# Two Wheeler Vehicle Modification Detection System

*A Project Report*

*Submitted to the APJ Abdul Kalam Technological University*

*in partial fulfillment of requirements for the award of degree*

***Bachelor of Technology***

*in*

***Computer Science and Engineering***

*by*


**NIVED K V(TLY21CS045)**

**OUCHITH RAJEENDRAN(TLY21CS046)**

**SANGEERTH A K(TLY21CS052)**

**SREENAND N(TLY21CS058)**


*under the supervision of*


**Prof. Ambili M P**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**COLLEGE OF ENGINEERING THALASSERY**

**KERALA**


**April 2025**

**DEPT. OF COMPUTER SCIENCE & ENGINEERING**

**COLLEGE OF ENGINEERING**

**THALASSERY**

**2024 - 25**

## CERTIFICATE

This is to certify that the report entitled **Two Wheeler Vehicle Modification Detection System** submitted by **NIVED K V** (TLY21CS045), **OUCHITH RAJEENDRAN** (TLY21CS046), **SANGEERTH A K** (TLY21CS052) and **SREENAND N** (TLY21CS058) to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Engineering is a bonafide record of the project work carried out by him under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

**Prof. Ambili M P**
(Project Guide)
Assistant Professor
Dept.of CSE
College of Engineering
Thalassery

**Prof. Dr. Mohamed Mubarak T**
(Project Coordinator)
Associate Professor
Dept.of CSE
College of Engineering
Thalassery

**Prof. Dr. Mohamed Mubarak T**
Professor and Head
Dept.of CSE
College of Engineering
Thalassery

# COLLEGE OF ENGINEERING, THALASSERY
# KANNUR – 670 107
# DEPT. OF COMPUTER SCIENCE & ENGINEERING

## Vision of College

Be an institute of excellence in technical education providing competent, efficient and committed engineers to meet the changing needs of the society.

## Mission of College

- To impart professional education endowed with human values to the younger generation.
- To transform younger generation to be competent & committed engineers capable of providing solutions to the global challenges.
- To train youngsters to deploy technology for the service of humanity.

## Vision of Department

Excel in Computer Science and Engineering education and research for overall sustainable development.

## Mission of Department

- Impart quality education to nurture globally competent and socially responsible Computer engineers.
- Promote research that addresses society's evolving technological needs.

## Program Specific Outcomes

PSO 1: Ability to apply theoretical foundations and fundamentals of computer science in modeling and developing software and hardware solutions to real-world problems.

PSO 2: Ability to use computational, algorithmic and programming skills, along with expertise in computer networks and emerging fields such as data science, AI, and cloud computing.

## Program Educational Objectives

1. Work productively as computer engineers and succeed in diverse career paths in multidisciplinary environments to solve real-world problems.
2. Engage in lifelong learning to adapt to continuously evolving technology.
3. Promote the design, research, and implementation of innovative products and services in Computer Science & Engineering, utilizing strong technical, communication, leadership, and entrepreneurial skills.

## DECLARATION

We hereby declare that the project report **Two Wheeler Vehicle Modification Detection System**, submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of Prof. Ambili M P

This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

We also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

**NIVED K V** (TLY21CS045)
**OUCHITH RAJEENDRAN** (TLY21CS046)
**SANGEERTH A K** (TLY21CS052)
**SREENAND N** (TLY21CS058)

Thalassery
26-03-2025

# Acknowledgement

We would like to express our sincere gratitude to all those who have contributed to the successful completion of this project. Firstly, we would like to thank our Principal, Professor Dr. Rajeev P, Mechanical Department, for providing us with the necessary infrastructure and resources to carry out this project. We extend our heartfelt appreciation to our project guide, Asst. Prof. Ambili M P, for providing us with invaluable guidance and support throughout the project. She has been a constant source of motivation and has provided us with insightful feedback that has helped us shape our project. We also extend our sincere thanks to our project coordinator, Asst. Prof. Nikhil Dharman and Asst. Prof. Amrutha B K, for coordinating the project and providing us with the necessary resources and facilities to complete the project. We are also grateful to the Head of the Department, Prof. Dr. Mohamed Mubarak T, for providing us with the opportunity to carry out this project and for supporting us throughout the project. Finally, we would like to thank all the faculties who have provided us with their valuable insights and feedback throughout the project. Their guidance and support have been instrumental in the success of this project. Once again, we would like to express our sincere gratitude to all those who have contributed to the successful completion of this project.

**NIVED K V**

**OUCHITH RAJEENDRAN**

**SANGEERTH A K**

**SREENAND N**

# Abstract

The increasing trend of unauthorized modifications in two-wheeler vehicles poses serious concerns for road safety, environmental pollution, and regulatory enforcement. Modified exhausts often produce excessive noise, while structural changes can compromise safety and make vehicles unfit for public roads. Current systems for detecting such modifications are largely manual, time-consuming, and dependent on the expertise of inspectors. In many cases, modified vehicles go unnoticed due to the lack of a standardized, automated inspection process, especially during routine checks. This results in regulatory gaps, increased noise pollution in urban areas, and a rise in road rule violations.

To address these challenges, we propose an intelligent, automated Two-Wheeler Vehicle Modification Detection System. This system leverages both image and audio-based analysis to determine whether a vehicle has undergone any unauthorized modifications. On the technical side, it uses a YOLOv8 deep learning model to detect key vehicle parts like the exhaust and headlight. If a part is not detected in the expected location, it is flagged as potentially modified. To ensure higher reliability, the system also uses an autoencoder-based anomaly detection model that examines visual deviations from standard components. Furthermore, the exhaust sound is analyzed using PyAudio; if the average decibel level over a 10-second recording exceeds 80 dB, the vehicle is considered to have a modified exhaust.

The system features an intuitive user interface built using Tkinter and enhanced with Pygame animations to ensure a smooth and engaging user experience. Users can capture or upload vehicle images, run exhaust sound tests, and view the real-time results. The system also generates a detailed PDF report that includes detection results, images, and sound analysis for future reference. By automating the detection process and minimizing the need for manual judgment, our solution ensures faster, more reliable inspections and supports regulatory bodies in maintaining road safety and compliance. It offers a scalable framework that can be adopted in vehicle testing centers, traffic monitoring systems, and enforcement units across cities.

# Contents

# List of Figures

# List of Tables

# List of Symbols

| | |
|---|---|
| **YOLO** | You Only Look Once |
| **ML** | Machine Learning |
| **DL** | Deep Learning |
| **AE** | Autoencoder |
| **GUI** | Graphical User Interface |
| **UI** | User Interface |
| **UX** | User Experience |
| **ResNet** | Residual Neural Network |
| **dB** | Decibel |
| **CNN** | Convolutional Neural Network |
| **AI** | Artificial Intelligence |

# Chapter 1

# Introduction

In recent years, vehicle modifications have become a growing concern for regulatory authorities and the general public. While some modifications are purely aesthetic, others—particularly modifications to exhaust systems—can lead to increased noise pollution, safety hazards and non-compliance with legal standards. These modifications can disrupt public spaces, violate noise regulations and contribute to environmental concerns. The lack of an efficient and automated system to detect such modifications has made enforcement difficult, as it often relies on manual inspections that are time-consuming, labor-intensive and prone to errors.

To address this issue, this project aims to develop an automated two-wheeler vehicle modification detection system utilizing advanced computer vision and anomaly detection techniques. By leveraging state-of-the-art deep learning methods, the system can efficiently and accurately detect modifications in vehicle components, particularly the exhaust system. The primary objective of this project is to provide a robust and scalable solution that reduces human intervention while improving the accuracy and reliability of modification detection. The system employs YOLO-based image detection to identify key vehicle parts, such as the exhaust and headlights, determining whether they remain in their original (stock) condition or have been altered. If the YOLO model does not detect a part where it should be, the system assumes it has been modified. Additionally, an anomaly detection model is integrated to analyze reconstruction errors, providing further verification of modifications and reducing false positives. To enhance detection accuracy, the system also integrates exhaust sound testing, measuring noise levels to determine if an exhaust system has been illegally modified. If the detected noise exceeds 80 dB, the system flags the exhaust as modified. This multi-layered approach ensures a comprehensive verification process, reducing misclassification rates and improving reliability.

One of the significant challenges in vehicle modification detection is ensuring that inspections are consistent and unbiased. Traditional manual inspections introduce variability in assessments, leading to potential discrepancies. This system addresses this issue by eliminating manual verification, enhancing enforcement efficiency and ensuring that assessments are objective and data-driven. The automation of the detection process not only streamlines inspections but also facilitates real-time reporting, allowing authorities or users to generate a comprehensive report detailing the findings.

By integrating computer vision, machine learning and anomaly detection, this system represents a significant step forward in automated vehicle inspection. It streamlines the detection process, reduces noise pollution and ensures compliance with regulatory standards. Additionally, the system has the potential for future expansion, accommodating more vehicle components and adapting to evolving modification trends. The implementation of such a system will significantly aid regulatory bodies, law enforcement agencies and vehicle owners in maintaining compliance with legal standards while mitigating the adverse effects of unauthorized modifications.

## 1.1 Problem Statement

Unauthorized modifications in two-wheeler vehicles, particularly in exhaust systems, have become a growing concern due to their contribution to excessive noise pollution, safety risks and legal non-compliance. Modified exhausts often exceed permissible noise limits, disrupting public spaces and violating regulations. Additionally, modifications to other vehicle components, such as headlights, can compromise safety standards. Traditional enforcement methods rely on manual inspections, which are inefficient, time-consuming and prone to inconsistencies. The lack of an automated and standardized detection system results in delayed identification of violations, allowing non-compliant vehicles to continue operating without consequences. This inefficiency not only burdens regulatory authorities but also impacts public health and safety. There is an urgent need for a reliable, real-time solution that can accurately detect vehicle modifications and assist in regulatory enforcement.

## 1.2 Scope

The scope of this system is centered on enhancing vehicle inspection and enforcement by integrating advanced technologies for detecting unauthorized modifications in two-wheeler vehicles. It automates the identification of modifications in key components such as exhausts and headlights using computer vision and anomaly detection, ensuring accurate and efficient assessments. The system streamlines inspection processes by providing real-time detection and sound analysis, helping regulatory authorities and enforcement agencies ensure compliance with noise and safety regulations. By eliminating reliance on manual inspections, it improves accuracy, reduces processing time and minimizes human error. Additionally, the system generates detailed reports with visual and sound-based evidence, supporting transparent enforcement and documentation. Prioritizing efficiency and reliability, it is designed to be scalable and adaptable, allowing future expansions to other vehicle modifications.

## 1.3 Objectives

TThe primary objective of this system is to automate the detection of unauthorized modifications in two-wheeler vehicles, enhancing efficiency, accuracy and compliance in vehicle inspections. By leveraging computer vision, anomaly detection and sound analysis, the system aims to provide real-time identification of modifications in key components like exhausts and headlights, reducing reliance on manual inspections. Another key goal is to assist regulatory authorities and enforcement agencies by streamlining the inspection process, ensuring quicker and more consistent enforcement of noise and safety regulations. The system also focuses on transparency and reliability, generating detailed reports with image and sound-based evidence to support regulatory actions. Ultimately, it seeks to improve road safety, reduce noise pollution and establish a scalable solution for automated vehicle compliance monitoring.

