted using the Librosa library, with each audio sample represented through forty MFCC coefficients. These coefficients effectively capture the spectral envelope and frequency characteristics of mechanical sounds, which are essential for distinguishing normal engine behavior from various fault conditions.

Once MFCC extraction is complete, each audio recording is segmented into approximately 130 time steps, allowing the model to learn the temporal dynamics of the engine sound. The resulting MFCC matrix undergoes feature normalization [6], where the mean is removed and the values are scaled according to the standard deviation. This step reduces disparities caused by recording devices or background noise. The processed audio is represented in a 40×130 MFCC format, which is then reshaped into a three-dimensional tensor suitable for CNN-based deep learning models. This representation captures both the frequency content and temporal progression of the engine signal, enabling the network to accurately classify mechanical issues such as knocking, misfiring, rattling or belt-related problems.

For all samples, the audio is processed using a uniform sample rate of 22,050 Hz, with each clip fixed to a length of three seconds. The final feature map of size 40×130×1 serves as the model's input, providing a dense and consistent feature structure that supports effective engine health prediction based on sound characteristics.

4.2.3 Image Feature Extraction for Interior Condition

To transform raw interior images into a form that can be understood by machine learning algorithms, several preprocessing and feature extraction steps are performed. Vehicle interior images contain unstructured visual information, so the system first standardizes and enhances them before passing them to the deep learning model. This process ensures uniformity in terms of image size, lighting normalization, and noise reduction, which improves the model's ability to detect relevant patterns.

The preprocessing phase begins with resizing all images to a fixed dimension suitable for deep learning architectures. Normalization is then applied to scale pixel values so the model can learn features consistently across all samples. Any unnecessary background noise or distortions are minimized during this step. Data augmentation techniques such as rotation, flipping, shifting, and brightness adjustment are also applied to increase the model's exposure to different visual variations. These operations make the model more robust and capable of handling real user-submitted images that may not always be perfectly aligned or evenly lit.

After preprocessing, images are passed through a convolutional neural network (CNN) that automatically extracts hierarchical features. Early layers in the network capture low-level visual characteristics such as edges, textures, and color gradients. Deeper layers extract more complex and abstract representations, including patterns related to seat wear, surface cracks, stains, dashboard cleanliness, and overall cabin condition. This layered extraction enables the model to differentiate between clean interiors and interiors that show signs of neglect or damage.

Feature maps generated from the CNN layers are then used by the classification component of the model. Based on the learned visual patterns, the system categorizes the interior as “Good,” “Moderate,” or “Poor,” and later applies price adjustments accordingly. These extracted features form the core of the automated decision-making process in the interior inspection module. Through this structured pipeline, the raw interior images are transformed into meaningful numerical representations that support accurate and efficient vehicle interior condition assessment.

4.2.4 Dent and Scratch Detection Feature Extraction

The dents and scratches detection unit is based on the Ultralytics YOLO framework, which is used to extract features, localize objects and classify damage in one step. The model is provided with labeled car exterior images with minor and major damage conditions. The dataset was preprocessed through a structured preprocessing workflow before training to make it consistent and accurate. All the pictures were scaled to the size of the input of YOLO which is usually 640x640 pixels but retained the aspect ratio to ensure that the damaged area is not distorted. The preprocessing activity involved normalization of pixel values to minimize the effects of lighting difference, reflection and color disparities usually present in vehicle images. The annotations of bounding-box of dents and scratches were formatted to the requirements of the YOLO, with conditions of class indexes and normalized coordinates of the center point, width, and height of the detected area. The model training was further enhanced with the inbuilt data augmentation pipeline of YOLOv8 so as to enhance robustness and generalization. Such augmentations added controlled distortions to the training images including random horizontal flips, scaling, cropping and color manipulations, which assisted the model to understand how dents and scratches look with various lighting, angle and environments. The addition of four training images together as one composite sample through a method known as mosaic augmentation also improved the capacity of the model to detect complex patterns of damage scattered across various body parts. After preprocessing and augmentation, the convolutional backbone of YOLO was used to do hierarchical feature extraction at a variety of depth levels. The fine edges, shallow surface lines and minor scratch textures were captured in the fine layers of the model. The medial layers were concerned with curvature, shadow effects and ridges. The bottom of the layers were taught the higher structural patterns that distinguish between the actual damage and the reflections, glare and natural body panel curves. This multi-scale feature representation enabled the model to identify (both subtle scratches and larger dents) well, and thus it is applicable in practical vehicle inspection situations.

4.2.5 Damage and Accidental Feature Extraction

The damage and accident detection module processes exterior car images to identify dents, scratches, cracks, broken parts, and signs of collision repair. Images are first resized, normalized, and formatted into the YOLO annotation style to standardize input for the detection model. YOLO's backbone extracts multi-scale visual features such as panel irregularities, sharp edges, deformation patterns, and shadow distortions that commonly occur in accidental damage [7]. The model uses these learned spatial features to localize damaged regions instead of classifying the entire image, allowing more accurate assessment of the severity and position of the impact. This preprocessing pipeline ensures that the detector remains consistent across different lighting conditions, camera angles, and vehicle surfaces.

CHAPTER 5

RESULTS AND DISCUSSIONS (or USER MANUAL)

5.1 Results of different model

5.1.1 Interior Condition Classification Model

The model of interior inspection demonstrates an excellent level of performance in all four categories, namely Poor, Fair, Good, and Luxurious. The overall accuracy of the model is 0.8989 which means that it is a reliable model in classifying interior condition based on user uploaded images. The classes of Good (Well Maintained) and Luxurious (Top Condition) give F1-scores of 0.9046 and 0.9307 respectively, which display a high degree of consistency in the recognition of interiors in good condition. The poor (Needs Attention) category has a perfect recall of 1.0, and this means that the model is effective in detecting all the badly damaged interiors. The acceptable accuracy, memory and weighted averages confirm that the model is very stable and works effectively on real life variations of interior images. The results are also shown in **Error! Reference source not found.**.

Table 5. 1: Performance results on Interior condition classification model

Class	Precision	Recall	F1-Score	Support
Poor (Needs Attention)	0.9512	1.0000	0.9750	39
Fair (Average Condition)	0.8415	0.9103	0.8745	379
Good (Well Maintained)	0.9289	0.8816	0.9046	608
Luxurious (Top Condition)	0.9400	0.9216	0.9307	102

The project applies a complex system of deep learning piping to classify the condition of the car interior and bases this approach on a transfer learning framework based on a pre trained version of DenseNet121. A bespoke-designed classifier head improves the architecture, which is four-layered fully connected network with dimensionality reduction that is progressive (1024 1024 512 256 4 neurons). The model inputs high-resolution (512x512) RGB data and uses ReLU activation functions in all hidden layers and is combined with a progressive dropout regularization scheme 0.6, 0.4, 0.3, 0.2) and batch normalization of each dense layer to provide stable and efficient training. It produces raw logits on four target classes; Poor_(Needs attention), Fair_(Average condition), Good_(Well maintained) and luxurious_(Top condition) using implicit softmax through a bespoke Guaranteed Focal loss function and a gamma of 2.0 to overcome the imbalance between classes.

