



**SCUOLA DI DOTTORATO**  
UNIVERSITÀ DEGLI STUDI DI MILANO-BICOCCA

Department of  
Informatics, Systems and Communication

Ph.D. in Computer Science  
XXXVI Cycle

# Assessment of Car Damage from Photographs

Surname: ORLOV

Name: IVAN

Registration number: 869195

Supervisor: Prof. Raimondo Schettini

Co-supervisor: Dr. Maurizio Rossi

Co-supervisor: Dr. Marco Buzzelli

Tutor: Prof. Rafael Peñaloza

Coordinator: Prof. Leonardo Mariani

**ACADEMIC YEAR 2022/2023**

# Table of Contents

Abstract . . . . .	iv
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Problem Statement . . . . .	2
1.2.1 Challenges in Damage Assessment from Photographs . . . . .	2
1.2.2 Aim of the Research . . . . .	3
1.3 End-to-End Approaches in Vehicle Damage Assessment: A Review . . . . .	4
1.4 Rationale for Module Selection and Division . . . . .	6
1.4.1 Vehicle Recognition/Identification . . . . .	7
1.4.2 Damage Recognition . . . . .	9
1.4.3 Repair Costs Estimation . . . . .	10
1.5 Organization of the Thesis . . . . .	11
1.5.1 Novelties and Contributions of the Research . . . . .	12
<b>I Vehicle Identification</b>	<b>14</b>
<b>2 Data and Dataset Characteristics</b>	<b>15</b>
2.1 Introduction . . . . .	15
2.2 Analysis of Existing Datasets . . . . .	15
2.2.1 DeepCar 5.0 Dataset . . . . .	16
2.2.2 Frontal-103 Dataset . . . . .	16
2.2.3 Stanford Cars Dataset . . . . .	16
2.2.4 LSUN Dataset . . . . .	18
2.2.5 VMMR Dataset . . . . .	19
2.2.6 CompCars Dataset . . . . .	19
2.2.7 Limitations of Existing Datasets . . . . .	20
2.3 Brumbrum’s Unique Dataset . . . . .	22
2.4 Annotation and Domain Knowledge . . . . .	24
2.4.1 Methodology and Glossary Creation . . . . .	24
2.4.2 Distinguishing Model Variants and Commercial Namings . . . . .	24
2.4.3 Dataset Characteristics . . . . .	25
2.4.4 Annotation Validation . . . . .	27
2.4.5 Discussion . . . . .	29
<b>3 Vehicle Identification and Classification</b>	<b>30</b>
3.1 Photograph Type Classification: Exterior vs. Interior . . . . .	31
3.1.1 Introduction and Related Works . . . . .	31
3.1.2 Dataset Creation . . . . .	32

3.1.3	Transfer Learning and Model Selection . . . . .	34
3.1.4	Results . . . . .	36
3.2	Vehicle Localization on Photographs . . . . .	38
3.2.1	Literature Review . . . . .	39
3.2.2	Need for Vehicle Localization in The System . . . . .	40
3.2.3	Model Selection and Qualitative Evaluation . . . . .	41
3.3	Vehicle Make/Model/Year Recognition . . . . .	43
3.3.1	Introduction . . . . .	43
3.3.2	Literature Review . . . . .	43
3.3.3	Dataset Utilization . . . . .	44
3.3.4	Restyling: Definition and Importance . . . . .	45
3.3.5	Two-Step Classification Approach . . . . .	46
3.3.6	Automated Dataset Creation for Restyling . . . . .	47
3.3.7	Classifier Design and Implementation . . . . .	47
3.3.8	Data Preprocessing . . . . .	48
3.3.9	Training and Results . . . . .	50
3.3.10	Discussion and Possible Improvements . . . . .	53
3.4	Conclusion . . . . .	54
<b>4</b>	<b>Vehicle Pose Detection</b>	<b>56</b>
4.1	Introduction and Related works . . . . .	56
4.1.1	Introduction to Pose Detection . . . . .	56
4.1.2	Traditional Image Processing vs. Deep Learning . . . . .	57
4.1.3	Car Pose Estimation . . . . .	57
4.1.4	Datasets for Car Pose Estimation . . . . .	58
4.2	Defining and Visualizing Azimuth . . . . .	60
4.3	Proposed approach . . . . .	61
4.3.1	Introduction and Motivation for Design Choices . . . . .	61
4.3.2	Architecture 1. Sin-Cos Representation . . . . .	61
4.3.3	Architecture 2. Directional Discriminators . . . . .	63
4.3.4	Evaluation method . . . . .	66
4.3.5	Training . . . . .	67
4.4	Results . . . . .	70
4.4.1	Quantitative Results . . . . .	70
4.4.2	Qualitative Results . . . . .	72
4.5	Discussion and Possible Improvements . . . . .	72
<b>II</b>	<b>Vehicle Damage Analysis</b>	<b>76</b>
<b>5</b>	<b>Component Recognition and Damage Presence Analysis</b>	<b>77</b>
5.1	Damage Dataset Description . . . . .	77
5.1.1	Structure of Expertise Documents . . . . .	78
5.1.2	Component Classification Dataset Creation . . . . .	79
5.2	Component Recognition . . . . .	81
5.2.1	Introduction and Related Works . . . . .	81
5.2.2	Selection of DCNN Architecture . . . . .	82
5.2.3	Integration of Vehicle Pose Information . . . . .	82
5.2.4	Training and Performance Analysis . . . . .	83

5.2.5	Ablation Study: Role of Pose Estimation . . . . .	86
5.2.6	Critical Evaluation and Possible Improvements . . . . .	88
5.3	Damage Presence Detection . . . . .	89
5.3.1	Related Works . . . . .	89
5.3.2	Objective and Approach Overview . . . . .	90
5.3.3	Dataset Description . . . . .	90
5.3.4	Analyzing the Influence of External Objects on Damage Recognition . . . . .	91
5.3.5	Development of Component-Specific Classifiers . . . . .	95
5.3.6	Performance Evaluation . . . . .	96
5.3.7	Critical Evaluation and Possible Improvements . . . . .	99
5.4	Conclusion . . . . .	100
<b>6</b>	<b>Vehicle Damage Repair Costs Estimation and System Evaluation</b>	<b>101</b>
6.1	Repair Costs Estimation . . . . .	102
6.1.1	Background and Related Work . . . . .	102
6.1.2	Dataset Description . . . . .	103
6.1.3	Characteristics of the Refined Dataset . . . . .	106
6.1.4	Methodology . . . . .	107
6.1.5	Evaluation Metrics . . . . .	108
6.1.6	Training Process and Performance Evaluation . . . . .	108
6.1.7	Results and Discussion . . . . .	110
6.2	End-to-End System Evaluation . . . . .	111
6.2.1	Test Setup and Dataset . . . . .	111
6.2.2	Evaluation Protocol . . . . .	113
6.2.3	Results Evaluation . . . . .	115
6.3	Conclusion . . . . .	118
<b>7</b>	<b>Conclusions</b>	<b>119</b>
	<b>Bibliography</b>	<b>122</b>



## Abstract

The used car dealer business model has traditionally been burdened with manual procedures for damage assessment from photographs. Even when damages are reported by external experts, a mandatory manual verification process is still necessary, often due to limited human resources leading to offer sampling. This inherent inefficiency prompted the exploration of automated solutions.

This research addresses the multifaceted challenges in automating the damage assessment process, including the variety of viewpoints, unrecognizable perspectives, poor lighting conditions, general image quality issues, and the presence of external objects in photographs. Further complexity arises from the subjective nature of damage assessment by experts in the field.

To tackle these challenges, the study explores the state of the art in damage recognition. Notably, most existing approaches rely on proprietary datasets, as open datasets are scarce and often lack representation of minor repairable damages. The research dissects the damage assessment problem into three interconnected subtasks: vehicle recognition/identification (comprising make, model, and production year determination), damage recognition (encompassing component identification, and damage presence classification), and the estimation of repair costs.

Several subsystems have been developed to construct a holistic solution. These include photograph type classification (discerning between exterior and interior images), vehicle detection and localization, vehicle make/model/year classification (offering comprehensive vehicle identification), vehicle pose detection (accurate azimuth estimation), component type classification (for precise damage localization), and damage presence classification.

This research presents an integrated framework that significantly enhances the efficiency of damage assessment processes within the used car dealership sector. In addition to developing various subsystems for comprehensive vehicle and damage analysis, this study has successfully completed an end-to-end evaluation of the entire system. This holistic evaluation demonstrates the practical applicability and robustness of the proposed solution, offering a significant leap in operational effectiveness and cost efficiency through advanced automation and optimization.

# Chapter 1

## Introduction

### 1.1 Background

The automotive industry has recently undergone significant transformations, characterized by advancements in vehicle technology and the evolution of sales and purchasing processes, particularly evident within the online contexts. This change is also observed in the Italian market, where traditional and digital environments intersect, creating new operational challenges and opportunities.

In this evolving landscape, various entities are seeking to adapt and innovate to meet contemporary demands. One such organization is brumbrum, a company operating primarily in the online second-hand car market. Established in Milan in 2016, brumbrum procures pre-owned vehicles, undertakes necessary refurbishments, and positions them within the consumer market, acknowledging the importance of vehicle quality in its operations. The company has invested in technological integration, evident through the establishment of a dedicated data science department. This move signifies a strategic pivot towards enhancing operational efficiency and customer service in the digital age.

The present research, conducted as part of an Executive Ph.D., addresses a practical business need while simultaneously contributing to academic discourse. Originating from the operational context of the used car dealership industry, the study focuses on the development of a system capable of assessing car damages through photographic images. Such a system is vital considering the industry's challenges, especially concerning damage assessment. For instance, with daily car offer volumes sometimes reaching up to 3,000, the magnitude of the assessment task becomes apparent. These offers, often presented in diverse formats, typically require manual verification of damage details due to the critical nature of the information. This endeavor not only aligns with the broader objectives of technological integration and enhanced digital service delivery but also upholds the rigorous standards and methodological robustness expected of scholarly research.

A standard car offer typically includes textual information regarding the vehicle, such as make, model, and other specifications. Nevertheless, the crux of these offers lies in the attached photographs. Situations where damage reports accompany the offers can further complicate the assessment, as these reports, though highlighting repair cost estimates, lack consistency. Given the direct implications of damages on the car's financial viability and the significance of repair costs on price determination, the role of expert evaluators in the organization is crucial.

Accurate damage assessment is not merely a procedural formality—it directly impacts the financial stability and efficiency of entities like brumbrum. Erroneous estimations or overlooks in the assessment can lead to unexpected repair costs, thereby affecting profitability. Such errors not only have financial implications but can also result in resource misallocation, causing operational disruptions. Consequently, an exhaustive damage assessment is paramount for sustaining profitability and ensuring operational continuity.

The manual damage assessment process in the used car industry, given its inherent inefficiencies, presents a compelling case for automation. By harnessing the potential of computer vision, this research aims to develop a system that can reliably evaluate car damages from photographs, fulfilling a crucial operational need for entities like brumbrum.

## 1.2 Problem Statement

### 1.2.1 Challenges in Damage Assessment from Photographs

Car damage assessment, particularly from photographs, presents a multifaceted challenge, accentuated by the eclectic variety in the car market and the unpredictable nature of photograph captures. Figure 1.1 provides visual examples of some of the challenges encountered in this context. The major ones can be enumerated as:

1. **Variability in Makes/Models and Accessories:** The diverse range of car makes and models, coupled with the multiplicity in colors and additional accessories, adds layers of complexity to the recognition and assessment task.
2. **Varying Damage Severities:** While the company primarily focuses on cars that can be relatively easily repaired, distinguishing between major damages, repairable damages, and minor aesthetic issues becomes pivotal.
3. **Lack of Standard Expertise Procedure:** The absence of a standard procedure for capturing damage photographs results in a plethora of poses, distances, and backgrounds, further complicating the assessment process.
4. **Photographic Artifacts:** Reflections, external object presence (e.g., fingers, rulers), and unorthodox perspectives add noise and ambiguity to the actual damage depiction.
5. **Image Quality Issues:** Poor lighting conditions, along with other generic image quality issues, can obscure damages or create false impressions.
6. **Subjectivity in Damage Reporting:** Experts sometimes have varied interpretations of damage severity—what might be severe for one might be perceived as minor by another, leading to inconsistencies in damage reporting.

To streamline the process of damage assessment, it is essential to classify damages based on their severity. While the company has no interest in cars with the most severe damages, understanding the spectrum can guide the automated system in discerning between acceptable and unacceptable damage levels. The categories, also shown on Figure 1.2, are:



Figure 1.1: Illustrative examples of challenges in automatic damage assessment from photographs: confusing reflections and unrecognizable perspective on the left, shadows and unclear declared damages in the center, and presence of external objects like fingers and rulers on the right.

1. **Worst/Unacceptable Conditions:** Damages in this category render a car unfit for the company’s operations. They usually comprise major structural or mechanical issues, which are not just cosmetic or superficial. Fortunately, such damages are sparse in the offers received.
2. **Poor Conditions but Repairable:** Cars in this category possess significant damages, yet they are repairable. It is essential to accurately assess them to estimate repair costs effectively.
3. **Good Conditions with Acceptable Damages:** These are minor imperfections or cosmetic issues that can be easily addressed. Distinguishing them from more severe damages ensures that the company can make informed decisions about refurbishing and pricing.



Figure 1.2: Examples showcasing the spectrum of damage severity: from non-repairable damages on the left to minor, sometimes even acceptable damages on the right.

## 1.2.2 Aim of the Research

The central aim of this research is to develop a system tailored to the company’s operations, capable of predicting repair costs for car damages from photographs.

This entails processing the input images, extracting relevant features, and leveraging them to provide an accurate and consistent estimation of repair costs, in line with industry standards and company benchmarks.

While the immediate goal is to address the company’s specific needs, the broader ambition of this research is to introduce a novel **methodological approach** to the problem of car damage assessment. By “method”, the research emphasizes a systematic decomposition of the problem, encompassing stages like data collection, annotation, subsystem implementation, and evaluation of each component. This modular approach not only ensures that each segment of the problem is addressed with precision but also offers scalability and adaptability to diverse scenarios.

Furthermore, the novelty of this research lies in its holistic approach. While many existing systems might address individual facets of the problem, this research aims to provide an end-to-end solution, integrating advanced computer vision techniques with structured data processing. Additionally, the research will explore data augmentation techniques, robustness against photographic artifacts, and the potential integration of auxiliary data sources to enhance the accuracy and reliability of the system.

In essence, this research aims to bridge the gap between practical industry needs and cutting-edge academic advancements, offering a solution that is both innovative and immediately applicable.

### 1.3 End-to-End Approaches in Vehicle Damage Assessment: A Review

The literature has shown a growing interest in the domain of car damage assessment. A majority of the existing methodologies follow a similar trajectory, encompassing car detection, damage identification, and damage categorization [1, 2, 3]. The debate in the literature revolves around whether damage detection should be approached as an object detection task [2] or a segmentation task [4, 5]. The recent trend leans towards segmentation.

A recurring challenge highlighted by multiple studies is the inherent difficulty in consistently and accurately categorizing damage types and severity during the creation of ground truth labels [2, 6]. The blurred boundaries between different damage classes further complicate this issue. This ambiguity can lead to inconsistent labeling in training datasets, posing challenges for model training. Additionally, the absence of a large, publicly available dataset for this domain has been a significant limitation, making direct comparisons between different methodologies challenging [2, 7].

One of the pioneering methods for car damage classification was presented by [7], which explored the potential of fine-tuning a network pre-trained on ImageNet [8], a large-scale database of annotated images used broadly for object recognition software. The study also compared fine-tuning with pre-training using a convolutional auto-encoder and ensemble techniques to address the challenge of small dataset sizes. In contrast, another approach treated damage detection as an object detection task, employing a YOLO-based object detector [9], with a focus on anti-fraud measures [3].

Recent methodologies often employ a two-step process: car detection followed

by damage assessment [2, 5]. Such pipeline-based strategies are particularly relevant for automating damage assessment in user-submitted images. The first step involves using a detection network to isolate the car from the broader image context. Subsequent steps involve identifying damage characteristics such as class, location, severity, and size. Data augmentation and transfer learning, especially from networks pre-trained on datasets like COCO [10], have been employed to enhance classifier performance in the face of limited training images [4].

Different pipeline-based methodologies have varied nuances. Some require specialized camera setups [6], while others, like [1], classify damage severity and location but do not provide explicit damage localization. A growing trend in recent pipeline-based strategies is to treat damage detection as a segmentation task, predicting segmentation masks for damaged areas [4, 11, 12]. The Mask R-CNN network [13], an instance-segmentation model, has been particularly popular for this purpose, allowing for the identification of distinct damage areas [4, 11]. However, its efficacy has been questioned by some studies, suggesting alternative specialized networks [14].

Beyond damage assessment, some studies have proposed comprehensive end-to-end pipelines for the entire damage assessment process. For instance, [15] focuses on predicting customer churn for a limited set of car models. In contrast, [16] integrates natural language processing to extract claimant information, aiming to automate the entire insurance claims process. Another notable approach by [17] employs Mask R-CNN for damage identification, subsequently combining metadata and image features for a final appraisal.

The advent of deep learning has ushered in a new era for car damage assessment. Several studies have evaluated and compared deep learning algorithms for semantic segmentation of car parts [18]. Transfer learning has been a common theme, with studies exploring its efficacy in damage detection, localization, and severity assessment using models like VGG16 and VGG19 [19]. The potential of Convolutional Neural Networks (CNN) in detecting and estimating various damage types has also been explored, with models like Inception V3, Xception, VGG16, VGG19, ResNet50, and MobileNet being evaluated for their performance [20]. The study by [21] presents a unique approach using CNN for identifying vehicle damage, emphasizing the importance of timely and accurate damage assessment for insurance claims.

Recent works like [5] have proposed innovative pipelines for car damage assessment, emphasizing the use of in-the-wild mobile images. Their approach uniquely combines semantic analysis with instance segmentation, offering a comprehensive solution for damage assessment and cost estimation. Figure 1.3 provides a visual representation of their pipeline, exemplifying the decomposition approach for damage assessment. Such methodologies highlight the potential of integrating advanced computer vision techniques with structured data to revolutionize the car insurance industry.

In addition to the general overview provided in this section, it is important to note that further in-depth analyses of the literature specific to each step of the car damage assessment process will be presented in subsequent chapters of this thesis. These analyses will delve into the nuances of each stage, including vehicle identification, damage recognition, and repair cost estimation. Detailed literature reviews for these specific stages can be found in their respective chapters, further

enriching the context and foundation of this study.

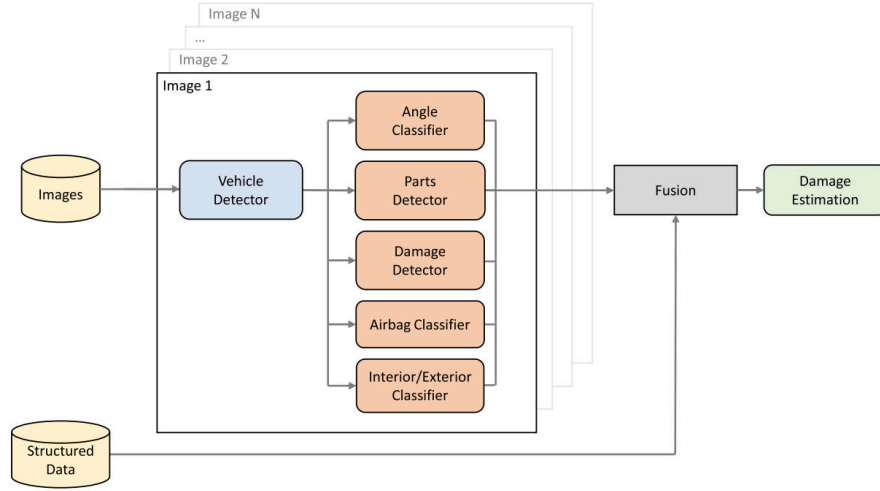


Figure 1.3: Pipeline for damage estimation as presented in [5], illustrating the decomposition approach to damage assessment.

## 1.4 Rationale for Module Selection and Division

Drawing from the extensive literature review, it becomes evident that the complexity of car damage assessment necessitates a multifaceted approach. Many of the discussed methodologies, whether they employ pipelines or end-to-end strategies, inherently break down the problem into discernible stages or components. This decomposition aligns with the broader trend in computer vision and machine learning, where complex tasks benefit from modular solutions.

Both modular and end-to-end approaches have their merits and demerits in the realm of car damage assessment. In a modular approach, which breaks down the task into separate subtasks, there are distinct advantages. These include the ability to analyze intermediate steps, leverage existing datasets specific to each subtask, and apply targeted improvements at each stage. However, this approach is not without challenges, such as the potential for error propagation across modules and the redundancy of analysis at different stages.

On the other hand, an end-to-end approach, which encompasses the entire process from input to final output, capitalizes on the strengths of deep learning. It learns features that are strictly necessary for the task, potentially leading to greater efficiency. However, this approach also presents difficulties, particularly in the analysis of intermediate reasoning errors, which can be complex and less transparent than in modular systems.

In light of these considerations, the decision to adopt a modular approach was made. This decision allows for an in-depth analysis and optimization at each stage, utilizing existing datasets and expertise specific to each subtask. However, to ensure a comprehensive understanding of the system's performance, an evaluation of the system in an end-to-end context is also included. This dual evaluation strategy aims to combine the benefits of both approaches, ensuring a robust and efficient solution for car damage assessment.

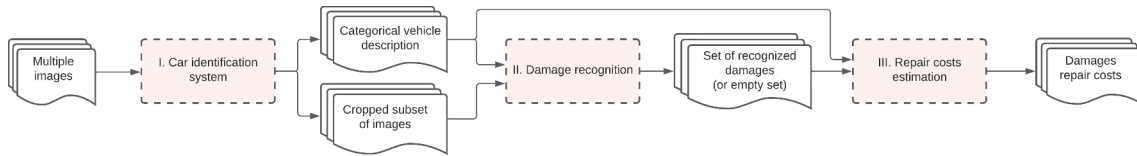


Figure 1.4: High-level flowchart depicting the three main steps in the vehicle damage assessment process as proposed in this thesis: 1) Vehicle identification, 2) Damage recognition, and 3) Repair costs estimation.

Referring to Figure 1.4, the process can be broadly categorized into three primary steps:

1. **Vehicle Identification:** The first crucial phase where the system discerns key vehicle details from the provided images.
2. **Damage Recognition:** Upon successful vehicle identification, the system shifts its focus to detect any present damages.
3. **Repair Costs Estimation:** With a clear understanding of the vehicle’s status and the damages identified, the system proceeds to estimate the associated repair costs.

These divisions not only streamline the process but also facilitate specialized optimization at each stage, ensuring overall accuracy and efficiency.

### 1.4.1 Vehicle Recognition/Identification

The task of accurately assessing vehicle damages begins with the fundamental step of correctly recognizing and identifying the vehicle in question. Such a foundational step is imperative not only for ensuring the specificity of subsequent assessments but also for streamlining the entire damage detection process. Given the diversity of vehicles and the variability in photographic conditions, this recognition process is inherently complex. To tackle this complexity head-on, it is essential to decompose the process into well-defined, sequential subtasks. Figure 1.5 provides a visual representation of this decomposition, elucidating the sequence and interrelations of the submodules. The ensuing sections delve into the crucial sub-modules within the scope of vehicle recognition: distinguishing the photograph’s type, pinpointing the vehicle’s location in the image, discerning the specific make, model, and year of the vehicle, and ascertaining its pose. Together, these modules cohesively work towards achieving a comprehensive and precise vehicle identification.

**Photograph Type Classification** The process of differentiating between photographs, be they of the vehicle’s exterior, its interior, or entirely unrelated subjects, stands as a pivotal initial step. The rationale behind this classification is twofold. Firstly, the primary focus of this research is centered around exterior vehicular damages. It is imperative to note that while interior damages represent a complex domain in their own right, they fall outside the purview of this study due to their inherent intricacies and unique challenges. As a result, such photographs, alongside unrelated subjects, should be dismissed at the earliest stage to maintain the specificity of the research. Secondly, from an efficiency standpoint, swiftly filtering out



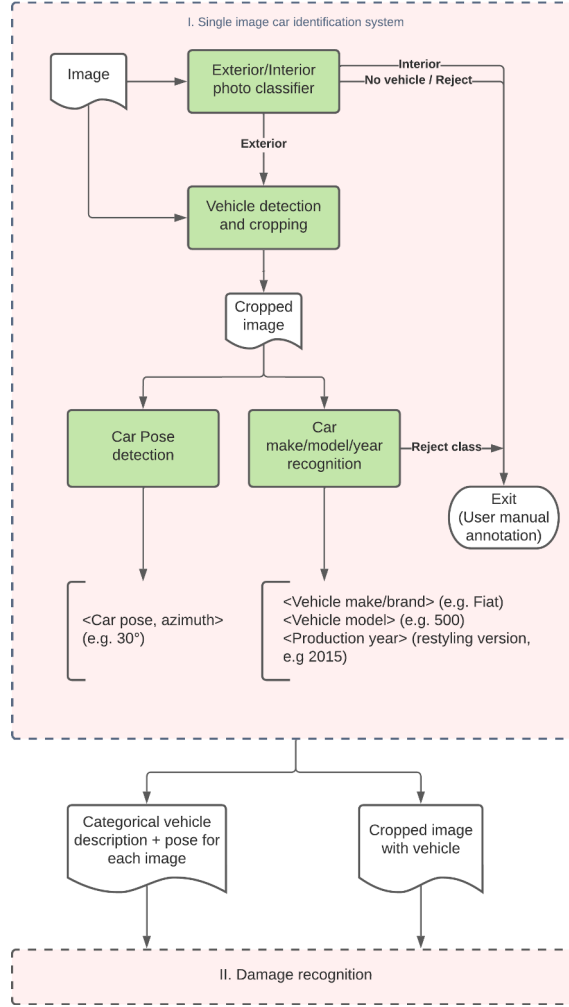


Figure 1.5: Flowchart illustrating the decomposition of the vehicle recognition process, highlighting the sequential steps and their interdependencies.

irrelevant images ensures optimal utilization of computational resources, thereby enhancing overall processing efficiency. The methodology and algorithms employed for this photograph type classification are comprehensively detailed in Section 3.1.

**Car Detection and Localization** Within the given image, it is paramount that the car remains the central subject. Achieving this not only heightens the precision of the assessment — by concentrating the region of interest solely on the vehicle, the subsequent modules are better positioned to function effectively, thereby minimizing the risk of false positives — but also aids in the elimination of any distracting backgrounds. By homing in exclusively on the vehicle, any extraneous elements in the periphery are systematically excluded, ensuring a more refined and targeted analysis. This module, which focuses on the vehicle localization within the image, is elaborated in the Section 3.2 of this thesis.

**Make/Model/Year Classification** The act of identifying specific vehicle details—such as make, model, and year—is crucial, particularly when accuracy in cost estimation is paramount. Different vehicle makes or models often carry distinct repair costs; thus, correctly recognizing them is an essential step in refining these

estimates. Additionally, being able to discern these specifics also ensures seamless compatibility with an existing database of past repairs, facilitating smoother historical comparisons and projections. A detailed discussion of this module, including its design and implementation, can be found in Section 3.3.

**Vehicle Pose Recognition** The orientation or “pose” of a vehicle in an image holds significant importance, especially when the task at hand involves identifying specific components to assess potential damage. Recognizing the vehicle’s pose not only aids in accurately pinpointing these individual car components, ensuring a heightened precision in subsequent damage detection, but also fosters view consistency. This consistency in recognition ensures that the damage detection system remains unfazed and is not misled by the varying perspectives from which the images might be taken. A comprehensive discussion of the methods and technologies employed for vehicle pose recognition is provided in Chapter 4.

### 1.4.2 Damage Recognition

After accurately identifying the vehicle, the next crucial phase is the recognition and characterization of potential damages. Recognizing damages is not just about detecting their presence; it involves a comprehensive understanding, starting with identifying the specific components affected and confirming the existence of damages. Each of these elements is vital in shaping the subsequent repair and cost estimation process. A visual representation of this damage recognition process is provided in Figure 1.6, illustrating the sequence and interconnectedness of the steps.

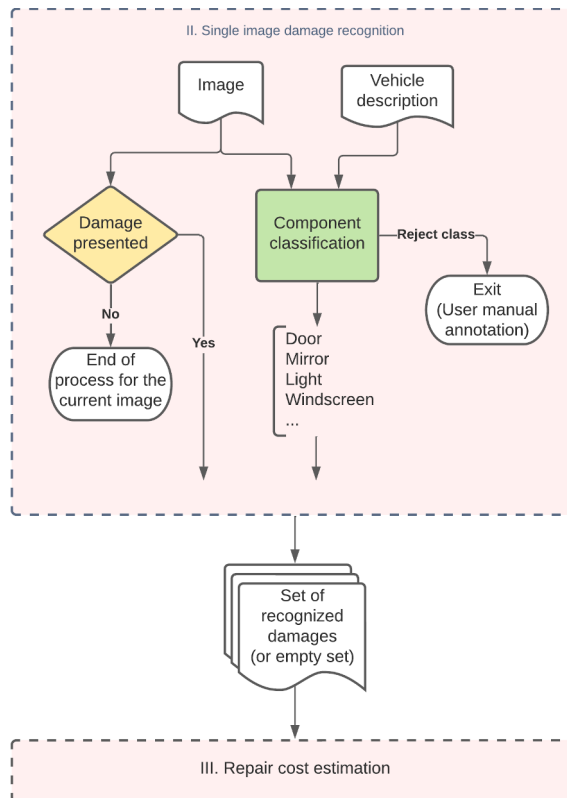


Figure 1.6: Flowchart delineating the stages involved in the damage recognition process, emphasizing their sequential and interdependent nature.

**Component Identification** In the journey of damage assessment, a fundamental step involves the accurate recognition of vehicle components. By pinpointing specific components, one can effectively localize where potential damage might reside, thereby streamlining the damage assessment process. Moreover, recognizing individual components also provides valuable context for repair costs, as certain components, due to their inherent complexity or function, might inherently carry higher repair costs than others. This module will be described in detail in Section 5.2.

**Damage Detection** Central to the damage assessment process is the essential task of discerning the existence of damages. This submodule serves a dual purpose. Firstly, it acts as a filter, ensuring that only vehicles with damages undergo more detailed analyses, optimizing resource allocation. Secondly, the act of damage detection offers preliminary insights, setting the stage for more detailed cost estimations in the following stages. This module is comprehensively discussed in Section 5.3.

### 1.4.3 Repair Costs Estimation

Upon gaining a holistic insight into the vehicle’s condition and the damages it has sustained, the culminating step in the system’s pipeline is the estimation of repair costs. This estimation is based on a comprehensive database of past repairs, grounding the predictions in real-world data. The goal here is to translate the identified damages into tangible monetary values. An illustrative representation of this repair costs estimation module can be seen in Figure 1.7.

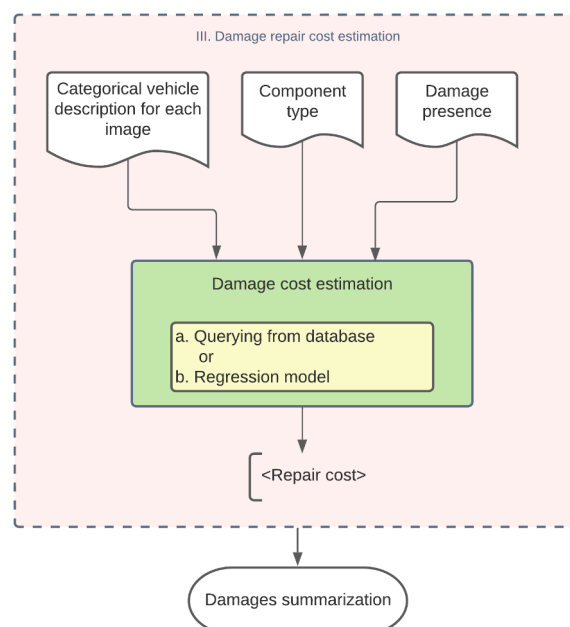


Figure 1.7: Diagram depicting the repair costs estimation module, emphasizing its data-driven approach.

Integral to the repair cost estimation is the “Cost Regression” submodule. This module harnesses the power of historical data, offering monetary estimates for the damages detected. Relying on past repair data imbues the process with a degree of

precision, ensuring that the predicted figures closely mirror actual repair costs experienced in real-world scenarios. Furthermore, this submodule is designed to evolve — as the database expands with more entries over time, the accuracy of the cost estimates is continually refined, enhancing the system’s reliability and robustness. A comprehensive discussion and detailed analysis of this module can be found in Section 6.1 of this thesis.

Decomposing the complex problem of vehicle damage assessment into these specific modules and submodules not only makes the challenge more tractable but also ensures that each stage is tackled with optimal precision and efficiency. A modular approach offers clarity and scalability. While automation seeks to improve consistency in the assessment process, it is essential to acknowledge that both manual and automated systems have their strengths and potential pitfalls.

## 1.5 Organization of the Thesis

This thesis is structured into two main parts, each focusing on distinct aspects of car damage assessment from photographs:

- **Part I: Vehicle Identification** - This part delves into the identification and classification of vehicles from photographs.
  - **Chapter 2: Data and Dataset Characteristics** - This chapter provides insights into the datasets utilized in this research, detailing their sources, characteristics, and the inherent challenges they present.
  - **Chapter 3: Vehicle Identification and Classification** - This chapter delves into the methodologies and techniques employed for identifying and classifying vehicles. It encompasses photograph type classification (distinguishing between exterior and interior shots), vehicle localization on photographs, as well as make, model, and year recognition. The core algorithms and their performance metrics for these tasks are presented.
  - **Chapter 4: Vehicle Pose Detection** - This chapter explores the techniques used to detect the pose of vehicles, an essential step for accurate damage assessment.
- **Part II: Vehicle Damage Analysis** - Building upon the vehicle identification, this part delves into the analysis of vehicle damage.
  - **Chapter 5: Component Recognition and Damage Presence Analysis** - This chapter explores two modules: component classification and damage presence detection. The methodologies employed to recognize specific vehicle components are presented, followed by approaches to detect the presence of damage on these components.
  - **Chapter 6: Vehicle Damage Repair Costs Estimation and System Evaluation** - This chapter delves into estimating the repair costs associated with detected damages, detailing the algorithms and their underlying rationale. Additionally, it presents an end-to-end evaluation, testing all the modules encompassed in the research.

The thesis is designed to provide a step-by-step understanding of the process, from vehicle identification to damage analysis and cost estimation.