## 1.4 Organization of the report

This report is structured to provide a comprehensive overview of the two-wheeler vehicle modification detection system and its role in improving vehicle compliance enforcement. The introduction discusses the growing concerns surrounding unauthorized vehicle modifications, particularly exhaust alterations and highlights the challenges faced in manual inspection processes, such as inefficiency and regulatory enforcement difficulties. The problem statement section defines these issues in detail, emphasizing the need for an automated and reliable solution to detect modifications in real-time.

The scope section outlines how this system addresses these challenges by leveraging computer vision, anomaly detection and exhaust sound analysis to provide real-time modification detection and verification. It also highlights the benefits of automated inspections, including improved efficiency, reduced human error and faster regulatory enforcement. The objectives section details the primary goals of the system, focusing on enhancing detection accuracy, streamlining compliance monitoring and generating transparent reports for regulatory use.

The following chapters will cover the system design, including the YOLO-based image detection model, anomaly detection methods and sound analysis techniques used for modification verification. Additionally, the report will discuss the feasibility study, implementation progress, functional requirements, external interface requirements, expected outcomes and conclusions. Each section builds upon the previous, providing a structured and in-depth understanding of how this system enhances vehicle modification detection and regulatory compliance.

# Chapter 2

# Literature Survey

## 2.1 Review of Related Work

### 2.1.1 Modified Vehicle Detection and Localization Model for Autonomous Vehicle Traffic System

**Author: Amit Juyal, Sachin Sharma, Shuchi Bhadula**

The paper [1] addresses the critical challenge of accurately detecting and localizing vehicles in complex traffic scenarios, which is essential for the safe operation of autonomous vehicles (AVs). The authors propose an enhanced version of the YOLOv5 model, termed YOLOv5-CBAM, which integrates a Convolutional Block Attention Module (CBAM) to improve feature extraction and focus on significant features during detection. The methodology involves a three-component architecture comprising a backbone for feature extraction, a neck for feature fusion, and a head for object detection, specifically tailored to handle the diverse and dynamic nature of Indian traffic. The model is evaluated on a custom dataset of 2,740 images, achieving a remarkable mean average precision (mAP) of 98.2%, significantly outperforming the original YOLOv5 model. This improvement highlights the model's effectiveness in real-time vehicle detection and localization, particularly in scenarios with modified vehicles, which are prevalent in the Indian context. The findings suggest that the YOLOv5-CBAM model not only enhances detection accuracy but also addresses the complexities of real-world traffic environments, although further research is needed to optimize computational efficiency for real-time applications.

### 2.1.2 Deep Learning-Based Anomaly Detection in Images: Insights, Challenges and Recommendations

**Author: Ahad Alloqmani et al**

The paper [2] provides a comprehensive analysis of the current state of deep learning-based anomaly detection in images, with a particular emphasis on its applications in the medical field. By reviewing twenty relevant studies, the authors categorize the methodologies, datasets, preprocessing techniques, and results, revealing a predominant reliance on unsupervised and semi-supervised learning methods. The findings indicate that deep learning models generally outperform traditional machine learning approaches, particularly in managing complex image data and addressing class imbalances inherent in medical datasets. However, the review also identifies significant challenges, such as the scarcity of labeled data and the presence of noise that complicates the identification of true anomalies. Many studies are constrained by small sample sizes, which may limit the robustness of their conclusions. Additionally, the paper highlights a gap in comparative analyses with state-of-the-art methods, suggesting that future research should focus on enhancing evaluation metrics and exploring preprocessing techniques to improve model performance. Overall, this paper underscores the need for continued innovation in anomaly detection methodologies to better serve the medical community and other fields reliant on image analysis.

### 2.1.3 Vehicle Brand Detection Using Deep Learning Algorithms

**Author: Mehmet Furkan Kunduraci, Humar Kahramanli Örnek**

The paper [3] investigates vehicle brand detection using deep learning algorithms. The study employs Faster R-CNN for vehicle brand classification, training the model on a dataset of 20 car brands. The results indicate that Faster R-CNN achieves moderate accuracy in brand recognition, with potential improvements through the integration of additional data and more robust feature extraction techniques. The paper underscores the importance of automated vehicle identification for traffic monitoring, security and intelligent transportation systems.

### 2.1.4 Vehicle Attribute Recognition by Appearance: Computer Vision Methods for Vehicle Type, Make and Model Classification

**Author: Xingyang Ni, Heikki Huttunen**

The paper [4] provides a comprehensive survey of vehicle attribute recognition methods, highlighting the shift from traditional hand-crafted feature extraction to deep learning approaches. The authors discuss the limitations of early datasets and the impact of larger, more diverse datasets on the performance of deep learning models. They also explore the potential of metric learning methods, which have shown promise in adapting to new vehicle models without extensive retraining. The study presents an experimental comparison of classification and metric learning approaches, demonstrating the superiority of the latter in certain scenarios. Furthermore, the authors identify future research directions, including few-shot learning, multi-task learning, and the integration of attention mechanisms, which could further enhance the accuracy and robustness of vehicle attribute recognition systems. This paper serves as a valuable resource for researchers and practitioners seeking to understand the current state of the field and its potential applications in traffic monitoring and automated vehicle identification.

### 2.1.5 Object Detection With Deep Learning: A Review

**Author: Zhong-Qiu Zhao, Peng Zheng, Shou-Tao Xu, Xindong Wu**

The review [5] examines deep learning-based object detection methods, focusing on two main approaches: region proposal-based and regression-based frameworks. Region proposal methods like R-CNN, Fast R-CNN and Faster R-CNN offer high accuracy but are computationally intensive. Faster R-CNN introduced the Region Proposal Network (RPN), which significantly improved speed and efficiency by generating region proposals internally. R-FCN further optimized this by using a fully convolution approach, enhancing computational efficiency. Regression-based methods such as YOLO and SSD prioritize speed. YOLO processes the entire image at once, making it extremely fast but less effective with small objects. SSD improves on this by using multiscale features, providing better accuracy across object sizes, though small object detection remains challenging.

### 2.1.6 Trademark Image Similarity Detection Using Convolutional Neural Network

**Author: Hayfa Alshowaish, Yousef Al-Ohali, Abeer Al-Nafjan**

The paper [6] presents a system to automate trademark similarity detection using deep learning. The system employs two pretrained CNNs, ResNet-50 and VGG-16, to extract features from trademark images, followed by PCA for feature reduction. Similarity is measured using Euclidean distance, with the system retrieving the most similar trademarks from a large dataset. The method improves accuracy and efficiency in trademark registration processes, achieving a mean average precision (mAP) of 0.774. However, the system faces challenges in handling text as shapes and requires significant computational resources, especially when dealing with small objects. Despite these limitations, the approach offers a promising solution for automating and improving trademark similarity detection. This study also emphasizes the growing importance of deep feature extraction in modern computer vision tasks.

### 2.1.7 Image Classification Based on RESNET

**Author: Jiazhi Liang**

The paper [7] explores image classification using deep learning, specifically focusing on ResNet (Residual Networks). Liang's research delves into the challenges faced by neural networks as they grow deeper, such as the vanishing gradient problem, where gradients decrease rapidly and slow down learning. The study leverages ResNet's architecture to overcome these issues by introducing a residual learning framework that enhances training for deep networks. Overall, the paper contributes valuable insights into deep learning's application in image classification, with ResNet demonstrating both robustness and limitations in handling very deep networks. The author also notes that proper model tuning and dataset quality play critical roles in achieving optimal classification accuracy.

### 2.1.8 Application of Deep Learning for Object Detection

**Author: Ajeet Ram Pathak, Manjusha Pandey, Siddharth Rautaray**

The paper [8] explores the effectiveness of deep learning techniques, particularly convolutional neural networks (CNNs), in object detection. The authors aim to assess the current state of deep learning in visual recognition systems, which encompass image classification, localization and detection—core components in applications like scene understanding, robotics and video surveillance. The methodology revolves around the use of CNNs, which do not rely on manually created feature extractors but instead learn directly from raw pixel data. The paper discusses how CNNs improve accuracy in object detection and compares various deep learning frameworks such as TensorFlow, Caffe and PyTorch. Several benchmark datasets, including Microsoft COCO and ImageNet, are utilized to evaluate the performance of deep learning-based object detection systems. The advantages of CNN-based approaches include their ability to automatically learn and optimize features for object detection, achieving remarkable accuracy even in complex tasks involving multiple objects. However, the disadvantages include the need for large labeled datasets to train models effectively. CNNs also face challenges with occlusions, varying scales and deformations, which can degrade performance. Moreover, deep learning models are computationally expensive, requiring powerful hardware like GPUs for training and inference.

### 2.1.9 Scratch Detection in Cars Using a Convolutional Neural Network by Means of Transfer Learning

**Author: César Giovany Pachón-Suescún, Paula C. Useche Murillo, Robinson Jimenez-Moreno**

The paper [9] presents a system for detecting scratches on cars using the AlexNet convolutional neural network (CNN) architecture. The authors employ transfer learning, where a pre-trained CNN is fine-tuned to classify car sections into two categories: scratched or unscratched. The training dataset is generated by splitting vehicle images into smaller sections to improve scratch visibility, followed by augmenting the data with lighting variations. The model achieved a validation accuracy of 88.29 and a test accuracy of 86.99. The advantages of this method include the ability to quickly adapt an existing network through transfer learning, reducing the need for massive datasets. This approach effectively handles the scratch detection problem by focusing on smaller sections of the vehicle, ensuring finer detail recognition. The system is also efficient in terms of processing time, making it applicable for real-time inspections in automotive contexts, such as parking lots or quality control during production. However, the disadvantages include potential misclassification due to lighting conditions, dirt, or non-car sections being analyzed, leading to false positives. The choice of AlexNet's architecture, designed for broader image classification tasks, may not be the most optimal for detecting small, detailed features like scratches, as evidenced by its large 11x11 convolutional filter size. Future improvements could include exploring alternative architectures with smaller filters or implementing region-based CNNs (R-CNN) to isolate car regions before scratch detection.

### 2.1.10 Vehicle Classification Using ResNets, Localisation and Spatially-Weighted Pooling

**Author: Rohan Watkins, Nick Pears, Suresh Manandhar**

The paper [10] explores the application of ResNet architectures for fine-grained vehicle classification. The authors investigate whether ResNet-18, ResNet-34 and ResNet-50 outperform traditional Convolutional Neural Networks (CNNs), specifically in the context of the Comprehensive Cars dataset. Additionally, they propose modifications to include Spatially-Weighted Pooling (SWP) and a localisation step, aiming to enhance classification accuracy. In their methodology, the authors first train ResNet models on the vehicle dataset without pre-training on ImageNet. They then introduce SWP and localisation to ResNet-50 to further improve performance. SWP refines feature extraction by focusing on critical image regions, while the localisation process enhances accuracy by isolating vehicle regions within an image. The advantages of this approach include an increase in classification accuracy, with the combination of SWP and localisation improving ResNet-50's performance by 3.7 percentage points, reaching a top-1 accuracy of 96.35. Additionally, the introduction of localisation significantly reduces the processing speed, which may be a disadvantage in real-time applications. Overall, the study highlights the trade-off between accuracy and computational efficiency.