The advanced techniques are also used in refining the training methodology, and these include the Automatic Mixed Precision (AMP) and the AdamW optimizer (lr=0.001) that are scheduled using a OneCycleLR policy. The approach uses a Guaranteed Augmentation System and an Enhanced Guaranteed Smart Data Balancer, where five different augmentation strategies are applied to the dataset to achieve artificial balance by using flip/rotate, color jitter, affine/perspective, noise/blur and combined transformations to achieve a predefined distribution of 700-900 images per class. A Professional Car Level Prediction Engine, used in order to make an inference, follows a multistrategy fusion mechanism. This engine derives profound features on all pictures of a car and employs a collection of fusion approaches, such as a factor of confident weighted feature fusion, majority-votes, probability averaging and a superior attention-based fusion, to combine image level forecasts in one and robust car level findings, practically averting data-leakage and validating real world utility.

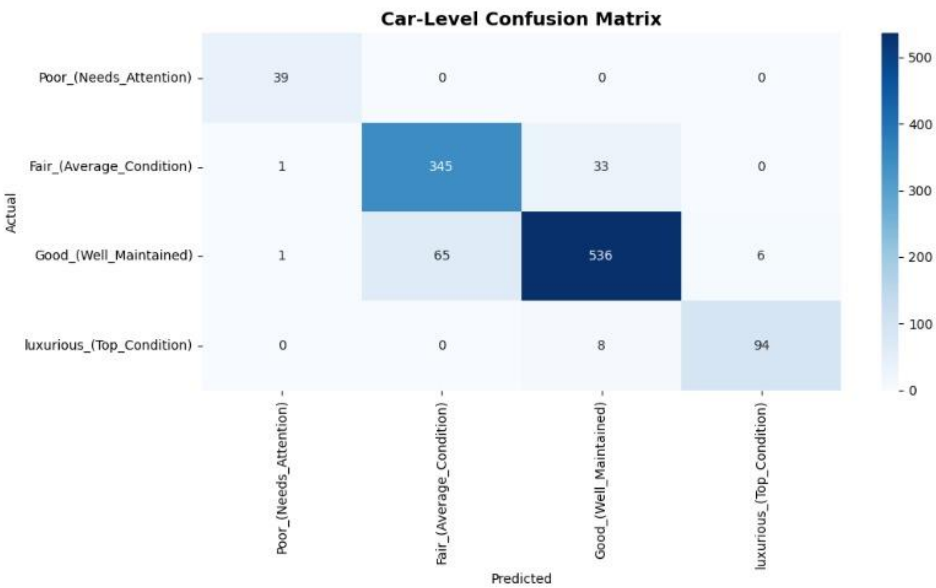


Figure 5. 1: Confusion Matrix for Car Levels (Validation)

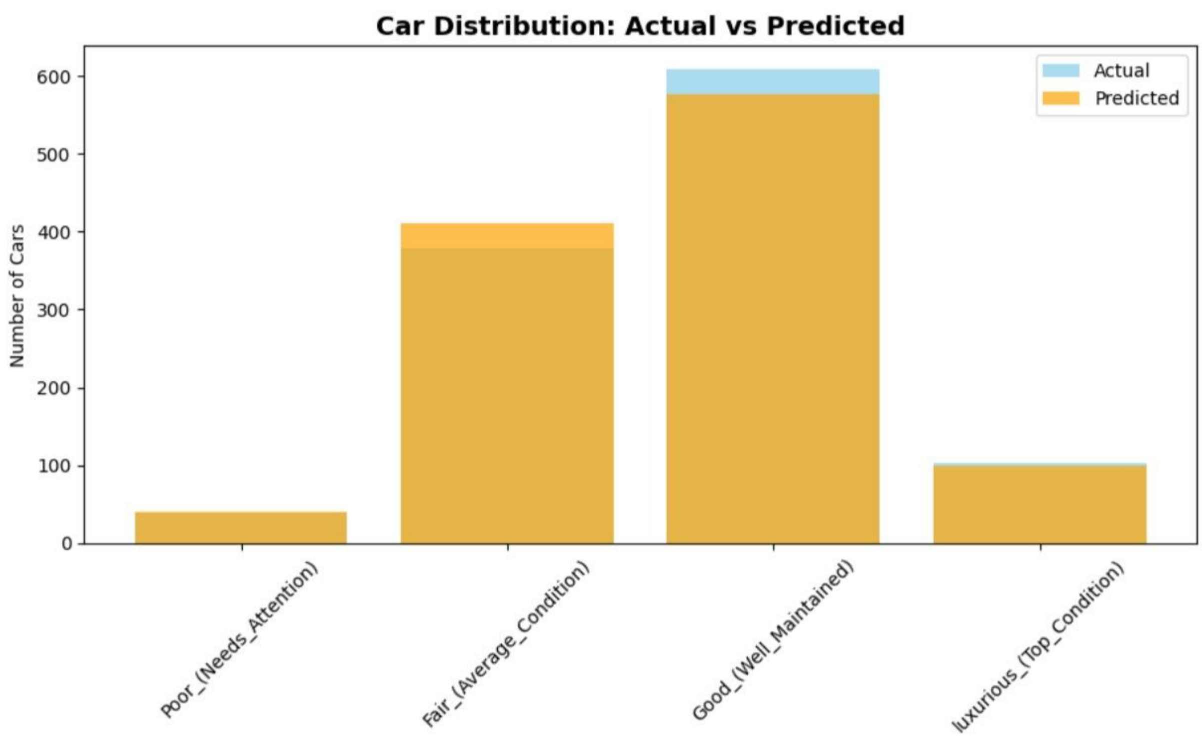


Figure 5. 2: Car Distribution for Actual vs Prediction (Validation)