### 1.5.1 Novelties and Contributions of the Research

This research introduces several novel contributions to the field of automated vehicle damage assessment, significantly advancing the state of the art. The key innovations are outlined as follows:

1. **Development of a Specialized Dataset with Annotations:** As detailed in Chapter 2, the dataset developed in this thesis represents a significant advancement in vehicle damage assessment, characterized by its enhanced accuracy and adaptability. The use of standardized annotations based on a well-defined glossary, coupled with the application of regular expressions, significantly elevates the precision and consistency of data categorization. This approach not only improves accuracy but also ensures scalability, allowing the dataset to expand without compromising quality. Its versatility is further enhanced by the comprehensive set of regular expressions, enabling it to adapt to a wide range of errors and variations in vehicle damage assessment.
2. **Advanced Vehicle Pose Estimation:** The thesis introduces two novel approaches for car azimuth estimation, both of which have demonstrated exceptional performance in state-of-the-art evaluations on the PASCAL3D+ dataset. These methods utilize the sinusoidal properties of orientations and the concept of directional discriminators, respectively. An intriguing finding is the minimal performance disparity between these approaches, indicating their comparable efficacy in various scenarios. Moreover, the models exhibit remarkable robustness, even in edge cases such as determining the orientation of cars under covers.
3. **Comprehensive Damage Identification Pipeline:** The thesis presents a comprehensive modular approach to vehicle damage identification, culminating in the development of an end-to-end pipeline. This pipeline intricately weaves together several key components: vehicle recognition, component classification, damage presence detection, and damage repair cost estimation. Its modular nature, as proposed and detailed in the thesis, represents a significant departure from traditional methods. By deconstructing the complex problem of car damage assessment into distinct, manageable subtasks, the thesis aligns with contemporary trends in computer vision and machine learning that favor modular solutions for complex challenges. By leveraging the strengths of specific models for each segment, the pipeline ensures a high degree of precision and efficiency, markedly improving the accuracy and functionality of car damage assessment system.
4. **End-to-End System Integration And Evaluation:** The thesis includes a novel end-to-end evaluation approach that incorporates all the modules and subsystems developed throughout the research. This comprehensive assessment utilizes a dataset of documents, each representing real-world cases of vehicles acquired and repaired by a company. These documents are enriched with images and technical metadata, such as full version names, registration dates, and gearbox specifications. The methodology for the end-to-end evaluation is designed to mirror real-world application scenarios. It starts with validating the exterior photography of the vehicles, progresses through various

classification and localization tasks, and culminates in the estimation of repair costs. This evaluation approach not only tests each individual module's effectiveness but also examines their integration and performance in a cohesive, practical context.

These innovations collectively represent a substantial leap forward in the automation of vehicle damage assessment processes. The implementation of these novel approaches promises to revolutionize the efficiency and accuracy of damage evaluations in the used car dealer sector, setting a new benchmark in the field.

**Part I**

**Vehicle Identification**

# Chapter 2

## Data and Dataset Characteristics

### 2.1 Introduction

The foundation of any machine learning or computer vision task lies in the quality and relevance of the dataset used. For the specific task of vehicle identification, especially in the context of the Italian car market, the dataset's nuances become even more critical. This chapter delves into the journey of curating a dataset tailored to the unique requirements of the company, highlighting the challenges faced, the decisions made, and the methodologies employed.

### 2.2 Analysis of Existing Datasets

In the rapidly evolving domain of vehicle recognition, the availability and quality of datasets play a pivotal role in determining the success of recognition systems. This section delves into an in-depth analysis of several prominent open-source car datasets. The primary objective of this exploration is to gauge the current state of the art and discern whether any of these datasets align with the specific requirements of the current task.

The evaluation is guided by a set of criteria, specified by the brumbrum company, which an optimal vehicle recognition system should meet:

- Focus on Italian/European cars.
- Include cars produced from 2010 onwards.
- Offer a comprehensive coverage of the market in terms of makes and models.
- Ensure high inter-class variance, implying multiple vehicles per class to capture diverse scenes, viewpoints, lighting conditions, car colors, and conditions (including both pristine and used cars).

Given these criteria, the subsequent sections provide a systematic evaluation of several datasets, assessing their characteristics, potential limitations, and relevance to the outlined requirements.



### 2.2.1 DeepCar 5.0 Dataset

The *DeepCar 5.0* dataset, introduced by Amirkhani et al. [22], presents a novel approach to vehicle make and model recognition (VMMR) based on front-view images of vehicles. The dataset is inspired by multi-agent systems (MASs) and ensemble models. The methodology emphasizes the importance of specific regions of interest (ROIs) on a vehicle, particularly the headlight, grill, scoop, and bumper sections. Unlike traditional methods that utilize the entire ROI, this approach extracts distinct ROIs from each image, with each ROI undergoing a unique preprocessing block and network, treated as an individual agent. Each agent is trained separately, and the final vehicle type is determined collaboratively using a blackboard classification system.

The DeepCar 5.0 dataset comprises 40,185 images, capturing both the front views and the front three-quarters of vehicles. These images span 480 different classes and are sourced from the top 50 automakers. Notably, all parts of this dataset have been manually labeled.

### 2.2.2 Frontal-103 Dataset

The *Frontal-103* dataset, as described by Lu et al. [23], is a significant contribution to the realm of fine-grained vehicle categorization, a crucial aspect of Intelligent Transportation Systems. The dataset consists of 65,433 web-nature images, spanning 1,759 fine-grained vehicle models across 103 vehicle makes. The dataset’s primary focus is on frontal views of vehicles, providing a unique perspective for categorization tasks.

Frontal-103 stands out from other vehicle image datasets in several ways. It boasts a larger scale and diversity, with a meticulous focus on accuracy at a fine-grained level. The dataset’s images are sourced from the Internet, ensuring a wide variety of real-world scenarios. A selection of sample images from the dataset is depicted in Figure 2.1. The dataset aims to address specific challenges inherent to fine-grained vehicle categorization, such as the nuanced differences between closely related vehicle models. However, a potential limitation of the Frontal-103 dataset is its exclusive focus on frontal views, which might not capture the complete essence of a vehicle from multiple angles.

### 2.2.3 Stanford Cars Dataset

The Stanford Cars dataset, introduced by Krause et al. [24], was developed to address the challenges associated with fine-grained categorization, particularly in the context of 3D object representations. While 3D object representations have seen applications in multi-view object class detection and scene understanding, their utilization in fine-grained categorization has been limited. The primary motivation behind the Stanford Cars dataset was to lift state-of-the-art 2D object representations to 3D, both in terms of local feature appearance and location.

#### Dataset Characteristics

The dataset consists of 16,185 images, categorized into 196 distinct classes. These classes are defined at the granularity of Make, Model, and Year (MMY). The dataset

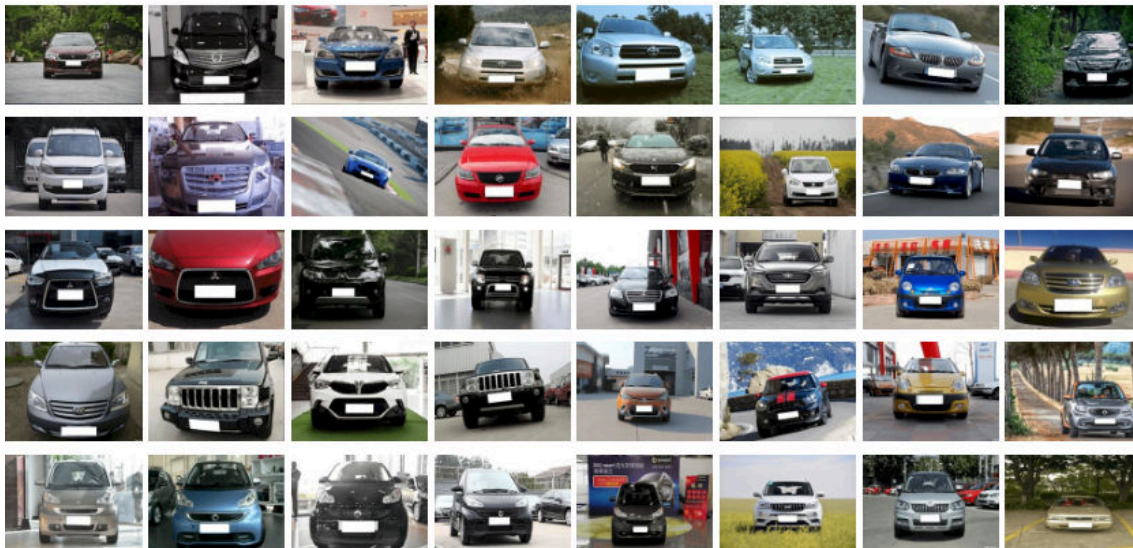


Figure 2.1: Sample images from the Frontal-103 dataset, showcasing the diversity and fine-grained categorization of vehicle frontal views.

has been divided into a training set with 8,144 images and a testing set with 8,041 images, maintaining an approximate 50-50 split for each class. Some of the MMY triplets also incorporate a concept similar to the car type within the label, offering a nuanced classification system.

Additionally, a smaller subset, known as the *BMW-10 Stanford dataset*, is available. This subset comprises 512 images, focusing exclusively on 10 models from different production years within the BMW car make.

### Data Quality and Annotation

Despite its structured approach to categorization, the Stanford Cars dataset is not devoid of challenges. An immediate observation reveals inconsistencies in labeling, as depicted in Figure 2.2.



Figure 2.2: Examples of mislabeled data in the Stanford Cars dataset. From left to right: An image labeled as “Audi TT hatchback 2011” which is actually an “Audi TT RS Coupe 2011”, an image labeled as “Audi RS4 convertible 2008” which is in reality an “Audi S5 convertible 2012”, and an image labeled as “Dodge Durango 2012” that is a “Jeep Grand Cherokee 2012”

Furthermore, the dataset contains images with noise, which can be attributed to ambiguous poses or the presence of multiple vehicles in a single image, as illustrated

in Figure 2.3.



Figure 2.3: Illustrative examples of noisy data from the Stanford Cars dataset. The images depict challenges such as ambiguous car poses and the presence of multiple vehicles in a single frame.

### 2.2.4 LSUN Dataset

The Large-scale Scene Understanding (LSUN) dataset, introduced by [25], was developed to address the increasing demand for large labeled training datasets, especially given the data-hungry nature of state-of-the-art visual recognition algorithms. The primary motivation behind LSUN was to keep pace with the rapid growth in model capacity, as existing datasets were quickly becoming outdated in terms of size and density.

#### Dataset Creation and Characteristics

The LSUN dataset [25] was constructed using a partially automated labeling scheme, which combined deep learning with human annotators in a loop. Starting with a vast set of candidate images for each category, a subset was sampled and labeled by human annotators. A trained model then classified the remaining images. Based on the classification confidence, the dataset was divided into positives, negatives, and unlabeled sets. This process was iteratively repeated with the unlabeled set until a comprehensive dataset was formed.

The LSUN classification dataset comprises 10 scene categories, including dining rooms, bedrooms, churches, and notably the “car” category (over 5,5 million of images), which is particularly relevant to this research.

#### Data Quality and Annotation

Despite its size and diversity, the LSUN dataset is not without its challenges. An immediate observation reveals that the dataset, particularly the car category, contains various types of noise, indicating that the data might not be entirely pruned. This noise is evident in Figure 2.4, which showcases some of the inconsistencies in the dataset.



Figure 2.4: Examples from the LSUN car dataset highlighting the presence of noise and inconsistencies. Adapted from [26].

The presence of such noise can be attributed to the use of Amazon’s Mechanical Turk service [27] for data annotation. Mechanical Turk, a crowdsourcing marketplace, outsources tasks to a distributed virtual workforce. However, as noted by [28], Mechanical Turk annotators can often be imprecise, suggesting that relying solely on humans for image classification and annotation might not be the most optimal choice.

### 2.2.5 VMMR Dataset

The *Vehicle Make and Model Recognition (VMMR)* dataset, introduced by Tafazzoli et al. [29], emerges as a pivotal contribution to the domain of Intelligent Transportation Systems (ITS) and its components, particularly Automated Vehicular Surveillance (AVS). The significance of VMMR lies in its potential to substantially reduce overhead costs by offering accurate and real-time recognition systems. The VMMR problem is inherently multi-class, presenting unique challenges such as multiplicity and ambiguities both within and between vehicle makes.

The VMMRdb dataset is expansive and diverse, comprising 291,752 images that span 9,170 distinct vehicle classes. These images encapsulate models manufactured over a broad time frame, from 1950 to 2016. A distinguishing feature of this dataset is its inherent variability: images have been sourced from different users, captured using varied imaging devices, and encompass multiple view angles. This ensures a comprehensive representation of real-world scenarios, accounting for potential challenges like occlusions, varying illumination conditions, and partial camera views. Furthermore, the images in the dataset often include misalignments, irrelevant backgrounds, and other imperfections, reflecting the complexities of real-life data capture. Geographically, the dataset is extensive, covering vehicles from 712 areas, which span all 412 sub-domains corresponding to US metro areas. Such diversity positions the VMMRdb dataset as a robust foundation for training models tailored to real-life traffic surveillance scenarios.

### 2.2.6 CompCars Dataset

#### General description

The CompCars (Comprehensive Cars) dataset [30] contains data from two scenarios: “surveillance-nature” and “web-nature”. The “surveillance-nature” set comprises 50,000 frontal-view car images, each associated with bounding boxes, car models, and colors. On the other hand, the “web-nature” set includes a total of 136,725

images, where each car model is labeled with its Make-Model-Year (MMY) triplet, spanning three levels of detail with 163 makes, 1716 models, and 4455 models from different years. This dataset exhibits classes with low cardinality and groups of similar images. The hierarchical annotation labels enable addressing classification problems at different levels of detail, including hierarchical classification. Additionally, the CompCars dataset features viewpoint annotations (frontal, rear, side, frontal-side, rear-side), and model-specific attributes such as maximum speed, displacement, number of doors, and number of seats. The dataset also provides a collection of car part images categorized into eight groups (headlight, taillight, fog light, air intake, console, steering wheel, dashboard, gear lever).

### **Critique of the CompCars Dataset**

The CompCars dataset [30] is undeniably rich in its collection of car images with detailed annotations. This dataset has been recognized as an essential resource for classifying car images at varying levels of granularity, spanning from general car types to specific makes, models, and years. Buzzelli et al. [31] showed that convolutional neural networks achieve a commendable accuracy above 90% on the dataset for finest-level classification tasks. However, they argued that this success is not truly indicative of real-world performance due to biases in the training/test split. To address this, they introduced a more representative split, resulting in a more grounded accuracy figure of 61% on their new test set.

Despite its contributions, several concerns arise with the CompCars dataset upon closer inspection. Firstly, it contains near-duplicates or actual duplicate images, as highlighted in Figure 2.5. The categorization sometimes appears inconsistent and not always intuitive. Distinctions like “BMW 7 Series” versus “BMW 7 Series hybrid” raise questions about the rationale behind such splits, especially when visual differences are minimal and other potential divisions, such as by fuel type, are overlooked. Buzzelli et al. also made efforts to improve the dataset by expanding type-level annotations and providing car-tight bounding boxes for each image [31].

Moreover, the dataset includes images with recurring advertisements and rendered car images, both depicted in Figure 2.5. Annotations pertaining to model years occasionally present ambiguity, with instances of “unknown” values or seemingly arbitrary year splits. Lastly, the accuracy of some labels remains dubious, with evident errors such as “BWM” instead of “BMW” and “Porsche Cayenne” instead of “Porsche Cayenne”.

### **2.2.7 Limitations of Existing Datasets**

In the pursuit of a robust and comprehensive dataset for vehicle make and model recognition, especially tailored for the Italian/European car market, several existing datasets were evaluated. However, each of these datasets presents certain limitations that make them less suitable for current specific requirements. As it was mentioned in the beginning of this section, the ideal dataset should satisfy the following criteria:

- Focus on Italian/European cars.
- Include cars produced from 2010 onwards.
- Offer a comprehensive coverage of the market in terms of makes and models.



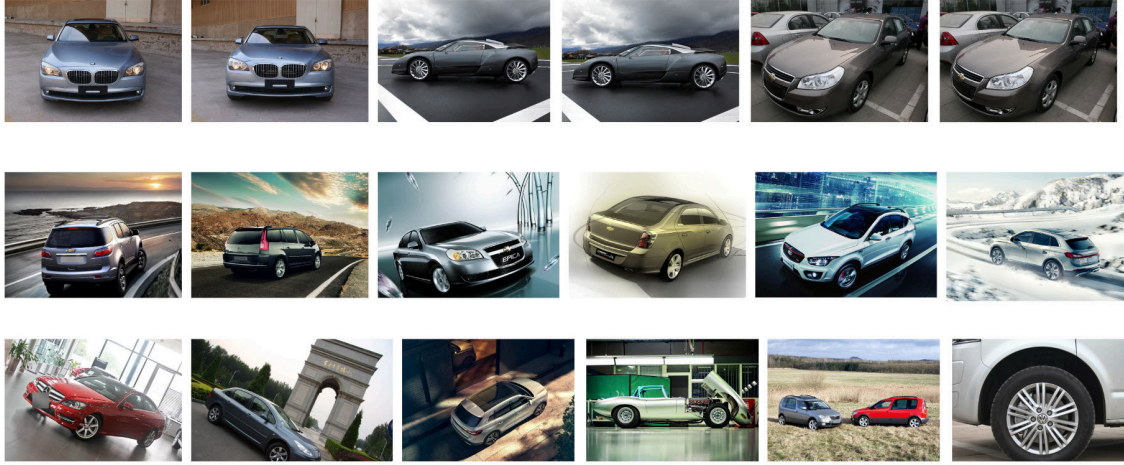


Figure 2.5: Anomalies observed in the CompCars dataset. **Top row:** Samples of duplicated or near-duplicate photos. **Middle row:** Unrealistic, rendered, or stylized images. **Bottom row:** From left to right: three images with unconventional viewpoints, an unmounted car during production, an image featuring two cars, and a zoomed-in photograph of a car wheel.

- Ensure high inter-class variance, implying multiple vehicles per class to capture diverse scenes, viewpoints, lighting conditions, car colors, and conditions (including both pristine and used cars).

Given these criteria, the limitations of the existing datasets are as follows:

1. **VMMR** [29]: Primarily contains US cars, many of which date back to the 1950s. Such old models are not relevant for current focus on modern cars.
2. **DeepCar 5.0** [22]: Exclusively focuses on cars from 2019, limiting its temporal scope.
3. **Frontal-103** [23]: Restricted to frontal viewpoints, which may not capture the complete essence of a vehicle.
4. **LSUN** [25]: Contains inconsistencies and anomalies in annotation (some samples are shown in Figure 2.4). The need for enhancement is evident as other works [26] have attempted to refine the dataset.
5. **Stanford Cars** [24]: Exhibits issues with annotation and noisy data (Figures 2.2 and 2.3). The dataset’s limitations led to efforts [26] to merge it with others to create a more robust collection.
6. **CompCars** [30]: Despite its common usage, this dataset is somewhat outdated (from 2015). It includes many concept cars, vehicles exclusive to the American/Asian market, and a plethora of non-European vehicles like vans, buses, and race cars. The dataset also suffers from limited photos per model, lack of visual differentiation for some restyling years, and issues like duplicates, unconventional viewpoints, and other outlier photos (as shown in Figure 2.5). The largest class has 175 images (depicted in Figure 2.6). Furthermore, many classes with fewer than 30 images often represent the same vehicle photographed in the same scene or event, indicating low inter-class diversity.

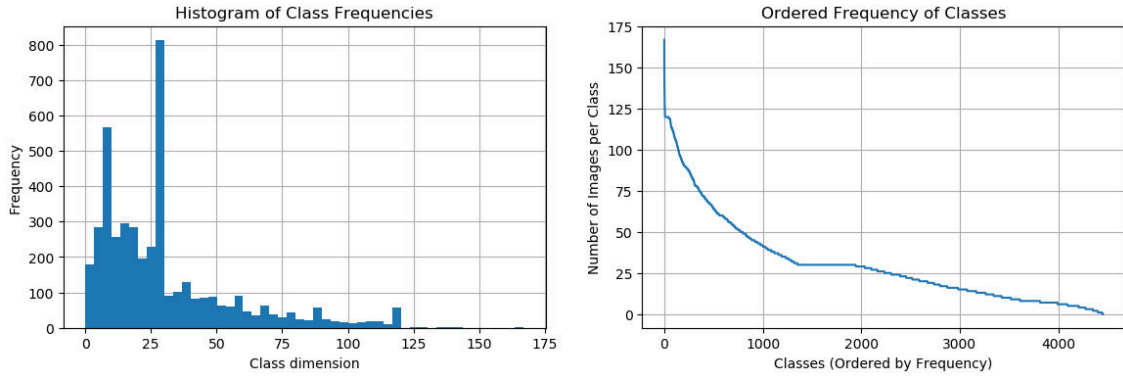


Figure 2.6: Class distribution in the CompCars dataset. **Left:** Distribution of images across different classes. **Right:** Classes ordered by their frequency, showcasing the number of images per class from the most frequent to the least frequent.

Given the limitations of these datasets, there is a compelling case for the creation of a new dataset that aligns more closely with current requirements.

## 2.3 Brumbrum’s Unique Dataset

Brumbrum’s dataset is a unique and extensive collection of car advertisements, primarily sourced from monitoring various car advertisement websites. This continuous monitoring provides the company with a vast amount of data, which, while primarily used for market and price analysis, offers a valuable resource for the car damage identification task.

The dataset is updated daily, with approximately 10,000 to 12,000 new car advertisements added. Each advertisement provides a wealth of structured information, including but not limited to the car’s make, model, version, registration date, gear type, fuel type, mileage, power, price, location, color, and a detailed description. Additionally, technical specifications such as emissions, consumption, optional features, and wheel drive are available. Each advertisement can have up to 16 associated images, showcasing the vehicle from various angles and under different lighting conditions. While there is no standardized protocol for capturing these images, the marketplace platform ensures they meet certain quality standards, such as the absence of unrelated objects or individuals in the foreground.

Given the vastness of the dataset, it is worth noting that advertisements can originate from both private and professional dealers. However, for the sake of reliability, the focus will be on the data from professional dealers, as their information tends to be more consistent and trustworthy.

The images in the dataset are primarily in the webp format, with a width of 1280 pixels. While the height varies due to differing aspect ratios, most images have a height of 960 pixels. A selection of these original images, illustrating the diversity and quality of the dataset, is presented in Figure 2.7.

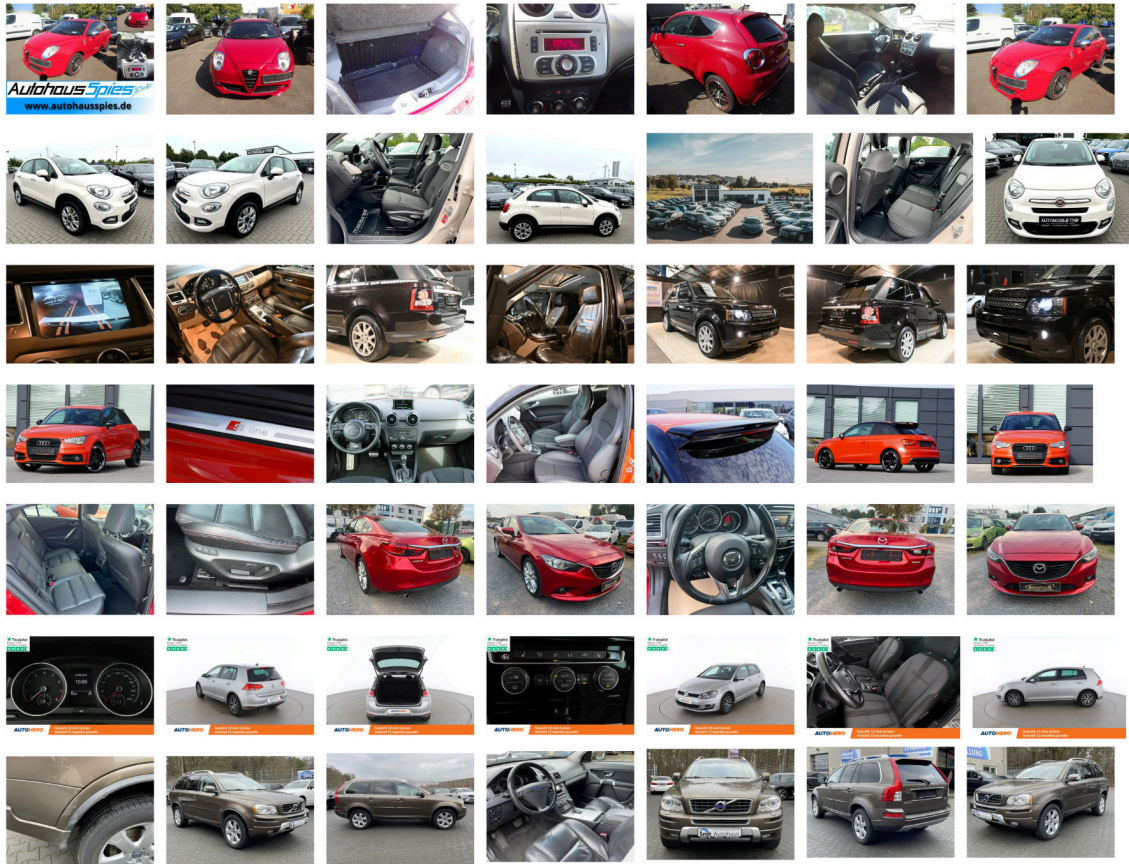


Figure 2.7: A sample of original, unprocessed images from the Brumbrum dataset. Each row showcases seven images from a single vehicle (advertisement), representing the diverse range of photographs uploaded by dealers. The images encompass various perspectives, including exterior views, interior shots, close-ups of specific details, and even promotional images from the dealers themselves, as seen in the second row, fifth column. The unordered arrangement of images in each row reflects the dataset’s initial and unstructured state.



## 2.4 Annotation and Domain Knowledge

### 2.4.1 Methodology and Glossary Creation

Web-based datasets, especially those derived from free-text inputs, present unique challenges in data annotation and validation. Dealers, when listing vehicles, often exhibit imprecision, omit crucial details, or introduce typographical errors. Such inconsistencies, if not addressed, can significantly impact the reliability and utility of the dataset. For instance, variations in naming conventions, such as “Mercedes” being listed as “Merc”, “Benz”, or even with typographical errors like “Mercedez”, can introduce ambiguity.

To address the aforementioned challenges, a foundational step involved creating a proprietary glossary of vehicle makes, models, and trim versions. This glossary serves as a reference point, ensuring consistency in annotations and reducing ambiguity.

Building on the glossary, a comprehensive set of regular expressions was developed to handle the diverse ways dealers might list vehicle details. Regular expressions, with their ability to match patterns in text, proved invaluable in capturing variations in naming conventions, handling common errors, and standardizing annotations. In total, approximately 5,000 regular expressions were crafted to cater to the myriad ways vehicle details might be listed.

### 2.4.2 Distinguishing Model Variants and Commercial Names

Consider the example of the Peugeot 308. Expert consultations led to the decision to split it into four distinct models: the standard 5-door version, the 3-door version, the CC (coupe cabrio), and the SW (station wagon). Given the textual data provided in each ad, the challenge was to accurately distinguish between these variants. Table 2.1 showcases examples of input strings and the corresponding model names that should be extracted.

The nomenclature for car variants can vary significantly across manufacturers. For instance, while Mercedes labels its station wagons as “Shooting Brake”, Audi uses “Avant”, BMW opts for “Touring”, Renault designates them as “Sporter”, Toyota prefers “Touring Sport”, and Volkswagen goes with “Sporter”. Complicating matters further, dealers often interchangeably use these terms across different makes, leading to potential misclassifications. Proposed methodology, therefore, had to account for these variations and potential confusions to ensure accurate annotations.

Certain car models and makes present unique challenges due to their history or branding. For instance, the Abarth 595, a tuned version of the Fiat 500, is occasionally mislabeled as the “Fiat 595”. Similarly, while Cupra cars were initially produced under the SEAT brand, some dealers continue to list them as “SEAT Cupra”. Another example includes the “Mini Mini” model, which should ideally be recognized as “Mini Cooper”, and the “Mini Cooper Countryman” is more accurately just “Countryman”. Additionally, the DS 4 model, which was known as “Citroën DS4” before 2015, is sometimes still associated with the Citroën brand in listings. Each of these nuances required careful consideration and the development of specific regular expressions to ensure accurate annotations.

Table 2.1 showcases various input strings from the ads and the corresponding

Table 2.1: Examples of Peugeot 308 model variations from ads.

Make	Model	Version	Redefined Model
Peugeot	308	hybrid 225 e-eat8 gt pack allure pack hybrid 180 e-eat8 nuova 308 bl - hybrid 180 e-eat8 allure pack hybrid 180 e-eat8 allure pack 1.6 PHEV 180 E-EAT8 ALLURE PACK 180 5P 308 Hybrid 180 e-EAT8 GT Pack 3 <sup>a</sup> serie Hybrid 225 e-EAT8 GT Pack PHEV 225ch GT Pack e-EAT8	308 5 doors
Peugeot	308	1.4 VTi 95CV 3p. Premium 308 1.4 VTi 95CV 3p. Comfort 308 1.6 e-HDi 115 CV Stop&Start Allure 3 doors 130 GT BVA GPS CAMERA I COCKPIT 3D 308 PureTech Turbo 130 S&S EAT8 Allure 3p	308 3 doors
Peugeot	308	CC 1.6 THP Sport Pack 140 Aut. 1.6 THP 140CV CC aut. Féline 308 2.0 HDi 136CV CC aut. Féline CC 2.0 HDI 136CV AUTOMATIQUE 308 CC 2.0 HDI136 F CC 2.0 hdi 16v Feline 136cv auto fap 308 CC 2.0 hdi 16v Feline 136cv auto fap	308 cc
Peugeot	308	SW 1.2 PureTech 130 PHEV 225ch GT e-EAT8 3 <sup>a</sup> serie Hybrid 180 e-EAT8 SW GT SW 3 <sup>a</sup> serie Hybrid 180 e-EAT8 GT SW 1.2 PureTech 130 308 Hybrid 180 e-EAT8 SW GT	308 sw

standardized model names extracted using the developed regular expressions.

This meticulous approach, involving manual crafting of regular expressions, was replicated for 48 makes that are representative of the Italian car market. Figure 2.8 illustrates the number of original models for each make and the redefined models in the entire dataset.

### 2.4.3 Dataset Characteristics

The dataset under consideration offers a comprehensive collection of car advertisements, providing a rich source of information for various analytical tasks.

**Volume of Data:** The dataset encompasses a total of 12.7 million advertisements, reflecting its expansive nature and potential for diverse analyses.

**Makes Distribution:** The dataset narrows its focus to 48 distinct car makes, ensuring a targeted approach to the most relevant and significant brands in the market. The distribution of these makes in terms of their frequencies can be visualized in Figure 2.9.

**Make-Model Combinations:** Further granularity is achieved by concentrating on 1,199 specific combinations of makes and models. Given the vast number of these combinations, a detailed representation is challenging. However, an ordered frequency distribution of these combinations is depicted in Figure 2.10. For a more detailed insight, the top 25 make-model combinations, based on their prevalence in the dataset, are shown in the Table 2.2.

**Registration Dates:** The dataset spans a temporal range starting from the year 2010, encapsulating the progression and shifts in car models and preferences over the years. A granular distribution, grouped by year and month, showcasing the

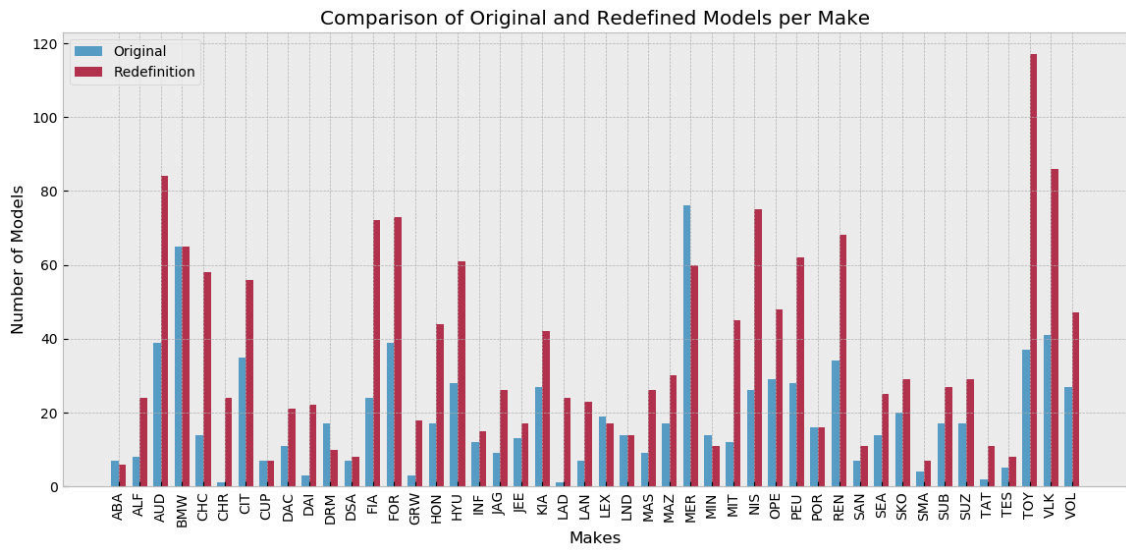


Figure 2.8: Comparison of original and redefined model counts for the dataset. Make codes translations may be found in Appendix 7

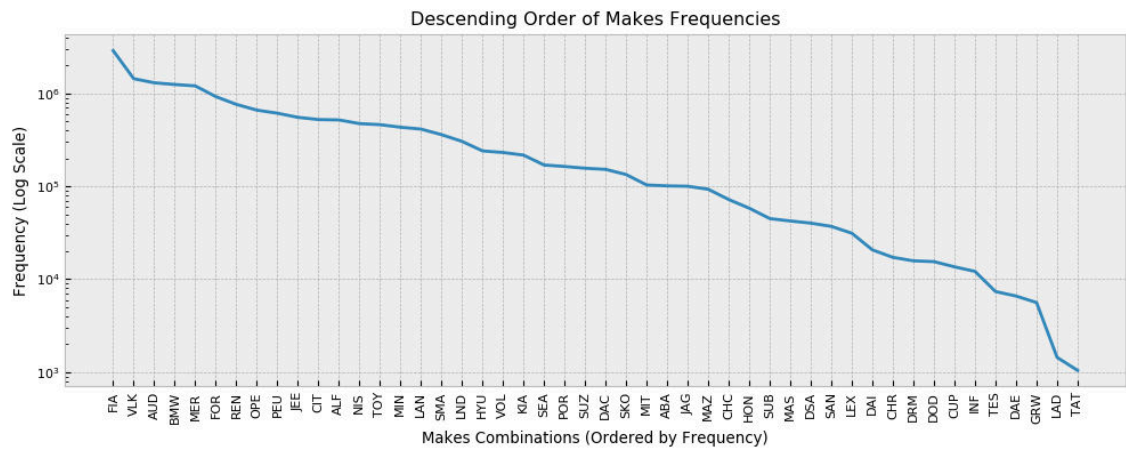


Figure 2.9: Plot visualizing the 48 makes ordered by their frequency in logarithmic scale

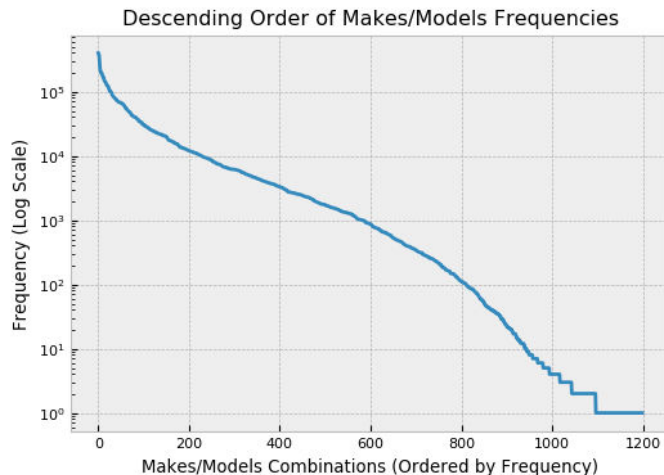


Figure 2.10: Plot visualizing the 1,199 makes/models combinations ordered by their frequency in logarithmic scale. This representation highlights the imbalance between classes.