### 2.1.11 A Survey Paper on Object Detection Methods in Image Processing

**Author: Manisha Vashisht, Brijesh Kumar**

The paper [11] presents an extensive review of various object detection techniques in image processing. The study explores different methodologies and evaluates their performance in real-world applications like face detection, pedestrian detection and medical imaging. In their methodology, the authors reviewed over 100 papers, focusing on 24 that were most relevant. These papers included machine learning and deep learning-based methods such as Convolutional Neural Networks (CNN), Single Shot MultiBox Detector (SSD) and region-based techniques. The review provides a comparative analysis of these methods in terms of accuracy, computational efficiency and usability. Overall, the survey provides a valuable foundation for future research in object detection.

### 2.1.12 A General Framework for Object Detection

**Author: Constantine P. Papageorgiou, Michael Oren, Tomaso Poggio**

The paper [12] presents a trainable framework for object detection in cluttered scenes. This framework utilizes a wavelet-based representation and statistical analysis to detect objects such as faces and pedestrians. The approach focuses on learning object classes using a support vector machine (SVM) and does not rely on pre-defined models or motion-based segmentation.The methodology involves extracting a compact representation of object classes using a subset of wavelet functions. Additionally, the authors introduce a motion-based extension for video sequences to enhance detection accuracy. Furthermore, the system may struggle with highly rotated objects, requiring additional training to improve performance.

### 2.1.13 YOLOv8-CAB: Improved YOLOv8 for Real-time Object Detection

**Author: Moahaimen Talib, Ahmed H. Y. Al-Noori, Jameelah Suad**

The paper [13] presents an enhanced version of the YOLOv8 model, termed YOLOv8-CAB, designed to improve real-time object detection, particularly for small and geometrically complex objects. The proposed framework incorporates the Context Attention Block (CAB) to effectively locate and identify small objects in images. Additionally, the model improves feature extraction by increasing the thickness of the Coarse-to-Fine (C2F) block and modifies Spatial Attention (SA) to accelerate detection performance. The methodology involves leveraging multi-scale feature maps and iterative feedback to optimize object detection mechanisms. Rigorous testing on the Common Objects in Context (COCO) dataset demonstrates a mean average precision of 97%, indicating a 1% improvement over conventional YOLO models. However, the system may face challenges in detecting highly rotated objects, requiring further training to enhance performance.

### 2.1.14 Auto-Encoders in Deep Learning—A Review with New Perspectives

**Authors: Shuangshuang Chen, Wei Guo**

This paper [14] provides a comprehensive review of auto-encoder (AE) algorithms within the context of deep learning, emphasizing their significance in unsupervised learning and non-linear feature extraction. The authors begin by introducing the basic structure and concept of auto-encoders, followed by an extensive examination of various AE variants, including denoising, sparse and variational auto-encoders. The review highlights the contributions and challenges of recent research, analyzing AEs from multiple perspectives, such as energy, manifold and information theory. The paper also discusses the relationships between AEs and other deep learning models, as well as their successful applications across diverse fields, including pattern recognition, computer vision and recommender systems. The authors conducted a thorough literature review, analyzing nearly 300 studies and identified future trends and challenges in the design and training of AEs. The findings suggest that while AEs have made significant strides in feature learning and data representation, there remain unresolved issues that warrant further exploration, particularly in hybrid model construction and the integration of attention mechanisms.

### 2.1.15 A Systematic Review of Anomaly Detection Using Machine and Deep Learning Techniques

**Authors: Sarfaraz Natha, Mehwish Leghari, Muhammad Awais Rajput, Syed Saood Zia, Jawaid Shabir**

The paper [15] presents a systematic review of machine learning (ML) and deep learning (DL) techniques for anomaly detection, focusing on advancements from 2019 to 2021. The authors provide a comprehensive analysis of various ML and DL models, including clustering-based, distance-based, statistical and classification-based methods, as well as deep learning approaches such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) and Generative Adversarial Networks (GANs). The review highlights the strengths and weaknesses of these models, particularly in handling challenges such as data complexity, noise and the unknown nature of anomalies. The authors also evaluate the accuracy and performance metrics of state-of-the-art works, using real-world datasets from domains such as surveillance, IoT and healthcare. The paper identifies key challenges, including the nature of input datasets, data stream complexities and time constraints and suggests future research directions, such as improving model performance, cross-validation and interdisciplinary collaboration. The review concludes that while significant progress has been made in anomaly detection, there is still room for improvement.

## 2.2 Conclusions

The reviewed literature provides a broad understanding of the growing role of deep learning in areas such as object detection, vehicle classification, and anomaly detection. Techniques like convolution neural networks, autoencoders and advanced models like YOLO and ResNet have significantly improved accuracy and efficiency across various applications. While each study focuses on different use cases—from vehicle monitoring to image classification—they collectively demonstrate how deep learning is transforming automated systems. Despite notable progress, challenges such as handling diverse real-world conditions, reducing dependency on large datasets, and improving real-time performance remain key areas for future research and innovation.

| Paper | Problem Statement | Methodology | Advantages | Disadvantages | Inference |
|---|---|---|---|---|---|
| Paper 1 | Detecting illegal vehicle modifications in traffic | YOLOv5-CBAM, convolutional attention mechanisms | High precision and recall | Computational complexity | Effective for traffic law enforcement |
| Paper 2 | Identifying image anomalies in various domains | Supervised, semi-supervised and unsupervised learning techniques | Versatile across applications | Requires large labeled datasets | Can assist in detecting rare modifications |
| Paper 3 | Classifying vehicle brands using deep learning | Faster R-CNN for vehicle classification | High accuracy in brand recognition | Large dataset requirement | Can enhance vehicle identification in surveillance systems |
| Paper 4 | Vehicle attribute recognition by appearance | Deep learning-based classification and metric learning | Precise classification of vehicle types | Computationally expensive | Useful for vehicle tracking and identification |
| Paper 5 | Object detection for vehicle modifications | R-CNN, Fast R-CNN, CNN for object detection | High accuracy in detecting modifications | Struggles with small objects | Effective for detecting major vehicle modifications |
| Paper 6 | Image similarity detection in trademarks | VGG-16, ResNet-50 for feature extraction | Efficient similarity matching | Affected by image distortions | Can be adapted for vehicle modification comparison |
| Paper 7 | Deep learning for vehicle modification detection | ResNet-20, CIFAR-10, Inception-ResNet | Handles complex vehicle modifications | Increased network depth leads to complexity | Useful for detailed vehicle part classification |
| Paper 8 | Object detection in vehicles using CNNs | Sliding window, region proposals, COCO dataset | High accuracy in vehicle detection | High computational cost | Strong for identifying modifications in real-world scenarios |

Table 2.1: Summary of Literature Survey

| | | | | | |
|---|---|---|---|---|---|
| Paper 9 | Scratch detection in vehicles | AlexNet, transfer learning, data augmentation | Fast and efficient detection | Susceptible to false positives (dirt misclassification) | Useful for assessing vehicle surface modifications |
| Paper 10 | Fine-grained vehicle classification | ResNet-18/34/50, Spatially Weighted Pooling | Improved classification accuracy | Computationally intensive | Effective for precise vehicle modification recognition |
| Paper 11 | Survey on object detection methods | Systematic analysis of CNN-based approaches | Comprehensive comparison of methods | Lacks implementation details | Provides strong background knowledge for further research |
| Paper 12 | General framework for object detection | Feature-based object detection with SVM | Established early foundations | Outdated compared to CNN-based methods | Basis for modern object detection approaches |
| Paper 13 | Real-time object detection for small and complex objects | YOLOv8-CAB, Context Attention Block (CAB) | High mean average precision (97%) | Challenges with highly rotated objects | Effective for real-time applications in various domains |
| Paper 14 | Unsupervised feature extraction and representation learning | Auto-encoders (AEs), various AE variants | Versatile in feature learning | Potential unresolved issues in hybrid models | Useful for data representation across multiple fields |
| Paper 15 | Anomaly detection in diverse applications | Machine learning and deep learning techniques | Comprehensive analysis of models | Requires large labeled datasets | Can enhance detection of rare anomalies in various domains |

Table 2.2: Summary of Literature Survey

# Chapter 3
# Project Objectives and Methodology

## 3.1 Project Objectives

Unauthorized modifications in two-wheelers, particularly in exhaust systems, pose regulatory and environmental challenges. Manual inspection methods are inefficient, time-consuming and prone to errors. This project leverages deep learning-based image analysis and sound level testing to automate the detection of vehicle modifications. The key objectives of this project are as follows:

### 3.1.1 Automated Image-Based Vehicle Part Detection

Traditional vehicle inspections rely on manual verification, making them slow and inconsistent. This system eliminates manual intervention by utilizing a YOLO-based deep learning model to automatically detect and classify vehicle parts. Users can upload an image allowing the system to analyze the vehicle part. If a part is missing or replaced with a non-stock version, it is flagged as modified.

### 3.1.2 Seamless Modification Detection and Verification

The system ensures accurate modification detection by leveraging separate YOLO and Autoencoder models for each bike part. Each detected part is compared against stock specifications-if discrepancies are found, the part is classified as modified. This approach ensures high precision in modification identification.

### 3.1.3 Real-Time Exhaust Sound Analysis

A key indicator of exhaust modifications is excessive noise levels. The system integrates an exhaust sound testing module that records 10 seconds of sound data and calculates the average decibel level. To enhance the user experience, a wave animation replaces traditional number updates, making the process both intuitive and visually engaging.

### 3.1.4 Intelligent Reporting and Documentation

The system automatically generates a comprehensive modification report containing high-resolution images of detected modifications, fine amount, modification classification results (Stock/Modified) and exhaust sound level analysis with a final verdict. This report can be exported as a PDF for regulatory use, helping authorities and vehicle owners maintain compliance with modification laws.

### 3.1.5 User-Friendly Interface and Interactive Experience

The system is designed for ease of use, ensuring a smooth and interactive experience for users. No manual input is required, as the system automatically detects the part. A visually engaging UI with real-time detection feedback ensures clarity. The wave animation for sound testing eliminates the need for number-based displays, while on-the-fly results allow users to see modification status instantly.

### 3.1.6 Scalability and Future Enhancements

This project follows a modular architecture, ensuring it can be scaled for additional vehicle models and parts in the future. Planned enhancements include expansion to other vehicle components such as headlights, indicators and chassis parts. UI/UX improvements will provide a more intuitive experience, while integration with regulatory databases will enable validation of modification legality.

## 3.2 Methodology

To achieve the project's objectives, a structured methodology is employed, integrating deep learning-based image detection, real-time sound analysis and a user-friendly interface. The methodology consists of seven key phases, ensuring systematic and efficient implementation.

### 3.2.1 Requirement Analysis and Feasibility Study

The initial phase involved analyzing the requirements for an automated vehicle modification detection system, focusing on two-wheelers. A feasibility study was conducted to assess the technical, operational and economic viability of the system. This phase helped define key functionalities, such as automatic part detection, anomaly-based modification verification and exhaust sound level testing. Additionally, potential challenges like real-time processing constraints and model accuracy were identified to ensure efficient implementation.