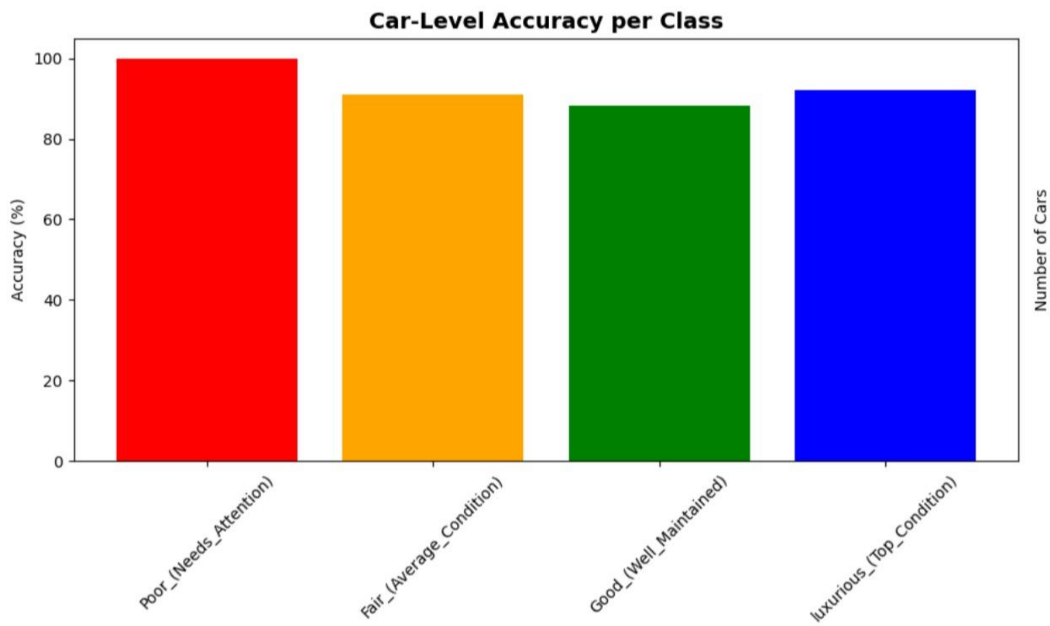


Figure 5. 3: Accuracy per Class (Validation)

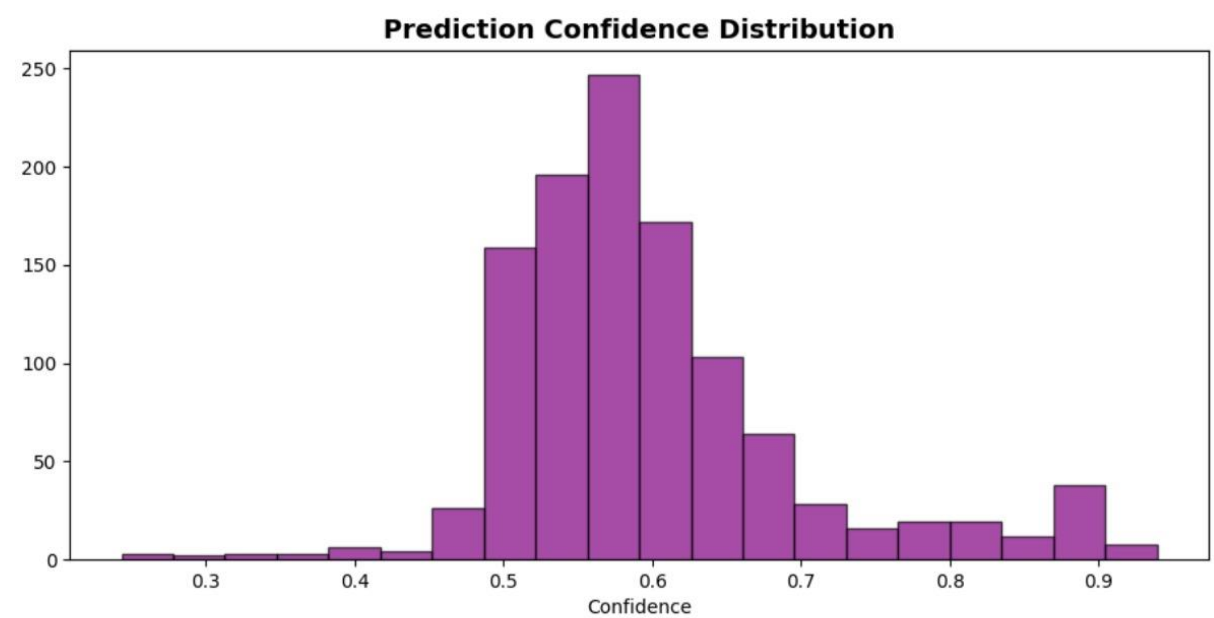


Figure 5. 4:Confidence Prediction (Validation)

5.1.2 Exterior Dent and Scratch Detection Model

The Exterior dent and scratch detector module is one of the fundamental parts of the Automobile inspector system. This model is trained with a semi-supervised YOLO-based pipeline which is created to identify 24 location-specific classes of scratches and dents. The idea is not to just determine how bad the damage is (minor, medium or major) but to determine which specific part of the vehicle was damaged, i.e. the door, fender, bonnet, bumper, roof, mirror, pillar, or trunk.

The training process is further differentiated into two phases: teacher model and student model. The manually labeled dataset is first used to train the teacher model (YOLOv8-L). This model then produces pseudo-labels to a big collection of unlabeled car images. Both labeled and pseudo-labeled samples are combined to create a larger training set of the student model. The largest and most powerful variant of the model is a student model (YOLOv8-X), which picks up this augmented dataset and has much better generalization performance.

Other automated preprocessing tasks carried out by the system include image resizing, image normalization, and conversion of labels into the YOLO format. Data will consist of small scratches, medium scratches, large scratches and mixed damages on more than one car surface. This variety enables the model to train the variations between superficial marks, deep scratches, and lines of dents and high-impact damage when the lighting, reflections, and camera angles change.

The final trained YOLOv8-X model is also accurate in all performance metrics. Its overall precision, recall, and mAP50 and mAP50-95 are 0.921, 0.898, 0.915 and 0.884 respectively, which are very good results in identifying and localizing exterior damage. Some classes, particularly minor and major bumper scratches, have the precision score near to 1.0, which demonstrates that the model is very reliable in detecting the actual damage without any false positives.

The model also has a steady performance on all 24 classes despite the problematic cases in terms of shadows, metallic paint reflections, or curved surfaces. Validation experiments indicate that the detector is able to reliably detect damaged regions and wrongly ignore clean surfaces. The findings confirm the idea that the exterior inspection system based on the YOLO is strong and can be applied to the real world when it comes to assessing vehicles, the image conditions of which differ greatly depending on the user.