Make	Model	Number of items in the dataset	Relative number of items (%)
FIA	panda	397,698	3.1214
FIA	500	383,562	3.0105
VLK	golf 5p	361,407	2.8366
JEE	renegade	275,482	2.1622
FIA	500l	224,606	1.7629
LAN	ypsilon	212,248	1.6659
FIA	500x	194,999	1.5305
AUD	a3 spb	193,622	1.5197
NIS	qashqai	193,307	1.5172
BMW	serie 1 5p	181,457	1.4242
SMA	fortwo	169,604	1.3312
MER	classe a 5p	168,077	1.3192
VLK	polo 5p	167,226	1.3125
AUD	a4 avant	151,888	1.1921
FOR	fiesta 5p	144,822	1.1367
MIN	countryman	143,340	1.1250
FIA	punto 5p	141,597	1.1114
BMW	serie 3 touring	136,248	1.0694
REN	clio 5p	132,515	1.0401
CIT	c3	124,599	0.9779

Table 2.2: Frequencies and relative percentage of top-20 combinations per Make/Model in the dataset.

frequency of car registrations and indicating the cars' age is depicted in Figure 2.11.

#### 2.4.4 Annotation Validation

The validation process is crucial to ensure the reliability and accuracy of the annotations derived from the automated system. Given the continuous growth of the dataset and the need for an automated system, it is imperative to periodically validate the results to maintain the integrity of the data.

**Validation Set Creation:** To ensure a robust validation process, a validation set was meticulously curated. From the entire dataset, which comprises over 14 millions entries, a subset (approximately 20,000 entries) was randomly selected to form the validation set. This subset size was chosen to provide a statistically significant

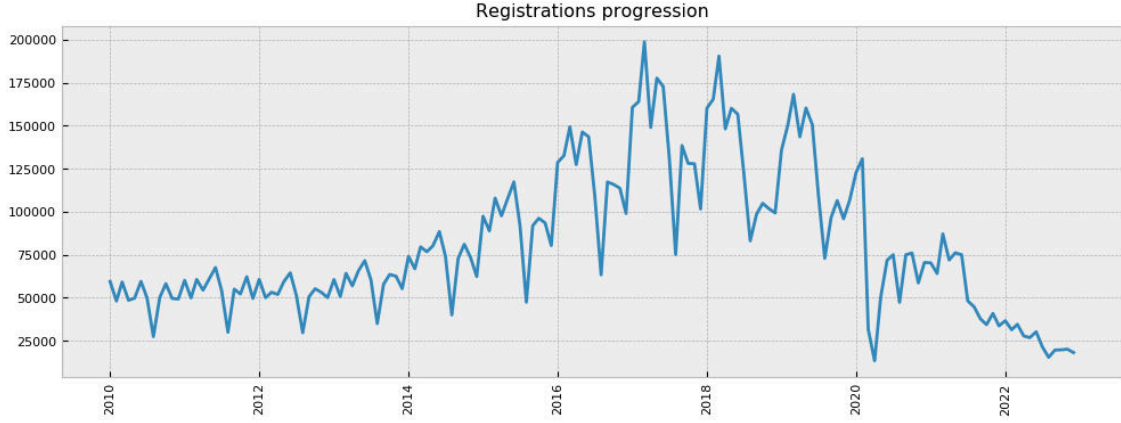


Figure 2.11: Distribution of car registrations grouped by year and month starting from 2010. Each data point represents the total number of cars registered within that specific month and year. Notably, an evident pattern of seasonality can be observed in the data, unintentionally reflecting the nature of the European car market throughout the years.

sample while remaining manageable for manual annotation.

To ensure that the validation set is representative of the entire dataset, stratified sampling was employed. This means that the validation set maintains the same distribution of car makes and models as the overall dataset. For instance, if 10% of the entire dataset consists of “Fiat” cars, then approximately 10% of the validation set will also be “Fiat” cars. This stratification ensures that the validation process is not biased towards any particular make or model and provides a holistic view of the annotation system’s performance across all categories.

Once the validation set was formed, it was then manually annotated by a team of domain experts. These experts meticulously went through each entry, ensuring that the make, model, and other attributes were correctly identified. This manually annotated validation set serves as the “gold standard” against which the automated annotations were compared, providing a benchmark for assessing the accuracy and reliability of the automated system.

**Metrics:** To quantify the performance of the automated annotation system, accuracy was chosen as the primary metric. Accuracy measures the proportion of correctly identified annotations to the total annotations. Additionally, the F1-score, which considers both precision (the number of correct positive results divided by the number of all positive results) and recall (the number of correct positive results divided by the number of positive results that should have been returned), was used to provide a more holistic view of the system’s performance.

Metric	Value on Validation Set
Accuracy	96.5%
F1-Score	95.8%

Table 2.3: Performance metrics of the automated annotation system on the validation set.

**Manual Verification Results:** Upon manual verification of the validation set, it was observed that the automated system achieved an accuracy of 96.5%,

indicating a high level of reliability in the annotations. The results can be found in Table 2.3. The few discrepancies that were identified were primarily due to rare naming conventions or exceptionally uncommon abbreviations that were not initially accounted for in the regular expressions. These findings were subsequently used to refine and enhance the system further.

### 2.4.5 Discussion

**Advantages:** The developed methodology for dataset collection and annotation offers several advantages:

- *Increased Accuracy:* By standardizing annotations based on a well-defined glossary and using regular expressions, the accuracy of annotations is significantly enhanced.
- *Scalability:* The approach can be easily scaled to handle larger datasets without compromising on annotation quality.
- *Versatility:* The comprehensive set of regular expressions ensures a wide range of errors and variations can be addressed.

**Limitations:**

- *Maintenance Overhead:* The glossaries encompassing vehicle makes and models necessitate periodic updates to remain current and comprehensive. This continual updating process requires a rigorous collaboration with automotive experts to ensure the inclusion of emerging car models and modifications.
- *Regular Expression Complexity:* Introducing new regular expressions to accommodate novel scenarios or data variations poses a challenge. Alterations or additions to the regular expressions might inadvertently broaden the scope of existing rules. Such broadening could lead to the reclassification of data instances that were previously recognized under different criteria. This dynamic nature demands a meticulous validation process each time the regular expression set undergoes modifications.

Future endeavors will focus on refining the annotation process. Potential avenues include integrating machine learning models for enhanced annotation accuracy, expanding the glossary to encompass newer vehicle models, and collaborating with domain experts for further validation and refinement.

# Chapter 3

## Vehicle Identification and Classification

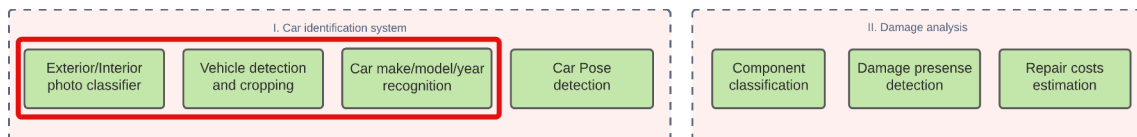


Figure 3.1: Highlighted segment of the whole system representing the focus of this chapter.

The objective of assessing car damages from photographs inherently requires a clear and definitive understanding of the vehicles in the images. To provide a comprehensive overview of the system’s structure and to position the focus of this chapter within the broader context, Figure 3.1 depicts the modular breakdown of the entire damage recognition system. This primary step, termed as *Vehicle Identification*, is foundational to the subsequent tasks of pose detection and damage analysis. Ensuring that the system accurately identifies the vehicle sets the stage for more intricate and detailed assessments later in the process.

Within the realm of Vehicle Identification, this chapter elucidates three crucial modules. First, the problem of **Photograph Type Classification** is addressed, which differentiates between exterior and interior photographs of vehicles. This classification ensures that the system is aware of the context of the image, thereby refining the subsequent analysis and assessments.

The subsequent section introduces the module responsible for **Vehicle Localization on Photographs**. Given a photograph, it is essential to pinpoint the vehicle’s position, which in turn facilitates both the pose and damage assessments. Correct and efficient vehicle localization is paramount as it eliminates potential background noise and emphasizes the subject of interest.

Lastly, the chapter delves into the **Vehicle Make/Model/Year Classification**. By ascertaining the make, model, and year of the vehicle, invaluable insights are garnered into potential structural nuances, typical vulnerabilities, and other vehicle-specific attributes that might influence damage patterns and repair costs.

These three modules collectively shape the first phase of the proposed approach to car damage assessment. They serve as the groundwork, ensuring that the system is equipped with all necessary vehicle-specific details before proceeding to the subsequent chapters on pose detection and damage assessment.

## 3.1 Photograph Type Classification: Exterior vs. Interior

### 3.1.1 Introduction and Related Works

In the process of assessing car damages from photographs, the nature of the photograph plays a pivotal role. While external damages are the primary focus of this thesis, it is imperative to segregate exterior photographs from the interior ones. The presence of interior photographs can introduce noise into the subsequent damage recognition phases, making it essential to have an automated system that can accurately differentiate between the two.

### Transfer Learning with DCNNs: A Robust Approach for Scene Classification

Deep Convolutional Neural Networks (DCNNs) have proven their prowess in a multitude of image classification tasks, especially when compared to traditional methods based on handcrafted or shallow learning-based features. According to [32], DCNNs, when pre-trained on vast datasets like ImageNet, can serve as effective universal feature extractors. They demonstrated the effectiveness of these networks by either using them directly for feature extraction or fine-tuning them for scene classification tasks on specific datasets, with both approaches yielding promising results. Such findings show that transfer learning, using models such as AlexNet, VGGNet, and GoogleNet [33], is beneficial for scene classification, especially when dealing with high-resolution images.

[34] built upon these principles, focusing on the challenges presented by complex context relationships and varying object scales in high-resolution images. They proposed an adaptive deep CNN-based method that recalibrates feature channels for better context understanding, ultimately enhancing spatial representation. The proposed method showcased its capability to extract high-level category features for scene classification when evaluated against other state-of-the-art CNN models .

### Indoor vs. Outdoor Scene Classification

Considering the absence of dedicated research on vehicle interior vs. exterior photograph classification, it is instructive to examine a parallel and somewhat analogous domain: indoor vs. outdoor scene classification. The rationale behind this is that the techniques and methodologies applied in discerning between indoor and outdoor scenes might offer insights, albeit indirectly, into classifying vehicle photographs.

Indoor scene classification, according to [35], is notably more challenging due to its inherent unpredictability. While various methods have been developed over the years, the accuracy remained a perennial issue. This paper suggests that DCNNs, especially models like VGG-19, offer a reliable solution as they can automatically filter features without compromising on performance. Furthermore, transfer learning using these pre-trained models, as corroborated by their experiments on datasets like SUN397 [36] and Places365 [37], has emerged as an efficient approach. Not only does transfer learning enable models to produce compelling results with limited datasets, but it also prevents the tedious process of building a CNN from scratch.



In a different domain, an attempt [38] was made to address the challenges of video scene classification with intricate backgrounds. An enhanced CNN model was proposed for video scene classification in mining environments, indicating the adaptability and potential of CNN models in handling diverse scenarios. This work emphasized the network’s structure, where the majority of the layers were dedicated to feature extraction, subsequently facilitating precise classification using the Softmax loss function.

### **Lack of Dedicated Datasets for Photograph Type Classification**

One of the challenges in the specific task of vehicle photograph type classification (exterior vs. interior) is the absence of dedicated datasets. While general scene classification problems have databases like SUN397 [36] and Places365 [37], specific datasets catering to the nuances of vehicle photographs are scant. This has necessitated the creation of proprietary datasets for more targeted training and evaluation.

The literature reinforces the potential of using transfer learning with deep convolutional neural networks for scene classification tasks. Even though there might not be direct references to vehicle interior/exterior classification, the task aligns closely with indoor/outdoor image classification. Additionally, the need to curate specialized datasets for this task underscores its novelty and the significance of the contributions made in this study.

### **3.1.2 Dataset Creation**

For an effective photograph type classification, a specialized dataset tailored to the unique requirements of the task is indispensable. Given the unavailability of pre-existing datasets specific to vehicle interior versus exterior photo classification, the need arose to curate a custom dataset. This curated approach was essential to ensure both the precision and the pertinence of the data to the classification challenge at hand.

#### **Data Collection**

The images used to construct this dataset were sourced from publicly available content on the internet, focusing predominantly on the Italian car market. This approach was adopted to ensure the images mirror the variety of makes and models endemic to the Italian market. To capture a holistic representation, efforts were made to incorporate images from a diverse range of scenarios – including varying poses, backgrounds, lighting conditions, and other defining characteristics.

#### **Class Definitions and Labeling**

Three distinct classes were defined for the purpose of the classification task:

- **Exterior:** Images predominantly showcasing the outside of a vehicle.
- **Interior:** Pictures detailing the vehicle’s interior.
- **Reject class:** Reserved for images that do not fit the primary classes, but might still be tangentially related to the automotive domain. The criteria for images falling into this class are:

- No vehicle is present.
- Multiple vehicles are present (more than one).
- A vehicle is present, but it is rendered or not realistic.
- Vehicle parts are present, but they are unrecognizable.

The images in the “Reject class” were not selected randomly, but rather were chosen based on the aforementioned criteria to ensure that while they do not belong to the primary classes of “Exterior” and “Interior”, they still remain loosely relevant to the domain of automotive imagery. Each image within the dataset was labeled manually to ensure the utmost precision in classification.

**Dataset Composition** The resulting dataset comprises a total of 3123 images. The class distribution is as follows:

- **Exterior:** 1375 images, (44% of the dataset).
- **Interior:** 1284 images, (41% of the dataset).
- **Reject class:** 464 images, (remaining 15% of the dataset).

To offer a tangible representation, Figure 3.2 provides select examples from each class. It is evident from the figure that the images span a comprehensive array of scenarios, reflecting the dataset’s richness and diversity.

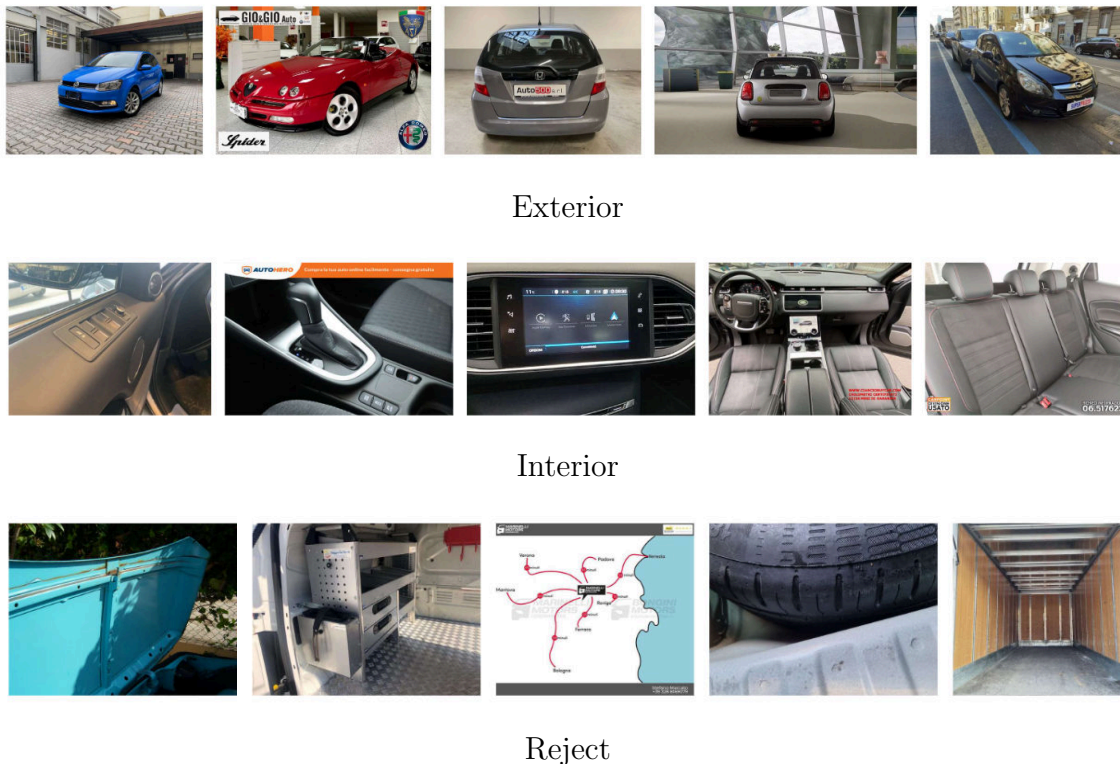


Figure 3.2: Sample images from the dataset, categorized into their respective classes.

### 3.1.3 Transfer Learning and Model Selection

Transfer learning, as evident from the review of related works, emerges as a preferred method in scene classification, and more importantly, in cases where there is limited data for a particular task. The main advantage of using transfer learning is to harness the knowledge obtained from training on a large dataset, like ImageNet, and apply this knowledge to a novel task.

Given that the classification task involves distinguishing between three classes—exterior, interior, and reject—a suitable model must be selected. A fundamental criterion for the system is its lightweight nature, thereby guaranteeing efficient performance even under resource constraints. To this end, the subsequent architectures, renowned for their compactness coupled with notable accuracy, were examined:

- MobileNetV2 [39]
- EfficientNetB0 [40]
- ResNet-18 [41]
- ShuffleNet [42]

A summary of their performance on the ImageNet classification task and complexity is presented in Table 3.1.

Table 3.1: Performance and complexity of considered architectures.

Name	Top1 Accuracy on ImageNet (%)	Number of Parameters
MobileNetV2	72.0	3.5M
EfficientNetB0	76.3	5.3M
ResNet-18	69.3	11.4M
ShuffleNet	71.5	3.4M

Each of the aforementioned architectures was pretrained on ImageNet, providing a solid foundation for feature extraction. For this specific task, the final pooling layer of these architectures is used as an image descriptor, which is then fed into a dense layer consisting of three neurons corresponding to the classes. For the training of the model, the loss function employed is categorical cross entropy.

**Evaluation Method** To gauge the efficacy of the selected model, the performance metrics chosen are accuracy and macro-averaged F1 score. Accuracy provides a direct proportion of correct predictions over the total predictions. The macro-averaged F1 score offers a balance between macro-average precision and macro-average recall, making it particularly valuable in multi-class scenarios. These metrics can be expressed as:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

$$\text{Macro-Averaged F1 Score} = \frac{2 \times \text{Macro-Average Precision} \times \text{Macro-Average Recall}}{\text{Macro-Average Precision} + \text{Macro-Average Recall}}$$

Where:

$$\text{Macro-Average Precision} = \frac{1}{N} \sum_{i=1}^N \frac{\text{True Positives}_i}{\text{True Positives}_i + \text{False Positives}_i}$$

and

$$\text{Macro-Average Recall} = \frac{1}{N} \sum_{i=1}^N \frac{\text{True Positives}_i}{\text{True Positives}_i + \text{False Negatives}_i}$$

Here,  $N$  represents the number of classes.

These evaluation methods will not only help in comparing the performances between the selected architectures but also ensure the chosen model meets the desired system requirements.

## Model Training

**Data Splitting** The dataset was divided into training and testing subsets using a 75/25% split. Ensuring a consistent representation of the classes in both sets, stratification was applied based on the class distribution. This method ensures that the training and testing sets have a similar distribution of classes, preventing any potential biases during model evaluation.

**Optimization and Learning Rate** For the training process, the Adam optimizer [43] was used, renowned for its efficiency and effective handling of sparse gradients. The initial learning rate was set to  $1 \times 10^{-3}$ , with a decay factor applied every epoch, reducing the learning rate by 0.1% of its previous value. This adaptive approach enables the model to converge faster initially, then refine its weights as it gets closer to the optimal solution.

**Early Stopping** To prevent overfitting and to save computational time and resources, an early stopping mechanism was implemented. The training process monitors the validation loss and halts if no significant improvement is noticed for five consecutive epochs. This technique ensures that the model does not over-optimize on the training data and retains its ability to generalize to unseen examples.

**Data Augmentation** In order to enhance the diversity of the training data and help the model generalize better, simple data augmentation techniques were applied. These techniques include random image rotation (within a range of  $-10^\circ$  to  $10^\circ$ ), horizontal flipping, and slight distortions. These transformations expand the variety of data the model encounters during training, aiding in its robustness against potential real-world variations.

**Training Sessions** Upon initiating the training process with the aforementioned configurations and leveraging the Nvidia Tesla T4 GPU, the model was subjected to multiple training epochs, with performance on the validation set gauged at the end of each epoch to ensure consistent improvement and to apply early stopping if necessary.

### 3.1.4 Results

#### Quantitative Results

The evaluation results for the task of photograph type classification across various architectures are presented in Table 3.2. As the data suggests, the MobileNetV2 model achieves a strong balance between model accuracy and computational complexity, with an accuracy of 0.92 and a Macro-Averaged F1 Score of 0.89. Despite EfficientNetB0 achieving slightly superior metrics, its increased parameter count signifies a higher computational demand. ResNet-18, with the highest parameter count, does not provide a justifiable increase in performance relative to its complexity. On the other hand, ShuffleNet, with a close parameter count of 3.4 million compared to MobileNetV2’s 3.5 million, has a slightly lower accuracy, achieving 0.715 compared to MobileNetV2’s 0.720.

Model	No. of Parameters	Accuracy	Macro-Averaged F1 Score
MobileNetV2	3.5M	0.92	0.89
EfficientNetB0	5.3M	0.94	0.91
ResNet-18	11.4M	0.88	0.82
ShuffleNet	3.4M	0.91	0.89

Table 3.2: Performance of various architectures for the task of photograph type classification.

In light of these results, the MobileNetV2 architecture was selected for further deployment and analysis, being the optimal combination of performance and computational complexity. For a more granular understanding of this model’s performance on individual classes, a detailed classification report (Table 3.3) and confusion matrix (Figure 3.3) related to the MobileNetV2 architecture are provided in the subsequent figures.

Classes (support)	Precision	Recall	F1-score
Exterior (140)	0.96	0.96	0.96
Interior (117)	0.86	0.97	0.91
Reject (55)	0.95	0.71	0.81

Table 3.3: Classification report for the MobileNetV2 architecture

#### Qualitative Results

For a comprehensive understanding of the model’s performance, it is essential to not just rely on quantitative metrics but also visually evaluate the predictions on the validation set. Figures 3.4 and 3.5 provide such a qualitative assessment.

In Figure 3.4, a selection of images from the validation set is presented that were correctly classified by the model. Each row represents one of the classes—Exterior, Interior, and Reject. It is evident that the model can reliably identify clear and distinguishable instances of each class.

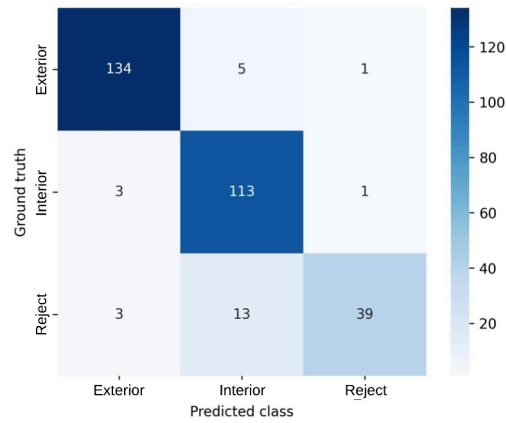


Figure 3.3: Confusion matrix for the MobileNetV2 architecture.

However, there are instances where the model fails to classify correctly, as depicted in Figure 3.5. Upon closer inspection of the misclassified images, some of them appear to be borderline in terms of content, making the classification task even challenging for human perception, particularly for the Reject class. Such borderline cases shed light on potential areas where model robustness could be improved, perhaps by curating a more discerning training set or refining the training process to handle such ambiguous instances more adeptly.

These qualitative results, in tandem with the quantitative metrics, offer a holistic view of the model’s capabilities and limitations, laying the groundwork for potential future refinements.



Figure 3.4: Sample validation set images demonstrating correct classifications. The top row represents images correctly classified as *Exterior*, the middle row as *Interior*, and the bottom row as *Reject*.



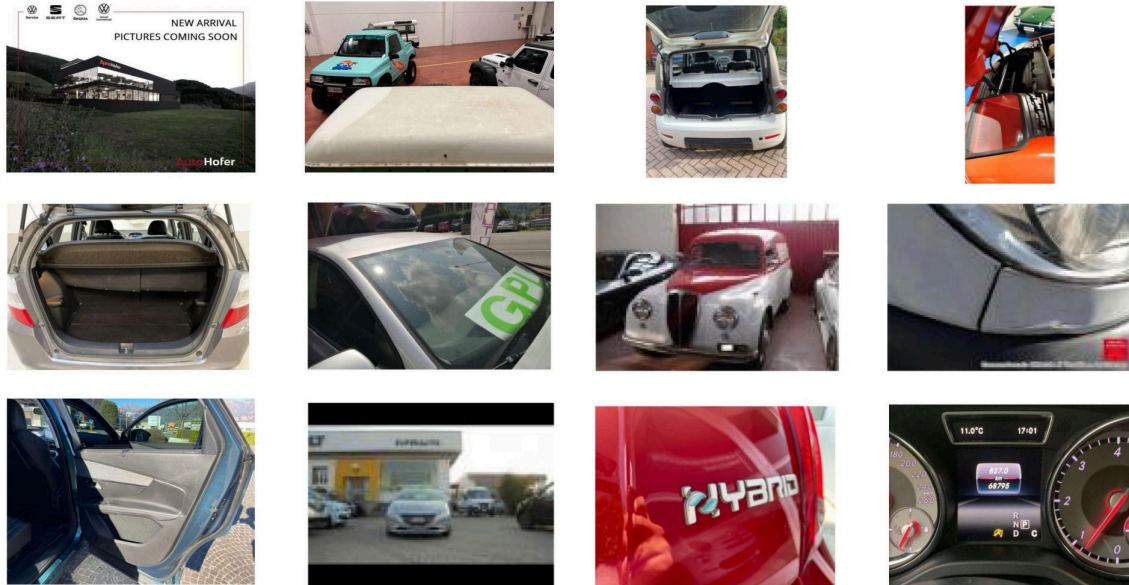


Figure 3.5: Sample validation set images that were misclassified by the model. The top row shows images mistakenly classified as *Exterior*, the middle row as *Interior*, and the bottom row as *Reject*. Some images, particularly in the *Reject* category, present borderline content that poses challenges even for human evaluators.

## Conclusion

The pursuit of an effective system for classifying vehicle photographs into exterior, interior, or reject categories required diligent efforts, spanning data creation to model selection. A customized dataset, indicative of the Italian car market, served as the foundation for this task. By harnessing transfer learning, the advantages of expansive datasets were utilized, leading to high performance even with limited custom data.

Among the various architectures evaluated, MobileNetV2 emerged as a balanced choice, combining classification accuracy with a lightweight design, making it suitable for deployments in resource-limited settings. Comprehensive metrics highlight the model’s ability to generalize across distinct classes.

This endeavor emphasizes the significance of careful dataset creation, strategic model selection, and the potential of compact architectures in specialized classification tasks. The adopted MobileNetV2 model, supported by empirical results, stands to improve vehicle photograph categorization, setting the stage for further advancements in this area.

## 3.2 Vehicle Localization on Photographs

With the rapid evolution of computer vision, several applications, from autonomous driving to vehicle classification, demand the precise localization of vehicles within images. The task of vehicle localization is fundamentally about determining the spatial presence of vehicles, typically represented as bounding boxes or regions of interest within a given photograph. By zeroing in on the vehicle, these systems can drastically reduce noise, enhance the focus, and thus improve the accuracy of

subsequent analyses.

### 3.2.1 Literature Review

The task of object localization in images has been a central concern in the domain of computer vision for several decades. Initial endeavors in this space often deployed traditional image processing techniques, with the primary objective being the detection and delineation of specific objects in a given image. The progression of methods in this domain can be broadly categorized into traditional image processing techniques, machine learning-based methods, and deep learning approaches.

Early solutions largely revolved around edge detection, contour-based methods, and background subtraction to identify objects [44]. Histogram of Oriented Gradients (HOG) combined with Support Vector Machines (SVM) was a popular method for detecting objects, particularly for pedestrian detection [45]. These methods, although effective in controlled environments, struggled in real-world scenarios with varied lighting, occlusions, and cluttered backgrounds.

With the advancement of computer vision techniques, especially in the 2000s, feature engineering became the centerpiece of object detection. Methods such as Scale-Invariant Feature Transform (SIFT) [46] and Speeded-Up Robust Features (SURF) [47] gained prominence. However, it is essential to note that these methods, particularly when aiming to detect generic cars, were primarily effective when employed in the “bag of words” variant. When combined with classifiers like SVM, they led to improved object localization performance over traditional techniques, particularly in the domain of vehicle detection.

The watershed moment in object and vehicle localization came with the advent of Convolutional Neural Networks (CNNs) [48]. With the capability to learn hierarchies of features directly from raw data, CNNs began to outperform the manually crafted feature-based methods.

RCNN [49] was among the pioneering models that combined the power of CNNs with region proposals for object detection. However, it was Faster R-CNN [50] that drastically improved efficiency by introducing the Region Proposal Network (RPN), making real-time object detection feasible. Parallely, models like YOLO (You Only Look Once) [9] and SSD (Single Shot MultiBox Detector) [51] proposed different paradigms for object detection, with YOLO focusing on predicting bounding boxes and class probabilities in a single forward pass of the network.

Vehicle localization, a subset of the broader object detection domain, has seen a multitude of task-specific applications tailored to the unique requirements of the environment and the demands of the problem at hand. Specific to vehicle localization, several benchmarks and challenges, such as the PASCAL VOC Challenge [52] and the COCO dataset [10], have facilitated the evolution of robust models. In the realm of autonomous driving, datasets like KITTI [53] have been instrumental in advancing vehicle localization techniques. These benchmarks often include a “car” or “vehicle” class, prompting the development and fine-tuning of models specifically for vehicular detection in various scenarios, from urban settings to highways.

### Surveillance Cameras

Vehicle detection from surveillance cameras is of paramount importance in urban planning, traffic management, and security applications. This involves identifying



vehicles from a top-down or off-angle perspective, often in high-density traffic scenarios or parking lots. Challenges include dealing with occlusions, varying light conditions, and static obstacles [54].

### **Autonomous Vehicles**

For autonomous vehicles, front-facing cameras primarily focus on detecting and localizing vehicles to navigate and make driving decisions. The images in this scenario are captured from an eye-level, car-front perspective, and the chief challenges are to deal with dynamic lighting, weather conditions, and fast-moving objects [55].

### **3D Detection**

3D vehicle detection aims to predict the 3D pose and shape of vehicles from 2D images. While providing a richer representation, it is inherently more complex than 2D bounding box detection. For tasks that require only 2D localization, using 3D detection models might introduce unnecessary complexity, often at the cost of real-time performance or accuracy on the simpler task [56].

### **Keypoints Detection**

Another related task is the detection of specific keypoints on vehicles, like headlights, tail lights, or license plates. Though providing a detailed understanding of vehicle orientation and parts, it may exceed the requirements of applications just needing basic localization. Moreover, such fine-grained detection tasks may also be computationally more intensive [57].

Other advanced methods, which integrate multi-sensor fusion involving RGB cameras, LiDAR, radar, and even thermal cameras, have been proposed for enhanced vehicle detection, especially in autonomous driving scenarios [58]. These methods benefit from the complementary information provided by different sensors, improving detection accuracy especially in challenging conditions. However, for scenarios where only 2D images are available, as in current research, these multi-sensor methodologies are not applicable.

The variety of tasks underscores the flexibility and adaptability of vehicle localization methods, allowing researchers to choose or design solutions best suited to their specific requirements.

## **3.2.2 Need for Vehicle Localization in The System**

Selecting an appropriate architecture for vehicle localization, given the constraints and the application’s specific requirements, is a nuanced process. The balance between performance, in terms of accuracy, and computational complexity is critical.

The architectures under consideration spanned a range of backbone architectures combined with object detection methodologies. These included the widely adopted Faster R-CNN [50], FCOS [59], RetinaNet [60], and SSD [51] architectures. While all these models are reputable in their own right, they offer different trade-offs in terms of accuracy, speed, and computational demands.

The architectures were benchmarked in terms of their operations (GFLOPS) which essentially provide an insight into the computational demands of the model

and the Mean Average Precision (Box mAP). It is a widely utilized metric for evaluating the performance of object detection models. Specifically, for bounding box detection tasks, it quantifies the model’s accuracy across all threshold levels, giving an aggregate view of the model’s detection capability. The Box mAP is computed by first calculating the precision-recall curve for the model’s detections. Precision (P) is the ratio of correct positive predictions to the total predicted positives, and recall (R) is the ratio of correct positive predictions to the total actual positives.

Given a set of precision and recall values, Average Precision (AP) is computed as:

$$AP = \sum_n (R_n - R_{n-1})P_n$$

where  $R_n$  and  $P_n$  are the recall and precision at the  $n$ th threshold. The mAP is then the mean of the AP values computed for each class in the dataset. Given a dataset with  $C$  classes, and letting  $AP_i$  denote the average precision for the  $i$ -th class, the mAP is given by:

$$\text{mAP} = \frac{1}{C} \sum_{i=1}^C AP_i$$

Model	Operations (GFLOPS)	Box mAP
FasterRCNN ResNet50 FPN	134	37.0
FasterRCNN ResNet50 FPNv2	280	46.7
FasterRCNN MobileNetV3Large FPN	4.5	32.8
FasterRCNN MobileNetV3Large 320FPN	0.7	22.8
FCOS ResNet50 FPN	128	39.2
RetinaNet ResNet50 FPNV2	152	41.5
SSD300 VGG16	35	25.1
SSDLite320 MobileNetV3Large	0.6	21.3

Table 3.4: Comparison of various architectures for the localization.