### 3.2.2 System Design and Architecture Development

Based on the requirements, a modular architecture was designed, ensuring scalability and maintainability. The system architecture includes:

- **User Interface**: Provides an interactive user interface for image upload, capturing and displaying results developed by using Python Tkinter and Pygame.
- **Detection and Analysis Module**: Utilizes a YOLOv8 model to detect vehicle parts and classify modifications. An autoencoder-based anomaly detection system further verifies modifications by analyzing reconstruction errors and statistical deviations.
- **Exhaust Module**: Records and evaluates exhaust sound levels over a 10-second test, determining modification status and displaying results with a wave animation. Implemented using PyAudio.
- **Result Display and Reporting**: Presents modification status, penalties and compiles findings, including detected modifications and corresponding images, into a PDF report.

### 3.2.3 Image-based Vehicle Part Detection

The system employs a YOLO model to automatically detect key vehicle components, such as exhausts and headlights, from input images. By analyzing the image, the model identifies and localizes these parts with bounding boxes, eliminating the need for manual selection. This automated detection ensures accurate identification, forming the basis for subsequent modification analysis.

### 3.2.4 Implementation of Multiple Models for Detection

The multi-model verification process starts with a primary YOLO model, which is trained on images containing all vehicle parts to ensure comprehensive initial detection. Once a part is identified, a separate YOLO model, specifically trained for that part, is used for precise verification. Finally, an autoencoder-based anomaly detection model analyzes reconstruction errors to detect subtle modifications, ensuring high accuracy in modification detection.

### 3.2.6 Testing, Evaluation and Performance Optimization

After development, Heisei underwent multiple testing phases to ensure reliability and efficiency:

- Unit Testing: Each detection module (YOLO-based part detection, modification analysis and sound testing) was tested individually to verify functionality.
- Integration Testing: Ensured smooth interaction between part detection, modification verification and result reporting components.
- Accuracy Validation: Evaluated the system's performance using real-world test images and sound samples, ensuring correct modification detection.
- Performance Optimization: Improved detection speed, reduced false positives and optimized system response time for real-time operation.

### 3.2.7 Deployment and Future Enhancements

The final phase involved deploying the two-wheeler modification detection system in a controlled testing environment to evaluate its real-world performance. Feedback was collected from test users and refinements were made based on accuracy and usability metrics. Some of the future enhancements that we wish to implement are:

- Real-Time Video-Based Detection: Extend the system to support live video input, enabling continuous monitoring of vehicle parts and modifications in real time.
- Expansion of Part Detection: Enhance YOLO-based detection to include additional components such as wheels, mirrors and body modifications.
- Cloud-Based Data Storage: Enable cloud storage for test results and historical tracking of vehicle modifications.
- Enhanced Report Generation: Improve the PDF reporting system with more detailed insights and comparative analysis.

# Chapter 4

# Work Plan and Approximate Budget

## 4.1 Work Plan

The development of the Two-Wheeler Vehicle Modification Detection System is guided by a structured and time-bound work plan, organized into five distinct phases. Each phase incorporates technical development, validation, and deployment tasks, ensuring that the system progresses from concept to implementation in a systematic and efficient manner. The plan emphasizes iterative refinement, cross-functional integration, and user-centered design to meet the intended goals of accuracy, usability, and robustness.

### 4.1.1 Phase 1: Project Initiation (Month 1-2)

This foundational phase involves establishing the vision and scope of the system. Emphasis is placed on identifying key user needs, understanding regulatory compliance requirements, and performing preliminary technical research. The aim is to ensure feasibility, stakeholder alignment, and a clear project direction. Key tasks include:

- **Defining Project Objectives:** Setting the overarching goal of developing a system capable of detecting unauthorized two-wheeler modifications using image processing and acoustic analysis.
- **Stakeholder Requirement Analysis:** Engaging with intended users such as RTO officers, law enforcement personnel, and inspection agencies to identify pain points and expectations.
- **Feasibility Study:** Evaluating hardware needs (camera, mic, GPU availability), software dependencies, and project constraints related to time, cost, and data availability.

- **Research on Existing Technologies:** Reviewing current vehicle inspection tools, including manual checks and commercial automated systems, to benchmark the proposed solution.

### 4.1.2 Phase 2: Planning and Design (Month 3-5)

The second phase translates conceptual goals into technical blueprints. It focuses on the detailed design of the detection pipeline, model architecture, user interface and backend integration. A modular and scalable system architecture is designed to accommodate future enhancements. Key tasks include:

- **Architectural Design:** Defining how components like the primary YOLO detector, part-specific YOLOs, anomaly detection modules, and the sound testing unit interact.
- **Pipeline Structuring:** Outlining the detection sequence—starting from image acquisition, part detection, modification analysis (image and sound), to final report generation.
- **User Interface Design:** Creating wireframes and interface flows for the GUI, focusing on simplicity, accessibility, and real-time feedback for users.
- **Tool Selection and Environment Setup:** Finalizing tools and frameworks such as Python, OpenCV, TensorFlow, PyTorch, PyAudio, and GUI libraries like Tkinter or Pygame.
- **Dataset Planning and Resource Allocation:** Initiating the creation and collection of image and sound datasets, identifying required classes (e.g., stock vs modified exhausts), and securing training resources.
- **Ethical and Legal Considerations:** Planning for data privacy, storage limitations, and ensuring that vehicle images or sounds used comply with ethical research guidelines.

### 4.1.3 Phase 3: Development and Integration (Month 6-7)

This phase is dedicated to the implementation of system components and integration of machine learning models. The focus is on achieving a working prototype with functional detection and analysis capabilities. Key tasks include:

- **YOLO Model Development:** Training and deploying a general YOLOv8 model to detect various vehicle parts from uploaded or captured images.
- **Specialized Detection Modules:** Incorporating part-specific YOLO models and anomaly detection autoencoders for parts like exhaust and headlights, trained separately per bike model.
- **Sound Analysis Integration:** Implementing PyAudio-based recording, average sound level computation, and classification logic for exhaust modification detection.
- **Frontend and Backend Development:** Establishing the logic to pass results between detection models and the UI, ensuring intuitive and responsive visual feedback to the user.
- **PDF Report Generation Module:** Automatically compiling images, results, timestamps, and remarks into structured, readable reports for download or printing.
- **Performance Optimization:** Ensuring the detection runs efficiently on mid-range systems, addressing issues like memory usage and processing lag.

### 4.1.4 Phase 4: Testing and Refinement (Month 8)

Once the core system is in place, rigorous testing is conducted to ensure that the system works accurately under varied conditions. This phase also focuses on optimizing the user experience. Key tasks include:

- **Unit Testing:** Verifying that each module, including UI elements, detection models and report generation, functions independently without errors.

- **System Integration Testing:** Checking that the modules work together seamlessly—for example, ensuring that image detection results correctly influence the anomaly model or report generation.
- **Validation of Models:** Using test datasets to evaluate YOLO model mAP scores and comparing autoencoder reconstruction errors between known stock and modified parts.
- **Sound Module Calibration:** Testing microphone sensitivity and validating sound thresholds (e.g., 80 dB as the cutoff for modified exhausts).
- **Real-World Simulation:** Running the system using varied images and real bike exhaust sounds to identify potential bugs or misclassifications.
- **UI Enhancements:** Refining layout, adding animations (e.g., sound wave visualizations), and simplifying the user journey based on feedback from test users.

### 4.1.5 Phase 5: Documentation and Finalization (Month 8)

The final phase consolidates all work into a professional project package for submission and possible deployment. It emphasizes quality documentation and readiness for real-world usage. Key tasks include:

- **Technical Report Preparation:** Drafting a comprehensive report covering objectives, design rationale, methodology, dataset creation, testing, and limitations.
- **Deployment Testing:** Conducting a real-world pilot run to observe system performance on actual vehicles, capturing any unexpected behaviors.
- **Stakeholder Feedback Collection:** Presenting the system to sample users (e.g., students, faculty, or vehicle inspectors) for feedback.
- **Final Bug Fixes and Enhancements:** Addressing any critical issues identified during deployment or user testing.
- **Presentation Preparation:** Designing slides and demos for project evaluation, highlighting core features and real-time detection scenarios.

## 4.2 Approximate Budget

The budget for Heisei is structured across the five development phases, covering expenses related to infrastructure, software tools and deployment. Below is the estimated cost breakdown:

### 4.2.1 Phase 1: Project Initiation

- Research and Requirement Analysis: ₹0
- Internet and Miscellaneous Expenses: ₹1000
- Total Cost: ₹1000

### 4.2.2 Phase 2: Planning and Design

- Data and Resource Collection: ₹0
- System Architecture & Workflow Planning : ₹0
- Total Cost: ₹0

### 4.2.3 Phase 3: Development and Integration

- Dataset Creation (Roboflow): ₹0
- YOLO Model Training(Google Colab): ₹0
- Autoencoder Model Training(Local Setup, No External Cost): ₹0
- Code and Model Integration: ₹0
- Total Cost: ₹0

### 4.2.4 Phase 4: Testing and Refinement

- Unit Testing for YOLO and Autoencoder Models: ₹0
- Integration Testing for Complete System Workflow: ₹0
- Debugging and Performance Optimization: ₹0
- Fine-Tuning Model Parameters and Threshold Adjustments: ₹0
- Total Cost: ₹0

### 4.2.5 Phase 5: Documentation and Finalization

- Report Printing and Binding: ₹1,500
- Presentation Preparation: ₹0
- Miscellaneous Expenses: ₹1,000
- Total Cost: ₹2,500

## Total Estimated Budget: ₹3,500

The budget is structured to efficiently support the development, testing and finalization of the two-wheeler vehicle modification detection system while keeping costs minimal. Additional funding may be required for enhanced model training, real-world deployment, scalability enhancements.

# Chapter 5

# Theory and Modeling

## 5.1 Theory

The theoretical foundations of the Two-Wheeler Vehicle Modification Detection System define the core principles guiding its design and functionality. This system integrates computer vision, anomaly detection and acoustic analysis to detect vehicle modifications accurately. The primary concepts include YOLO-based object detection, autoencoder-based anomaly detection and sound analysis for exhaust evaluation.

### 5.1.1 Object Detection using YOLO

Object detection is a crucial component of the Two-Wheeler Vehicle Modification Detection System, enabling automatic identification of vehicle parts and modifications. The system employs YOLO (You Only Look Once), a real-time deep-learning model, to detect and classify various vehicle components. The key properties include:

- Real-Time Detection: YOLO processes images in a single pass, allowing fast and efficient detection of vehicle parts.
- High Accuracy: The model is trained on a dataset of two-wheeler images, ensuring reliable detection of stock and modified components.
- Automated Part Identification: Eliminates the need for manual input by automatically recognizing the exhaust, headlight and other parts.

### 5.1.2 Anomaly Detection using Autoencoders

While YOLO is effective for detecting modified vehicle parts based on visual differences, the system further verifies modifications using an autoencoder-based anomaly detection model. This method enhances accuracy by analyzing patterns and detecting deviations from stock components.

- Unsupervised Learning: The autoencoder is trained on normal (stock) vehicle parts and learns to reconstruct them accurately.
- Reconstruction Error Measurement: If a detected part significantly deviates from expected reconstruction patterns, it is flagged as modified.

This approach ensures a robust verification process, reducing false positives and enhancing detection reliability. By combining YOLO-based visual detection with autoencoder-driven anomaly detection, the system achieves higher accuracy in identifying modified components.