Table 5. 2: Performance results on Exterior Dent Scratch classification model

Class	**Instances**	**Precision**	**Recall**	**mAP50**	**mAP50-95**
minor_scratch_bumper	47	0.994	0.957	0.978	0.919
minor_scratch_trunk	1	0.000	0.000	0.000	0.000
medium_scratch_door	1	1.000	1.000	0.995	0.895
medium_scratch_fender	1	0.834	1.000	0.995	0.995
medium_scratch_bonnet	9	1.000	0.778	0.889	0.881
medium_scratch_bumper	19	1.000	0.926	0.974	0.933
medium_scratch_roof	5	0.960	1.000	0.995	0.982
medium_scratch_mirror	11	1.000	1.000	0.995	0.946
medium_scratch_pillar	1	1.000	1.000	0.995	0.895
major_scratch_fender	2	1.000	1.000	0.995	0.995
major_scratch_bonnet	1	1.000	1.000	0.995	0.995
major_scratch_bumper	67	1.000	0.925	0.963	0.944
major_scratch_roof	21	0.955	1.000	0.995	0.963
major_scratch_mirror	9	1.000	0.778	0.889	0.889
major_scratch_pillar	17	1.000	1.000	0.995	0.970
major_scratch_trunk	4	1.000	1.000	0.995	0.934

The project is based on a complex semi-supervised learning pipeline using the YOLOv8 (You Only Look Once version 8) object detection architecture to classify the location of an automotive scratch as a scratch. It uses a dual model teacher student design, and a preliminary YOLOv8l (large) teacher model is trained upon 100 epochs with labeled data and a second YOLOv8x (extra-large) student model trained upon 150 epochs using a mixture of labeled and pseudo-labeled data. The architecture takes advantage of transfer learning by using both models with pre-trained weights of the COCO datasets (yolov8l.pt and yolov8x.pt), which immediately gives it strong features extraction features.

The YOLOv8 model has a CSPDarknet backbone which uses SiLU (Swish) activation functions in all layers, and takes 640 indicators, six 640 indicators of RGB input to a sequence of convolutional blocks with channel expansion. The neck part employs PAN-FPN (Path Aggregation Network - Feature Pyramid Network) to make predictions at three scale feature fusions on three detection heads (80×80, 40×40, and 20×20 grids), and the anchor-free detection head that produces three predictions of 24 location-sensitive scratch classes (minor scratch door, medium scratch fender, major scratch bonnet, etc.) of automotive components combinations. The output layer produces 87 channels per detection (4 bounding box coordinates with no activation, 1 abjectness score with sigmoid activation and 24 class probabilities with sigmoid activation) at each of the three detection scales, so that the output is 25200 predictions per image.

The training methodology uses a progressive learning with teacher model of 8-batch size, 20-epoch patience (early stop) and the standard YOLOv8 model optimization, whereas the student model uses the improved parameter setting of smaller learning rate (0.001), larger patience (25 epochs), and more extensive regularization in terms of weight decay (0.0005). Semi-supervised pipeline uses the 0.7confidence threshold to give pseudo-labels to unlabeled data and effectively increases the training data and therefore generalization of the model. This advanced methodology allows an accurate localization and categorization of scratches on selected parts of the car without sacrificing computational efficiency due to the single-stage detection framework and feature hierarchy optimization of the YOLO framework.

5.1.3 Engine Sound Classification Model

The car sound classification system uses an architecture of Convolutional Neural Network (CNN) to train and classify 24 different types of mechanical faults in vehicles. The model is used to process audio signals by a complex set of feature extraction processing in which Mel-Frequency Cepstral Coefficients (MFCCs) are calculated to represent the spectral properties of mechanical sounds. The deep learning model is fed on these 40 dimensional MFCC features which are a result of 130 time frames of 3 seconds audio clips.

The CNN model has several 2D convolutional layers that have batch normalization to ensure a strong feature learning, and max-pooling layers that reduce the dimensions. The learned features are condensed with global average pooling and eventually classified with dense layers using dropout regularization. This architecture is successful in learning hierarchical patterns of audio data, and allow the discrimination between such subtle acoustic signature of various mechanical problems.

The model has a good performance on all 24 mechanical fault categories, with a precision and recall range mostly found between 0.72 and 0.82. This stability shows the effectiveness of the model in identifying subtle sound configurations related to different car issues. Much better performance is registered on the components such as "Failing Water Pump" (F1: 0.767), "AC Compressor" (F1: 0.805), "Ball Joint" (F1: 0.809) and "Brake Squeaking" (F1: 0.778) with the model recording F1-scores above 0.76.

The macro average general accuracy of 78.1% proves that the model can be used to classify short audio segments as mechanical faults. This level of performance gives a good ground to introduce price deduction threshold of the valuation of vehicles, where identified problems can be quantitatively associated to the approximated repair costs and their value corrections.

This recognition ability allows automated vehicle evaluation systems to detect certain mechanical issues by using acoustics analysis only. The capability to distinguish 23 distinct types of faults using the short audio samples is especially useful with used cars valuation, pre-purchase inspections, and maintenance diagnostics to have the objective data-driven information on the basis of which the pricing decisions can be made.

Table 5. 3: Performance results on Engine Trouble Sound condition classification model

	Class	Precision	Recall	F1-Score	Accuracy	Support
0	AC compressor	0.7874	0.8229	0.8048	0.7874	23
1	Alternator bearing	0.7831	0.7908	0.7869	0.7831	23
2	Bad alternator	0.7741	0.7475	0.7606	0.7741	23
3	Bad transmission	0.7165	0.7428	0.7294	0.7165	23
4	Ball joint 2	0.8012	0.8179	0.8094	0.8012	23
5	Battery problem	0.7659	0.8019	0.7835	0.7659	23
6	Brake squeaking 2	0.7892	0.7665	0.7777	0.7892	23
7	Brake wear	0.7496	0.7259	0.7376	0.7496	23
8	CV joint 2	0.7831	0.7851	0.7841	0.7831	23
9	Differential noise	0.7909	0.7744	0.7826	0.7909	23
10	Dying battery	0.8018	0.7729	0.7871	0.8018	23
11	Engine belt	0.7328	0.7230	0.7279	0.7328	23
12	Engine knock on idle	0.7442	0.7654	0.7547	0.7442	23
13	Engine mounts	0.7264	0.7274	0.7269	0.7264	23
14	Engine problem indication	0.7537	0.7195	0.7362	0.7537	23
15	Engine problem indication 2	0.7547	0.7299	0.7421	0.7547	23

	Class	Precision	Recall	F1-Score	Accuracy	Support
16	Engine rod knock	0.7170	0.7492	0.7328	0.7170	23
17	Engine starter 2	0.7796	0.8037	0.7915	0.7796	23
18	Failing water pump	0.7832	0.7517	0.7671	0.7832	23
19	Fuel warning	0.8062	0.8014	0.8038	0.8062	23
20	Inner tie rod	0.7210	0.7206	0.7208	0.7210	23
21	Low engine oil	0.7149	0.7442	0.7292	0.7149	23
22	Low power steering fluid	0.7804	0.7931	0.7867	0.7804	23