### 3.2.3 Model Selection and Qualitative Evaluation

Considering the results presented in Table 3.4, the FasterRCNN MobileNetV3Large FPN architecture emerged as a compelling choice for vehicle detection. With 4.5M GFLOPS, it presents a good balance between computational cost and accuracy, achieving a box mAP of 32.8. This makes it relatively efficient for processing a vast dataset on limited computational resources, without significantly compromising the accuracy of vehicle localization. Other architectures, though presenting better accuracies, also come with significantly higher computational demands, making them less suited for the application’s constraints.

Upon a hands-on evaluation on a subset of the dataset described previously in this chapter, the chosen architecture visually demonstrated competent vehicle localization capabilities. As presented in Figure 3.6, while the detections were not immaculate, they were sufficiently accurate to meet the objectives of the research. The visual consistency across the detections, even if not perfect, substantiates the model’s potential to generalize well across the dataset. Such observations further strengthen the decision to select this particular architecture.

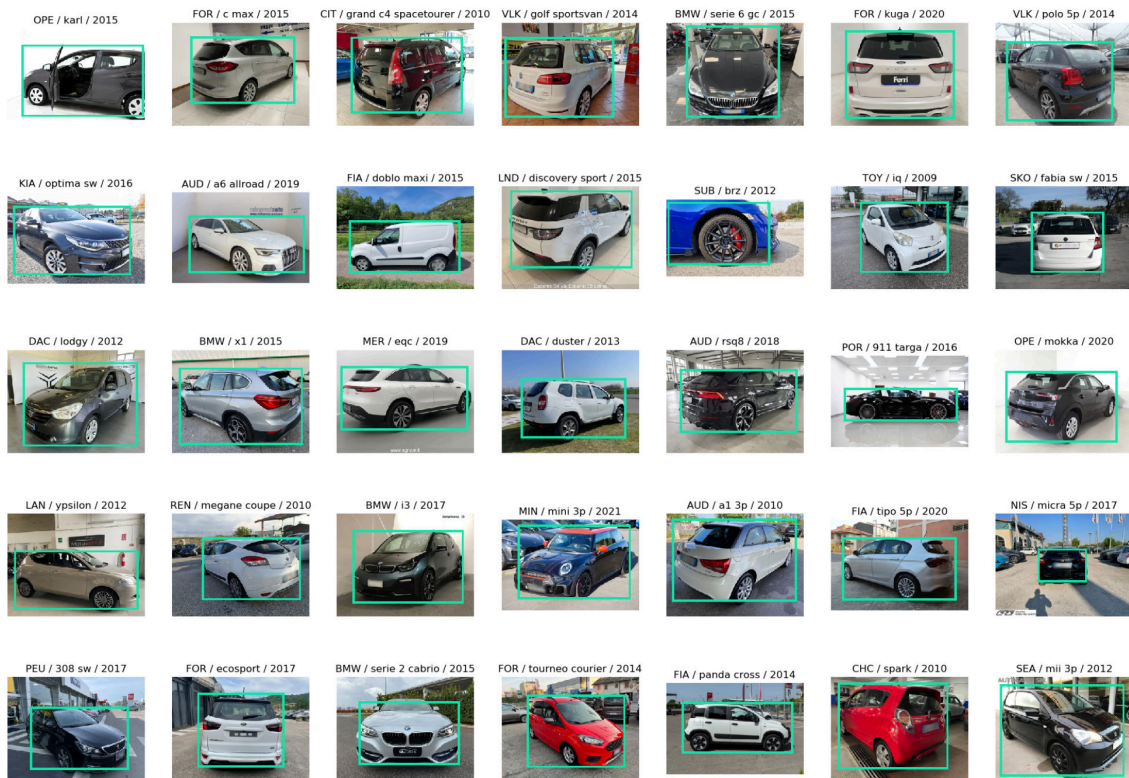


Figure 3.6: Illustrative examples of vehicle localization results using the Faster-RCNN MobileNetV3Large FPN architecture on a subset of the main dataset. The bounding boxes depict the detected vehicle regions, showcasing the model’s ability to efficiently identify and highlight vehicles across varied backgrounds and orientations.

## 3.3 Vehicle Make/Model/Year Recognition

### 3.3.1 Introduction

In the intricate landscape of vehicle recognition, the granularity with which a vehicle can be identified plays a pivotal role. While broader categorizations such as vehicle type have their applications, the detailed classification encompassing make, model, and the production year of a vehicle holds significant implications in various domains.

The primary objective of this section is to elucidate the methodological approach undertaken for this granular classification task. Leveraging a specialized dataset, as delineated in Chapter 2 of this thesis, the approach adopted is comprehensive, factoring in the unique characteristics and nuances of car makes and models, especially those that have undergone restylings or facelifts over the years.

Moreover, the intricacies of restylings, which involve subtle yet impactful visual changes while retaining the commercial name, pose a unique challenge. These restylings not only influence the visual aesthetics of vehicles but also have economic implications, impacting market prices and consumer preferences.

In the subsequent sections, the detailed approach, spanning from dataset utilization to classifier design and implementation, will be presented. The emphasis will be on elucidating the two-step classification strategy, which encompasses an initial make and model classification followed by a detailed restyling recognition.

### 3.3.2 Literature Review

The problem of vehicle recognition, particularly at a granular level involving car makes, models, and years, has been a prevalent research topic in recent years. The growth of intelligent transportation systems and the increasing demands for efficient traffic management have necessitated advancements in this field [61]. The following is a review of notable works in this domain.

Deep learning, particularly convolutional neural networks (CNNs), has been the cornerstone of most recent advancements in vehicle recognition. [62] illustrates the application of CNNs for vehicle type recognition using surveillance images. Their approach, however, tackles a significant challenge - the assumption that training and test data come from similar imaging systems. By leveraging transfer learning, they utilize labels from web data to train their system, sidestepping the need for manual annotations from surveillance images. This adaptation of transfer learning, especially from web-natured data to surveillance data, showcases its potential, resonating with the approach adopted in the current work.

[63] emphasizes the capability of deep convolutional neural networks (DCNNs) in recognizing fine-grained car details. Their work introduces a spatially weighted pooling (SWP) strategy to enhance the robustness of DCNNs. The SWP layer improves feature representation by weighing spatial units based on their discriminative power, which is particularly crucial for detailed tasks like make/model/year recognition.

While traditional vehicle recognition methods may focus broadly on vehicle types, the granular classification involving makes, models, and years adds complexity to the task. [61] highlights the challenges posed by the sheer number of car models and their similarity in appearance. Their solution involves a multi-path DCNN model that captures both holistic and part-based vehicle information. Emphasizing

the importance of car fronts, they underscore the value of leveraging different vehicle parts for improved recognition.

[64] further expands on the importance of additional information beyond just the vehicle image. By incorporating 3D bounding boxes and vehicle orientation data, their CNN model achieves significant performance boosts. Their approach emphasizes that recognizing subtle details, critical for make/model/year classification, requires auxiliary information beyond the primary vehicle image.

For many applications, including traffic surveillance and real-time monitoring, efficiency is paramount. [65] focuses on make and model recognition (MMR) using an optimized SqueezeNet [66] architecture. Not only does their model achieve a high recognition rate, but its compact nature makes it viable for real-time applications. This underscores the need for a balance between accuracy and efficiency, particularly in real-world scenarios.

Due to varying traffic camera configurations, vehicle images can differ widely in terms of viewpoints, lighting conditions, resolution, and color depth. [67] proposes a framework that first detects cars using a part-based detector. Cardinal anchor points, such as license plates and headlamps, are then utilized to rectify projective distortion, emphasizing the importance of adapting to the inherent variability in traffic images.

[68] also addresses the challenges posed by varying car appearances in images. Their unique approach involves predicting a confidence score for each detected bounding box, illustrating the importance of accurate vehicle localization in classification tasks.

In understanding the broader landscape, [69] offers a comprehensive categorization of automated vehicle classification studies. Their categorization based on granularity — from vehicle type to make and model — offers valuable insights into the challenges and strategies at each level, providing a holistic view of the research domain.

The reviewed literature underscores the dynamic nature of vehicle recognition tasks, particularly the increasing emphasis on granular classifications involving make, model, and year. Deep learning, especially convolutional neural networks, remains pivotal in advancing this domain. Among the methodologies explored, transfer learning emerges as a particularly potent strategy. As illustrated by [62], harnessing pre-trained models on diverse data sources, such as web-natured images, and adapting them to specific tasks, like surveillance-based vehicle recognition, can mitigate challenges associated with data annotation and domain discrepancies. This adaptation of transfer learning, bridging web-derived data with surveillance imagery, showcases the versatility and potential of such approaches. As vehicle recognition tasks continue to evolve, the integration of transfer learning and fine-grained classification methodologies will likely remain central to future advancements.

### 3.3.3 Dataset Utilization

The dataset selected for the task of car make/model/year classification holds a pivotal role in ensuring the robustness and accuracy of the classifiers. Drawing from Chapter 2, this dataset possesses several distinctive advantages that set it apart from others:

1. **Expert Domain Knowledge Annotation:** Unlike many datasets available,

this dataset was annotated with the expertise of domain specialists. Such rigorous annotation ensures that the subtle differences between car makes and models are accurately captured, laying a solid foundation for the subsequent classification tasks.

2. **Web-Nature Based on Ads:** Sourcing data from web-based advertisements ensures that the dataset mirrors the current market distribution of cars. This is in stark contrast to datasets that might include rare or prototype cars which, while interesting, do not reflect the real-world distribution and can introduce noise into the classification process.
3. **Focus on Visual Aesthetics:** The dataset prioritizes vehicles' exterior visual characteristics, which are paramount for the task at hand. By grouping together cars with different internal characteristics such as transmission or fuel type, the dataset narrows its focus to visual cues which are most pertinent to the classification objective.
4. **Inclusion of Registration Dates:** A unique feature of this dataset is the inclusion of vehicle registration dates. These dates serve a dual purpose: they not only provide an indirect measure of the vehicle's production year but also facilitate the identification and classification of restylings for the annotation phase. As restylings are closely tied to specific time frames, having accurate registration dates is invaluable.

### 3.3.4 Restyling: Definition and Importance

The automotive industry is constantly evolving, with manufacturers continually seeking to improve their vehicles while catering to changing consumer preferences. One such evolution, often subtle yet impactful, is the practice of “restyling”.

Restyling, often referred to as a “facelift” within the automotive domain, is defined as the practice wherein a carmaker maintains the commercial name of a model but introduces visual changes. These modifications can range from minor adjustments in design details to more pronounced changes in body shape, lighting configurations, or front and rear fascia designs. This practice is common among manufacturers to refresh a model's appearance without launching an entirely new model.

A visual representation of restyling can be seen in the case of the Toyota Aygo. Over the past decade, Toyota Aygo has undergone four distinct versions, each representing a different restyling phase. This can be illustrated in Figure 3.7, which showcases the subtle yet distinct visual differences among the various versions.

The importance of recognizing restylings extends beyond aesthetics. From a consumer perspective, restylings often signify improvements or updates in vehicle features. Moreover, restylings can influence the market value of a car, often impacting its resale value and even insurance premiums.

From a technical standpoint, and particularly relevant to this research, different restylings might involve the use of varied parts during production. These variations can lead to differences in repair costs due to the availability, demand, or complexity of certain parts. Recognizing these subtle differences becomes crucial when the ultimate goal is to assess damages, as the repair costs and procedures can significantly vary based on the specific restyling version of a vehicle.



Figure 3.7: Visual representation of the restylings of the Toyota Aygo model across different years. The columns, from left to right, showcase the design evolutions in 2009, 2012, 2014, and 2018. The top row presents the front view of blue cars, while the bottom row illustrates the rear-quarter view of white cars, highlighting the distinct design changes over the years.

In the context of this research, understanding and accurately classifying restylings is paramount. The granular level of classification ensures that damage recognition algorithms are attuned to the specific nuances of each vehicle version, leading to more precise and accurate assessments.

### 3.3.5 Two-Step Classification Approach

#### Primary Classification

The foundational step in the classification methodology involved the deployment of a Deep Convolutional Neural Network (DCNN) to discern and categorize vehicles based on their make and model. DCNNs, by virtue of their deep layered architecture, are adept at learning hierarchical features, making them particularly suited for intricate tasks such as vehicle recognition. The choice of the specific DCNN architecture, along with any relevant hyperparameters, was driven by a combination of empirical evaluations and domain-specific considerations, ensuring optimal feature extraction and classification performance for the dataset in question.

#### Restyling Recognition

Post the primary classification, the challenge of recognizing restylings—or facelifts—came to the fore. Instead of adopting a monolithic classifier approach, the methodology pivoted to a more granular strategy. For every car model that underwent a restyling in the last decade, a dedicated classifier was trained. In total, 357 individual classifiers were developed, each tailored to discern between 2 to 5 classes, corresponding to the number of restylings a particular model underwent.

The rationale behind this design decision is rooted in the inherent advantages of specificity. While a singular, all-encompassing classifier might be generic and

less sensitive to the nuances of specific model restylings, having 357 lightweight, dedicated classifiers ensures a higher degree of precision. These classifiers are more attuned to the subtle visual distinctions between different versions of the same car model, thereby enhancing the granularity and accuracy of the recognition process. Further details on the methodology and architecture will be elaborated in Section 3.3.7.

### 3.3.6 Automated Dataset Creation for Restyling

Creating an annotated dataset for restylings necessitated an automated methodology. A pivotal step in this process involved leveraging cars' registration dates. These dates were matched against a meticulously curated glossary of restyling periods. An exemplar of this methodology is observed in the Toyota Aygo, which witnessed four distinct model versions over the past decade. These versions correspond to four separate production periods, as referenced in table 3.5.

Each production period is characterized by a unique registration interval. These intervals, sourced from commercial car codebooks, are essential in discerning the specific model versions. However, it is imperative to consider periods where overlaps in registration might occur. These overlaps introduce ambiguity, as they represent timeframes where multiple versions could have been registered.

To mitigate potential annotation inaccuracies, cars from overlapping registration periods were systematically excluded. This overlap-free approach aims to eliminate the possibility of erroneously labeling a car model that, while no longer in production, might still be registered during that period. Consequently, for each car model that underwent restyling, a distinct dataset was generated using this method, subsequently serving as the foundation for training the classifiers.

Restyling Class	Restyling Production Interval	Allowed Registration Period
2009	2009-02-09 - 2012-06-01	2009-02-09 - 2012-03-01
2012	2012-03-01 - 2014-07-01	2012-06-01 - 2014-07-01
2014	2014-07-01 - 2018-06-05	2014-07-01 - 2018-06-05
2018	2018-06-05 - Now	2018-06-05 - Now

Table 3.5: Detailed restyling classes and associated production intervals for the Toyota Aygo. The “Restyling Production Interval” represents the period during which a specific version of the model was produced. The “Allowed Registration Period” denotes the time frame during which vehicles from a specific restyling class are most likely to be registered, eliminating overlaps to ensure accurate model identification.

### 3.3.7 Classifier Design and Implementation

The classification approach for recognizing car makes, models, and years is formulated as a two-step process.

#### Primary Make/Model Classifier

The initial phase focuses on classifying vehicles based on make and model. The decision to employ the MobileNetV2 architecture was grounded in its proven performance and efficiency. As detailed in the previous section dedicated to Interior /



Exterior classification, a comprehensive comparison of lightweight architectures was conducted. MobileNetV2 emerged as a suitable choice given its balance between computational complexity and classification performance. This analysis is tabulated in 3.1 for reference. The architecture, pretrained on the ImageNet dataset, is designed to distinguish among 236 classes. These classes represent combinations of the most prevalent makes and models in the market. Notably, all restylings of a particular model are grouped under a single class at this stage.

### **Restyling Recognition**

Upon determining the make and model, the system proceeds to the second phase, which aims at recognizing the specific restyling or generation of the identified model. Instead of a singular classifier catering to all models, an individual classifier is assigned to each model that has undergone restyling in the past decade. In total, 357 such classifiers have been developed, corresponding to each car model with at least one restyling or generation in the last ten years. The granularity of these classifiers varies, with each classifier handling between 2 to 5 classes, indicating that certain models have witnessed up to 4 restylings or 5 distinct versions in the considered time frame.

Features extracted from the penultimate layer of the primary make/model classifier serve as the input to these Multi-Layer Perceptrons (MLPs). This design choice ensures that the features used for restyling recognition are consistent with the primary classification, thereby ensuring cohesion in the two-step process.

The rationale for deploying 357 individual classifiers, as opposed to a singular global classifier, stems from the need for precision. A global classifier, while encompassing, might be too generic, potentially compromising accuracy in discerning between specific model restylings. In contrast, the array of lightweight, model-specific classifiers provides a more tailored and precise recognition mechanism.

### **Implementation**

Given a vehicle image, the system initiates the recognition process with the primary make/model classifier. Upon determining the make and model, the corresponding restyling classifier is invoked to ascertain the specific generation or version of the identified model. This sequential approach ensures a hierarchical classification, where the broad categorization of make and model guides the subsequent granular recognition of restyling. The architecture of this two-step classification process is illustrated in Figure 3.8.

## **3.3.8 Data Preprocessing**

### **Image Filtering**

To ensure the relevance of training data, only exterior images were included in the training dataset. The distinction between exterior and interior images was made using a dedicated exterior/interior model, as described in a previous section. Further, to hone the focus on vehicles, a localization model was employed, ensuring that only images containing vehicles were retained for training.

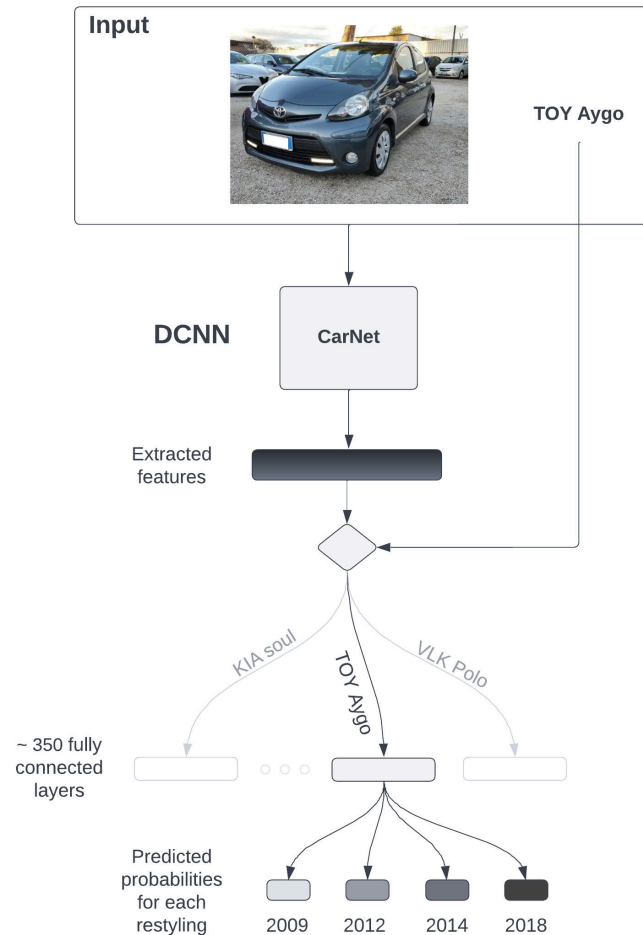


Figure 3.8: The two-step classification process for car make, model, and year recognition. Initially, an input image undergoes preprocessing. Subsequently, features are extracted using the primary classifier, termed as CarNet, which is analogous to ImageNet but tailored for car classes. Upon determining the make and model, in this case exemplified by Toyota Aygo, the corresponding MLP classifier is selected. This dedicated classifier then predicts the specific restyling or generation of the identified model.

### Near-Duplicates Removal

Maintaining dataset quality and minimizing redundancy is essential for effective training and robust model performance. To this end, an aggressive procedure to remove near-duplicates from the dataset was adopted. This process involved the application of perceptual hashing techniques [70]. Each image in the dataset was mapped to a binary vector using perceptual hashing. The similarity between images was then determined by calculating the Hamming distance between these binary vectors.

To further refine the dataset, classes were merged if the minimum distance between their sets of example images was less than a predefined threshold  $\tau$ . This ensured that subtle variations between similar images, which could lead to overfitting or misclassification, were effectively managed.

### Data Preprocessing Visualization

The preprocessing process for the dataset can be visualized as a sequence of three fundamental steps, as illustrated in Figure 3.9. The initial step involves detecting and removing near-duplicates to ensure dataset uniqueness. This is followed by filtering out interior images, retaining only those that depict the exterior of vehicles. The final step focuses on car localization, where the specific area containing the vehicle is cropped, ensuring a concentrated focus on the vehicle itself in subsequent analyses.



Figure 3.9: Flowchart depicting the sequential preprocessing steps for the dataset. Starting with near-duplicates removal, followed by an exterior image filter, and culminating with precise car localization to crop the vehicle’s area.

## 3.3.9 Training and Results

### Make/Model Classifier

**Training Procedure** The primary classifier targets the identification of vehicle makes and models, spanning across 236 classes. The dataset was partitioned into 193,424 training images and 23,601 test images, maintaining a 10% hold-out for testing.

The architecture of choice was MobileNetV2, leveraging weights pretrained on the ImageNet dataset. The model was trained for 16 epochs with a batch size of 64, utilizing the Adamax optimizer and a learning rate set at 0.001. Observations from the validation metrics, calculated at the end of each training epoch, indicated that the minimum loss was achieved at the 8th epoch.

**Quantitative Results** Upon evaluation, the make/model classifier demonstrated commendable performance metrics. The accuracy achieved was 0.819. In terms of precision, recall, and F1-score, the classifier yielded:

Metric	Precision	Recall	F1-score	Support
Accuracy	-	-	0.82	23601
Macro Average	0.81	0.81	0.81	23601
Weighted Average	0.82	0.82	0.82	23601

**Qualitative Results** For a more visual understanding of the classifier’s performance, refer to Figure 3.10, which showcases a selection of images and their corresponding ground truth and predicted labels. This 6x6 grid provides insights into the classifier’s ability to discern between various makes and models.

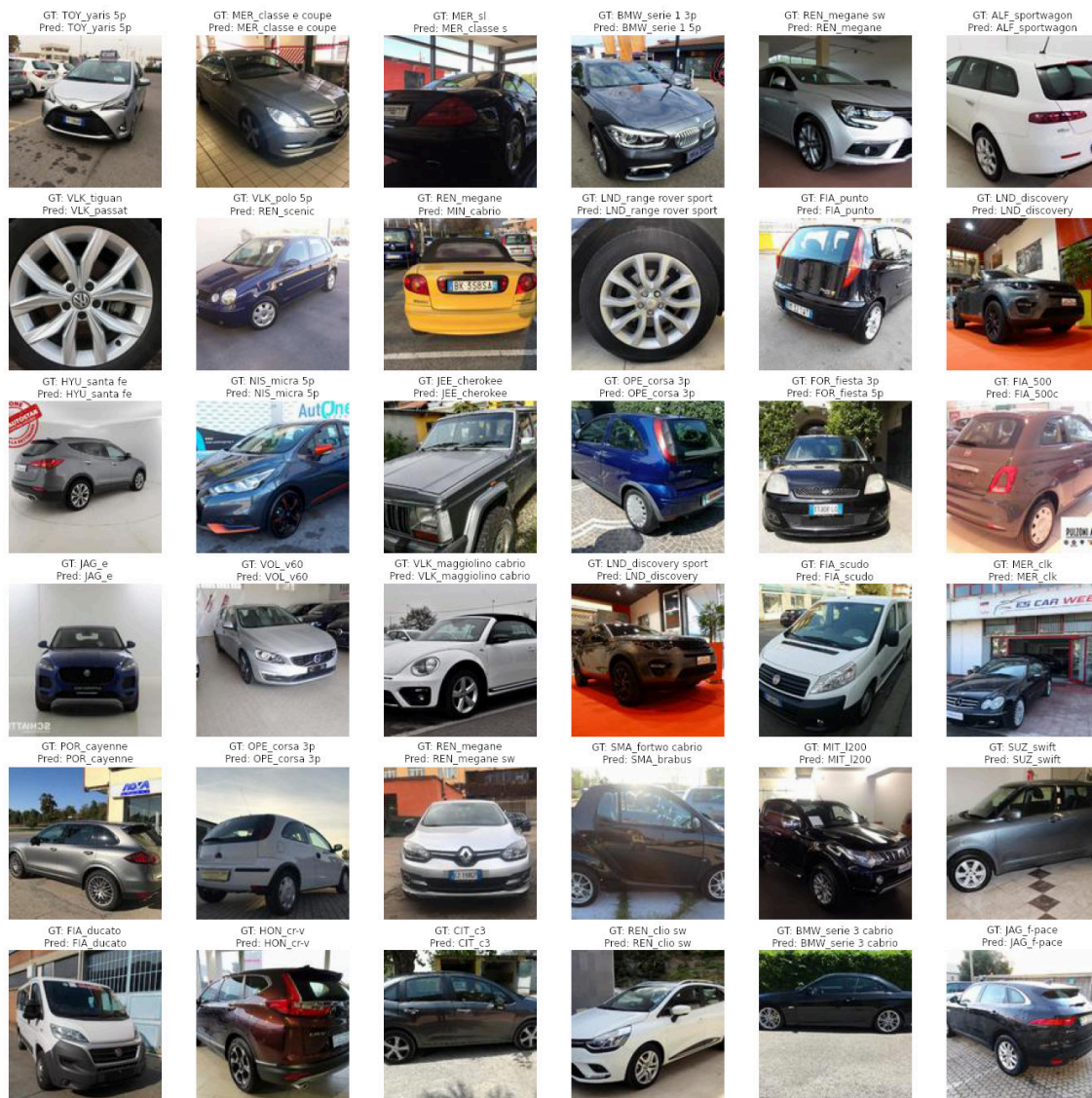


Figure 3.10: Grid representation of the make/model classifier’s predictions. Each cell displays an image, its ground truth label (GT), and the predicted label (Pred) from the classifier.

### Restyling Classifiers

For the fine-grained classification of restylings, a comprehensive set of 357 classifiers was constructed. These classifiers are tailored to specific vehicle models, with the number of discernible restylings ranging from 2 to 5. As an example, the Hyundai i20 5doors model has five distinct restylings, corresponding to the versions from 2010, 2012, 2014, 2018, and 2020.

**Classifier Performance Distribution** The performance distribution of the classifiers, based on the number of restyling classes they differentiate, is visualized in Figure 3.11. This figure provides insight into the varying levels of accuracy achieved for classifiers with different numbers of restyling classes.

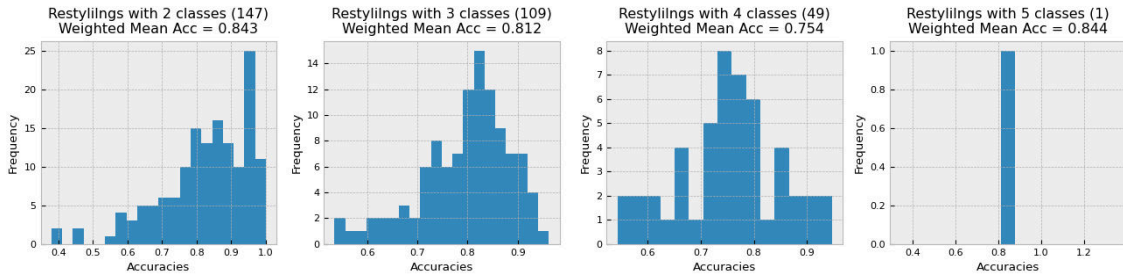


Figure 3.11: Distribution of classifier accuracies based on the number of restyling classes (2 to 5). Each plot represents the accuracy distribution for classifiers differentiating a specific number of restylings. Atop each plot, the corresponding weighted mean accuracy value is displayed.

The table 3.6 presents a summary of the weighted mean accuracies for the restyling classifiers. These weights are calculated based on the size (number of items) of each classifier. This summary is organized based on the number of restyling classes each classifier is designed to differentiate.

Number of Restylings	Weighted Mean Accuracy	Number of Classifiers
2	0.843	147
3	0.812	109
4	0.754	49
5	0.844	1

Table 3.6: Weighted mean accuracies for restyling classifiers based on the number of restyling classes they differentiate. The table also indicates the count of classifiers for each category of restyling classes.

**Case Studies: Weak and Strong Models** To provide a more detailed perspective, two case studies are presented: one showcasing a weaker performing model and another highlighting a strong model.

For the weaker model, the AUDI A3 Sedan was chosen. Conversely, the Mercedes Classe A (5 doors) serves as an example of a robust classifier. Detailed confusion matrices for these models will be provided, offering insights into the specific challenges and strengths of each classifier.



Further, to understand the practical implications of these classifiers, figures with correct and incorrect predictions for both the AUDI A3 Sedan and the Mercedes Classe A (5 doors) will be presented. These figures will visually encapsulate the accuracy and potential pitfalls of each classifier in real-world scenarios.

**Case Studies: Weak and Strong Models** To offer a granular understanding, two distinct case studies are examined:

1. **AUDI A3 Sedan:** This model exemplifies a classifier with suboptimal performance. As detailed in the confusion matrix (refer to Figure 3.12), it is evident that the classifier often confuses the 2013 version with the 2016 version. The 2016 restyling introduced only minor facelifts, specifically the shape of the front lights and subtle changes to the front grilles, making it challenging to differentiate. The achieved accuracy was 0.737.
2. **Mercedes Classe A (5 doors):** Representing a classifier with superior results, the performance metrics for this model are illustrated in Figure 3.13. Conversely, Mercedes classifier achieved a commendable accuracy of 0.965.

For each of these models, two figures are provided:

1. A confusion matrix detailing the classification outcomes for the restylings of the vehicle model.
2. A classification report outlining precision, recall, and F1 score metrics for each of the restylings.

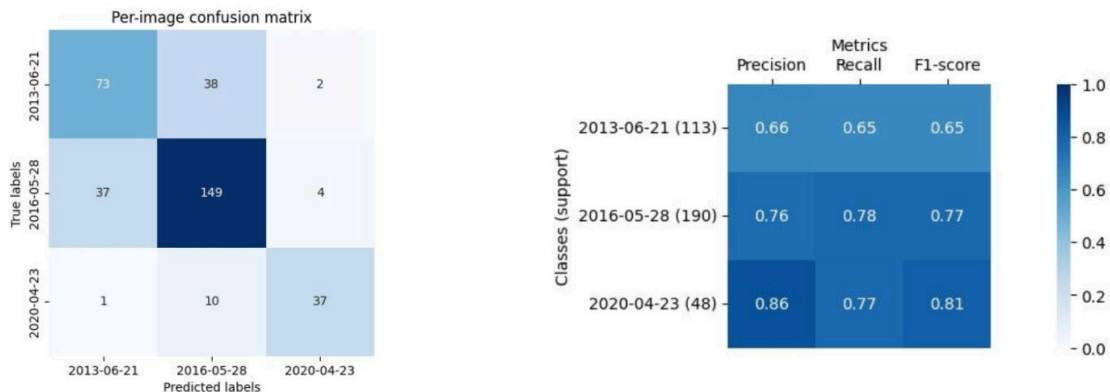


Figure 3.12: Confusion matrix for the AUDI A3 Sedan classifier detailing classification results across its restylings.

### 3.3.10 Discussion and Possible Improvements

The Make/Model/Year recognition system, as designed, hinges on a two-step approach that, while introducing architectural intricacies, offers potential advantages in granular classification tasks. Intuitively, a more compartmentalized classification mechanism, wherein different heads cater to smaller sets of classes, can provide

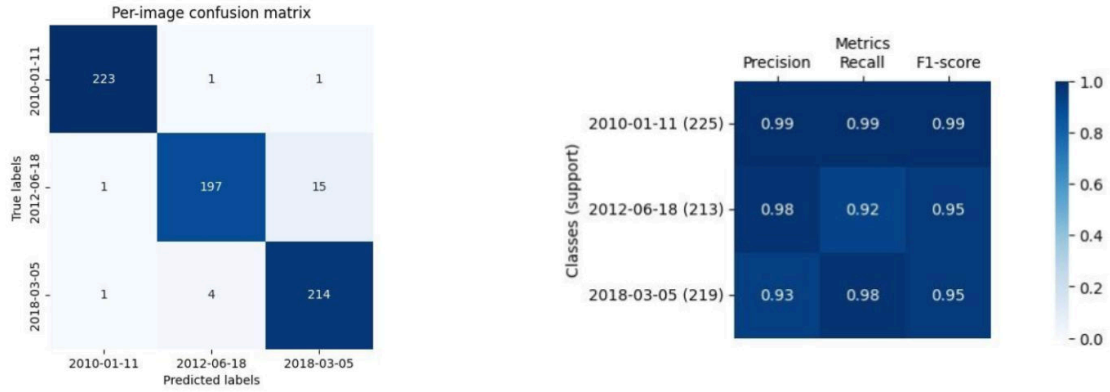


Figure 3.13: Confusion matrix for the Mercedes Classe A (5 doors) classifier showcasing the classification outcomes across its restylings.

more tailored and precise classification boundaries. Rather than having a monolithic classifier managing a vast number of classes, leveraging multi-head classifiers, especially for groups of 2-5 classes, can be more efficient. This hypothesis is further supported by the choice of lightweight classifiers for the second step, using 1-layer MLPs, which ensures that despite the multi-head approach, resource consumption remains minimal.

However, it is essential to acknowledge the areas that beckon further refinement. The current architecture, although promising, has displayed varying performance across models, with some exhibiting sub-optimal accuracy levels. Future endeavors could focus on enhancing the classifier’s ability to discern subtle facelift features more effectively. Incorporating additional cues, such as the vehicle’s pose—a subject explored in subsequent chapter—could serve as valuable input, potentially refining the classification process and further bolstering the system’s efficacy.

### 3.4 Conclusion

In the intricate process of car damage assessment from photographs, the initial phase of *Vehicle Identification* has been underscored in this chapter as a bedrock for subsequent, more detailed analyses. The systematic breakdown of this phase into three pivotal modules offers a comprehensive insight into the nuances and intricacies of identifying vehicles from images.

The differentiation of photographs into exterior and interior types via **Photograph Type Classification** establishes the initial context for the system. This foundational knowledge is further built upon by **Vehicle Localization on Photographs**, ensuring the precise positioning of the vehicle in the image, thus eliminating extraneous details and sharpening the focus on the subject.

While these modules set the context and scope, the depth of the system’s understanding is elevated by the **Vehicle Make/Model/Year Classification**. This module identifies the vehicle that can be instrumental in later stages of assessment.

While the modules elucidated in this chapter lay a robust foundation for the proposed car damage assessment system, it is worth noting that there exists potential for refinement, especially in the granularity of vehicle identification. As research

progresses, further enhancements can be envisioned, making the process more precise and efficient.

In essence, this chapter has laid the groundwork, ensuring the system is well-prepared with comprehensive vehicle-specific details, setting the stage for the subsequent phases of pose detection and damage assessment.