### 5.1.3 Exhaust Sound Analysis for Modification Detection

Exhaust sound analysis is a crucial component of the Two-Wheeler Vehicle Modification Detection System, enabling the identification of modified exhausts based on sound levels. The system utilizes PyAudio to record exhaust noise for 10 seconds, analyzes the sound pressure level (SPL) in decibels (dB) and determines if the exhaust has been modified.

- **Sound Recording**: The system captures 10 seconds of exhaust noise using PyAudio, ensuring sufficient data for analysis.
- **Real-Time Visualization**: Instead of displaying raw numerical values, the system presents a wave animation to represent sound levels dynamically.

These features ensure an intuitive and efficient user experience by combining accurate sound analysis with real-time visualization, making modification detection more accessible and reliable.

## 5.2 Modeling

### 5.2.1 System Architecture

The vehicle modification detection system follows a multi-layered architecture to ensure efficient detection, analysis and visualization of modifications. The architecture consists of five primary components:

1. **User Layer**

   The User Layer represents the primary users of the system:

   - **Vehicle Owners**: Individuals who use the system to check if their vehicle's exhaust system is modified.
   - **Inspectors/Authorities**: Authorities or mechanics who analyze vehicles for compliance with legal exhaust regulations.

   Users interact with the system through the interface to upload images, record sound samples and receive modification status results.

2. **Application Layer**

   The Application Layer serves as the primary interface between users and the system, consisting of:

   - **GUI (Tkinter-based)**: Tkinter-based GUI that allows users to upload images, start exhaust sound testing, view modification results, generate PDF reports and interact with the system for seamless vehicle modification detection.

   This layer facilitates real-time interactions between users and the detection models, ensuring seamless operation and visualization.

3. **Processing Layer**

   The Processing Layer serves as the computational core of the system, handling image and sound analysis to detect vehicle modifications. This layer:

   - Utilizes the YOLO model to detect vehicle parts and assess modifications (e.g., exhaust, headlights).

- Applies anomaly detection using an autoencoder to validate detected parts and identify unusual modifications.
- Conducts sound-based analysis via PyAudio to measure exhaust noise levels and detect excessive sound output.

By integrating computer vision, anomaly detection and sound analysis, this layer ensures accurate and efficient identification of modifications.

4. **Result Layer**

The Result Layer is responsible for interpreting detection outcomes and presenting final modification status to the user. This layer:

- Combines results from image-based detection, anomaly detection and sound analysis.
- Classifies the vehicle as ”Modified” or ”Not Modified” based on the detection findings.
- Sends final results to the Application Layer for display.
- Generates and stores a PDF report containing detected modifications, analyzed images and sound test results.

By consolidating and presenting analysis results, this layer ensures transparency and provides actionable insights for users and authorities.

5. **Admin Layer**

The Admin Layer is responsible for maintaining and improving the system's accuracy and efficiency. It operates in the background, ensuring that the Processing Layer (YOLO & Sound Analysis) functions optimally. The key responsibilities of this layer include:

- **Model Retraining**: Admins can retrain the YOLO and anomaly detection models to improve detection accuracy when necessary.
- **System Monitoring**: Tracks modification detection trends, system logs and overall performance to identify potential improvements.
- **Dataset Management**: Ensures that training datasets are updated and maintained for better future detections.

While the Admin Layer does not directly interact with the user interface, it plays a crucial role in optimizing the Processing Layer, ensuring that users receive accurate and reliable modification detection results.

The Two-Wheeler Vehicle Modification Detection System is designed for accuracy, automation and compliance. Its multi-layered architecture ensures seamless user interaction while maintaining highly reliable vehicle modification detection through image processing, anomaly analysis and sound evaluation. By integrating YOLO-based detection, autoencoder validation and real-time sound analysis, the system enhances trust, enforcement efficiency and road safety compliance. With structured data storage, real-time decision-making and automated report generation, the system provides a scalable, secure and transparent solution for vehicle modification assessment, benefiting users, regulatory bodies and law enforcement agencies.