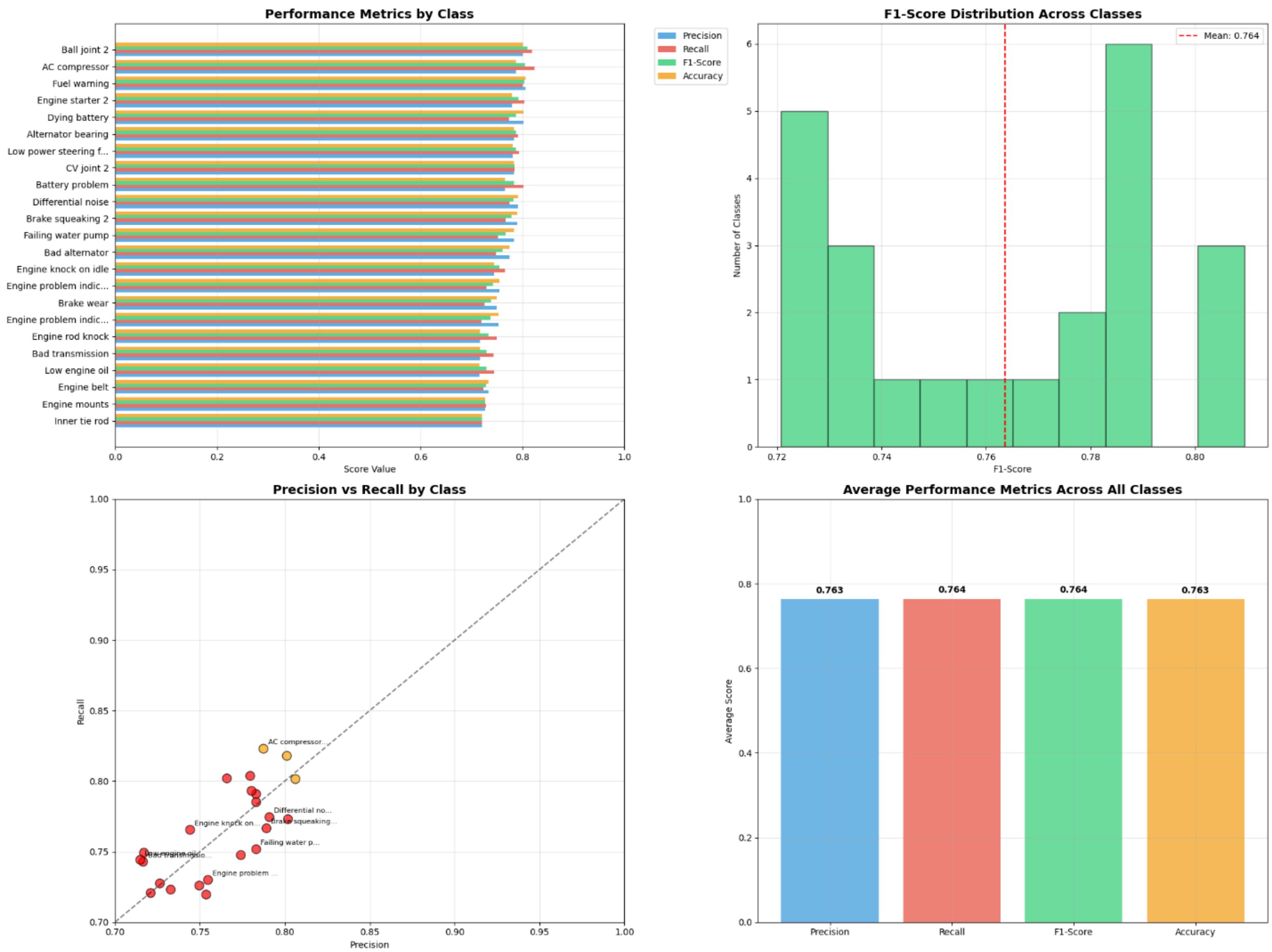


Figure 5. 5: Performance evaluation metrics for the vehicle fault detection model (Validation)

The current project applies a complex Convolutional Neural Network (CNN) in the classification of automotive sounds with the utilization of MFCC (MelFrequency Cepstral Coefficients) features generated by audio signals. The architecture uses a single-handwritten CNN without transfer learning that uses 40×44×1 MFCC features as input and uses a sequential network with two convolutional blocks of 32 and 64 filters respectively each using 3x3 kernel with ReLU activation functions and 2x2 MaxPooling layers and Batch Normalization to stabilize training. The backbone of the feature extraction is connected to a Global Average Pooling layer to reduce the spatial dimensions then a fully connected 128 neuron (with ReLU activation) then a final 23 neuron (corresponding to the number of classes of car trouble) and Softmax activation probability distribution.

The training methodology is based on manual splitting of datasets with stratified sampling to guarantee the representation of classes using 100 training cycles with a batch-size of 8 and the Adam optimizer (learning rate: 0.0005) and extensive callbacks (Checkpointing of models (monitoring val accuracy), Early stopping (patience: 20 epochs), ReduceLROnPlateau (patience: 8 epochs). The process of regularization is done with Dropout (0.3 rate following both Global Average Pooling and the 128-neuron dense layer) and L2 regularization using weight decay, and the loss function uses Sparse Categorical Crossentropy with integer-encoded labels. The input pipeline reads and extracts MFCC using librosa (22050Hz sampling rate, 3 seconds duration, 40 coefficients) and z score-normalized input audio to produce 40×44 maps features, which are channel-expanded to be compatible with CNNs. This system of end-to-end audio analysis shows test accuracy of 100 per cent to detect 23 different automotive problems such as failures of AC compressor, problems of bearings in the alternator, malfunctioning of the brake system, and other anomalies related to the engine due to the use of optimized spectral feature learning and strong architectural design.

5.1.4 Text Input Regression Model

The regression outcomes regarding the structured features-based price estimation reveal that the ensemble model is the most effective. The XGBoost model has an RMSE of 2189.28 and R 2 of 0.913, indicating that the model is effective in estimating the prices of vehicles in real-life scenarios. The same stability is also validated by cross-validation with a mean CV R 2 of 0.896. These findings indicate that XGBoost does better in capturing non-linear associations among the characteristics of the vehicle (mileage, model year, fuel type and engine capacity) compared to the standard regression models. This renders it quite suitable in the creation of base price, prior to condition based deductions.