# Chapter 4

## Vehicle Pose Detection

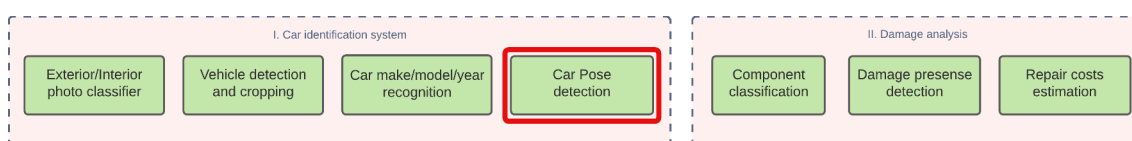


Figure 4.1: Highlighted segment of the whole system representing the focus of this chapter.

In car damage assessment from photographs, determining the vehicle’s pose is crucial. Detecting the pose accurately helps localize damages and infer potential causative events. Figure 4.1 showcases the role of the pose detection module within the damage recognition system.

The chapter begins with the **Related Works** section, reviewing historical and current methodologies in vehicle pose detection. This sets the stage for **Defining and Visualizing Azimuth**, which dives into the concept of azimuth, its visual representations in vehicle orientation. The **Proposed Approach** section introduces two distinct architectures for encoding the azimuth. Their effectiveness, limitations, and areas for improvement are then evaluated in the **Results** section.

This chapter is based on the vehicle pose estimation discussions presented in [71].

## 4.1 Introduction and Related works

### 4.1.1 Introduction to Pose Detection

Pose detection revolves around the process of determining the position and orientation of specific parts or features of an object or entity in images or videos. Historically, the primary motivation for developing pose detection algorithms was to detect and analyze human body parts and their relative positions. Over time, these methodologies have evolved and have been adapted to cater to various objects, including cars, enabling applications in fields as varied as animation, augmented reality, sports analytics, and vehicle damage assessment.

Early techniques employed to estimate pose made use of part-based models, where individual parts of an entity (like limbs in humans) were detected and then assembled to deduce the overall pose [72]. The advent of deep learning, particularly

Convolutional Neural Networks (CNNs), ushered in a transformation in this domain. CNNs, with their hierarchical structure, can capture complex patterns and spatial hierarchies, making them immensely effective for tasks like pose estimation [73]. Representative models like OpenPose [74] and DensePose [75] have set benchmarks in human pose estimation, extracting skeletal structures with impressive accuracy. Adapting such methodologies for car pose detection necessitates accounting for the unique challenges posed by vehicular structures, like varied designs and reflective surfaces, but the foundational principles remain analogous.

### 4.1.2 Traditional Image Processing vs. Deep Learning

The progression of pose estimation techniques can be broadly categorized into two phases: the era of traditional image processing and the subsequent advent of deep learning methodologies. Both approaches offer distinct advantages and have been instrumental in shaping the trajectory of pose detection algorithms.

Traditional image processing techniques primarily relied on handcrafted features. These features, meticulously designed, were intended to be invariant to transformations like scaling, rotation, and partial occlusion. Prominent among these techniques are SIFT (Scale-Invariant Feature Transform) [76] and SURF (Speeded-Up Robust Features) [47]. Both SIFT and SURF are designed to detect and describe local features in images, making them robust against transformations and thus suitable for tasks like object recognition and pose estimation. To determine the pose of objects from these features, researchers often turned to geometric techniques. The PnP (Perspective-n-Point) problem, which involves deducing an object’s pose based on a set of 2D-to-3D point correspondences, played a pivotal role in this phase [77].

The rise of deep learning and particularly, Convolutional Neural Networks (CNNs), brought a paradigm shift in pose detection methodologies. Unlike traditional methods, where features had to be meticulously crafted, CNNs allowed for automatic feature learning from data. These networks, with their deep architectures, could learn intricate and hierarchical patterns from vast datasets. As a result, they proved to be remarkably effective in detecting subtle nuances in images, surpassing the performance of conventional techniques in many benchmarks. Deep learning models, such as PoseNet [78] and Mask R-CNN [13], are representative examples that have showcased the potential of CNNs in pose estimation tasks.

### 4.1.3 Car Pose Estimation

Historically, the process of car pose estimation was rooted in techniques derived from traditional image processing. Features were extracted, typically edge-based or texture-based, from car images, and matched with 3D car models to deduce the pose. As cars have a rigid structure as opposed to the flexible human anatomy, these methods initially showcased promise. However, the vast array of car models, designs, and colors, coupled with real-world factors like occlusions, varying illumination, and reflections, often compromised the accuracy of these techniques.

With the advent of deep learning, car pose estimation witnessed a substantial advancement. CNNs, inherently adept at managing diverse patterns and structures in images, have been instrumental in this progress. Deep learning models tailored for car pose estimation, such as [79], offer impressive accuracy in predicting the 3D

bounding box of cars from 2D images, further refining the pose estimation process. [80] similarly aimed at estimating the continuous six-degree of freedom (6-DoF) pose of objects from a single RGB image. Another significant deep learning model tailored for car pose estimation is MonoGRNet [81]. This model offers a unified approach for 3D vehicle detection and pose estimation using only monocular RGB images. By leveraging geometric relationships between the 2D and 3D bounding boxes, MonoGRNet not only predicts the 3D location and dimensions of vehicles but also accurately estimates their poses, making it especially pertinent for scenarios where stereo or depth information is unavailable. [82] ventured into a generic deep pose estimation approach that does not rely on category-specific training. This method, dynamically conditioned with a 3D shape representation, offers flexibility and outperforms other techniques across multiple benchmarks.

The scarcity of annotated training data and the need for powerful features are issues in viewpoint estimation. To tackle these, [83] proposed a combination of render-based image synthesis and Convolutional Neural Networks (CNNs). [84] took a step further by estimating 3D pose and subsequently retrieving 3D models that accurately match objects in RGB images. Both methods harness the growing availability of 3D models to improve performance.

Some novel methodologies and data augmentation are represented in several works. [85] put forward a characteristic view selection model (CVSM) that integrates a reinforcement learning framework for efficient 3D pose estimation. [86] integrated Riemannian geometry into CNN-based monocular orientation estimation, offering a mix of regression and classification frameworks that account for nearly symmetric objects.

A key aspect of car pose estimation is its criticality in various applications. From autonomous driving systems, where understanding the orientation and position of surrounding vehicles is crucial, to insurance sectors assessing vehicle damages, accurate car pose detection plays an indispensable role.

Furthermore, the importance of car pose estimation is underscored in scenarios where direct sensor data, like LiDAR or radar, might be unavailable or compromised. In such situations, visual cues become the primary source of information, emphasizing the significance of accurate pose estimation techniques for cars [87].

#### 4.1.4 Datasets for Car Pose Estimation

An essential component in developing, training, and benchmarking car pose estimation algorithms is the availability of comprehensive and high-quality datasets. These datasets offer a collection of annotated images or sequences that represent varied car models, poses, lighting conditions, and occlusions, providing a foundation for researchers to innovate and evaluate their methodologies.

**KITTI** [87]: Established as one of the premier datasets for autonomous driving tasks, the KITTI dataset comprises a rich collection of annotated images, LiDAR point clouds, and other sensor data captured in urban settings. For car pose estimation, KITTI provides annotated 3D bounding boxes that are valuable for both training and benchmarking pose estimation algorithms.

**PASCAL3D+** [88]: An extension of the PASCAL VOC dataset, PASCAL3D+ augments the original dataset with 3D annotations for objects, including cars. This dataset is particularly notable for its varied set of car poses, making it a valuable

resource for car pose estimation. Figure 4.2 illustrates the azimuth distribution for each object category in the PASCAL3D+ dataset.

**CompCars** [30]: Specifically curated for car-related tasks, the CompCars dataset offers a vast collection of over 200,000 photos covering 171 car makes. Annotations include car models, types, and viewpoints, thereby making it a resourceful dataset for pose estimation and related tasks.

**ApolloCar3D** [89]: Stemming from the Apollo autonomous driving project, ApolloCar3D offers a collection of images from urban driving scenarios. This dataset provides rich 3D car instance annotations, catering specifically to 3D car understanding tasks including pose estimation [4].

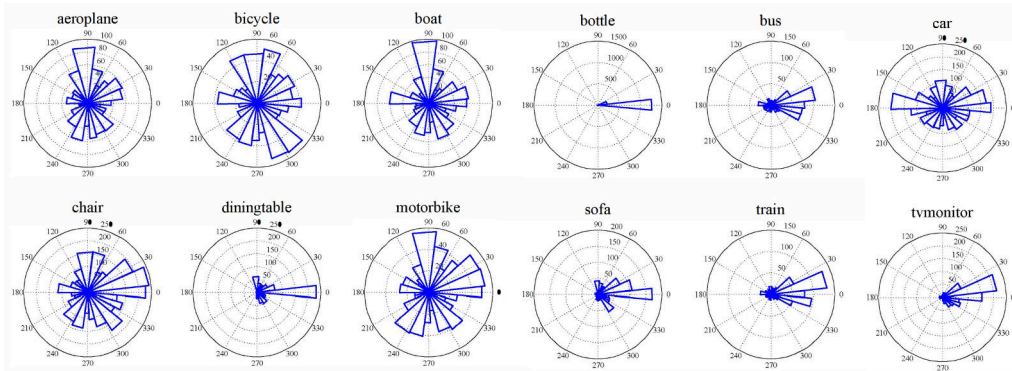


Figure 4.2: Polar histograms show the distribution of azimuth among the PASCAL3D+ images for each object category.

The availability of these datasets has substantially accelerated research advancements in car pose estimation. By offering diverse representations of cars and associated annotations, they empower researchers to train more robust models and evaluate them under varied conditions. As technology evolves and the nuances of pose estimation become more intricate, the continuous enhancement and creation of datasets will remain crucial.

### PASCAL3D+ Dataset

Selecting an apt dataset is pivotal in guiding the research process and ensuring the derived outcomes are reflective of the research objectives. For this investigation into car pose estimation, with a particular focus on azimuth estimation, the PASCAL3D+ dataset emerged as a front-runner. A driving factor behind this choice was the detailed annotations the dataset offers for each image, notably the azimuth values. Azimuth estimation, a critical facet of pose detection, provides insights into an object’s orientation within a 3D space, as detailed later on in section 4.2. PASCAL3D+ alleviates the complexities of deriving these angles by offering direct data for azimuth estimation, ensuring a more precise and streamlined research methodology.

The PASCAL3D+ dataset, a stellar extension of the revered PASCAL VOC dataset, augments the original images with intricate 3D annotations, laying the foundation for 3D object detection and pose estimation tasks.

A prominent feature of this dataset is its compilation of 5,475 car images, sourced directly from ImageNet, presenting a myriad of scenarios for researchers to explore.

Each car in this dataset is meticulously annotated with a corresponding 3D CAD model, which enables researchers to juxtapose pose estimations against a standardized 3D reference. For cars, the annotations delve deep, offering viewpoints, bounding boxes, and crucially, azimuth angles.

Several nuances make PASCAL3D+ a challenging yet rewarding dataset. The presence of occluded objects simulates real-world complications that algorithms need to account for. Furthermore, the dataset showcases a wide variance in car makes and models, capturing the diversity of the automotive world. However, it is essential to note that while the dataset offers this diversity, it does not explicitly label the specific makes or models.

The comprehensive nature of PASCAL3D+ combined with its direct azimuth data solidifies its position as an invaluable tool for detailed investigations into car pose estimation, especially when dealing with the intricacies of diverse car images and occlusions.

## 4.2 Defining and Visualizing Azimuth

**Problem definition:** In the domain of vehicle pose detection, one of the paramount tasks is the precise estimation of the vehicle’s orientation in a given image or frame. The key orientation parameter being focused upon in this research task is the azimuth, often denoted as  $\varphi$ .

**Definition of Azimuth:** The azimuth,  $\varphi$ , is defined as the angle in the range  $[-\pi, \pi]$  that represents the orientation of a vehicle with respect to the viewer. Originating from the front of the car, this angle describes how much the vehicle has rotated from this frontal viewpoint. For instance,  $\varphi = 0$  would indicate a car directly facing the viewer, while  $\varphi = \frac{\pi}{2}$  would signify the car turned  $90^\circ$  to the right. This definition is depicted in Figure 4.3



Figure 4.3: Azimuth  $\varphi$  definition for car pose estimation. In this image, an azimuth of  $\varphi = -30^\circ$  corresponds to the car slightly turned to present its right side (passenger side) towards the viewer. The right dashed line indicating  $0^\circ$  represents the reference axis for this calculation.

It is noteworthy to mention the deliberate exclusion of other viewpoint characteristics from this study, such as elevation, distance, and the roll equivalent from roll pitch and yaw. While these parameters can offer further granularity to pose detection, the primary focus here remains the continuous estimation of azimuth.

Pose estimation, especially of vehicles, often brings with it a multitude of challenges, ranging from variances in lighting to occlusions. However, when distilled to its essence, the problem tackled in this research is one of regression. Instead of the conventional classification-based approach where discrete classes represent different poses or orientations, the goal here is continuous azimuth estimation. This involves predicting a specific value of  $\varphi$  for a given vehicle image. The advantage of this method is that it allows for a much finer granularity of orientation prediction, catering to even minute variations in vehicle orientation, and thereby enhancing the accuracy of pose detection.

Accompanying this description is a visualization, showcasing a car image annotated with its corresponding azimuth  $\varphi$ , shown on Figure 4.4.



Figure 4.4: Some examples of azimuth  $\varphi$  values and corresponding pose visualizations

## 4.3 Proposed approach

### 4.3.1 Introduction and Motivation for Design Choices

Vehicle pose estimation, especially focusing on the azimuthal angle, is a multifaceted challenge. While most regression tasks in deep learning provide continuous values within a predictable range, the angular nature of azimuth presents cyclic constraints that require special consideration.

The motivation behind adopting unique approaches to predict the azimuth  $\varphi$  emerges from the inherent challenges of angular regression. Traditional regression models would treat angles such as  $\varphi = \pi$  and  $\varphi = -\pi$  as distinct, ignoring their equality due to the cyclic nature of angles.

In the context of this research, two distinct methodologies have been adopted. The subsequent sections will delve into each approach, confronting obtained results with the state of the art methods.

### 4.3.2 Architecture 1. Sin-Cos Representation

One of the pivotal tasks in vehicle pose estimation is to represent the azimuthal angle,  $\varphi$ , in a format that can be effectively estimated using deep convolutional neural networks (DCNNs). To this end, the first proposed architecture adopts a Sin-Cos representation.

**Model Construction:** The designed DCNN architecture is partitioned into two primary segments. Initially, a *backbone* is utilized as an image feature descriptor. This backbone captures intricate patterns and details from the input images,

converting them into a condensed feature map. Following this feature extraction phase, a custom multi-layer perceptron (MLP) is stacked atop the backbone. This MLP consists of a hidden layer comprising 100 neurons, activated by the ReLU (Rectified Linear Unit) function. To enhance generalization and curtail overfitting, a dropout layer with a rate of 10% is integrated into the architecture [90]. The whole architecture is presented on Figure 4.5.

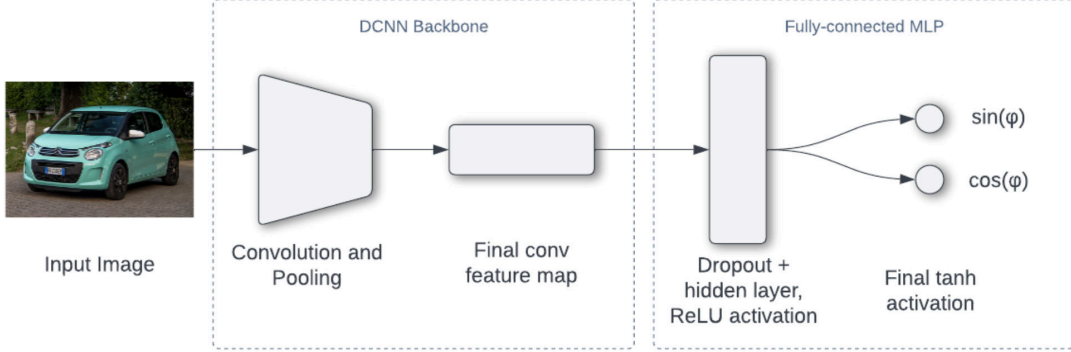


Figure 4.5: Proposed Sin-Cos architecture

**Output Mechanism:** The crux of this architecture lies in its output mechanism. The network culminates in two output neurons that are activated by the hyperbolic tangent (tanh) function. The tanh activation ensures that the output values lie within the range  $[-1, 1]$ , which aligns with the natural range of sine ( $\sin(\varphi)$ ) and cosine ( $\cos(\varphi)$ ) functions. Thus, these neurons are adeptly designed to predict the sine and cosine values of the azimuthal angle. Consequently, the estimated azimuth  $\varphi$  can be derived using the inverse tangent function as:

$$\varphi = \text{atan2}(o_1, o_2)$$

where  $o_1$  and  $o_2$  correspond to the outputs of the sine and cosine neurons respectively.

**Loss Function:** The training process aims to optimize the mean squared error (MSE) between the predicted values and the true sine and cosine values. Mathematically, the loss  $L$  is represented as:

$$L = \frac{1}{N} \sum_{i=1}^N ((o_{1i} - y_{1i})^2 + (o_{2i} - y_{2i})^2)$$

where  $N$  is the number of samples,  $o_{1i}$  and  $o_{2i}$  are the predicted sine and cosine values respectively, and  $y_{1i}$  and  $y_{2i}$  are the true sine and cosine values.

**Azimuth Calculation from Sine and Cosine:** To estimate the azimuth  $\varphi$  from the predicted sine and cosine outputs, the inverse tangent function, typically represented as  $\text{atan2}$ , is employed. Given the nature of this function, it is capable of determining the correct quadrant for the resulting angle based on the signs of the sine and cosine values. Specifically, the formula is:

$$\varphi = \text{atan2}(y_{\sin}, y_{\cos}) \quad (4.1)$$

**Drawbacks:** While the Sin-Cos representation offers a unique approach to tackle the cyclic nature of azimuth angles, it is not devoid of challenges. The most significant is that the predicted sine and cosine values, when considered in isolation,



do not guarantee a resultant unit vector. This problem is illustrated on Figure 4.6. Specifically, when reconstructing the azimuth using  $\text{atan2}(y_{\sin}, y_{\cos})$ , only one of the sine or cosine values dominantly determines the resultant angle, while the other mainly influences the sign and quadrant determination. Thus, even if one value is significantly off, it might not drastically affect the angle's magnitude but can change its direction. This can lead to errors, especially when the predicted values drift away from forming a unit vector.

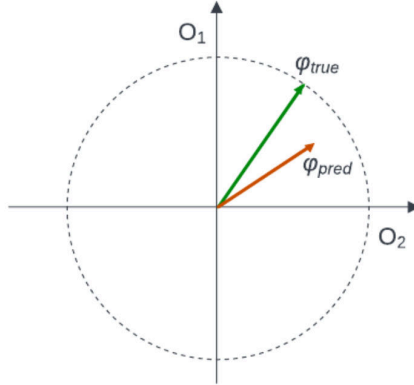


Figure 4.6: Architecture drawback: sine and cosine values, when considered in isolation, do not guarantee a resultant unit vector

### 4.3.3 Architecture 2. Directional Discriminators

To introduce more nuance and precision in the estimation of the azimuthal angle,  $\varphi$ , the second architecture employs a distinctive double-discriminator approach. While it retains the same backbone as the first architecture, it refines its head to present an innovative mechanism for pose determination.

**Output Interpretation:** In contrast to the previous architecture, the network culminates in two output neurons activated by the sigmoid function. This choice ensures that the predictions are bounded within the  $[0, 1]$  range. These outputs correspond to the normalized absolute values of two novel angles:  $\alpha$  and  $\beta$ .

**Alpha Discriminator ( $|\alpha|$ ):** The  $\alpha$  angle represents the azimuthal view from the car's front position. Specifically:

- $\alpha = 0$  depicts a direct frontal view of the car.
- $\alpha = \pi$  corresponds to a direct rear view.
- $\alpha = \pi/2$  represents the left side view.
- $\alpha = -\pi/2$  equates to the right side view.

Given the absolute interpretation  $|\alpha|$ , it inherently serves as a front/rear discriminator. However, this absolute representation also forfeits its ability to distinguish between the car's left and right sides.

**Beta Discriminator ( $|\beta|$ ):** The  $\beta$  angle complements  $\alpha$  and serves a similar function but with different reference points:



- $\beta = 0$  signifies the car's left side (driver's seat) view.
- $\beta = \pi$  corresponds to the car's right side (passenger seat) view.
- $\beta = \pi/2$  indicates the car's rear view.
- $\beta = -\pi/2$  represents the direct frontal view.

Being an absolute representation  $|\beta|$ , it naturally acts as a left/right discriminator, but similarly loses distinction between front and rear views.

A visualization is provided to elucidate these angles and their orientation concerning the car's image on a Figure 4.7. Additionally, a schematic of the network architecture highlights the structural design and flow, Figure 4.8

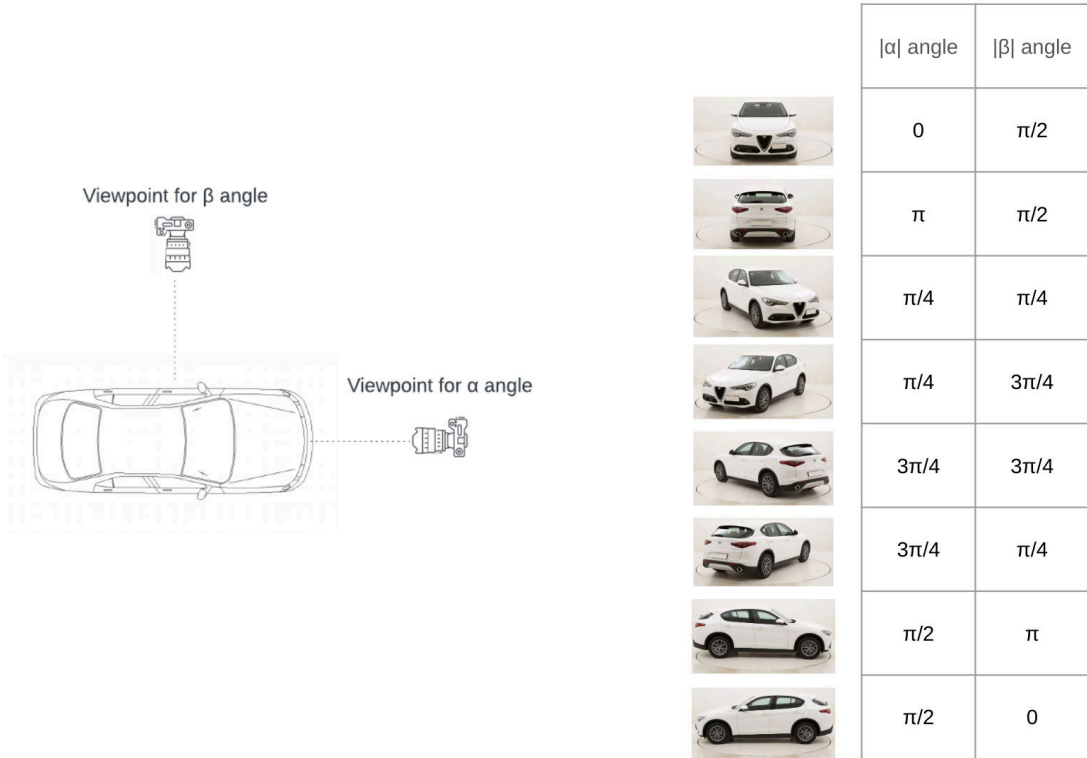


Figure 4.7: On the left: viewpoints visualization for the  $\alpha$  and  $\beta$  angles. On the right: Viewpoint of a car and corresponding values of  $|\alpha|$  and  $|\beta|$

**Loss Function:** The network optimizes a composite loss function derived from the binary cross-entropy (BCE) loss for both  $\alpha$  and  $\beta$  predictions. Formally, the loss  $L$  is given by:

$$L = \text{BCE}(\alpha_{\text{pred}}, \alpha_{\text{true}}) + \text{BCE}(\beta_{\text{pred}}, \beta_{\text{true}})$$

Where the binary cross-entropy (BCE) is defined as:

$$\text{BCE}(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

In the above:

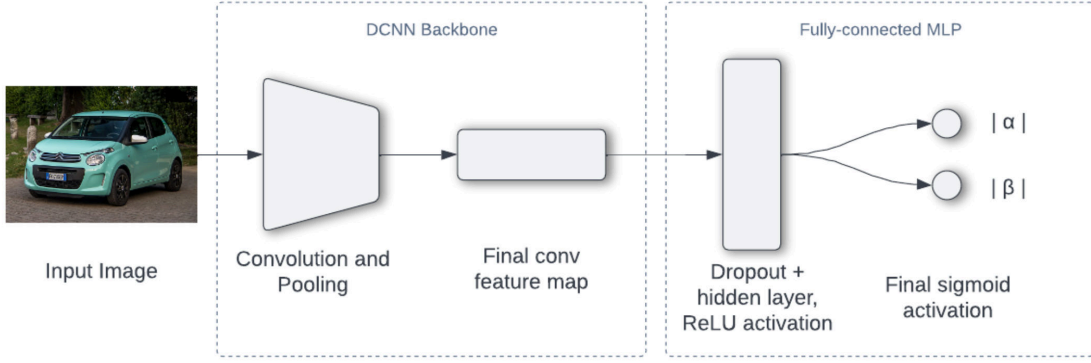


Figure 4.8: Proposed Directional Discriminators architecture

- $y$  represents the true labels (ground truths).
- $\hat{y}$  denotes the predicted values from the network.
- $N$  is the total number of samples.

#### Azimuth calculation from the sigmoids predictions

To estimate the azimuth  $\varphi$  from the sigmoid outputs, it is necessary to transform these outputs to angles within the range  $[0, \pi]$ .

$$\alpha_{\text{abs}}, \beta_{\text{abs}} = y_{\text{sigmoids}} \times \pi \quad (4.2)$$

Here,  $\alpha_{\text{abs}}$  and  $\beta_{\text{abs}}$  represent the absolute angles corresponding to the front/rear and left/right discriminators, respectively. The next step is to determine the specific quadrant of the azimuth angle based on the values of  $\alpha_{\text{abs}}$  and  $\beta_{\text{abs}}$ :

$$Q_1 \leftrightarrow \alpha_{\text{abs}} < \frac{\pi}{2} \wedge \beta_{\text{abs}} < \frac{\pi}{2} \quad (4.3)$$

$$Q_2 \leftrightarrow \alpha_{\text{abs}} \geq \frac{\pi}{2} \wedge \beta_{\text{abs}} < \frac{\pi}{2} \quad (4.4)$$

$$Q_3 \leftrightarrow \alpha_{\text{abs}} \geq \frac{\pi}{2} \wedge \beta_{\text{abs}} \geq \frac{\pi}{2} \quad (4.5)$$

$$Q_4 \leftrightarrow \alpha_{\text{abs}} < \frac{\pi}{2} \wedge \beta_{\text{abs}} \geq \frac{\pi}{2} \quad (4.6)$$

Having determined the quadrant, it is necessary to compute the secondary angle,  $\alpha_{2,\beta}$ , based on the quadrant and the value of  $\alpha_{\text{abs}}$  and  $\beta_{\text{abs}}$ :

$$\alpha_{2,\beta} = \begin{cases} \frac{\pi}{2} - \beta_{\text{abs}}, & \text{if } Q_1 \\ \frac{\pi}{2} + \beta_{\text{abs}}, & \text{if } Q_2 \\ 3\frac{\pi}{2} - \beta_{\text{abs}}, & \text{if } Q_3 \\ -\frac{\pi}{2} + \beta_{\text{abs}}, & \text{if } Q_4 \end{cases} \quad (4.7)$$

The mean angle,  $\bar{\alpha}$ , is then computed by averaging  $\alpha_{\text{abs}}$  and  $\alpha_{2,\beta}$ :

$$\bar{\alpha} = \frac{\alpha_{\text{abs}} + \alpha_{2,\beta}}{2} \quad (4.8)$$

Lastly, the azimuth  $\varphi$  is obtained by adjusting the sign of  $\bar{\alpha}$  based on the quadrant:

$$\varphi = \bar{\alpha} \times (-1)^{\delta(Q_3 \vee Q_4)} \quad (4.9)$$

In this formula,  $\delta$  is the Kronecker delta function, which assigns a value of 1 if either condition  $Q_3$  or  $Q_4$  is true, and 0 otherwise.

**Drawbacks:** The introduction of two discriminators for azimuth representation can make the network’s prediction mechanism less intuitive and more intricate than the more direct sin-cos representation. Moreover, by utilizing absolute values and confining outputs to the range  $[0, \pi]$ , there is potential for a loss of precision in angle estimation, especially when the real angle hovers near the defined boundaries.

### 4.3.4 Evaluation method

Viewpoint estimation, especially for automobile orientation, distinguishes itself from traditional classification tasks by predicting a continuous variable instead of categorical outputs. In this work, by decomposing the target into two variables (e.g., sin/cos or alpha/beta), it is possible to employ classical regression error metrics for evaluation. Therefore, besides the commonly used Median Error (MedErr) and Accuracy within  $\pi/6$  ( $\text{Acc}_{\pi/6}$ ), regression evaluation metrics like Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and  $R^2$  have been incorporated, given their significance in assessing models yielding continuous predictions.

**Mean Absolute Error (MAE)** The Mean Absolute Error is a measure of the average magnitude of errors between predicted and actual values, without considering their direction. It is defined as:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |\hat{\varphi}_i - \varphi_{\text{true},i}| \quad (4.10)$$

where  $n$  is the number of predictions. A lower MAE indicates that the model’s predictions are close to the actual values on average, which is preferable.

**Root Mean Square Error (RMSE)** RMSE represents the square root of the second sample moment of the differences between predicted values and observed values or the quadratic mean of these differences. It emphasizes larger errors over smaller ones. It is given by:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{\varphi}_i - \varphi_{\text{true},i})^2} \quad (4.11)$$

A lower RMSE indicates a better fit of the model to the data.

**R squared**  $R^2$ , also known as the coefficient of determination, represents the proportion of the variance in the dependent variable that is predictable from the independent variables. For regression models, this metric provides an indication of how well observed outcomes are replicated by the model.

The sum of squares of residuals (SSR) represents the aggregate squared difference between the observed outcomes and the values predicted by the model:

$$SSR = \sum_i (y_i - \hat{y}_i)^2 \quad (4.12)$$

The total sum of squares (SST) quantifies the overall variance in the dependent variable:

$$SST = \sum_i (y_i - \bar{y})^2 \quad (4.13)$$

The  $R^2$  value is then defined as:

$$R^2 = 1 - \frac{SSR}{SST} \quad (4.14)$$

A higher  $R^2$  value suggests that the model explains a higher proportion of the variance in the output variable.

**Median Error (MedErr)** The Median Error (MedErr) metric provides a robust measure of central tendency of the viewpoint estimation errors across the dataset. Given the predicted viewpoints  $\hat{\varphi}$  and the ground truth viewpoints  $\varphi_{\text{true}}$ , the MedErr is computed as:

$$\text{MedErr} = \text{median}(|\hat{\varphi} - \varphi_{\text{true}}|) \quad (4.15)$$

The usage of the median offers resistance to outliers, ensuring that sporadic large errors do not dominate the evaluation. A lower MedErr indicates that the central error distribution of the model’s predictions is close to the ground truth, which is desirable.

**Accuracy within  $\pi/6$  ( $\text{Acc}_{\pi/6}$ )** This metric provides a complementary perspective to MedErr by focusing on the percentage of predictions that are reasonably close to the ground truth. Specifically,  $\text{Acc}_{\pi/6}$  measures the proportion of predictions where the error is within  $\pi/6$  radians (or 30 degrees) of the true viewpoint:

$$\text{Acc}_{\pi/6} = \frac{\text{number of predictions with } |\hat{\varphi} - \varphi_{\text{true}}| \leq \pi/6}{\text{total number of predictions}} \times 100\% \quad (4.16)$$

A high  $\text{Acc}_{\pi/6}$  value suggests that a significant proportion of the model’s predictions are not only correct but also highly accurate, falling within a tight margin around the ground truth.

### 4.3.5 Training

#### Data Preparation

**Dataset Split** The PASCAL3D+ dataset, which was employed for this research, inherently provides a train/validation split. The total number of images in the dataset amounts to 5,475. Of these, 2,763 belong to the training set, while 2,712 are earmarked for validation, representing a nearly even 50/50 split.

**Data Augmentation** To boost the robustness of the trained models and to mitigate overfitting, an array of data augmentation techniques was integrated into the pipeline:

- **Rotation:** Images were rotated with a random angle constrained to a maximum of  $10^\circ$ .
- **Barrel/Pincushion Distortions:** These were introduced to simulate lens distortions.
- **Brightness and Contrast Adjustments:** Random adjustments were made to image brightness and contrast levels.
- **Horizontal Flips:** Images were horizontally flipped. It is essential to note that the azimuth angle needs adjustment when flipping.

**Azimuth Adjustment for Horizontal Flips** When an image is flipped horizontally in the Sin-Cos approach, the sine value of the azimuth changes its sign while the cosine value remains the same. Given the original pose  $[\sin(\varphi), \cos(\varphi)]$ , the adjusted pose after a horizontal flip becomes:

$$[ -\sin(\varphi), \cos(\varphi) ] \quad (4.17)$$

In the Directional Discriminators approach, the value for  $\alpha$  remains unchanged after the horizontal flip, but the value for  $\beta$  is subtracted from 1. Given the original pose  $[\alpha, \beta]$ , the adjusted pose post horizontal flip becomes:

$$[ \alpha, 1 - \beta ] \quad (4.18)$$

**Network Backbone** For the neural network backbone, the EfficientNetB0 architecture [40] was chosen, pre-trained on ImageNet dataset [91]. EfficientNetB0 is acknowledged for delivering state-of-the-art performance while maintaining a relatively compact model size. Its design philosophy makes it an ideal choice for this research, ensuring efficient training without compromising accuracy.

### Training Parameters & Hardware Configuration

The training process was governed by the following parameters:

- **Learning Rate:**  $5 \times 10^{-3}$
- **Optimizer:** Adam
- **Learning Rate Decay:** 0.96
- **Batch Size:** 32

The models were trained for a maximum of 50 epochs. However, an early stopping mechanism was integrated to halt training if the validation performance did not improve for 7 consecutive epochs (patience parameter).

The training was facilitated on a hardware setup powered by an Nvidia Tesla T4 GPU, ensuring swift and efficient computation throughout the training process.

## Training Curve Plots

Training curve plots offer essential visual insights into the model's behavior throughout its training phase. Through these plots, one can discern patterns indicative of overfitting, underfitting, or a stable learning progression.