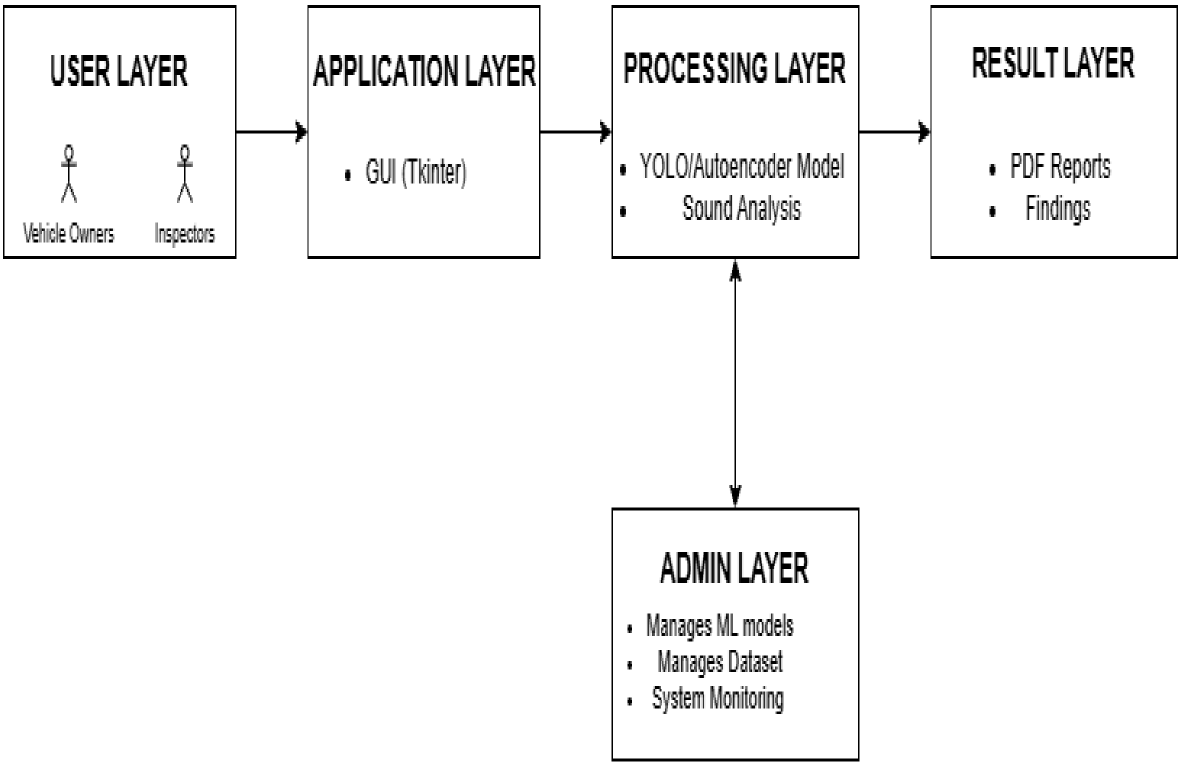


Fig 5.1. Architecture Diagram

### 5.2.2 Use Case Diagram

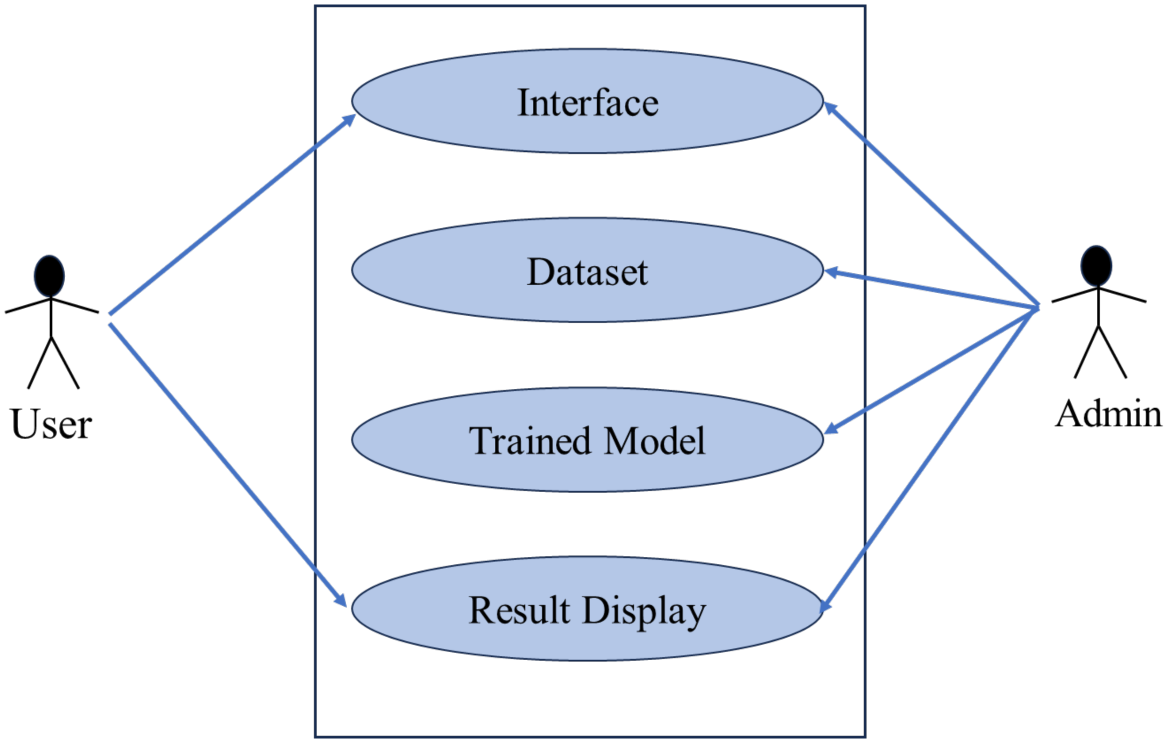


Fig 5.2. Use Case Diagram

### 5.2.3 Data Flow Diagram

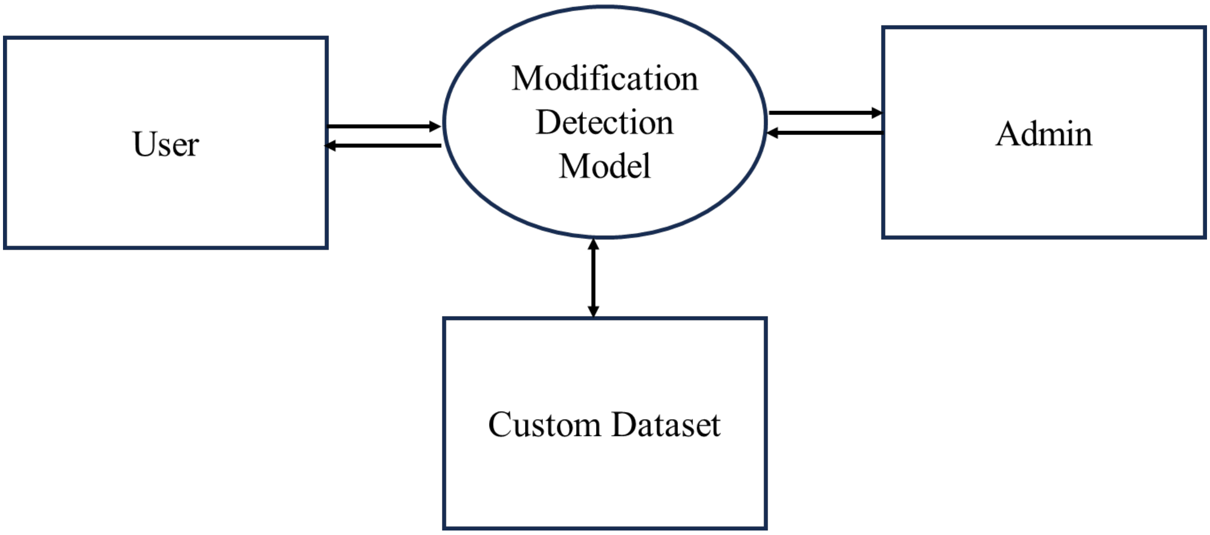


Fig 5.3. Level 0 Data Flow Diagram

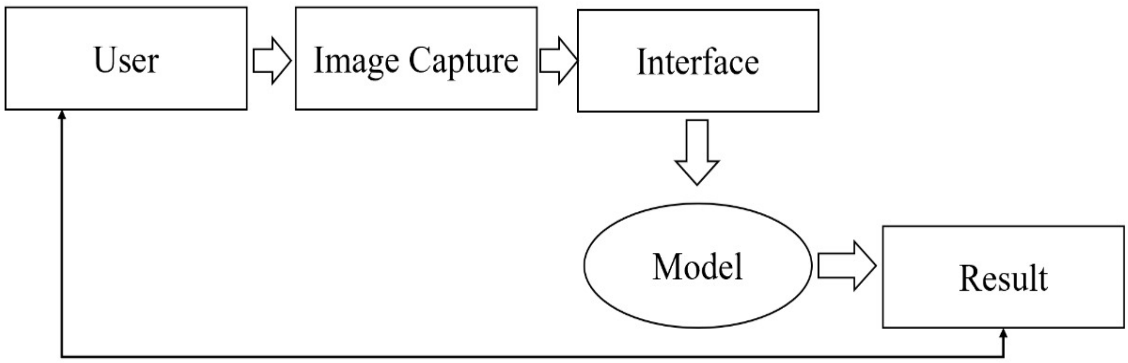


Fig 5.4. Level 1 Data Flow Diagram

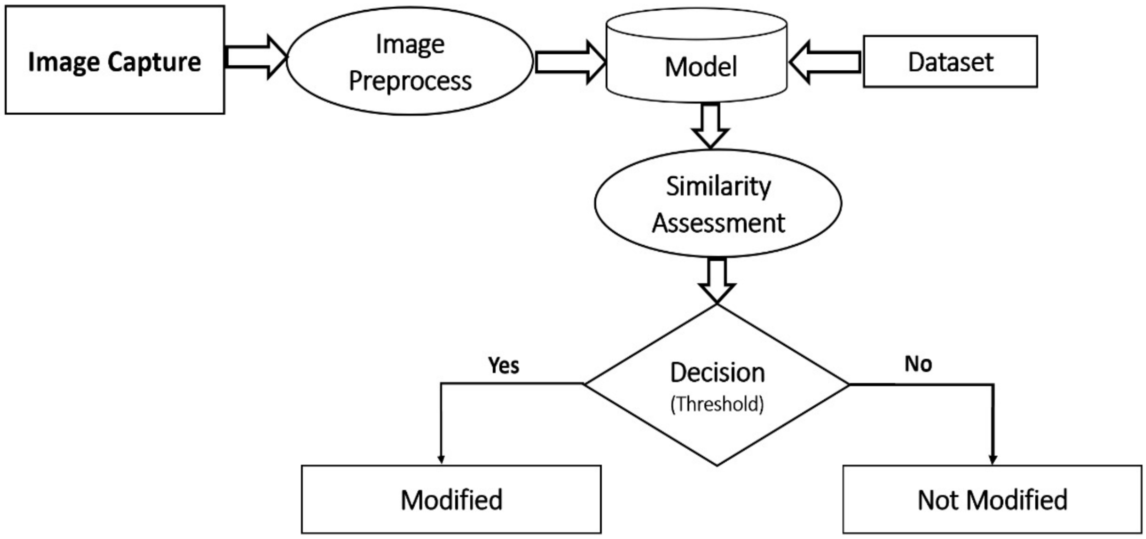


Fig 5.5. Level 2 Data Flow Diagram

### 5.2.4 Algorithm Used

The following algorithm is designed to detect modifications in two-wheeler vehicles by analyzing images. The system uses a combination of YOLO-based part detection and an autoencoder-based anomaly detection model to determine if a vehicle component has been modified.

**Working:**

1. **Image Upload and Selection**: The user uploads an image of the vehicle or captures it using the system's interface.
2. **Image Preprocessing**: The uploaded image is resized and normalized to ensure compatibility with the YOLO detection model.
3. **YOLO Detection**:
   i. If the specific vehicle part (e.g., exhaust, headlight) is detected, it is assumed to be in stock condition.
   ii. If the part is **not detected**, it is classified as **modified**.
4. **Anomaly Detection using Autoencoder**:
   i. If the part is detected, its features are extracted and passed to the autoencoder.
   ii. The reconstruction error is calculated to determine how much the detected part deviates from a normal (stock) sample.
5. **Threshold Comparison**:
   i. If the reconstruction error is below a predefined threshold, the part is classified as **Not Modified**.
   ii. If the reconstruction error exceeds the threshold, the part is classified as **Modified**.
6. **Display Results**

# Chapter 6

# Modular Division

The Two-Wheeler Vehicle Modification Detection System is meticulously designed using a modular architecture to promote scalability, maintainability, and robustness. The system is structured into two major components:

1. **Frontend**: An interactive platform that enables users to capture or upload vehicle images, initiate exhaust testing, and visualize the results in a user-friendly format.
2. **Backend (Inference Module)**: A powerful processing engine that analyzes images and sound data using deep learning models to determine whether vehicle components have been modified.

This modular division not only promotes parallel development and ease of debugging but also allows for the future integration of additional modules without disrupting the existing framework. Each component works independently while maintaining seamless intercommunication to ensure high performance and accuracy.

## 6.1 Frontend Module

The Frontend Module acts as the bridge between users and the system's intelligence. It ensures an intuitive and visually engaging experience for vehicle inspectors, enforcement officers, and end users. The module is developed using:

- **Tkinter** – For building structured, responsive, and easily navigable GUI components.
- **OpenCV** – For managing image capture, input selection, and preliminary preprocessing.
- **Pygame** – For rendering animated sound wave effects and improving visual feedback.

**Core Features of the Frontend Module:**

- **Image Capture & Upload:** Users can either capture a live image through a connected camera or upload an existing image for analysis. This flexibility supports diverse real-world scenarios.
- **Real-Time Detection Display:** Once an image is processed, the frontend immediately displays whether each vehicle part is stock or modified, along with relevant visual indicators.
- **Exhaust Sound Testing Interface:** A dedicated window powered by Pygame presents a dynamic wave animation during the sound recording phase, providing a visual cue and enhancing user interaction.
- **Interactive UI Elements:** Smooth transitions, color-coded feedback, and intuitive button placement make the interface highly accessible and responsive even for non-technical users.
- **Live Modification Count:** The frontend dynamically updates the modification tally in real time based on detection outcomes, eliminating the need for manual tracking.
- **PDF Report Generation:** At the end of the session, users can export a structured PDF report that includes images, sound test results, modification status, timestamps, and compliance notes.

By combining structured UI elements from Tkinter with the animated capabilities of Pygame, the Frontend Module delivers a streamlined and engaging experience while maintaining strong communication with the backend inference engine.

## 6.2 Backend/Inference Module

The Backend Module serves as the intelligence hub of the system, orchestrating the analysis of image and audio data to detect unauthorized modifications. It houses all major machine learning models and handles data flow, decision-making, and integration with the frontend.

**Core Functions of the Backend Module:**

- **Vehicle Part Detection:** A general YOLOv8 model is first used to identify the presence of major vehicle parts (e.g., exhaust, headlight). If a part is not detected in its expected position, it is automatically flagged as modified.
- **Anomaly-Based Verification:** To further verify part authenticity, each detected component is analyzed using a specialized autoencoder model trained on stock images. The system uses reconstruction error thresholds to distinguish between stock and modified components.
- **Exhaust Sound Analysis:** Using PyAudio, the system records 10 seconds of exhaust noise. The average decibel level is calculated in real time. If the level exceeds 80 dB, the exhaust is classified as modified.
- **Seamless Frontend Integration:** The backend transmits detection results, sound classification, and decision flags to the frontend immediately after processing. This ensures minimal latency and a smooth user experience.
- **Automated Report Generation:** At the end of the session, the backend consolidates all detection data into a PDF report. This includes captured images, sound analysis, system decisions, and a summary for regulatory use.

The Backend Module thus acts as a comprehensive inference engine that combines image-based object detection, deep learning–driven anomaly analysis, and real-time audio classification into a unified system. Its modular, extensible nature ensures long-term adaptability to new regulations and emerging technologies.