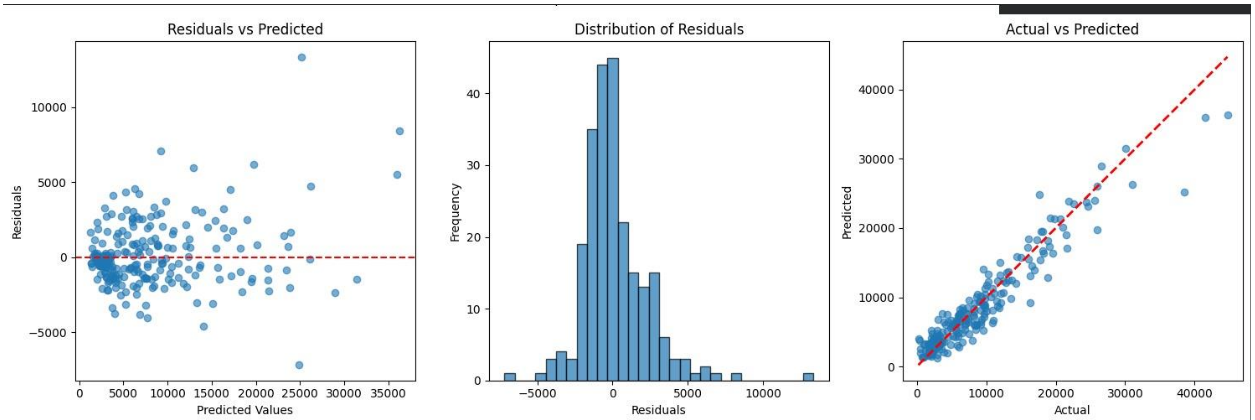


Figure 5. 6: Diagnostic plots residual analysis and prediction accuracy

5.1.5 Exterior Damage Detection Model

The professionally engineered YOLOv8-based pipeline, trained on the model of the Car Damage and Accidental Detection, is oriented on the full-body accident damage and the damage, which is specific to the right and left sides, e.g., left and right impact. The model has been trained following a long series of datasets refining steps and re-structuring the raw accidental-car images into clean images followed by balancing them with the help of an advanced augmentation system, which is an Albumentations driven system. This augmentation module used varying strength of the transformations based on the damage: heavy geometric and photometric distortion in the case of side impacts, moderate transformation in the case of front and rear accidents, and less significant modifications in the case of classes with sufficient sample. This balancing served to eliminate bias in a class and the model was able to learn at a steady rate across all the types of accidents. The training pipeline was based on a two-model approach.

The primary model (YOLOv8m) was trained on all the accidental-damage classes such as front damage, back damage, left damage, right damage and other accident-related classes. On a focused dataset generated out of the main set, a specialized model was trained separately on left- and right-damage. This enabled the system to enhance the identification of side impacts, which are usually challenging due to the different camera angles and coverages. A mechanism of ensemble was introduced to combine the predictions of the two models. When left or right damage is detected by the specialized model, it gives more confidence weight, which results in more stable and accurate detection in difficult angles.

The model demonstrated high localization and good classification performance of all classes of the accidents during the evaluation. The improvement in the detection of the rare categories was of great help owing to the augmentation strategies since initially the categories had extremely low representation in the dataset. The per-class performance measures and the confusion matrix showed that the model could identify most types of accidents with high precision and recall. The primary YOLO model proved to be a stable general detector and the damage-specific side model increased the sensitivity of left and right impact areas. The ensemble created cleaner predictions, fewer false positives and improved calibration of confidence particularly on complex accidental cases. In general, Car Damage and Accidental Detection Model is stable in the detection of accidents in the vehicle body and heavily relies on the professional augmentation pipeline strategy, balanced dataset creation, and dual-model ensemble method. This will enable the Automobile Inspector system to identify front, back, left, and right accidental damage in real-life pictures with a uniform performance.

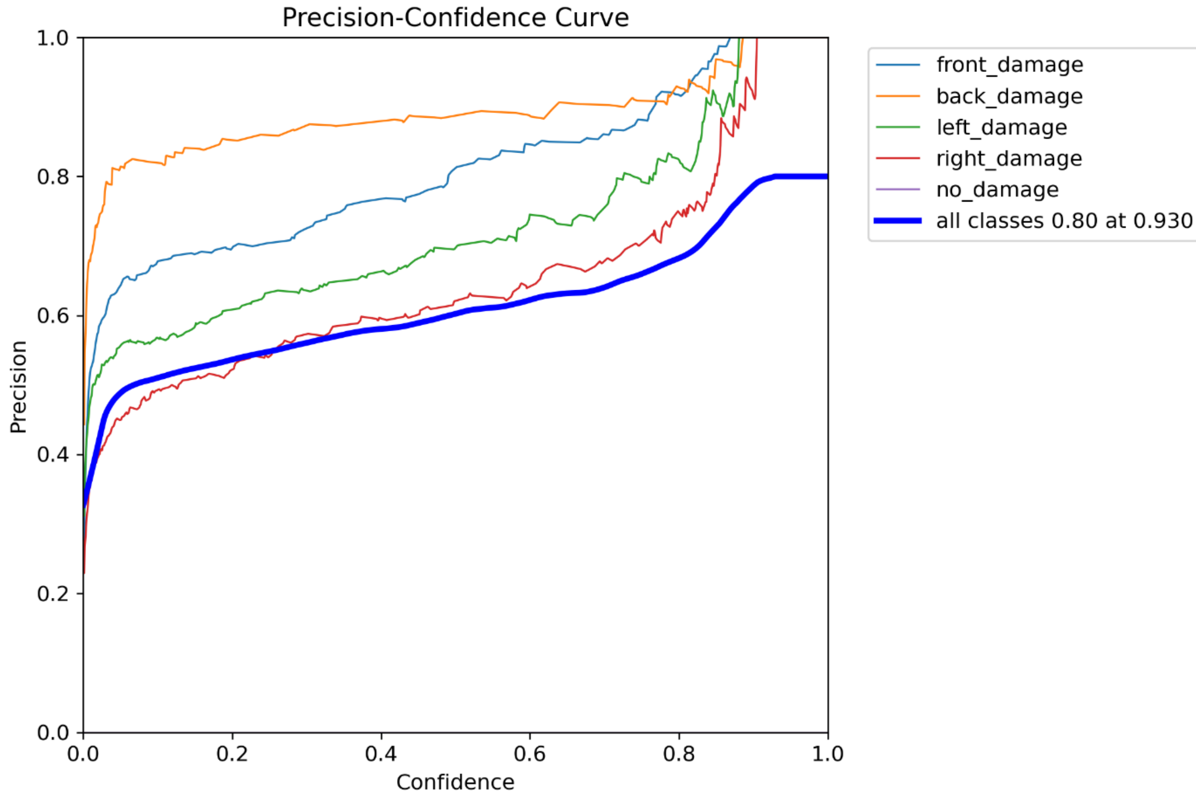


Figure 5. 7: Precision-Confidence curve illustrating detection performance

This project uses an object detecting pipeline with advanced YOLOv8 architecture based on extensive transfer learning using the pre-trained weights of COCO (yolov8m.pt). The system uses pure convolutional neural network with CSPDarknet backbone with SiLU (Swish) activation functions per pixel where it processes 640 640 RGB input in a series of CSP blocks with channel expansion of (64>128> 256 -5 1024) channels. It has a PAN-FPN architecture, which features multi scale feature fusion across three detection heads with each containing 40x 40x 20 grids (80x) and an architecture where one detection head produces 27 channel per detection (4 bounding box coordinates, no activation), 1 objectness score (sigmoid activation), and 4 class probabilities (sigmoid activation per class) at the end of the target classes (front-damage, back-damage, left-damage, and right-damage).