**Approach 1: Sin-Cos Representation, Figure 4.9** The minimum validation loss is pinpointed around the 35th epoch, registering at an approximate value of 0.027. The distinction between the training and validation loss here serves as a testament to the model's ability to generalize on unseen data.

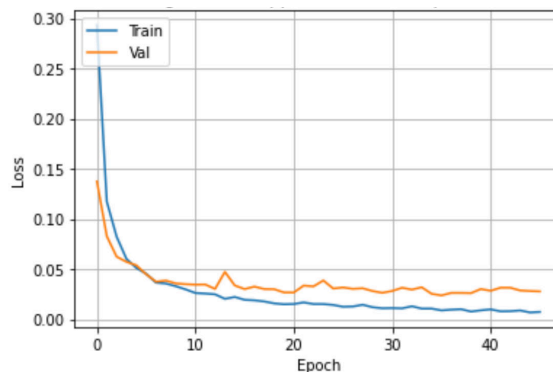


Figure 4.9: Training curve for the Sin-Cos approach

**Approach 2: Directional Discriminators, Figure 4.10** In the Directional Discriminators approach, the validation loss curve arrives at its minimum close to the 25th epoch, with a reading of about 1.05. Considering the unique loss function used for this approach, it is pivotal to interpret this loss value within its specific context, rather than in direct comparison to other models or approaches.

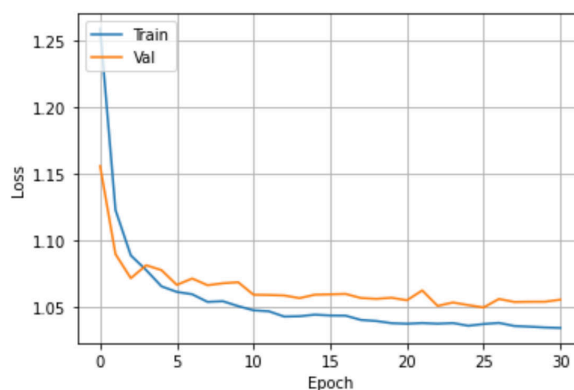


Figure 4.10: Training curve for the Directional Discriminators approach

For both methods, the plots emphasize the judiciousness of employing the early stopping technique, ensuring that training is halted once optimal validation loss is achieved, preempting any overfitting.

## 4.4 Results

### 4.4.1 Quantitative Results

The quantitative assessment of the viewpoint estimation performance comprises two tables. Table 4.1 provides a detailed performance evaluation of the proposed methods using all five metrics—Median Error, Accuracy within  $\pi/6$ , Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and  $R^2$ . In contrast, Table 4.2 exclusively compares the proposed methodologies on the PASCAL3D+ category-specific viewpoint estimation for cars with several state-of-the-art methods using the two metrics that are widely reported in existing literature.

Table 4.1: Comprehensive Performance Metrics for Viewpoint Estimation Methods

Approach	MAE	RMSE	$R^2$	$Acc_{\pi/6}$	MedErr
Sin-Cos	7.3	14.8	0.95	0.97	3.5
Directional Discriminators	7.2	14.5	0.95	0.97	3.4

- Comprehensive Performance Assessment:** Table 4.1 showcases the full breadth of performance metrics for both of the described methodologies. The Directional Discriminators approach demonstrates a slightly superior performance with an MAE of **7.2**, RMSE of **14.5**, and  $R^2$  of **0.95**. In comparison, the Sin-Cos representation achieves an MAE of **7.3**, RMSE of **14.8**, and an equivalent  $R^2$  score of **0.95**.
- Benchmark Achievement:** Both of the presented methodologies—the Sin-Cos representation and the Directional Discriminators approach — surpass all the prior methods documented. Remarkably, both of the described methods reach an  $Acc_{\pi/6}$  score of **0.97**, which stands as the top performance among the evaluated techniques. Further emphasizing the accuracy of the proposed methods, the MedErr metric—which gauges the median error—registers its lowest values for the discussed approaches. The Directional Discriminators leads with a MedErr of **3.4**, closely followed by the Sin-Cos representation at **3.5**.
- Intra-comparison of the Two Approaches:** A side-by-side examination of the two techniques reveals closely aligned results. The Directional Discriminators slightly outperforms the Sin-Cos representation in terms of MedErr. Nonetheless, the difference is a mere 0.1, which, in practical applications, might fall within an acceptable margin of error. This tight competition underscores the robustness and reliability of both approaches.
- Residuals Analysis:** One powerful diagnostic tool to assess the accuracy and reliability of the viewpoint prediction model is to inspect the distribution of residuals — the differences between the observed orientations and their predicted values. Mathematically, for a given true orientation  $\varphi$  and its predicted orientation  $\hat{\varphi}$ , the residual  $r$  is given by:

$$r = \varphi - \hat{\varphi} \tag{4.19}$$



The histogram of residuals for the Directional Discriminators, shown on Table 4.11 approach reveals a compellingly centered distribution around 0, indicating a generally accurate prediction by the model.

However, the presence of non-zero residuals in extreme intervals such as  $r < -150^\circ$  and  $r > 150^\circ$  signifies occasional outlier predictions. These outliers emphasize that, despite the model’s overall strong performance, there remains room for further refinement. Such sporadic, significantly erroneous predictions underscore the need for ongoing research to perfect the model and minimize these anomalies.

Table 4.2: Results on PASCAL3D+ category-specific viewpoint estimation (car).  $Acc_{\pi/6}$  measures accuracy (the higher the better) and MedErr measures error (the lower the better)

Method	$Acc_{\pi/6}$	MedError
Prokudin et al. [92]	0.91	4.5
Su et al. [83]	0.88	6.0
Mousavian et al. [79]	0.90	5.8
Pavlakos et al. [80]	-	5.5
Grabner et al. [84]	0.94	5.1
3DPoseLite [93]	0.92	-
Xiao et al. [82]	0.91	5.0
Klee et al. [94]	-	4.9
Nie et al. [85]	0.92	5.1
Mahendran et al. [86]	0.95	4.5
Ours (Sin-Cos)	0.97	3.5
Ours (Directional Discriminators)	<b>0.97</b>	<b>3.4</b>

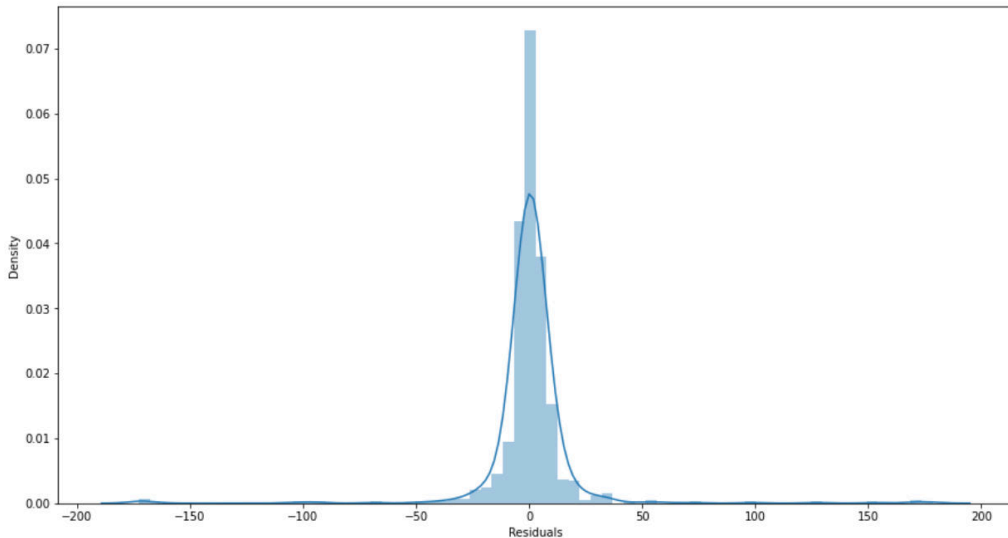


Figure 4.11: Residuals distribution for the Directional Discriminators approach

To conclude this subsection, the quantitative data reinforces the efficacy of the proposed methods, making them promising candidates for precise viewpoint estimation tasks.

### 4.4.2 Qualitative Results

**PASCAL3D+ Validation Set:** Figure 4.12 presents a  $5 \times 5$  grid showcasing predictions made on the validation set of the PASCAL3D+ dataset. Each image in this grid is accompanied by an azimuth diagram situated at the right top corner, in which the predicted azimuth is marked with a red line while the ground truth is indicated by a green line. A closer inspection of the images reveals the striking proximity between the predicted and actual orientations across the majority of samples, highlighting the model’s effectiveness.

However, it is essential to recognize instances like the sample in the second row and third column where the divergence between the prediction and the ground truth is nearly  $30^\circ$ . Contrary to initial impressions, this deviation does not necessarily reflect an inaccuracy in the model. Upon closer inspection, it becomes evident that the ground truth provided for this particular image does not align seamlessly with the actual orientation of the car, hinting at occasional noise and inconsistencies in the PASCAL3D+ dataset. Such observations underline the importance of maintaining a critical approach when evaluating predictions, especially in the context of potentially noisy datasets.

**Internet-sourced Images:** The versatility and generalizability of the proposed model are further demonstrated in Figure 4.13. This figure showcases a  $5 \times 5$  grid of car images sourced from the internet, beyond the boundaries of the PASCAL3D+ dataset. As these images come without any associated ground truth, only the predicted azimuth, denoted by a red line, is illustrated on the azimuth diagrams. Notably, even in the absence of ground truth for comparison, the predictions appear highly plausible, resonating well with the visual orientations of the cars.

An intriguing observation from this set is the image located in the first column and fourth row, where a car is obscured by a car cover. Despite this blanket obscuring the intricate details and distinctive features of the vehicle, the model still manages to deduce the azimuth quite accurately. This exemplifies the model’s ability to generalize and make predictions based on broad contextual cues, even when faced with unconventional scenarios.

**Model Interpretability and Utility:** Visual results, as presented in the aforementioned figures, are vital for offering an intuitive sense of model performance. They not only establish confidence in the model’s quantitative metrics but also showcase its utility in real-world, diverse scenarios. Moreover, such qualitative results facilitate potential troubleshooting and refinement strategies by revealing situations where the model might underperform or when external factors, like dataset noise, come into play.

## 4.5 Discussion and Possible Improvements

**Discussion:** The study has presented two approaches for car azimuth estimation, leveraging both the sinusoidal properties of orientations and the concept of directional discriminators. The results, both quantitative and qualitative, demonstrate the prowess of the proposed methods in achieving state-of-the-art performance on

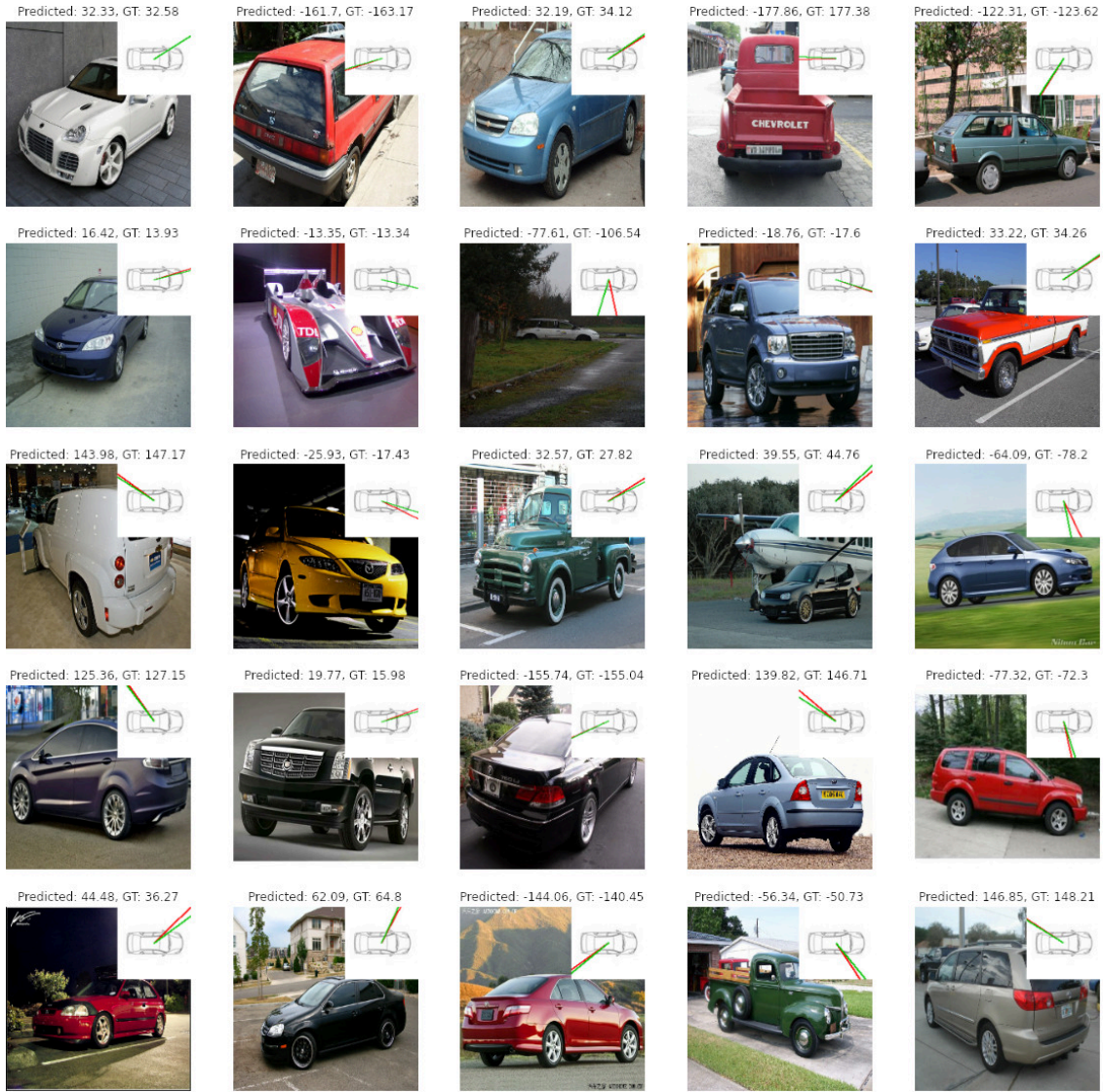


Figure 4.12: Sample predictions on the PASCAL3D+ validation set. The red and green lines on the azimuth diagrams correspond to predicted and ground truth azimuths, respectively.

the PASCAL3D+ dataset. One key observation is the minimal difference in performance between the two proposed methods, which suggests that while both methods have their individual merits, their practical outputs can be very similar under certain conditions.

Furthermore, while the proposed model demonstrates robustness in handling edge cases, like predicting the orientation of a car covered with a car cover, it is essential to underline the inherent noise in datasets like PASCAL3D+. Such noise may impact the absolute measures of performance, and it is crucial for future researchers to consider this when comparing against benchmarks.

**Possible Improvements:** While the proposed model achieves commendable results, there is always room for improvements and refinements:

- **Data Augmentation Expansion:** Further exploring diverse and more ag-



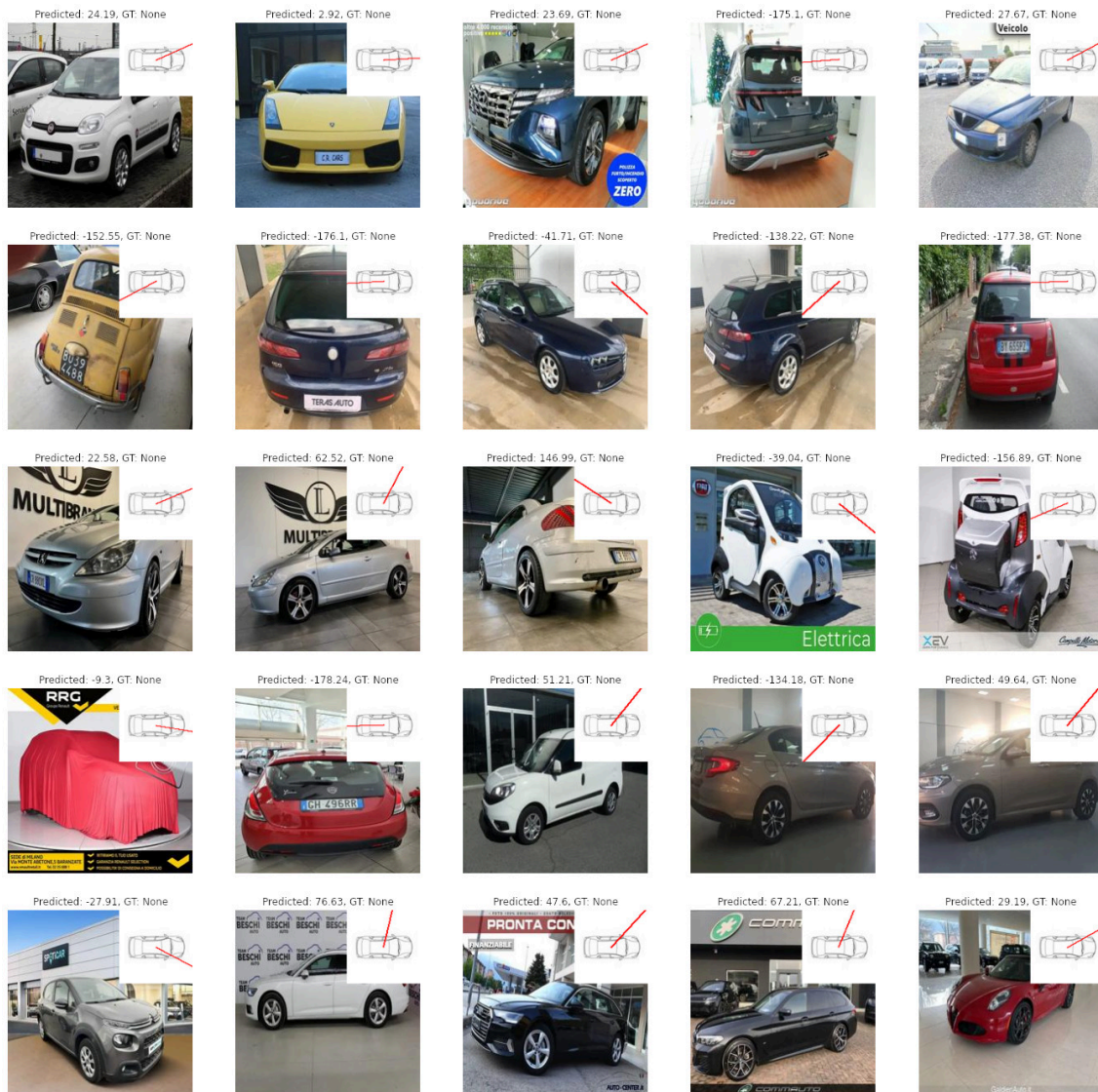


Figure 4.13: Sample predictions on car images sourced from the internet. Only the predicted azimuth (red line) is depicted due to the absence of ground truth.

gressive data augmentation techniques might make the model even more robust, especially for real-world scenarios.

- **Model Ensembling:** Combining predictions from multiple models or iterations could potentially improve accuracy and reduce outlier predictions.
- **Fine-tuning on Noisy Datasets:** Given the noted noise in the PASCAL3D+ dataset, fine-tuning the model on a manually curated, noise-free subset might enhance performance metrics.
- **Exploring Alternative Network Architectures:** While EfficientNetB0 serves as a strong backbone, there is potential in experimenting with newer architectures or custom-tailored network designs for this specific task.

In summary, the pursuit of optimal viewpoint estimation remains a continuing endeavor. Although significant progress has been made, numerous avenues remain

unexplored. Subsequent research may leverage the established foundation, introducing advancements and refinements to further expand the realm of possibilities.

**Part II**

**Vehicle Damage Analysis**

# Chapter 5

## Component Recognition and Damage Presence Analysis

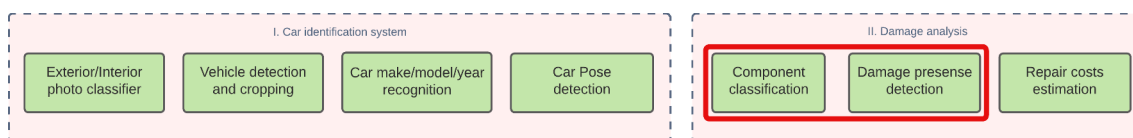


Figure 5.1: Highlighted segment of the whole system representing the focus of this chapter.

Vehicular damage assessment has traditionally relied on manual evaluations by trained professionals, leading to potential inconsistencies and inefficiencies. As outlined in Chapter 1, the work undertaken in this research moves towards automation of this process. The automated system, as established in preceding chapters, spans from vehicle identification to pose detection. The current chapter delves into the crucial phase of damage assessment, specifically focusing on the recognition of vehicle components and the identification of damage presence. Accurate component recognition is not only fundamental for the subsequent chapter, which discusses estimating repair costs but also sets the stage for the immediate task of detecting the presence of damage, without delving into the severity of the damage. A visual representation of the entire system’s pipeline, highlighting the current focus on component classification and damage presence detection, is provided in Figure 5.1.

In detailing the sophisticated methodologies adopted, the unique dataset curated for this research is introduced, accompanied by comprehensive performance metrics that evaluate the models’ success. The influence of integrating vehicle pose information into the component recognition model is also investigated, contributing to a more nuanced understanding of model behavior under varying inputs. Further, the discussion explores the intricacies involved in developing binary classifiers for identifying damage presence, highlighting the challenges and the innovative solutions devised to enhance the accuracy of these systems.

### 5.1 Damage Dataset Description

The assessment of vehicular damages, particularly from photographs, remains a crucial yet intricate task in automotive industries. While the significance of this task



is undoubted, a surprising observation is the conspicuous absence of comprehensive public datasets dedicated to car damage identification and evaluation. Numerous studies have emphasized the scarcity of such datasets, compelling researchers to resort to proprietary or self-compiled datasets. This trend is evident in works such as [1, 7, 2, 3, 6, 11, 15, 16], among others. In the light of this gap, the expertise documents curated by *brumbrum*, assume paramount importance. Sourced from two external damage assessment companies, these documents provide a rich repository of car damages, annotated with a granularity not commonly found in many datasets.

### 5.1.1 Structure of Expertise Documents

Expertise documents from external damage assessment companies present a structured and hierarchical layout designed for detailed vehicular damage documentation. Each document enumerates the damages identified and for each entry it provides:

1. The suggested *Operation* for the damage, indicating either “Acceptable” for damages where no repair is required due to insignificant or almost unnoticeable damage appearance, “Repair”, or “Replacement”.
2. A series of photographs capturing the damage from various distances and angles, typically ranging from 2 to 5 images, to provide a comprehensive view of the inflicted damage.

Figure 5.2 displays sample photographs from an expertise document, emphasizing the diversity and detail of the visual documentation.

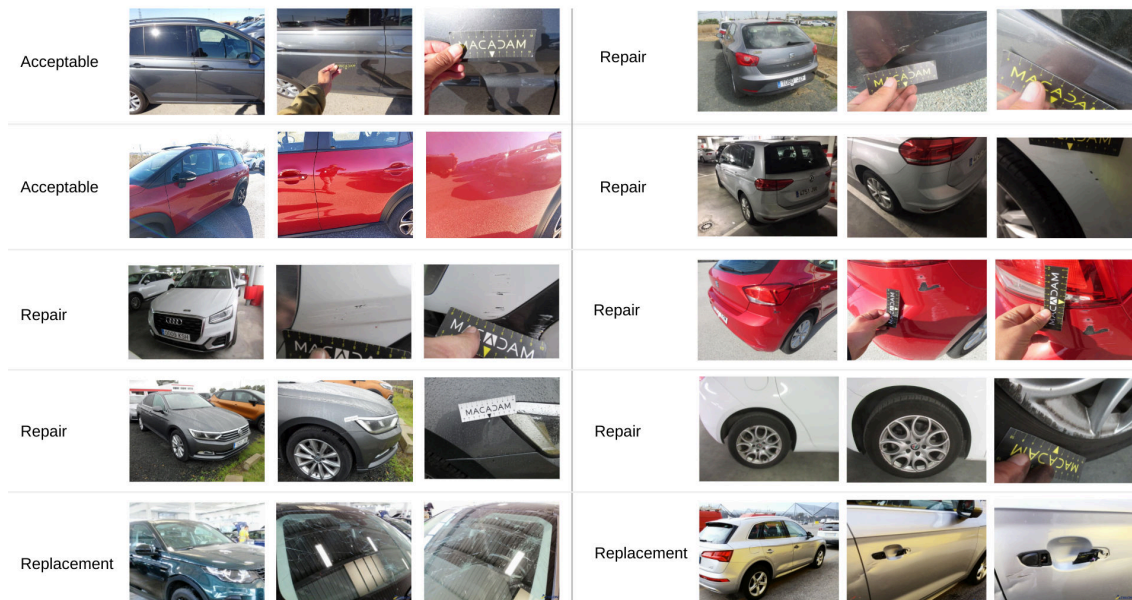


Figure 5.2: Samples of damages from expertise documents. Each sample displays an associated operation (either “Acceptable”, “Repair”, or “Replacement”) accompanied by three photographs showcasing the same damage from different distances and angles, ensuring a clear representation of the damage’s location.

Such structured documentation not only ensures uniformity across different reports but also sets the groundwork for subsequent data extraction, processing, and analysis.

### 5.1.2 Component Classification Dataset Creation

The pivotal first step in the dataset creation was the selection of car components. Collaboration with automotive experts facilitated the identification of 16 exterior car components deemed most common and relevant for this study. This strategic choice was based on their prevalence in repair scenarios and their significance in the damage assessment process. Each selected component represented a class for which images were exhaustively gathered and annotated. Additionally, a “reject” class was introduced to encompass scenarios where components were either unrecognizable or photographs were taken at an extremely close range, making component identification unfeasible. A detailed enumeration of the classes is presented in Table 5.1.

Class Index	Component Name
1	Side Mirror
2	Rim
3	Hood
4	Tail Light
5	Headlight
6	Grille
7	Rear Window
8	Handle
9	Molding
10	Windshield
11	Fender
12	Bumper
13	Door
14	Tailgate / Trunk Lid
15	Sill
16	Roof
17	Reject Class

Table 5.1: List of identified car components used for classification, including a specific “Reject Class” for unidentifiable instances.

A substantial effort was invested in curating a balanced dataset, entailing the acquisition of 600 images for each of the 17 classes, summing up to 10,200 images in total. These images were extracted from the original pool of expertise documents, ensuring a diverse representation of damages, angles, and lighting conditions. Such diversity was crucial for preparing the model for real-world scenarios where such variables are unpredictable. Figure 5.3 showcases samples from the dataset, illustrating the variety of images and the representation of each component class.

CHAPTER 5. COMPONENT RECOGNITION AND DAMAGE PRESENCE ANALYSIS

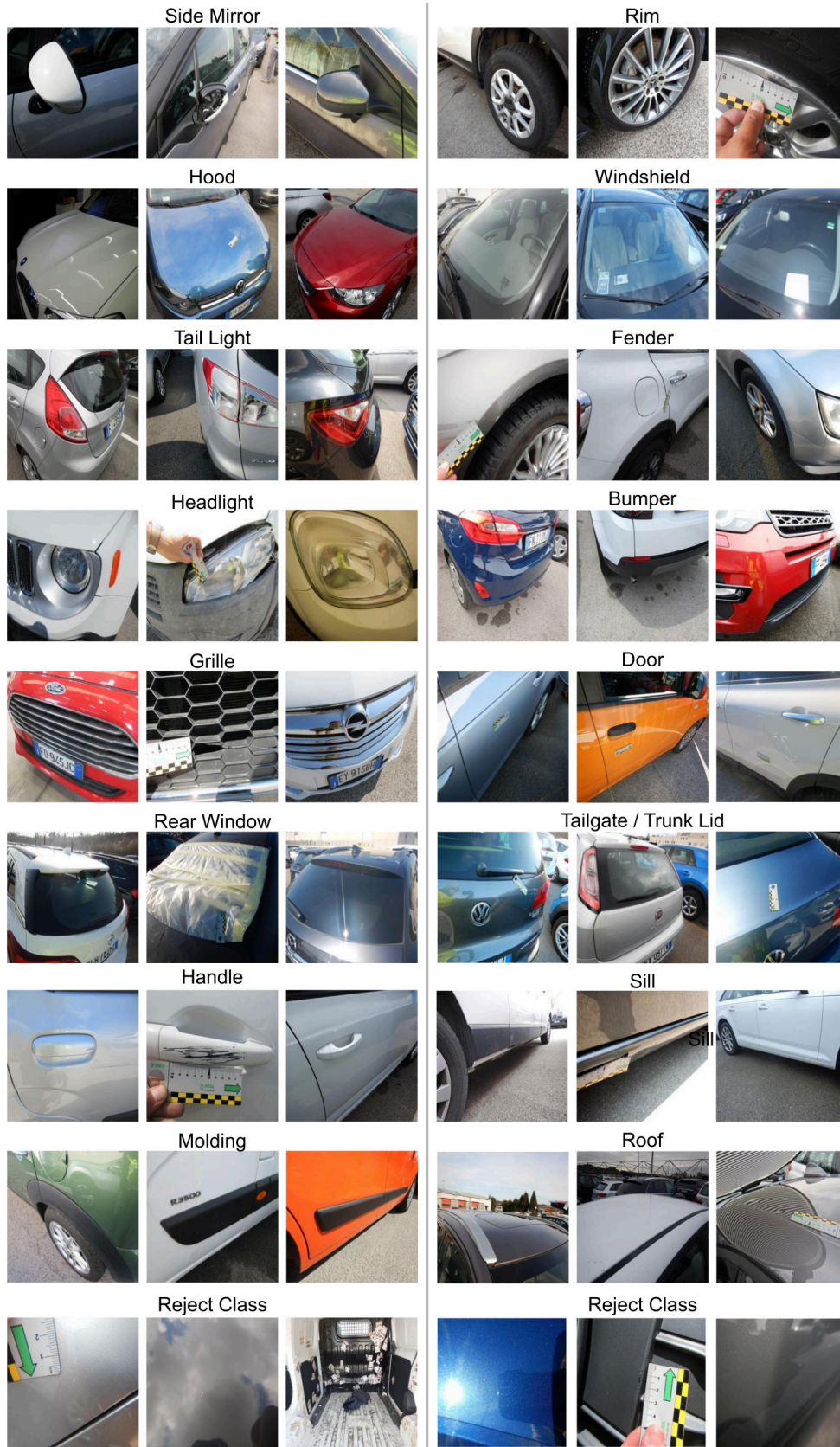


Figure 5.3: Representative images for each of the 16 component categories: three sample images per component with the final row dedicated to the “Reject” class, illustrated with six images.

## 5.2 Component Recognition

### 5.2.1 Introduction and Related Works

The task of understanding and analyzing the damages in vehicles, particularly through the lens of digital images, is of increasing significance in the automotive and insurance sectors. Automating such tasks not only reduces manual effort but also holds potential for increasing accuracy and speed in assessment, which can streamline the overall claim process for insurance companies.

Environmental understanding and the semantic analysis of vehicle components are emerging as prime research areas, especially with the rapid advancements in deep learning techniques. For instance, Jurado et al. [95] delved into the realm of 3D semantic segmentation, focusing on cars. By harnessing photogrammetric processing of UAV-based imagery, they proposed an automatic procedure for the segmentation of 3D car models, achieving noteworthy results in identifying sixteen distinct car parts. The integration of the U-Net [96] architecture and Inception V3 [97] encoder showcases the effectiveness of modern convolutional neural networks (CNNs) in tasks traditionally approached by earlier computer vision techniques.

However, the majority of contemporary works lean more towards 2D image analysis. Chua et al. [98] developed a two-tier machine learning system focused on car parts' identification and subsequent damage detection. Using CNNs, their research yielded an impressive accuracy of over 94% in car parts classification, underscoring the potential of deep learning in this field.

In another extensive study, Pasupa et al. [18] embarked on evaluating and comparing five distinct deep learning algorithms for the semantic segmentation of car parts. Despite their focus on segmentation, their insights into the resilience and adaptability of these models, especially under varying environmental conditions, hint at their utility in a wide range of applications, including classification.

Another relevant study by Khanal et al. [99] sheds light on the criticality of automatic car part recognition, especially in contexts like quality inspection and auto-assembly. Utilizing the VGG-16 [100] deep learning architecture, they managed to attain an average accuracy nearing 94% in recognizing eight different car parts, reinforcing the notion that CNNs are aptly suited for such classification challenges. It should be noted that the data used for this study was based on their own proprietary dataset.

Lastly, the work by Dwivedi et al. [101] explicitly targeted the problem of vehicle damage classification, which is closely aligned with the overarching objective of this study. Utilizing CNN models that were pretrained on the ImageNet dataset, they reported an impressive accuracy rate of over 96% in vehicle damage recognition. Importantly, the dataset utilized in their research was also proprietary.

Collectively, these works underscore the applicability and success of deep convolutional neural networks in vehicle component analysis, whether it is in the realm of segmentation or classification. The repeated success of DCNNs in various studies, spanning different car parts and damage types, consolidates the belief that DCNN-based classification is a promising and forward-thinking approach for the automated assessment of car damages. However, it is crucial to highlight that the success achieved in these studies was largely reliant on proprietary datasets, underscoring the significance of dataset quality and specificity in the realm of car damage assessment.



### 5.2.2 Selection of DCNN Architecture

The selection of an appropriate DCNN (Deep Convolutional Neural Network) architecture is a critical decision, influenced primarily by the internal requirements of the system, particularly in terms of computational efficiency. While the task involves classifying images into one of seventeen distinct categories (sixteen representing various vehicle components and one as “Reject”), the complexity of the task is not the only factor guiding the choice. More imperative is that the model operates within the acceptable boundaries of computational resources, ensuring it is sustainable and efficient in processing large volumes of data without compromising on performance accuracy.

In section 3.1.3, a comparative study was conducted to weigh the merits of four lean DCNN architectures. Each was evaluated based on its computational complexity and benchmark accuracy, underscoring the need for a model that is both resource-efficient and effective. The culmination of this analysis pointed to MobileNetV2 [39] as the most fitting candidate. Notably, MobileNetV2 has carved a niche for itself in the machine learning community, known for its smaller size and deft balance between speed and accuracy. Its design employs compound model scaling, a principle that uniformly scales all dimensions of depth, width, and image resolution, contributing to its efficiency.

### 5.2.3 Integration of Vehicle Pose Information

The integration of vehicle pose into the component recognition task hypothesizes an enhancement in the model’s ability to accurately classify car components. The pose of a vehicle provides contextual information, potentially aiding the differentiation between components, especially when they are captured from similar angles. However, the hypothesis also acknowledges the inherent challenges posed by close-view photographs, where pose estimation may not only be irrelevant but could lead to incorrect model inferences.