# Chapter 7

# Results and Discussion

## 7.1 UI Interfaces

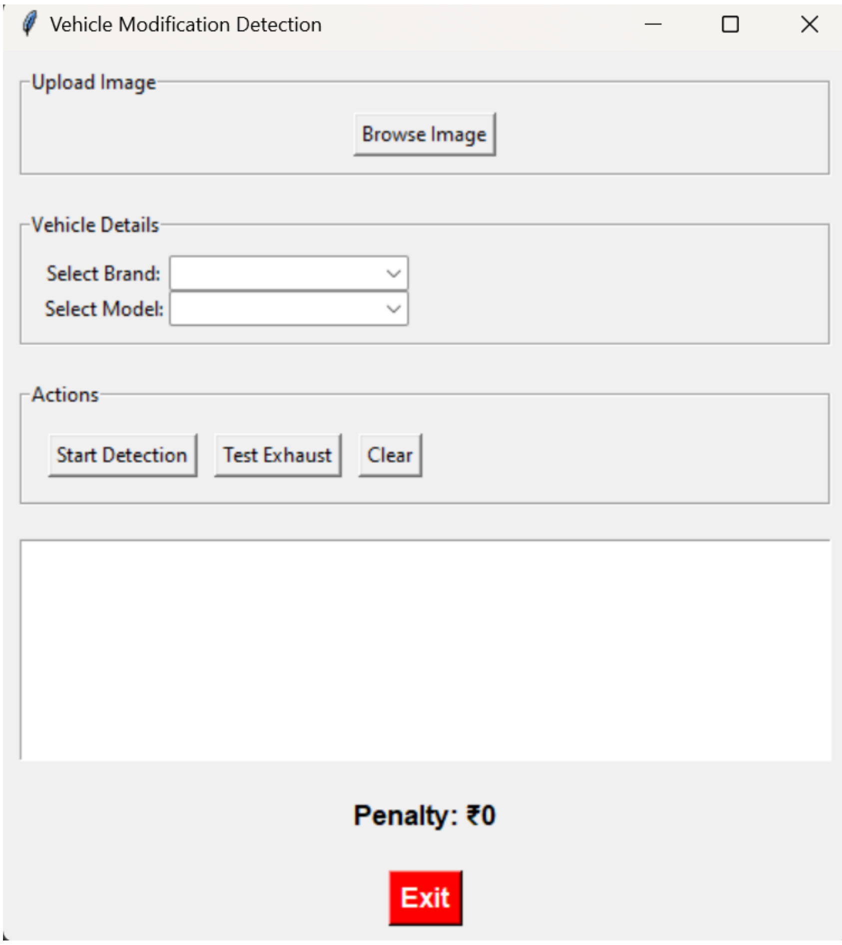


Fig 7.1. Initial Page

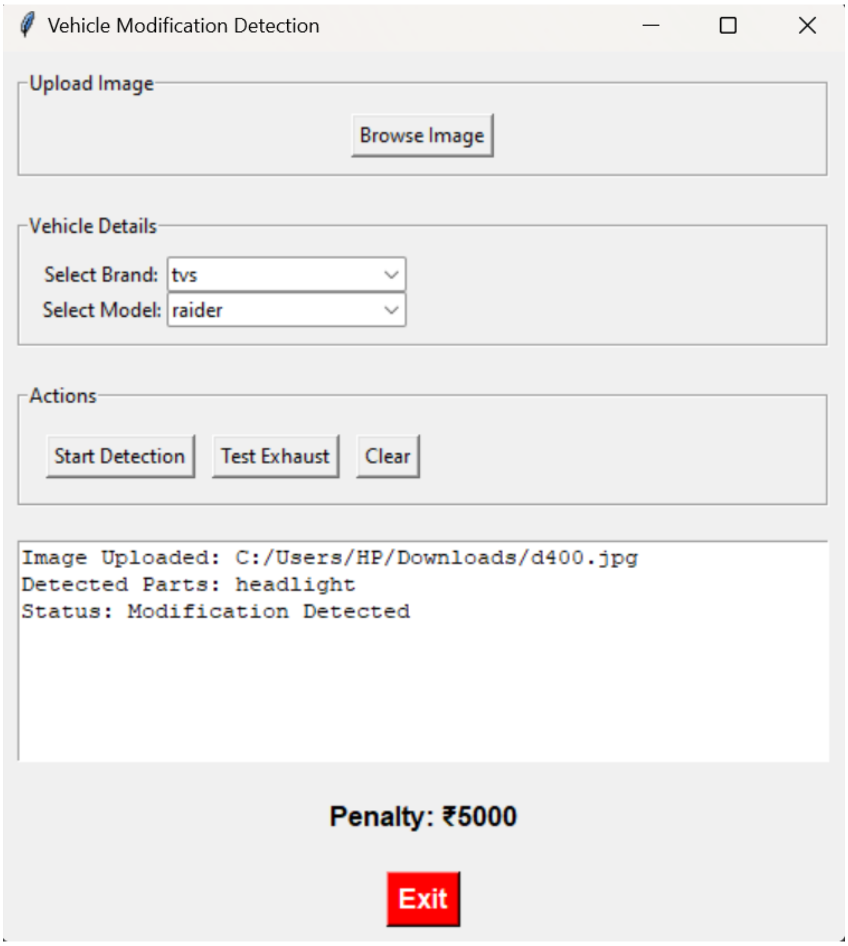


Fig 7.2. Detection Process

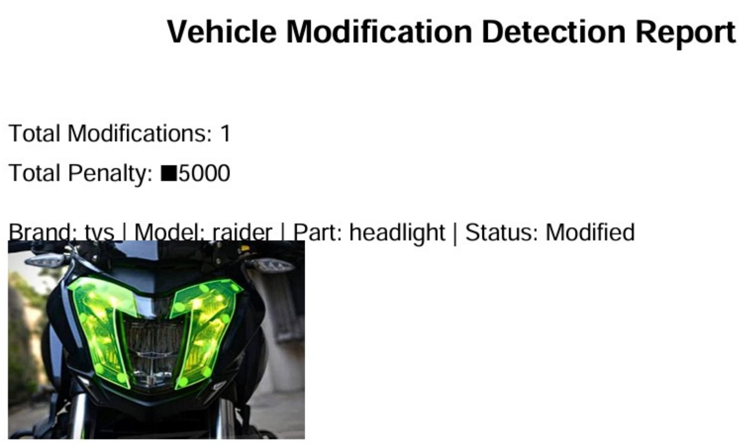


Fig 7.3. Penalty Report

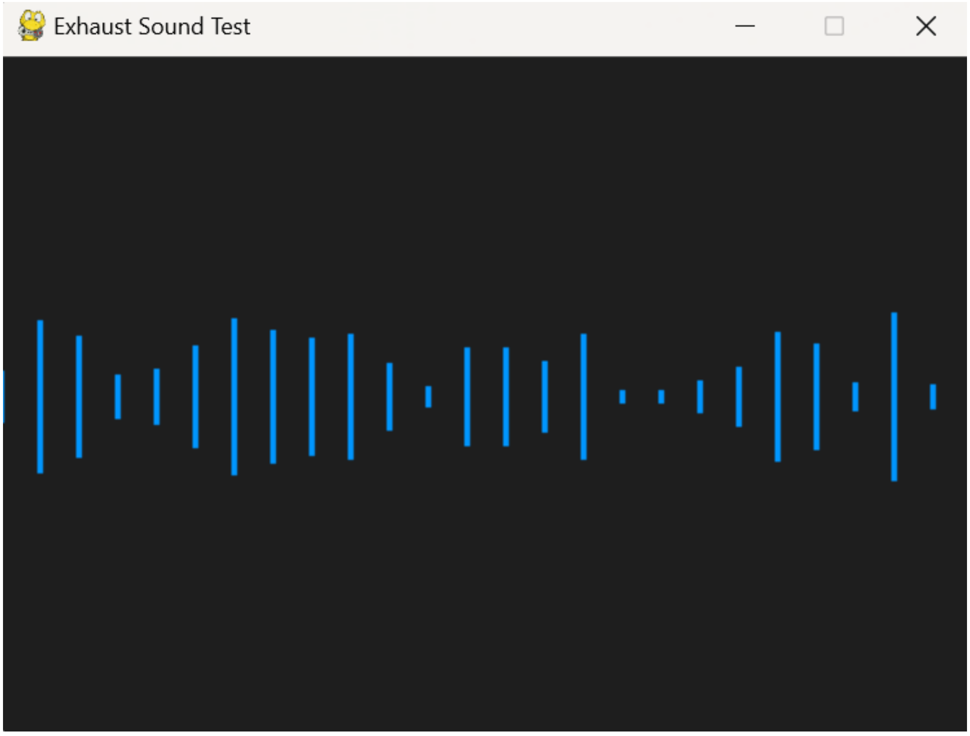


Fig 7.4. Exhaust Sound Test

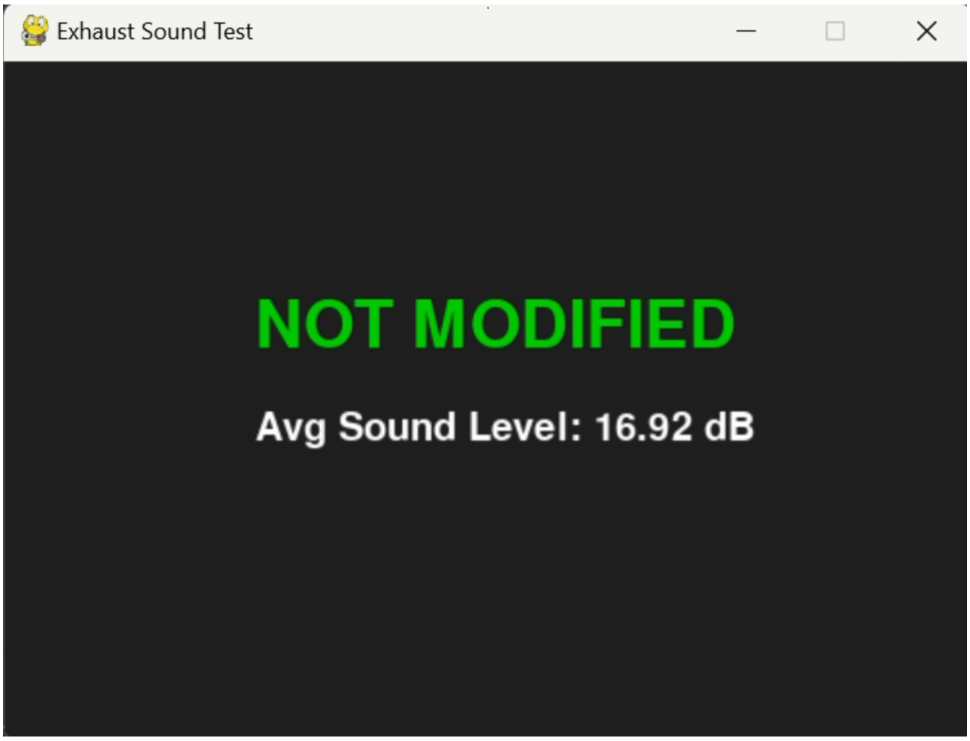


Fig 7.5. Exhaust Sound Result

## 7.2 Important Codes

```
def calculate_decibels(data):
    rms = audioop.rms(data, 2)
    if rms > 0:
        db = 20 * math.log10(rms / 32768)  # Convert to decibels
        return max(0.1, db + 100)  # Avoid returning 0
    return 0.1  # Small non-zero value to prevent incorrect averaging

# Function to generate wave animation
def draw_waveform(amplitude): ...
    pygame.display.flip()

# Collect sound readings for 10 seconds
decibel_readings = []
start_time = time.time()

while time.time() - start_time < TEST_DURATION:
    data = stream.read(CHUNK, exception_on_overflow=False)
    decibels = calculate_decibels(data)
    decibel_readings.append(decibels)

    # Draw wave animation based on sound level
    draw_waveform(decibels)

# Calculate final values
average_db = sum(decibel_readings) / len(decibel_readings)
modification_status = "MODIFIED" if average_db > THRESHOLD_DB else "NOT MODIFIED"
status_color = RED if modification_status == "MODIFIED" else GREEN
```

Fig 7.6. Exhaust Sound Analysis

```
from torchvision import transforms

def preprocess_image(image):
    transform = transforms.Compose([
        transforms.Resize((64, 64)),
        transforms.ToTensor(),
        transforms.Normalize((0.5,), (0.5,))
    ])
    return transform(image)
```