The training model deploys a dual-model framework where the primary model is trained with 100 epochs (batch size 8, learning rate 0.01) and a specialized left/right damage model is trained with 40 epochs (batch size 16, learning rate 0.001) with the enhanced parameters such as higher box loss weight (10.0) and aggressive left right flipping augmentation (0.8). The system has professional augmentation engine with damage specific pipelines, smart class balancing with focus of about 350 instances per class and ensemble prediction system with fusion of model outputs based on confidence weight and non-maximum suppression. Interestingly, there exist no so called connected layers within the architecture as everything consists of convolutional operations that are optimally regularized with weight decay (0.0005), batch normalization and a large use of data augmentation instead of dropout. It is a multi-scale object detector with a strong foundation to support automotive damage localization and classification.

5.1.6 Splash Screen

Figure 5. 8:Automobile Inspector Splash Screen

5.1.7 Authentication Screen

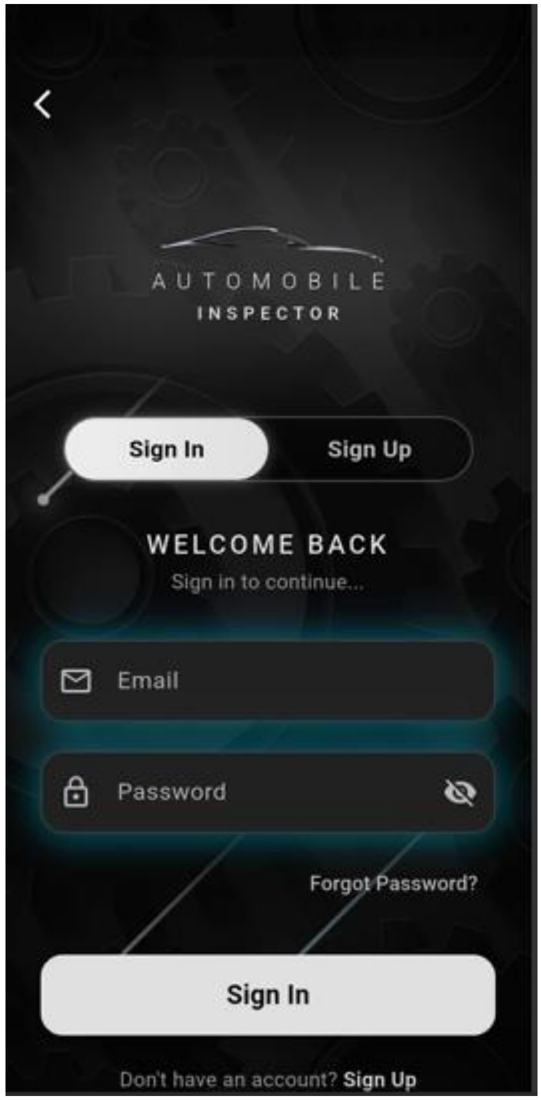


Figure 5. 9:Automobile Inspector Authentication Screen

5.1.8 Main Dashboard

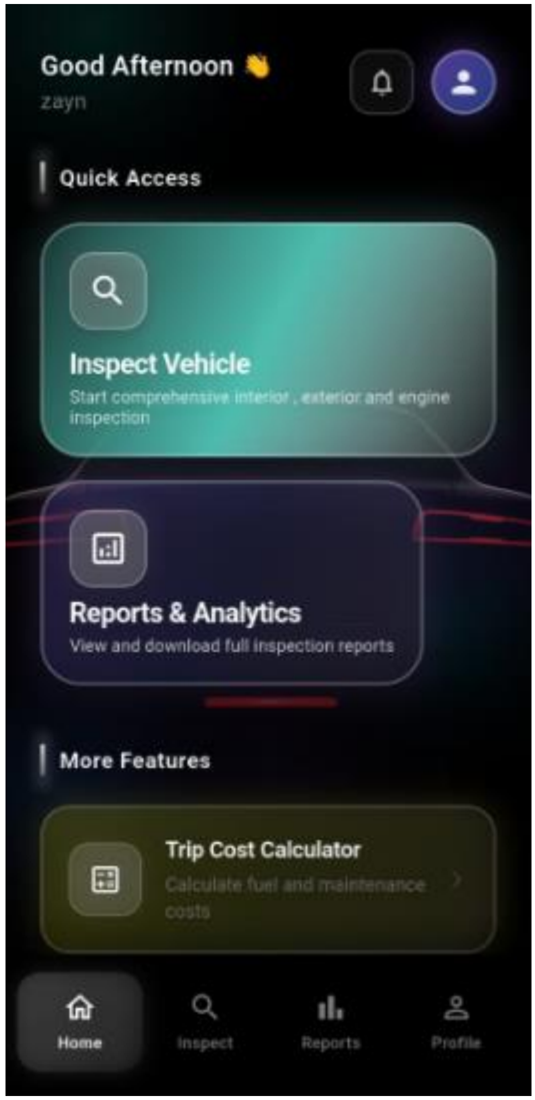


Figure 5. 10:Automobile Inspector Main Dashboard

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

The Automobile Inspector system is a full AI-based solution to assessing used vehicles that consists of computer vision, audio analysis, and structured data processing into a single automated inspection system. The system conducts multi-level vehicle inspection which includes exterior damage detection, interior conditions, engine sound diagnostics and feature price estimation. Every module is constructed through the application of specific machine learning and deep learning methodologies which collaborate to produce an accurate price forecast and inspection report to the user.

The exterior damage and paint modules use the YOLO-based object detection to detect the scratches, dents, accidental marks in the various body panels. The model has a good performance in its 24 scratch and dent classes with a high mAP, recall value and thus it can be said that it has consistency in real world situations when the lighting, angle and surface reflections vary. This semi-supervised pipeline also enhances this module by utilizing both labeled and pseudo-labeled images to enhance generalization.

The engine inspection module examines the mechanical health using Mel-Frequency Cepstral Coefficients (MFCC) of audio acquired by the user when listening to the engine. This method records the frequency pattern that can depict the knocking, misfiring, wearing of the belts, or other problems that affect the market value of the vehicle. Equally, the interior inspection module measures the state of the seat, the health of the dashboard, and the state of the cabin by using CNN features based on images.

The last price forecast is a combination of all inspection modules. The base valuation is made of structured inputs (model, mileage, manufacturing year, fuel type and engine capacity) and is deduced by exterior, interior, paint and engine results. This multi-model design guarantees precise, open and data-driven pricing.