#### Hypothetical Benefits and Limitations of Pose Integration

Integrating vehicle pose data is premised on the assumption that additional spatial information can refine the deep convolutional neural network’s (DCNN) decision-making criteria. For photographs capturing the vehicle from a moderate distance, indicating the orientation could help discern features that are otherwise non-distinctive. Conversely, for close-view photographs, the pose information becomes less reliable, even detrimental, as the limited field of view restricts accurate pose estimation. It is postulated that the DCNN would learn to selectively weigh this input, diminishing its influence when it contradicts the primary visual data.

#### Custom DCNN Architecture for Component Recognition with Pose Estimation

The proposed DCNN architecture combines feature extraction from images with auxiliary input from vehicle pose estimation. As illustrated in Figure 5.4, the architecture extends the pre-existing MobileNetV2 model, known for its efficiency and accuracy in mobile vision applications.

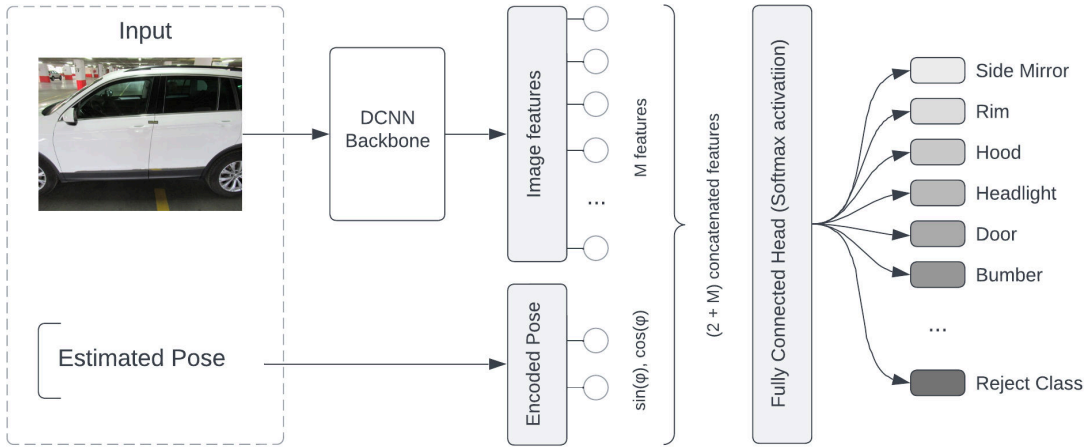


Figure 5.4: Enhanced DCNN Architecture Incorporating Vehicle Pose Estimation: This diagram depicts the sophisticated architecture of the proposed deep convolutional neural network (DCNN), emphasizing the integration of vehicle pose information (sine and cosine values of the azimuth angle) with traditional image descriptors extracted from the MobileNetV2’s penultimate layer.

The procedure begins with the utilization of MobileNetV2 up to its penultimate layer, harnessing the global average pooling layer with 1280 features as a comprehensive image descriptor. This layer effectively captures the essential characteristics present in the image, providing a foundation for the subsequent layers to process. Subsequently, two additional features, representing the sine and cosine values of the vehicle’s azimuth angle, are concatenated with these image descriptors. This augmentation embeds the contextual information provided by the vehicle’s pose directly into the feature set used for component classification.

The combined feature set, now enriched with spatial context, proceeds to a final softmax layer, which plays a crucial role in classifying the input into one of the seventeen possible classes. These classes include sixteen representing distinct vehicle components and one dedicated to unidentifiable instances (the “Reject” class).

In training the model, categorical cross-entropy loss is employed as the loss function, a standard choice for classification problems with multiple classes. This loss function quantifies the disparity between the predicted probability distribution across the classes and the true distribution, thereby guiding the optimization of the model’s parameters.

### 5.2.4 Training and Performance Analysis

The dataset, comprising 10,200 images, was partitioned into training and testing sets, maintaining an 80/20 split. This division resulted in 8,147 images for training and 2,053 for testing, with allocations based on unique damage identifiers to obviate data leakage and ensure model generalizability.

To enhance the model’s robustness to visual variances in real-world scenarios, data augmentation techniques were employed during training. These techniques included Horizontal Flip, Rotation (up to 10 degrees), Optical Distortion, and Random Brightness Contrast adjustments, thereby introducing controlled variability and reducing the potential for overfitting. The training process, facilitated by the

computational power of an Nvidia Tesla T4 GPU, was optimized to handle complex tasks and large volumes of data, ensuring an efficient training phase with reliable performance metrics.

### Training

The model was trained using the Adam optimizer with a learning rate set at  $1 \times 10^{-5}$ , and a batch size comprising 64 images per iteration. The MobileNetV2 component of the architecture was initialized with pretrained weights derived from ImageNet benchmark training. These weights were not frozen, but instead, they were updated throughout the training. Over 48 epochs, the model’s performance was diligently scrutinized, with a notable lowest validation loss of 0.525 achieved during the 38th epoch. Correspondingly, the validation accuracy peaked at 0.838, indicative of the model’s learning efficacy. These dynamics are visually represented in the training progress curves depicted in Figure 5.5.

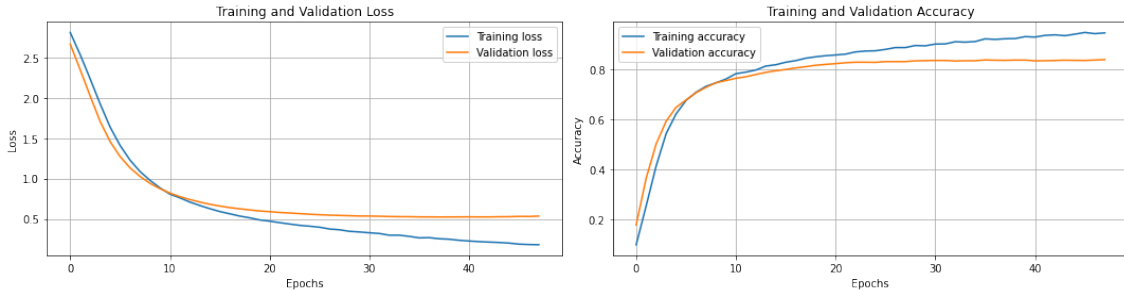


Figure 5.5: The evolution of the model’s performance over 48 epochs of training. The graph delineates the trajectories of both training/validation loss (on the left) and accuracy (on the right)

### Performance Evaluation

Post-training, the model’s competence in accurately classifying vehicle components was evaluated using a confusion matrix and a detailed classification report. The confusion matrix, illustrated in Figure 5.6, revealed a notable challenge in the accurate classification of the “Molding” class, with frequent misclassifications occurring with similar components such as “Sill” and “Bumper”. This ambiguity underscores the complexity of distinguishing finely nuanced features.

Further, the classification report in Table 5.2 quantified these observations: the “Molding” class demonstrated a recall of 0.51 and an F1-score of 0.57, the lowest across all categories. Additionally, the “Reject” class exhibited subdued performance, potentially attributable to the inherent ambiguity in classifying indistinct or visually obfuscated components, emphasizing the subjective challenge that even human experts might encounter.

### Qualitative Assessment

To complement the quantitative analysis, a qualitative assessment was conducted to visually interpret the model’s performance, particularly focusing on the misclassifications within the “Reject”, “Molding”, and “Bumper” classes. These categories



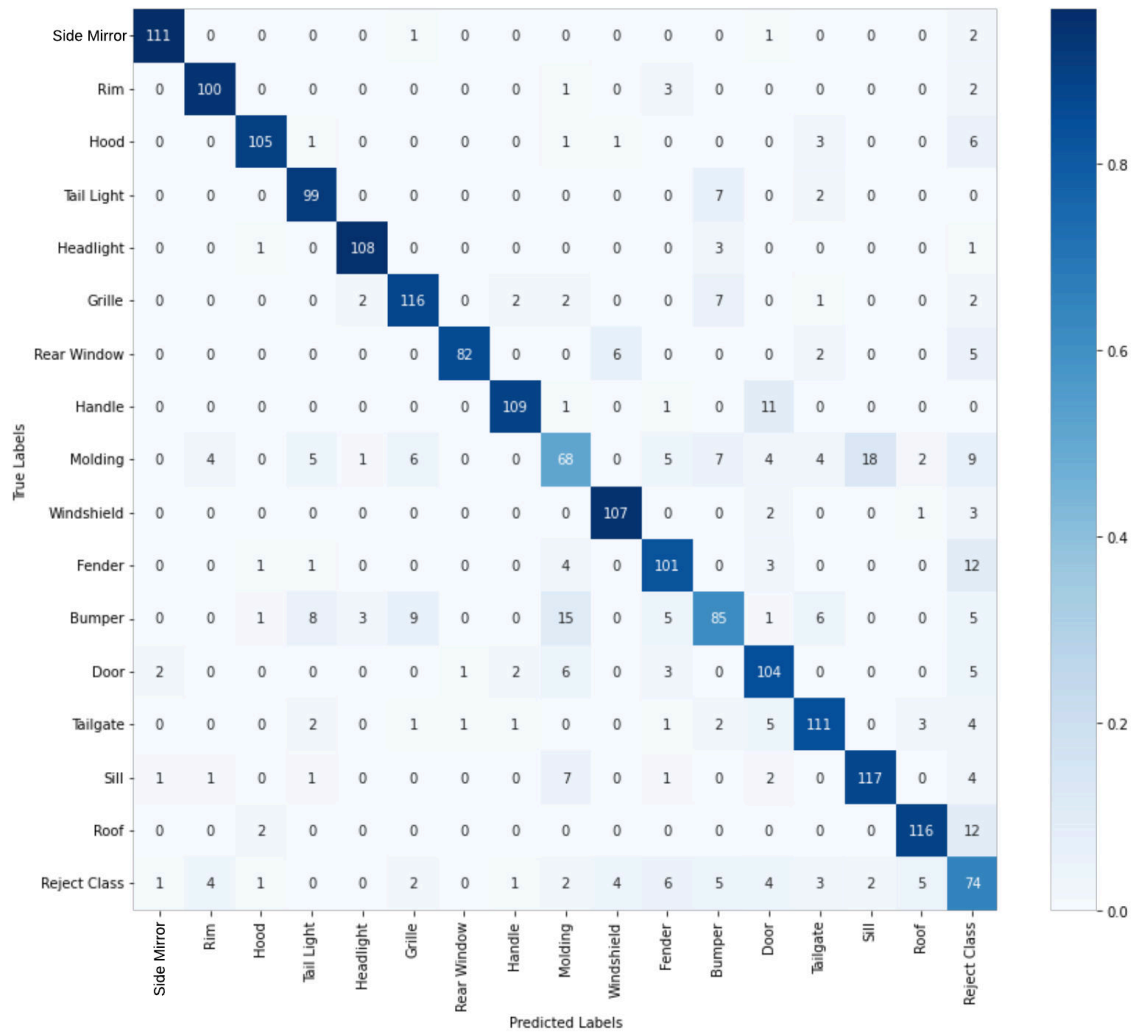


Figure 5.6: Confusion matrix derived from the model’s predictions on the validation set. The figure highlights the pronounced misclassification challenges encountered with the “Molding” category, often confused with “Sill” and “Bumper”

Table 5.2: Classification report showcasing the precision, recall, and F1-score for each of the identified vehicle components. The results highlight the varying degrees of model proficiency in recognizing specific components, with particular challenges encountered in accurately classifying the “Molding” and “Reject” classes.

Class	Precision	Recall	F1-score
Side Mirror	0.97	0.97	0.97
Rim	0.92	0.94	0.93
Hood	0.95	0.90	0.92
Tail Light	0.85	0.92	0.88
Headlight	0.95	0.96	0.95
Grille	0.86	0.88	0.87
Rear Window	0.98	0.86	0.92
Handle	0.95	0.89	0.92
Molding	0.64	0.51	0.57
Windshield	0.91	0.95	0.93
Fender	0.80	0.83	0.81
Bumper	0.73	0.62	0.67
Door	0.76	0.85	0.80
Tailgate	0.84	0.85	0.84
Sill	0.85	0.87	0.86
Roof	0.91	0.89	0.90
Reject Class	0.51	0.65	0.57

were specifically selected for scrutiny due to their underwhelming performance in the previous evaluations.

Figure 5.7 exhibits a series of misclassified samples for each identified problematic class. Intriguingly, the images presented often appear to be more congruent with the model’s predicted class than their true labels. This discrepancy suggests the presence of inherent ambiguities within the dataset itself, emphasizing the subjective nature of certain classifications even for human perception.

For instance, several “Reject” instances were misinterpreted as valid components, likely because the damages or perspectives shown in the photographs resembled features typically associated with certain classes. This observation is particularly salient for close-up shots or images lacking clear contextual indicators, complicating the classification task. Similarly, “Molding” and “Bumper” misclassifications often involved components with overlapping or subtly distinct features.

These insights from the qualitative assessment reinforce the necessity for a robust, well-curated dataset where class definitions are unambiguous and consistently applicable. They also highlight the potential need for an enhanced training phase, possibly incorporating human expertise for validation, to mitigate subjective biases and improve the model’s interpretative accuracy.

### 5.2.5 Ablation Study: Role of Pose Estimation

A crucial aspect of assessing the effectiveness of integrated pose information within the component recognition model involves conducting an ablation study. This study removes the pose data, considered a potential confounding variable, to analyze its impact on the model’s performance metrics. Such an approach helps determine whether the pose data provides meaningful improvements or merely introduces noise into the system.

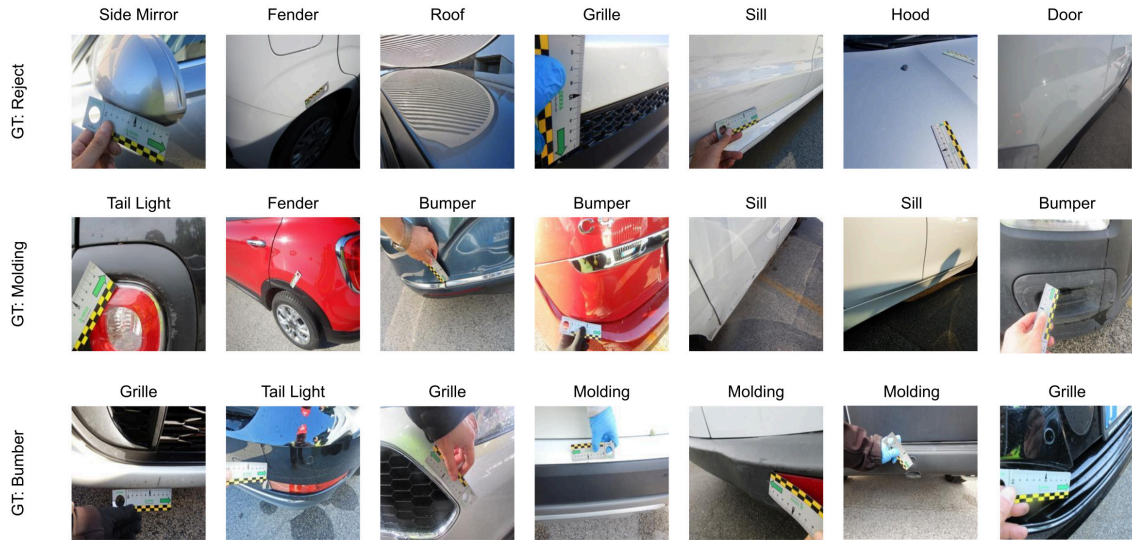


Figure 5.7: Misclassification examples for the “Reject”, “Molding”, and “Bumper” classes, each row dedicated to one category. Images depict cases where the model’s predictions, indicated above each sample, seem more characteristic of classes other than the true labels.

The initial hypothesis posited that including vehicle pose information would enhance the model’s ability to identify components, grounded in the presumption that spatial orientation aids in distinguishing similar features. However, the reliability of this assumption is challenged by the conditions under which the pose estimator was trained. The training dataset predominantly consisted of images where vehicles are not only fully visible but also devoid of any damage. This specificity raises concerns about the estimator’s applicability and accuracy when confronted with images that diverge from this standard, such as close-up shots or visuals of damaged components, conditions frequently encountered in real-world scenarios of vehicle assessment.

To assess the validity of incorporating pose estimation, an ablation experiment was conducted wherein the model was retrained in the absence of pose data. This process ensured that any variation in performance metrics could be attributed directly to the presence (or absence) of pose information. The comparative analysis focused on four critical metrics: accuracy, weighted recall, weighted precision, and weighted F1-score. As summarized in Table 5.3, the ablation study revealed a marginal degradation in performance when pose information was omitted.

Model	Accuracy	Recall	Precision	F1-Score
With Pose	0.838	0.838	0.839	0.837
Without Pose	0.829	0.829	0.831	0.829

Table 5.3: Performance comparison of the component recognition model with and without pose information.

The negligible decline in these metrics suggests that while the pose data contributes to the model’s decision-making process, its role is not critically significant. This underwhelming contribution points towards the potential inaccuracy or irrelevance of pose information when dealing with close-up views or damaged areas, situations where the pose estimator’s predictions may inherently be less reliable.

The findings from the ablation study indicate that the current model’s utility in leveraging pose information is limited, suggesting avenues for future enhancement. One prospective improvement involves adapting the pose estimator to the specific domain of close-view photographs and those depicting damages.

### 5.2.6 Critical Evaluation and Possible Improvements

The qualitative and quantitative analyses indicate that one of the primary factors contributing to misclassification is the inherent ambiguity in the images themselves. Instances where components are visually obfuscated, damaged, or presented in close-up shots without clear contextual features often lead to errors. Furthermore, the “Molding” and “Bumper” classes proved particularly challenging due to their nuanced and subtle distinctiveness, which is sometimes even subjective to human interpretation.

The ablation study provided additional insights, particularly concerning the role of pose estimation data. While initial assumptions suggested that this information would significantly enhance the model’s ability to recognize components accurately, the results indicated a marginal contribution. The exclusion of pose information led to only a slight decrease in performance, suggesting that the pose estimator’s current state may not be entirely suitable for this application domain.

Given these insights, several improvements are recommended for future iterations of the model:

- **Dataset Refinement:** To address ambiguities in component appearance, especially for categories like “Molding” and “Bumper”, the dataset requires further curation. This refinement involves a more meticulous image selection and labeling process, ensuring that each category is consistently and unambiguously represented. This strategy may also include the incorporation of additional features or annotations to provide clearer indicators for accurate classification.
- **Pose Estimator Adaptation:** The limited impact of pose information on the model’s performance suggests a need for a more tailored pose estimator. Future work should explore training the estimator on a dataset that more closely aligns with the target application, including images of damaged vehicles and close-up shots of components. Enhancing the estimator’s accuracy in these scenarios could significantly improve the overall model’s performance.
- **Advanced Feature Recognition:** Implementing more sophisticated feature extraction and recognition techniques could further refine the model’s classification capabilities. Deep learning advancements, such as attention mechanisms [102] or region-based convolutional neural networks (R-CNNs) [49], could offer more nuanced understandings of complex visual contexts, potentially improving classification accuracy in challenging scenarios.
- **Human-in-the-loop Validation:** To mitigate the subjectivity in class definitions and model interpretations, a human-in-the-loop approach could be beneficial. This process would involve human experts reviewing and correcting the model’s predictions during or post-training, helping to refine the algorithm’s decision-making criteria and address potential biases or misinterpretations.

In conclusion, while the current model establishes a foundation for vehicle component classification, these recommended enhancements aim to address the identified shortcomings. Through continued refinement, the model can achieve higher accuracy levels, making it an even more valuable tool for automated vehicle inspection and assessment systems.

## 5.3 Damage Presence Detection

In the realm of automated vehicle damage assessment, the accurate identification of damage presence represents a critical juncture in the analysis pipeline. Building upon the foundational modules delineated in the preceding chapters, which proficiently extract detailed information regarding the vehicle's make, model, and year, the focus now shifts to a nuanced aspect of the assessment: discerning the presence of damage itself. This process marks a significant transition from preliminary vehicle identification and component classification, covered in earlier discussions, to a stage where precision is paramount in recognizing and categorizing actual physical damage evident in the vehicle images.

The following sections address distinct challenges in the interference of external objects and variability in damage manifestations across different vehicle components. The adopted methodology, refined to accommodate component-specific peculiarities, fortifies this crucial phase of the automated system.

### 5.3.1 Related Works

Recent studies have leveraged deep learning methodologies, particularly Convolutional Neural Networks (CNNs), for automating vehicle damage detection, a critical need in the automotive insurance industry. These approaches vary in sophistication and application, focusing on different aspects of the damage detection process.

Deep learning models with transfer learning have gained traction for their ability to efficiently classify car damages. Works by Dwivedi *et al.* [101], Kyu and Woraratpanya [19], Sruthy *et al.* [20], and Gandhi [103] utilize models pre-trained on extensive datasets like ImageNet, employing transfer learning for feature extraction and damage classification. Specifically, Dwivedi *et al.* [101] and Gandhi [103] extended their analysis to damage localization using the YOLO object detector.

Emphasis on hybrid approaches for enhancing classification accuracy is evident in studies such as Rio-Torto *et al.* [104] and Waqas *et al.* [105]. These works combine deep learning with other technological facets, like simulated data and authenticity verification through metadata analysis, to improve system reliability.

In terms of comparative analyses, Anwer *et al.* [106] and Chaudhari [107] experimented with various network architectures to identify superior models for damage detection tasks. Interestingly, Chaudhari [107] highlighted the effectiveness of ensemble learning and domain-specific pre-training in enhancing model performance.

Concerning damage segmentation, research by Dhieb *et al.* [108] proposed a combination of deep learning with instance segmentation for precise damage localization, emphasizing the system's potential in mitigating claims leakage in insurance.

Lastly, the integration of CNNs with traditional machine learning models showcases the evolving landscape of damage assessment methodologies. Tian and Han

[109] exemplified this by fusing deep learning feature extraction with logistic regression and support vector machines for efficient damage assessment.

The aforementioned works demonstrate the versatility of deep learning applications in vehicle damage detection. However, a noticeable trend is the prevalent use of DCNNs for classification tasks, providing a reliable foundation for further refinement and adoption in industry-specific applications.

### 5.3.2 Objective and Approach Overview

The paramount goal of this phase in the research is the development of robust binary classifiers capable of accurately identifying the presence of damage in various vehicle components. The classifiers are tailored for each specific component listed in Table 5.1, excluding the Reject class, thereby facilitating nuanced and detailed damage assessment.

Each classifier is forged with the intent to discern between two states: the absence of damage and the presence of damage, which necessitates repair or replacement. These states are extrapolated from the “Operation” field within the expertise documents. The “repair” and “replacement” categories are conglomerated into a singular class to signify the presence of damage, simplifying the classification into a binary system.

The analysis and subsequent classification leverage the dataset detailed in section 5.1. This rich compilation of images presents a lot of scenarios covering the spectrum of damage severities and component types. The use of real-world data, rife with typical inconsistencies and challenges, fortifies the classifiers’ ability to perform in practical, real-world scenarios.

### Component-Specific Strategies

Given the diverse nature of vehicle components and the damages they may incur in, a one-size-fits-all approach is eschewed in favor of component-specific strategies. Each component potentially presents unique damage manifestations, influencing both the visual cues available for classification and the consequent decision-making process. By training individual classifiers for each component, the system can cater to these nuances, thereby enhancing accuracy and reliability.

### 5.3.3 Dataset Description

The dataset comprises images of 16 distinct vehicle components, as itemized in table 5.1, with the explicit exclusion of the Reject Class. The rationale behind the selection of these specific components, rooted in their relevance and frequency of damage, has been expounded upon in section 5.1. Each component is represented by 600 images, capturing the component in various states and lighting conditions, thus offering a comprehensive view necessary for effective training.

For the sake of analytical clarity and methodological effectiveness, the images are divided into two primary classes: “Damage” and “No Damage”. The distribution of images between these classes, with approximately 59% in the “No Damage” category and 41% in the “Damage” category, might initially appear unaligned with real-world observations, where damages might be more prevalent. However, this distribution was intentionally chosen, as the “No Damage” class also encompasses acceptable or

barely perceptible damages. A detailed explanation regarding this classification can be found in section 5.1.1. The specifics of this distribution are visualized in Figure 5.8, shedding light on the dataset’s structure.

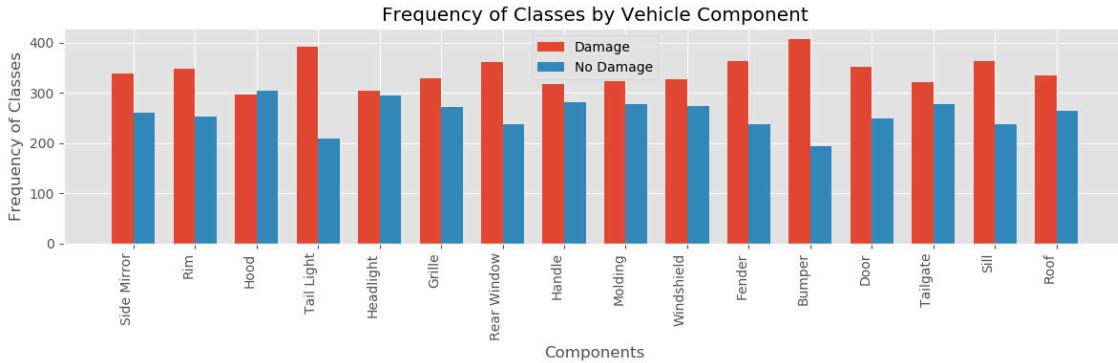


Figure 5.8: Frequency of classes by vehicle component. The figure illustrates the distribution of the “Damage” and “No Damage” classes across all 16 components, highlighting the variability in the prevalence of damage.

To elucidate the nature of the images used in this study, Figure 5.9 showcases a selection of the dataset. This visual representation includes two instances of damaged components and two instances of components with no damage, across eight different types of components. These examples underscore the variety in damage types and severities as well as the subtleties the model must discern in components deemed to have no damage.

### 5.3.4 Analyzing the Influence of External Objects on Damage Recognition

#### Introduction to the Problem

The integrity of image classification systems, particularly in contexts of vehicle damage assessment, is pivotal to their operational efficacy. One of the pertinent challenges encountered during the dataset compilation involves the frequent presence of external objects, such as rulers or hands, in the photographs of damages, presented in the Figure 5.10. These rulers are traditionally used by professionals to gauge the extent of damage, serving as a reference scale.

However, the inclusion of these objects raises a critical question: could their presence inadvertently influence the automatic detection of damage, thereby skewing the results? This concern stems from the possibility that the machine learning models might not solely abstract the characteristics of the damage, but also the ancillary elements present, such as the ruler. If not addressed, this could culminate in a model with high precision in scenarios where these objects are present but reduced accuracy otherwise, thereby compromising the generalizability and reliability of the system.

To safeguard against this potential source of bias and to validate the model’s ability to discern damage irrespective of these additional elements, it becomes imperative to conduct an analytical study. The use of the Chi-squared test for independence surfaces as a suitable method for this investigation. This statistical test



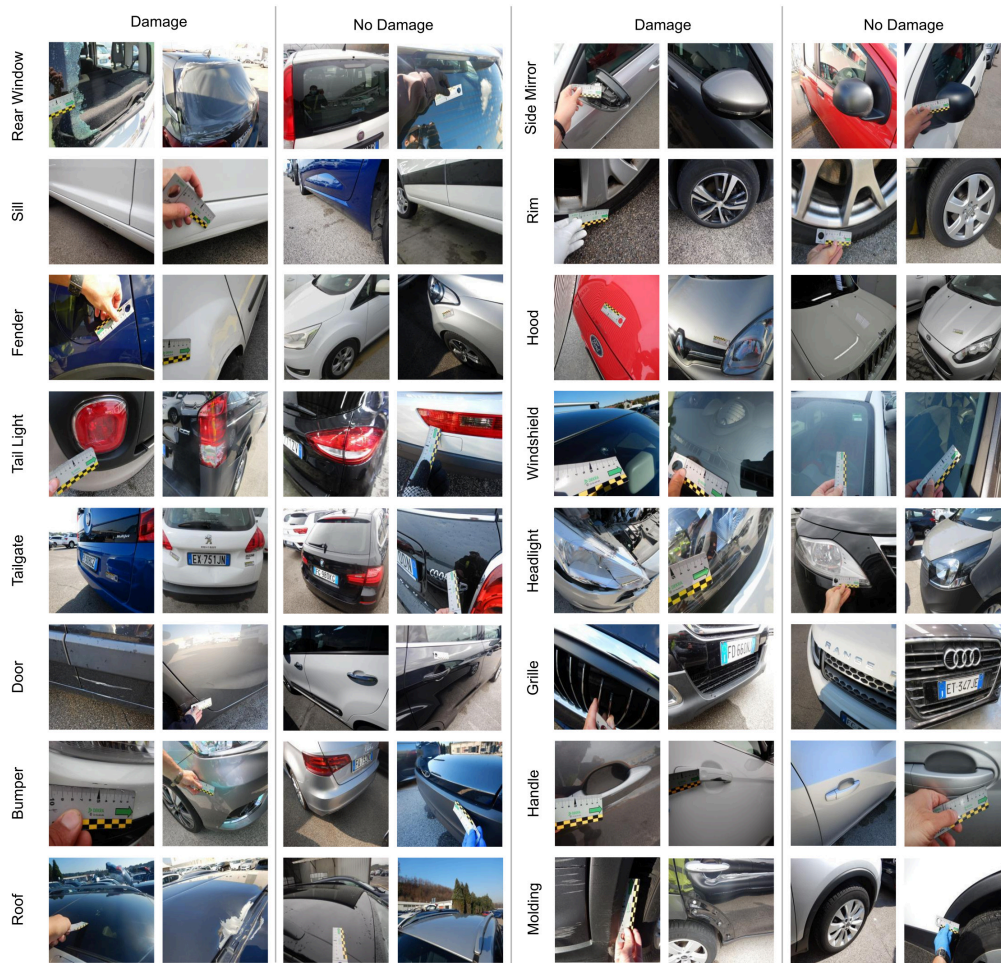


Figure 5.9: Examples of vehicle components from the dataset. Each row represents a different component, with the first two images in each row showing examples of “Damage” and the last two images demonstrating “No Damage” instances.

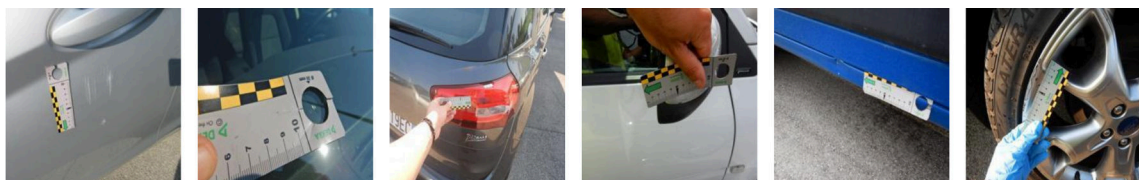


Figure 5.10: Sample images from the dataset illustrating the challenge posed by the inclusion of external objects, such as rulers, in the visual assessment of vehicle damages.



evaluates the likelihood that the recognition of damage is associated with the presence of these objects by examining the distribution of observed frequencies across categories defined by these two variables. A significant Chi-squared result would indicate a dependency, prompting a need for further analysis or model adjustment to ensure that damage recognition is attributed correctly to the damage itself, not confounded by external variables.

### Methodology for Independence Testing

An essential aspect of validating the robustness of damage detection models involves ensuring that the identification of damage is not unduly influenced by extraneous variables. The hypothesis under consideration was formulated as follows:

H0: There is no relationship between the presence of a ruler (or other external objects) and the identification of damage (i.e., the two factors are independent).

To test this hypothesis, the Chi-squared test for independence was employed. This test is widely utilized in statistical analyses to examine the independence of two categorical variables. The procedure involves the following steps:

1. **Construction of a Contingency Table:** The first step in the analysis is the construction of a contingency table for each component. A  $2 \times 2$  contingency table is created, which categorizes the observations into four outcomes based on two categorical variables: the presence or absence of damage, and the presence or absence of a ruler (or any external object).
2. **Calculation of Expected Frequencies:** Under the null hypothesis of independence between the two categorical variables, the expected frequency for each cell in the contingency table is calculated using the formula:

$$E_{ij} = \frac{(R_i \cdot C_j)}{N}, \quad (5.1)$$

where  $E_{ij}$  represents the expected frequency for cell  $(i, j)$ ,  $R_i$  is the total count of row  $i$ ,  $C_j$  is the total count of column  $j$ , and  $N$  is the grand total of all observations.

3. **Computation of the Chi-squared Statistic:** This step involves calculating the Chi-squared statistic to test the independence of the two categorical variables. It is computed as:

$$\chi^2 = \sum \frac{(O_{ij} - E_{ij})^2}{E_{ij}}, \quad (5.2)$$

where  $O_{ij}$  represents the observed frequency in cell  $(i, j)$ . The summation is carried out over all cells in the contingency table.

4. **Derivation of P-value and Statistical Inference:** The Chi-squared statistic is associated with a p-value, which is derived considering the degrees of freedom of the contingency table, calculated as  $(\text{number of rows} - 1) \times$

(number of columns  $- 1$ ). This p-value indicates the probability of observing the given (or more extreme) distribution of frequencies, assuming the null hypothesis of independence is true. A standard threshold ( $\alpha = 0.05$ ) is set for the significance level, and the p-value is compared against this threshold. If the p-value is less than or equal to  $\alpha$ , the null hypothesis is rejected, indicating that there is a statistically significant relationship between the two variables. Conversely, if the p-value exceeds  $\alpha$ , there is insufficient evidence to reject the null hypothesis, suggesting no significant dependence between the categorical variables under investigation.

If the computed Chi-squared statistic exceeds the critical value from the Chi-squared distribution, the null hypothesis is rejected, indicating a significant association between the two categorical variables. This methodology was applied iteratively for each of the 16 components, analyzing the potential influence of external objects on the accuracy of damage detection.

### Statistical Findings

For each of the 16 components, a contingency table was prepared delineating the presence or absence of damages against the presence or absence of an external object. This formed the basis for conducting the Chi-squared test for each component.

The summarized findings from these tests are presented in Table 5.4. Specifically, this table encompasses:

- The component under examination.
- Count of images with damage and an external object.
- Count of images with damage and without an external object.
- Count of images without damage but with an external object.
- Count of images without damage and without an external object.
- The computed Chi-squared statistic value for the test.
- The corresponding p-value for the test.

A critical observation from these results is that none of the components yielded a p-value less than the conventional significance threshold of 0.05. While these findings underscore the absence of a statistically significant association between the presence of an external object in annotations and the identification of damage, it is pivotal to highlight that this study primarily centered on annotations. The actual behavior of a classification model, which might utilize these annotations for training, may present nuances that weren't explicitly evaluated in this context. A model could, theoretically, leverage subtle patterns from such annotations – even in the absence of overt statistical significance – to influence its predictions. Therefore, the outcomes of this study should be interpreted in light of its specific focus on annotations and not as a holistic evaluation of a classification model's performance or its potential biases.

Table 5.4: Chi-squared statistics and p-values for the independence test between damage detection and the presence of external objects across different vehicle components.