Fig 7.7. Preprocessing Step

```python
# Mapping class IDs to part names
CLASS_ID_TO_PART = {
    0: "exhaust",
    1: "headlight"
}

# Expand bounding box size
EXPAND_RATIO = 0.15

class YOLOModel:
    def __init__(self, model_path):
        self.model = YOLO(model_path)  # Load the YOLO model

    def detect(self, image_path, conf_threshold=0.2):  # Adjust confidence threshold
        results = self.model(image_path)  # Run detection
        detections = []

        # ✅ Load image size using OpenCV
        image = cv2.imread(image_path)
        image_height, image_width = image.shape[:2]

        for result in results:
            if not hasattr(result, 'boxes') or result.boxes is None:
                continue  # Skip if no boxes detected

            for box in result.boxes:
                conf = box.conf.item()
                class_id = int(box.cls.item())

                if conf >= conf_threshold:
                    x1, y1, x2, y2 = box.xyxy.tolist()[0]

                    # ✅ Expand bounding box before cropping
                    width = x2 - x1
                    height = y2 - y1
                    x1 -= width * EXPAND_RATIO
                    x2 += width * EXPAND_RATIO
                    y1 -= height * EXPAND_RATIO
                    y2 += height * EXPAND_RATIO

                    # Ensure bounding box remains within image bounds
                    x1, y1, x2, y2 = max(0, x1), max(0, y1), min(x2, image_width), min(y2, image_height)

                    part_name = CLASS_ID_TO_PART.get(class_id, "unknown")
                    print(f"[DEBUG] Class ID: {class_id}, Mapped Part: {part_name}, Expanded Bounding Box: {x1, y1, x2, y2}")

                    detections.append({
                        "bbox": [x1, y1, x2, y2],
                        "part": part_name,
                        "confidence": conf
                    })

        return detections
```

Fig 7.8. Part Detection by YOLO

```
yolo_model = ObjectDetection(yolo_model_path)
print("[DEBUG] Running second YOLO detection on cropped image...")
part_detections = yolo_model.detect(cropped_image_path)
print(f"[DEBUG] Second YOLO Detection Results: {part_detections}")

if not part_detections:
    print("[INFO] Modification Detected: No object found in cropped image.")
    append_modification(brand, model, detected_part, "Modified", cropped_image_path)
    return {"status": "Modification Detected."}
```

Fig 7.9. Modification Detection by YOLO

```
# Load Autoencoder Model
print(f"\n[INFO] Loading Autoencoder Model for {brand} {model} {detected_part}...")
anomaly_detector = AnomalyDetector(autoencoder_model_path)
cropped_image = Image.open(cropped_image_path).convert("L")
preprocessed_image = preprocess_image(cropped_image).unsqueeze(0)

# Compute Reconstruction Error
reconstruction_error = anomaly_detector.compute_reconstruction_error(preprocessed_image)
print(f"[INFO] Reconstruction Error: {reconstruction_error}")

# Load error statistics
with open("C:/Users/HP/OneDrive/Desktop/vehicle_modivication_detection - Copy/config/error_statistics.json", "r") as f:
    error_statistics = json.load(f)

if brand in error_statistics and model in error_statistics[brand] and detected_part in error_statistics[brand][model]:
    mean_error = error_statistics[brand][model][detected_part]["mean_error"]
    std_error = error_statistics[brand][model][detected_part]["std_error"]
else:
    print(f"[ERROR] Error statistics not found for {brand} {model} {detected_part}")
    return {"status": f"Error statistics not found for {brand} {model} {detected_part}"}

# Compute Dynamic Threshold
threshold = mean_error + (2 * std_error)
print(f"[INFO] Dynamic Threshold: {threshold}")

# Classify Anomaly
classification = anomaly_detector.classify_anomaly(reconstruction_error, threshold)
print(f"[RESULT] Final Classification: {classification}")

if classification == "Modified":
    append_modification(brand, model, detected_part, "Modified", cropped_image_path)  # ✅ Added PDF logging

return {f"{brand} {model} {detected_part}": classification}
```

Fig 7.10. Autoencoder and Rest of Modification Detection Logic

## 7.3 Discussion

### System Performance and Efficiency

The Two-Wheeler Vehicle Modification Detection System was designed to automate the identification of modified vehicle parts using computer vision and sound analysis techniques. By leveraging YOLO-based part detection, anomaly detection and exhaust sound testing, the system provides an efficient and structured approach to identifying unauthorized modifications. Performance evaluations indicate that the system accurately detects modifications in vehicle exhausts and other parts while maintaining a user-friendly interface built with Tkinter and Pygame. The real-time analysis capabilities ensure quick decision-making, while automated report generation enhances usability for inspectors and authorities.

### Impact on Vehicle Modification Detection

The system introduces an automated and standardized method for detecting vehicle modifications, offering several key benefits:

- Automated Detection: Eliminates the need for manual inspection by leveraging machine learning to analyze images and sound data.
- Improved Accuracy: Combines YOLO-based part detection, anomaly detection and exhaust sound analysis to enhance modification identification precision.
- User-Friendly Interface: The integration of Tkinter for structured UI and Pygame for animations provides an engaging and intuitive user experience.
- Efficient Report Generation: The system automatically generates PDF reports, documenting detected modifications with corresponding images and sound data, improving regulatory enforcement.

## Challenges and Limitations

Despite its advantages, the system faces certain challenges that could be improved in future iterations:

- Scalability Constraints: The system currently operates on a limited dataset. Expanding it to detect modifications across a wider range of vehicle models would require additional training data and model optimization.
- Lighting and Angle Sensitivity: The YOLO-based part detection may be affected by lighting conditions, occlusions and camera angles, potentially impacting detection accuracy.
- Sound Variability Issues: Exhaust sound analysis depends on environmental noise levels and microphone quality, which may introduce inconsistencies in sound-based modification detection.
- Accuracy Issues: The ML model may misclassify or fail to detect vehicle parts, especially in low-resolution images or unconventional exhaust designs, leading to false modification alerts.
- Hardware Dependency: Real-time detection performance may vary based on system hardware capabilities, especially when running deep learning models on lower-end devices.

## Future Scope

To further improve system effectiveness, several enhancements are proposed:

- **Real-Time Video-Based Detection**: Extend the system to support live video input, enabling continuous monitoring of vehicle parts and modifications in real-time.
- **Automatic Bike Brand and Model Detection**: Implement an automated system to identify the brand and model of the bike, reducing the need for manual input and improving detection accuracy.

- **Expansion of Part Detection**: Enhance YOLO-based detection to include additional components such as wheels, mirrors and body modifications, providing a more comprehensive modification detection system.
- **Cloud-Based Data Storage**: Enable cloud storage for test results, allowing users to store and retrieve historical modification records. This will assist in tracking vehicle modifications over time.
- **Enhanced Report Generation**: Improve the PDF reporting system by adding detailed insights, comparative analysis and trend tracking. Allow users to compare past and present modifications.
- **Integration with Law Enforcement**: Link the system with traffic law enforcement databases for automated detection of illegal modifications. Automated report generation can be connected to a centralized system for better monitoring.
- **Mobile Application for On-the-Go Detection**: Develop a mobile app version that allows users to capture images and record sound directly from their smartphones, making the system more accessible for vehicle owners, inspectors and enforcement agencies.

# Chapter 8

# Conclusion

The Two-Wheeler Vehicle Modification Detection System offers a robust, automated solution for identifying unauthorized modifications in components such as exhausts and headlights. By combining advanced deep learning techniques—including YOLO-based object detection and autoencoder-based anomaly detection—with real-time exhaust sound analysis, the system provides a multi-faceted approach to compliance monitoring. It removes the dependency on manual inspections, offering faster and more reliable detection of modified vehicle parts. The system architecture supports seamless integration of multiple detection models, each tailored for specific vehicle parts and models, which enhances its adaptability and accuracy across various use cases.

With a clean and intuitive user interface, the system ensures ease of use for law enforcement agencies and regulatory bodies. It delivers real-time analytics and generates comprehensive PDF reports that include visual and sound-based evidence of modifications, allowing for informed and transparent decision-making. Designed with scalability and future readiness in mind, the system can easily accommodate additional features or regulatory updates. In doing so, it not only streamlines the enforcement process but also contributes to safer roads, reduced noise and emissions, and greater public compliance with vehicle modification laws.

# References


[1] Amit Juyal, Sachin Sharma and Shuchi Bhadula. ”Modified Vehicle Detection and Localization Model for Autonomous Vehicle Traffic System.” 2024.

[2] Ahad Alloqmani, Yoosef B. Abushark, Asif Irshad Khan and Fawaz Alsolami. ”Deep Learning-Based Anomaly Detection in Images: Insights, Challenges and Recommendations.” International Journal of Advanced Computer Science and Applications, Vol. 12, No. 4, 2021.

[3] Mehmet Furkan Kunduraci and Humar Kahramanli Örnek. ”Vehicle Brand Detection Using Deep Learning Algorithms.” International Journal of Applied Mathematics, Electronics and Computers, Vol. 7, No. 3, 2019.

[4] Xingyang Ni and Heikki Huttunen. ”Vehicle Attribute Recognition by Appearance: Computer Vision Methods for Vehicle Type, Make and Model Classification.” Journal of Signal Processing Systems, Vol. 93, 2021. https://doi.org/10.1007/s11265-020-01567-6

[5] Zhong-Qiu Zhao, Peng Zheng, Shou-Tao Xu and Xindong Wu. ”Object Detection With Deep Learning: A Review.” IEEE Transactions on Neural Networks and Learning Systems. https://doi.org/10.1109/TNNLS.2018.2876865

[6] Hayfa Alshowaish, Yousef Al-Ohali and Abeer Al-Nafjan. ”Trademark Image Similarity Detection Using Convolutional Neural Network.” Applied Sciences, Vol. 12, No. 3, 2022. https://doi.org/10.3390/app12031752

[7] Jiazhi Liang. ”Image Classification Based on RESNET.” Journal of Physics: Conference Series, Vol. 1634, 2020. https://doi.org/10.1088/1742-6596/1634/1/012110

[8] Ajeet Ram Pathak, Manjusha Pandey and Siddharth Rautaray. ”Application of Deep Learning for Object Detection.” Procedia Computer Science, Vol. 132, 2018, pp. 1706–1717. https://doi.org/10.1016/j.procs.2018.05.144

[9] César Giovany Pachón-Suescún, Paula C. Useche Murillo and Robinson Jimenez-Moreno. ”Scratch Detection in Cars Using a Convolutional Neural Network by Means of Transfer Learning.” International Journal of Applied Engineering Research, Vol. 13, No. 16, 2018, pp. 12976-12982.

[10] Rohan Watkins, Nick Pears and Suresh Manandhar. ”Vehicle Classification Using ResNets, Localisation and Spatially-Weighted Pooling.” Preprint submitted to Elsevier, 2018. https://arxiv.org/abs/1810.10329

[11] Manisha Vashisht and Brijesh Kumar. ”A Survey Paper on Object Detection Methods in Image Processing.” Manav Rachna International Institute of Research And Studies.

[12] Constantine P. Papageorgiou, Michael Oren and Tomaso Poggio. ”A General Framework for Object Detection.” Sixth International Conference on Computer Vision, 1998.

[13] Moahaimen Talib, Ahmed H. Y. Al-Noori and Jameelah Suad. ”YOLOv8-CAB: Improved YOLOv8 for Real-time Object Detection.” Karbala International Journal of Modern Science, Vol. 10, No. 1, 2024.

[14] Shuangshuang Chen and Wei Guo. ”Auto-Encoders in Deep Learning—A Review with New Perspectives.” Journal of Machine Learning Research, Vol. 24, No. 1, pp. 1–45, 2023.

[15] Sarfaraz Natha, Mehwish Leghari, Muhammad Awais Rajput, Syed Saood Zia and Jawaid Shabir. ”A Systematic Review of Anomaly Detection Using Machine and Deep Learning Techniques.” Quaid-e-Awam University Research Journal of Engineering Science & Technology, Vol. 20, No. 1, pp. 83–94, 2022.