On the whole, the automation of the entire used car inspections is accomplished by the system. It lowers the need to rely on a manual review and minimizes the human error by delivering objective and consistent and AI-driven information to buyers, sellers, and dealerships. The system allows scalability and real-world automobile marketplace adaptability due to the modular design..

6.2 Recommendations

Although the Automobile Inspector system works well in all the modules, it is possible to improve on them by adding better accuracy, generalization, and user experience. Increasing the exterior-damage data by accommodating a larger range of car types, lighting setups, and camera angles would improve the overall performance of the detectors in all circumstances. The body-paint module can be refined by providing more and more different cases of accidental cases so that vehicles that were repaired using non standard color blends can have certain cases. Other newer model architectures like YOLOv10, SAM (Segment Anything Model) or diffusion-based vision models may also be adopted to enhance the system with better micro-scratches and hairline cracks segmentation and fine-grained detection. To analyze the engines, it will be useful to add more and varied engine sounds and variety of microphones to get better resilience against the noise of the environment. The app can have a user feedback loop that it can use to gather corrections on issues related to price predictions, damages detection issues, or report inaccuracies. Regular retraining of the models can utilise this feedback to ensure that the system is in line with the market trends and the current state of the cars. Application wise, role-based access control (RBAC) can be implemented to isolate the user roles like buyers, sellers, dealers and the admin staff. A dynamic analytics dashboard that can be used by users and administrators would also be included to track the inspection history, usage statistics, and model performance trends. Lastly, a real-time cloud supported deployment of the system will enhance speed and scalability particularly when it comes to performing multiple inspections at a given time. As these enhancements occur, the Automobile Inspector will be able to become a high end, enterprise-scale car-inspection ecosystem that is applicable to professional automotive websites and used car markets.

References

[1] T. G. A. &. M. G. Diwan, “Object detection using YOLO: challenges, architectural successors, datasets and applications,” *Multimedia Tools and Applications,* vol. 81(7), p. 9243–9275, 2022.

[2] S. T. A. &. N. S. Rath, “Prediction of new active cases of coronavirus disease (COVID-19) pandemic using multiple linear regression model,” *Diabetes & Metabolic Syndrome: Clinical Research & Reviews,* vol. 1467–1474, p. 14(5), 2020.

[3] S. A. R. M. &. A. M. R. Syed, “Comparative Analysis of CNN and RNN for Voice Pathology Detection,” *BioMed Research International,* p. Article ID 5582281, 2021.

[4] G. B. Garcia, Learning Image Processing with OpenCV, 1st edition ed., Birmingham–Mumbai: Packt Publishing, 2015.

[5] T. Diwan, G. Anirudh and J. V. Tembhurne, “Object detection using YOLO: challenges, architectural successors, datasets and applications,” *Multimedia Tools and Applications,* vol. 82(6), p. 9243–9275, 2022.

[6] H. a. B. H. a. Z. Yan, X. a. Wu, Z. a. Wen, L. a. Jia and Xisheng, “Combination of VMD Mapping MFCC and LSTM: A New Acoustic Fault Diagnosis Method of Diesel Engine,” vol. 22, 2022.

[7] P. S. K. R. Y. V. K. Neeta Verma, “Car Damage Detection Analysis using Deep Learning, Computer Vision Techniques,” *International Conference on Artificial Intelligence and Data Science Applications - 2023,* vol. ICAIDSC2023, no. January 2025, pp. 11-15, 2025.

[8] Ramazhan, M. R. S. a. Bustamam, A. a. Buyung and R. Anwar, “Smart Car Damage Assessment Using Enhanced YOLO Algorithm and Image Processing Techniques,” *Information,* vol. 16, 2025.

FYP REPORT

ORIGINALITY REPORT

6%	5%	2%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	1%
2	Submitted to Kookmin University Student Paper	1%
3	www.coursehero.com Internet Source	<1%
4	www.frontiersin.org Internet Source	<1%
5	Thangaprakash Sengodan, Sanjay Misra, M Murugappan. "Advances in Electrical and Computer Technologies", CRC Press, 2025 Publication	<1%
6	koreascience.or.kr Internet Source	<1%
7	seyboldpublications.com Internet Source	<1%
8	link.springer.com Internet Source	<1%
9	www.springermedizin.de Internet Source	<1%
10	pmc.ncbi.nlm.nih.gov Internet Source	<1%
11	Udara Yedukondalu, V Vijayasri Bolisetty. "Advancing Innovation through AI and	<1%

	Machine Learning Algorithms - Computational Intelligence for Virtual System Optimization. A proceeding of ICMVRCET – 2025", CRC Press, 2025 Publication	
12	arxiv.org Internet Source	<1 %
13	Submitted to The University of the West of Scotland Student Paper	<1 %
14	downloads.hindawi.com Internet Source	<1 %
15	thesai.org Internet Source	<1 %
16	www.restack.io Internet Source	<1 %
17	suspace.su.edu.bd Internet Source	<1 %
18	vdocument.in Internet Source	<1 %
19	www.pure.fo Internet Source	<1 %
20	accedacris.ulpgc.es Internet Source	<1 %
21	wiredspace.wits.ac.za Internet Source	<1 %
22	www.acadlore.com Internet Source	<1 %
23	avestia.com Internet Source	<1 %
24	core.ac.uk	

Internet Source <1 %

25 ro.ecu.edu.au
Internet Source <1 %

26 www.mdpi.com
Internet Source <1 %

27 Asali, Ehsan. "Spatiotemporal Multimodal Representation Learning for Activity Understanding and Behavior Monitoring.", University of Georgia
Publication <1 %

28 Durgesh Kumar Mishra, Nilanjan Dey, Bharat Singh Deora, Amit Joshi. "ICT for Competitive Strategies", CRC Press, 2020
Publication <1 %

29 Felipe de Lima Peressim. "Machine learning for the prediction of 28-day cement compressive strength", Universidade de São Paulo. Agência de Bibliotecas e Coleções Digitais, 2025
Publication <1 %

30 adoc.tips
Internet Source <1 %

31 kipdf.com
Internet Source <1 %

32 uia.brage.unit.no
Internet Source <1 %

Exclude quotes Off
Exclude bibliography On
Exclude matches Off

Bscs S25-010

FYP REPORT

Quick Submit

Quick Submit

COMSATS University Islamabad, Lahore Campus

Document Details

Submission ID

trn:oid:::1:3426586958

Submission Date

Nov 28, 2025, 11:10 AM GMT+5

Download Date

Nov 28, 2025, 11:16 AM GMT+5

File Name

Automobile_Inspector_BSCS-S25-010.docx

File Size

4.6 MB

56 Pages

10,545 Words

59,907 Characters

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

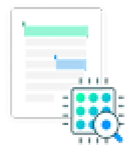
Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?

The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.



False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.