Component	Damage & Ruler	Damage & No Ruler	No Damage & Ruler	No Damage & No Ruler	$\chi^2$ Statistic	P-Value
Side Mirror	272	67	213	48	0.179	0.672
Rim	262	86	176	76	2.199	0.138
Hood	220	76	222	82	0.130	0.718
Tail Light	292	99	152	57	0.270	0.603
Headlight	228	77	217	78	0.112	0.738
Grille	233	95	200	72	0.460	0.498
Rear Window	281	81	197	41	2.350	0.125
Handle	239	79	208	74	0.154	0.695
Molding	237	86	190	87	1.662	0.197
Windshield	255	72	204	69	0.878	0.349
Fender	259	104	168	69	0.015	0.902
Bumper	323	84	155	38	0.073	0.787
Door	252	99	174	75	0.260	0.610
Tailgate	248	74	212	66	0.048	0.826
Sill	264	99	171	66	0.024	0.877
Roof	259	76	220	45	2.992	0.084

### 5.3.5 Development of Component-Specific Classifiers

In continuation with the findings from the comparative studies of deep convolutional neural network architectures detailed in Section 3.1.3, the MobileNetV2 architecture was adopted as the backbone for the classifiers. This architecture strikes an optimal balance between computational efficiency and predictive performance, essential for the final applications envisioned for the system.

A distinguishing feature of the model is the addition of a custom classification head comprising a single neuron with a sigmoid activation function. This setup is particularly suited for the binary classification task at hand, facilitating the prediction of the probability of damage presence on an input image.

#### Training Methodology

The training set preparation involved a stratified 75/25 train/test split, ensuring a representative mix of various classes and conditions in both sets. Crucially, the split was executed per vehicle to preclude data leakage, as multiple photographs often depict the same instance of damage.

The loss function employed for this binary classification task is the binary cross-entropy. Binary cross-entropy is a loss function commonly used for binary classification problems with probabilistic predictions, such as the presence or absence of damage.

To enhance the model’s ability to generalize and mitigate overfitting, data augmentation techniques were extensively applied during training. These techniques included random horizontal flips, rotation, and various image distortions such as brightness, contrast, and saturation adjustments to simulate diverse lighting conditions encountered in real-world scenarios.

The MobileNetV2 backbone was initialized using pretrained weights on the ImageNet dataset. All layers of the network were set to be trainable, allowing for updates to the weights throughout the entire model during the training process.

The models were trained with an initial learning rate of  $1 \times 10^{-5}$ , leveraging the adaptability of adaptive learning rate methods. Implementation of early stopping based on validation loss further refined the training process, preventing unnecessary computation and overfitting by monitoring model performance on unseen data.

### 5.3.6 Performance Evaluation

#### Quantitative Results

The efficacy of the developed models was evaluated using several metrics that reflect the nuanced performance characteristics in classifying the “Damage” category. These metrics include ROC AUC, Accuracy, Precision, Recall, and F1-Score. Comprehensive evaluation allows for a nuanced understanding of model performance, highlighting areas of strength and potential weakness.

A critical component of the performance evaluation is the analysis of the Receiver Operating Characteristic (ROC) curves, consolidated in Figure 5.11. Each curve visualizes the trade-off between the true positive rate and the false positive rate of a classifier, providing insight into the balance between sensitivity and specificity. The area under the ROC curve (AUC) is particularly informative, serving as an aggregate measure of a model’s performance across all possible classification thresholds.

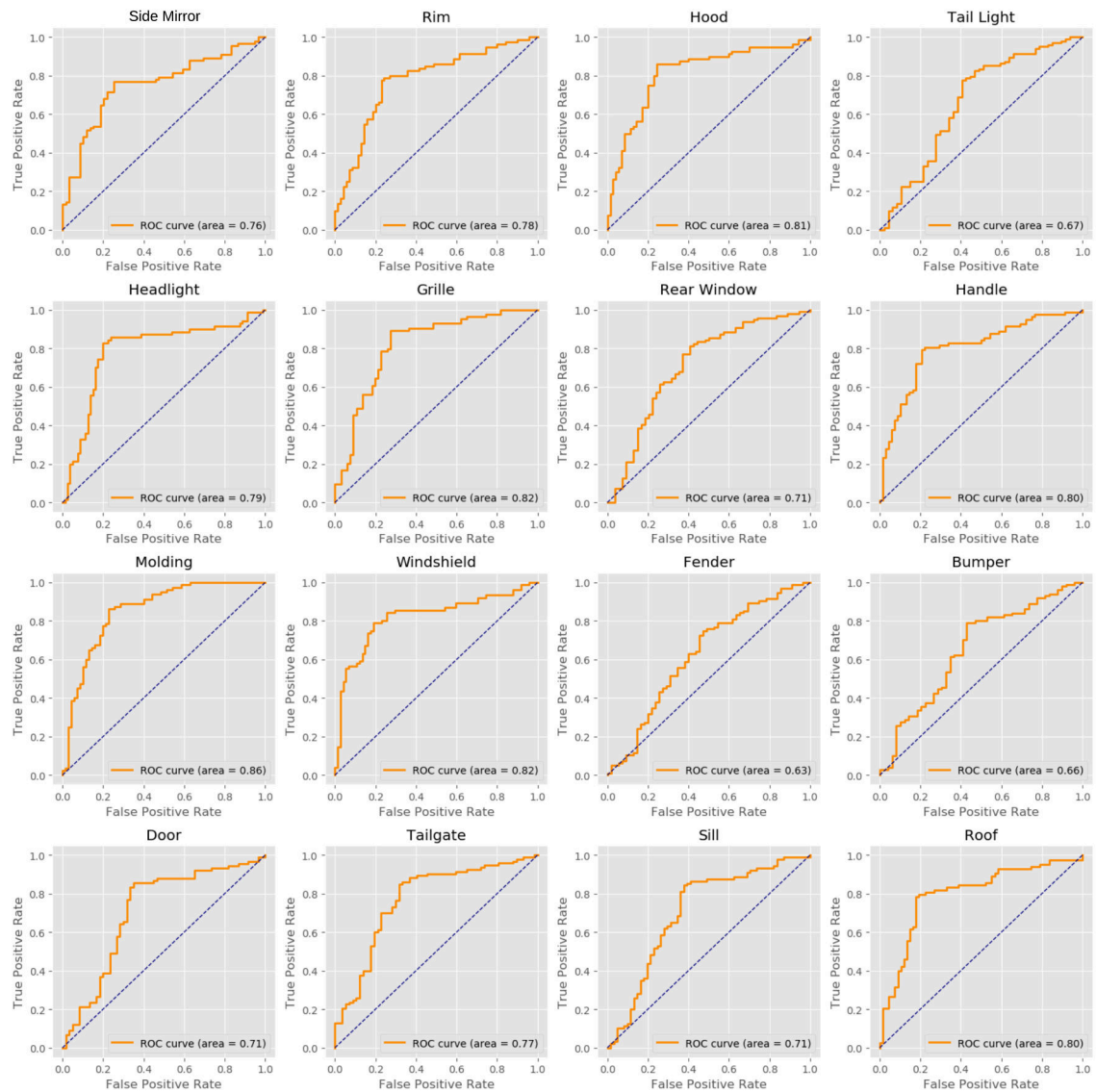


Figure 5.11: Composite visualization of the Receiver Operating Characteristic (ROC) curves for the 16 component-specific classifiers.

In the context of damage detection, a high AUC value indicates a high likeli-

hood that the model can distinguish between classes of present damage or absent damage with minimal confusion. Figure 5.11 reveals that the models exhibit commendable discriminative capacity, with varying degrees of success across different vehicle components.

Further insight into the models’ performance is provided in Table 5.5. Notably, models exhibit substantial variation in their precision, recall, and F1-scores, indicative of the complexities associated with damage detection in different vehicle components.

Table 5.5: Performance metrics across the 16 classifiers, offering a qualitative assessment of Accuracy, Precision, Recall, and F1-Score for each component-specific model. The “Damage Perc” column corresponds to the proportion of items of the Damage class.

Class	Damage Perc	ROC AUC	Accuracy	Precision	Recall	F1-score
Side Mirror	0.56	0.76	0.75	0.82	0.76	0.79
Rim	0.58	0.78	0.77	0.79	0.78	0.78
Hood	0.49	0.81	0.81	0.80	0.86	0.83
Tail Light	0.65	0.67	0.73	0.79	0.82	0.80
Headlight	0.51	0.79	0.81	0.78	0.83	0.81
Grille	0.55	0.82	0.82	0.81	0.89	0.85
Rear Window	0.60	0.71	0.73	0.77	0.82	0.80
Handle	0.53	0.80	0.79	0.82	0.79	0.81
Molding	0.54	0.86	0.81	0.80	0.88	0.83
Windshield	0.55	0.82	0.79	0.77	0.83	0.80
Fender	0.60	0.63	0.66	0.73	0.73	0.73
Bumper	0.68	0.66	0.72	0.79	0.79	0.79
Door	0.58	0.71	0.75	0.78	0.81	0.80
Tailgate	0.54	0.77	0.79	0.81	0.85	0.83
Sill	0.60	0.71	0.73	0.76	0.79	0.77
Roof	0.56	0.80	0.80	0.84	0.80	0.81

Certain components, such as Fenders, Bumpers, Rear Window, and Tail Lights, present particular challenges, evidenced by their relatively low ROC AUC and Accuracy values. These components might share certain textural or shape characteristics with surrounding areas, leading to lower distinctiveness in damage features and, consequently, a higher likelihood of misclassification. Conversely, other components achieve higher metric scores, underscoring the effectiveness of the classification approach in those instances.

The implications of these findings are twofold. Firstly, they confirm the capability of the deep learning models to effectively distinguish between damaged and non-damaged states across a majority of vehicle components. Secondly, they highlight specific areas where further refinement of the models is warranted, possibly through targeted data augmentation, fine-tuning, or application of more sophisticated feature extraction techniques.

### Qualitative Results

The qualitative analysis complements the quantitative evaluation by scrutinizing instances of misclassifications, particularly for the Fender, Tail Light, Bumper, and Rear Window components. Figure 5.12 illustrates several cases where the model incorrectly identified the presence or absence of damage. These instances underscore

the nuanced challenges inherent in automated damage detection, and they can be attributed to several factors, each contributing to the uncertainty in classification.

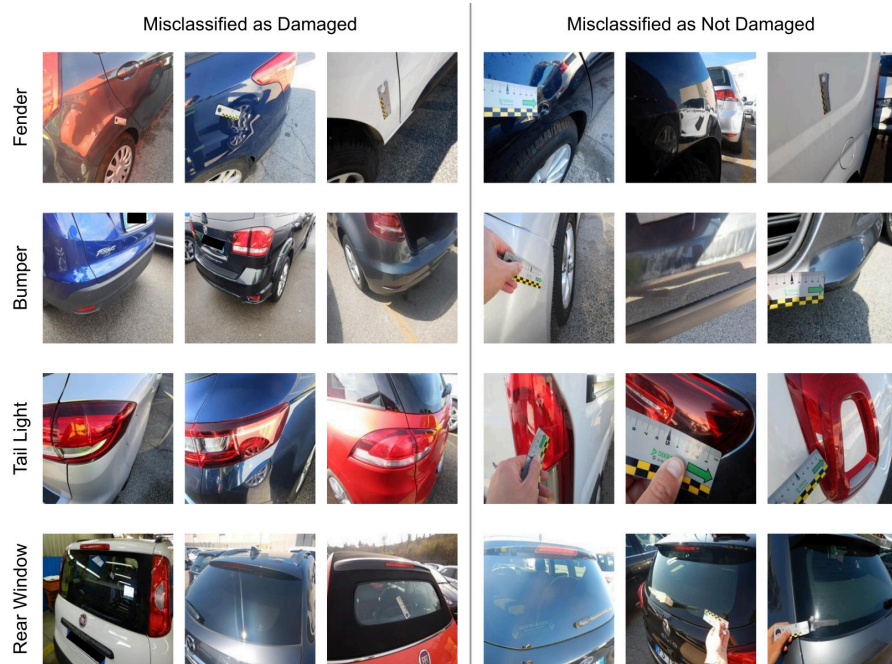


Figure 5.12: Illustration of common misclassification instances encountered by the damage detection model. Each panel presents examples from four components that most frequently confound the classifier: Fender, Tail Light, Bumper, and Rear Window. For each component, the figure contrasts three instances where damages were erroneously flagged as present (Damage misclassifications) with three instances where actual damages were overlooked, being wrongly categorized as absent (No Damage misclassifications). These examples underscore the challenges in discerning authentic damage from various interferences such as reflections, neighboring component damages, image quality constraints, and anomalous visual features.

**Impact of Reflections on Classification** Reflections on vehicle surfaces pose a significant challenge to accurate damage identification. In numerous instances, confusing reflections – ranging from bright highlights to mirrored surroundings – mimic the appearance of scratches or dents, leading the model to false-positive errors. Conversely, actual damage can be obscured by overlying reflections, resulting in false negatives. The extensive variability of real-world lighting conditions exacerbates this issue, as the training dataset cannot possibly encompass all potential scenarios.

**Interference from Adjacent Components** Misclassifications often occur when damage is present on an adjacent component rather than the target of analysis. The model, trained to discern damage based on features specific to certain components, might either overlook the damage impacting its accuracy or falsely identify it due to the close proximity to the targeted area. This situation not only challenges the prediction capability but also calls into question the precision of ground truth labels, which may be based on expert assessments focused on the primary component in question.

**Limitations Due to Photo Quality** The resolution and quality of photographs directly influence the model’s ability to detect minor damages. In several cases, diminutive damages, critical for accurate assessments, go unnoticed in the images due to insufficient resolution, poor lighting, or suboptimal angles. This limitation highlights the need for high-quality image data that can capture sufficient detail to allow the model to make accurate determinations.

**Subjectivity in Damage Assessment** Instances have emerged where the subjective nature of damage assessment by human experts leads to inconsistencies in ground truth labeling, particularly for minor damages that may or may not warrant repair. This subjectivity introduces ambiguity into the training process, potentially leading the model to learn from and perpetuate these inconsistencies.

**Complexity of Damage Patterns** In real-world scenarios, damages can exhibit a complex array of patterns, influenced by the nature of the impact, material properties of the component, and environmental conditions. These complexities may lead to unpredictable appearances of damages, not adequately represented in the training dataset, thus leading to misinterpretations by the model.

### 5.3.7 Critical Evaluation and Possible Improvements

The current models demonstrate promising capabilities in identifying damage across various vehicle components. However, the analysis also uncovers several areas necessitating improvement to enhance overall accuracy and reliability.

**Addressing Reflections and Lighting Variability** The prevalent issue of misleading reflections and inconsistent lighting conditions can be mitigated by expanding the training dataset to include a diverse range of lighting scenarios. Additionally, incorporating image pre-processing techniques to normalize lighting conditions and reduce reflection-related noise could enhance the model’s robustness.

**Refining Damage Localization** Errors due to interference from adjacent components suggest a need for improved damage localization. Future iterations could benefit from integrating more sophisticated object detection frameworks, such as region-based Convolutional Neural Networks (R-CNNs), to more precisely delineate damage areas.

**Enriching Training Data for Complex Patterns** The unpredictability of damage patterns requires a more comprehensive dataset that encapsulates the myriad forms of vehicle damage. This enhancement means sourcing images that illustrate a broader spectrum of damage types, severities, and impacted materials, potentially achieved through synthetic data generation methods.

In summary, while the models achieve notable success in damage detection, these improvements are integral to advancing the system’s operational efficacy. By addressing these specific challenges, future iterations could move significantly closer to the nuanced perceptual abilities of human inspectors, thereby elevating the reliability and precision of automated vehicular damage assessment.



## 5.4 Conclusion

The exploration within this chapter revealed critical insights into the challenges of visual classification tasks. For component recognition, ambiguities inherent in image data and the subtle distinctiveness of certain vehicle parts presented considerable hurdles. Although the integration of pose estimation data was presumed to enhance performance, its impact was marginal, prompting consideration for a more application-specific pose estimation technique. Meanwhile, the damage presence detection exercise uncovered the complexities of discerning damages, where factors such as confusing reflections, damage proximity to adjacent components, photographic quality, and human subjectivity in damage assessment play significant roles in influencing the model's performance.

The qualitative and quantitative evaluations underscored the necessity for advanced techniques and continued model refinement. Reflections on the misclassifications stressed the need for a comprehensive, well-curated dataset and advanced feature recognition strategies, highlighting areas for potential enhancement of the system's accuracy and reliability.

In conclusion, the investigations and findings presented reaffirm the potential of machine learning in automating vehicular damage assessment. However, they also emphasize the continued need for advancements in visual recognition technologies and methodologies.

# Chapter 6

## Vehicle Damage Repair Costs Estimation and System Evaluation

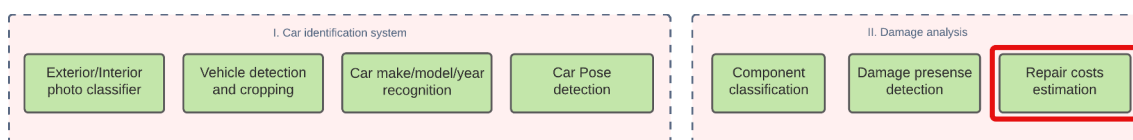


Figure 6.1: Highlighted segment of the whole system representing the focus of this chapter.

Building upon the foundational modules established in previous chapters, this chapter zeroes in on the ultimate task of the thesis: providing a robust estimate for vehicle damage repair costs. By utilizing insights such as the vehicle’s make, model, year, and the specific damaged components identified earlier, this section aspires to derive an approximate repair cost. This translation from technical damage metrics to monetary values offers practical utility for service providers and consumers alike.

Building upon the foundational modules established in previous chapters, this chapter zeroes in on the ultimate task of the thesis: providing a robust estimate for vehicle damage repair costs. By utilizing insights such as the vehicle’s make, model, year, and the specific damaged components identified earlier, this section aspires to derive an approximate repair cost. It’s important to note that while the broader scope of this thesis emphasizes visual analysis, the repair cost estimation presented here is primarily based on numerical and categorical data, and not directly on visual inputs. The translation from technical damage metrics to monetary values offers practical utility for service providers and consumers alike.

Distinct from traditional classification challenges, the repair cost estimation operates within the ambit of regression, predicting a continuous value rather than discrete outcomes. The chapter delves into the methodology behind this approach, charting the course from initial model considerations to the refinement of the chosen regression model.

Chapter 6 is structured into two principal sections:

- **Repair Costs Estimation:** Beginning with a brief overview of related works, this section further explores dataset characteristics, the employed methodology, and the ensuing process of model training and evaluation.

- **End-to-End System Evaluation:** Offering a comprehensive system review, this segment assesses the integration and performance of all modules, culminating in an end-to-end test of the entire damage assessment mechanism.

The chapter concludes with an analysis and discussion on the presented results, illuminating both the achievements and areas warranting further exploration.

## 6.1 Repair Costs Estimation

### 6.1.1 Background and Related Work

The automation of vehicle damage assessment and repair cost estimation has been a focus of considerable research, particularly due to its implications in insurance and leasing industries where rapid, consistent, and accurate evaluations are essential. Various studies have explored different aspects of this issue, employing a range of methodologies that integrate deep learning, computer vision, and machine learning algorithms, particularly focusing on cost estimation post damage detection.

Mohammed et al. [110] proposed an end-to-end solution that makes use of the Mask Region-based Convolutional Neural Network (Mask RCNN) [13] to classify vehicle damage costs. Two Mask RCNN models were employed: the first to detect vehicle sides which affect damage cost estimation, and the second to detect the area of the damage. Their approach solely focuses on the area of damage for estimating repair costs, multiplying the recognized damage area by a side-dependent factor to determine the estimated cost.

Contrarily, Fernando et al. [16] and Jameel et al. [111] adopted more comprehensive strategies. Fernando et al. integrated Convolutional Neural Networks (CNNs) and Natural Language Processing (NLP) to identify the vehicle and its damaged components before using rule-based classifications for cost estimation, grounding the assessment in the context of damage severity and type. Meanwhile, Jameel et al. demonstrated the effectiveness of a multi-task image regression model that utilizes vehicle configuration data, reducing the error in repair cost estimates. They emphasized the utility of machine learning, particularly Random Forest [112] and XGBoost [113], in analyzing non-image data for cost prediction.

Mallios et al. [5] and Poon et al. [17] presented methods that merge computer vision with feature importance analysis. Mallios et al. employed semantic segmentation in damage detection before using an XGBoost model for final cost estimation, factoring in the vehicle's historical data and the extent of damage. Poon et al. combined deep learning technology with a statistical analysis of appraisal metadata, demonstrating the efficiency enhancements in the claim process when using advanced neural networks coupled with regression modeling for cost predictions.

Ul et al. [114] introduced an innovative object regression model that synchronizes damage detection and cost prediction. They leveraged a combination of Faster-RCNN [50] and ResNet50 [41] for detection tasks, followed by various regression models for cost prediction, noting the superior performance of robust methods like Random Forest and XGBoost over linear regression.

Analyzing these studies reveals a significant trend towards the integration of image processing for damage assessment with machine learning algorithms for cost estimation. This synthesis of visual cues and computational intelligence is particularly

prominent in contributions where the image data serves not just as a supplementary feature but often plays a pivotal role in the predictive modeling [110, 16]. However, a comprehensive approach to automated cost estimation transcends the use of visual information alone.

While the fusion of damage imagery with cost data presents a highly promising frontier for precision in repair estimates, the absence of a paired photographic dataset in the current study necessitates the exploration of alternative, albeit robust, methodologies. In this context, the XGBoost algorithm emerges as a compelling second avenue, capable of harnessing a set of non-visual features for insightful predictions, as accentuated in diverse studies [5, 17, 111, 114].

### 6.1.2 Dataset Description

The crux of the damage repair cost estimation module pivots on the quality and detail of the data available. This study utilizes a unique dataset provided by the brumbrum company, comprising comprehensive logs from their repair factory. This section delves into the specifics of these records, shedding light on their original structure, inherent challenges, and the consequent preprocessing required to render them suitable for the task at hand.

#### Initial Dataset Structure

The foundational dataset is an aggregation of detailed operations carried out in the brumbrum repair facilities. Initially, it encompasses various pieces of information concerning each repair action undertaken for distinct components of a wide array of vehicles. The dataset, structured in a tabular format, features several pertinent fields per operation as follows:

- **Date of the Event:** The specific date when the repair operation was conducted.
- **Vehicle ID:** Unique identifier for the vehicle.
- **Vehicle Model and Version:** Detailed information concerning the vehicle's model and specific version.
- **Component:** The particular vehicle part that underwent the repair process.
- **Severity of the Operation:** Categorized levels of severity for the operation, labelled as Low, Medium, or Standard.
- **Duration:** The time taken to complete the repair, recorded in hours and minutes.
- **Repair Cost:** The total expenditure for the repair work, denoted in euros.

Each row in the dataset corresponds to a singular operation, with the entire compilation consisting of 41,846 such records. A visual representation of the dataset's initial sample rows is provided in Figure 6.1.

## CHAPTER 6. VEHICLE DAMAGE REPAIR COSTS ESTIMATION AND SYSTEM EVALUATION

Date Event	Vehicle ID	Vehicle Model Version	Component	Difficulty	Timings	Repair Cost
12/06/2020	137191	Ford Kuga 2018 1.5 TDCI 120 CV S&S 2WD ST-Line	Carpet floor	Low		50
12/06/2020	137191	Ford Kuga 2018 1.5 TDCI 120 CV S&S 2WD ST-Line	Bumper moulding rear	Severe	0:40	143,14
12/06/2020	137191	Ford Kuga 2018 1.5 TDCI 120 CV S&S 2WD ST-Line	Position lamp lateral	Severe	0:20	20,32
12/06/2020	137433	BMW Serie 3 320d xDrive Business 2018	Tailgate	Severe	5:00	646,03
15/06/2020	137465	Ford Edge 2018 2.0 tdcI Titanium S&S	Rear rim and right front (redo)	Severe		160
15/06/2020	137518	Volkswagen Passat 2014 Var. 1.6 TDI Comfortline	Fender rear left	Severe	3:20	469,14
15/06/2020	137518	Volkswagen Passat 2014 Var. 1.6 TDI Comfortline	Windshield	Severe	4:40	380,83
15/06/2020	137734	Audi A3 2015 Sportback 1.6 tdi Business 110cv	Windshield	Severe	4:30	315,2
15/06/2020	137734	Audi A3 2015 Sportback 1.6 tdi Business 110cv	Rear wheel rim repair	Severe		80
17/06/2020	137823	Toyota Auris 2018 5 Porte 1.8 Hybrid	Moulding	Severe	0:30	30,98
17/06/2020	137823	Toyota Auris 2018 5 Porte 1.8 Hybrid	Door rear left	Medium	3:50	483,95
17/06/2020	137880	Ford Mondeo 2018 Full Hybrid 2.0 187 CV	Hood, hail	Severe	4:20	353,26
22/06/2020	138605	Opel Insignia 2017 sports tourer 2.0 CDTI 170 CV S&S	Windshield-repair	Low	0:40	50
22/06/2020	138613	Opel Astra 2017 Sports Tourer 1.6 cdti Business	Shell Left	Low	0:50	80
22/06/2020	138627	Ford Fiesta 2012 1.5 TDCi 3 Porte	Fender rear right	Medium	3:30	289,6

Table 6.1: A sample of car repair records in the initial dataset

### Data Preprocessing and Refinement

**Initial Data Translation** The preprocessing phase was inaugurated with the indispensable task of translating the dataset’s raw data values into a consistent and standardized format. This initial step was firmly anchored in the glossary terms meticulously outlined in earlier chapters (Section 2.4), which became the cornerstone for this translation process. The glossaries were comprehensive, precise definitions and categorizations for 1199 make and model combinations. These definitions were accompanied by an array of carefully crafted regular expressions, each serving as a potential categorization tool for the raw string values representing the vehicle makes and models in the dataset.

The objective of this transformation was twofold: first, to eradicate any ambiguities that could cloud the dataset’s integrity, and second, to homogenize the terminologies used, ensuring they were in strict conformity with the established standards from the glossaries.

However, not all entries could be seamlessly translated into the glossary terms—a reality that led to the pragmatic decision to discard certain rows. Specifically, entries that defied categorization, where no association could be established with the glossary’s make and model definitions, were considered outliers and thus eliminated from the dataset. This exclusion was critical in preserving the dataset’s coherence and readiness for the subsequent stages of preprocessing, ensuring that only the data points adhering to the recognized standards would proceed.

### Strategic Decisions in Data Handling

Following the initial translation, several strategic decisions were essential to refine the dataset further and ensure its compatibility with machine learning models, specifically focusing on the handling of make, model, production year information, and component categorization.

**Utilization of the Codebook** Given the extensive variety of make and model combinations, amounting to 1199, a simplistic dummy encoding approach was impractical due to dimensional concerns. Instead, the study leveraged a “codebook” — a comprehensive commercial database encompassing approximately 200,000 records of distinct vehicle trims or versions registered within the European Union. This resource was instrumental in translating the “model” variable into analyzable features. The codebook offered a wealth of technical and commercial information about each

vehicle, ranging from mechanical specifications to market segmentation and listing prices.

**Refinement of Vehicle Attributes** The “model” variable underwent a transformation through its association with the codebook, resulting in a set of derived variables that offered more depth and relevance for the analysis:

- **Vehicle Category:** The original “model” information was crucially transformed, associating each entry with one of the several distinct categories illustrated in Table 6.2. These categories encompass types such as Station Wagon, SUV, Sedan, Convertible, among others. These categories were then subjected to dummy encoding.
- **Vehicle Dimensions (in centimeters):** Dimensions for each vehicle, specifically width, length, and height, were also included. Given the slight variations across different versions, average dimensions for each make and model were calculated and used.
- **Average Listing Price:** For each make, model, and production year, the corresponding average listing price (known also as Price-on-New) was extracted from the codebook.
- **Year:** This attribute remained untouched, retaining its numerical form, considering its significance and the nature of being already quantifiable.

Vehicle Category
Subcompact
Crossover
Compact sedan
City car
Convertible
Station wagon
SUV
Minivan
Multi-purpose vehicle
Sedan

Table 6.2: Possible categorizations for the “Vehicle Type” attribute in the dataset.

**Component Encoding and Standardization** The “component” field in the dataset was streamlined to reflect the 16 main components identified in 5.1.2, and was accordingly dummy-encoded to transform categorical data into a format suitable for machine learning algorithms. Post encoding, all features underwent standardization, ensuring they were centered around a mean with a standard unit of deviation. This process is paramount in machine learning to balance the scales of measurement and provide each feature an equal opportunity to influence the model.

**Compatibility with System Design** It is noteworthy that these transformations align impeccably with the end-to-end system design. They serve merely as mapping procedures, translating the make, model, and production year into specific, analytically relevant variables without necessitating additional input, thus preserving the system’s intended operational simplicity.

### 6.1.3 Characteristics of the Refined Dataset

#### Dataset Structure and Attributes

The refined dataset features 31,454 rows, each representing a unique repair operation, with several attributes and a target variable. The attributes include “Make” and “Vehicle Type” (both encoded as dummy variables), “Average Listing Price”, “Average Width”, “Average Length”, “Average Height”, and “Year”. The “Component” attribute, representing the specific part of the vehicle involved in the repair operation, is also encoded as a dummy variable. The target variable in this dataset is the “Repair Cost”, expressed in euros, which encapsulates the essence of this study.

In the transformed dataset, the dimensionality reflects the comprehensive nature of the data used for this analysis. Specifically, the dataset comprises 47 dummy variables for the “Make”, 15 for the “Component”, and 9 for the “Vehicle Type”. These are in addition to 3 numerical variables representing the vehicle’s dimensions, 1 continuous variable for the “Year”, and 1 for the “Average Listing Price”, amounting to a total of 76 explanatory variables.

#### Variable Distributions

To provide a visual representation of the dataset’s categorical variables, Figure 6.2 features three pie charts depicting the distributions of “Components”, “Vehicle Types”, and “Makes”. These charts, arranged from left to right, underscore the non-uniform distribution of values across these categories, thereby reflecting the diverse nature of the dataset.

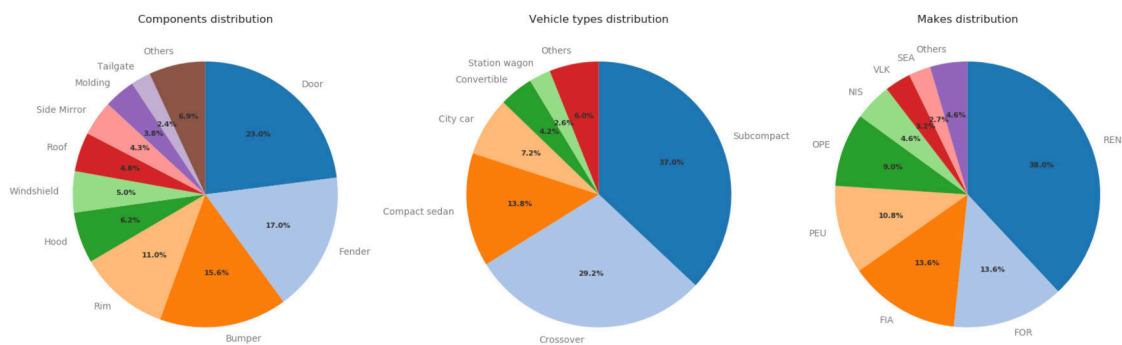


Figure 6.2: Pie charts illustrating the distribution disparity among various categorical variables. From left to right: distribution of components, vehicle types, and makes.

Moreover, the distribution of the “Repair Cost”, the target variable, is showcased in Figure 6.3. The form of this distribution bears resemblance to a log-normal distribution, indicative of a wide-ranging set of values with a concentration around the lower cost spectrum. The mean repair cost stands at 778 euros, with a standard deviation of 525 euros, highlighting the variability in repair expenses encountered in real-world scenarios.



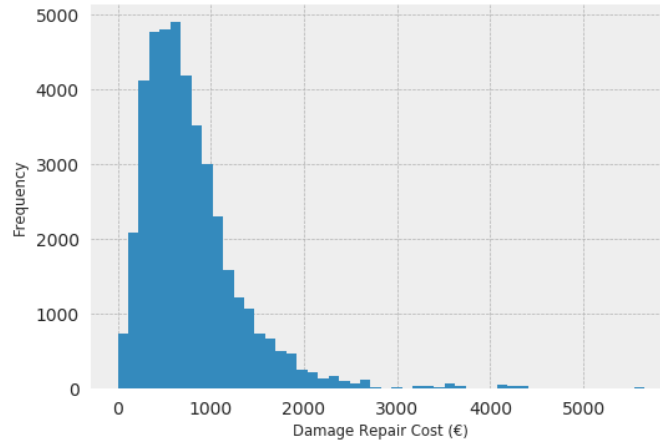


Figure 6.3: Distribution of repair costs, exhibiting a log-normal-like shape, underscoring the variability in vehicle repair expenses.

### 6.1.4 Methodology

The methodology employed in this research was orchestrated with a focus on exploring and comparing various regression models to predict repair costs based on the processed dataset. In this study, four distinct models were examined, each with its unique assumptions, strengths, and computational intricacies: Linear Regression, Feedforward Neural Networks (FFN), Random Forest, and XGBoost. These models were chosen for their widespread usage and proven effectiveness in handling high-dimensional data, nonlinear relationships, and model interpretability requirements.

#### Linear Regression

Linear Regression was utilized as the baseline model owing to its simplicity and interpretability. The implementation involved constructing a linear equation that predicts the repair cost as a function of the 76 input features. The model was optimized using ordinary least squares to minimize the sum of the squared differences between the actual and predicted repair costs.

#### Feedforward Neural Networks (FFN)

The Feedforward Neural Network employed was a multi-layered network designed to model the potentially complex non-linear relationships between the input features and the repair costs. The network comprised an input layer that accepted the 76 features, followed by two hidden layers, each with a rectified linear unit (ReLU) activation function, and an output layer with a single neuron and ReLU activation to predict the continuous cost value. The model parameters were optimized using backpropagation with a mean squared error loss function.

#### Random Forest

Random Forest regression was applied due to its capability of handling high-dimensional datasets and providing importance scores for the features. This ensemble model uses multiple decision trees, operating on randomly selected subsets

of the data and features, aggregating their outputs. The model mitigates overfitting, common in individual decision trees, and enhances prediction accuracy.

### **XGBoost**

XGBoost, an advanced implementation of gradient boosted decision trees, was chosen for its efficiency and effectiveness, especially in regression tasks with a complex structure. The model operates by iteratively correcting the residuals of the preceding trees, focusing more on difficult-to-predict instances, thereby boosting the overall prediction accuracy.

Each of these methodologies was subjected to training and evaluation processes, with their performance metrics compared to ascertain the most suitable model for predicting car repair costs accurately and reliably.

### **6.1.5 Evaluation Metrics**

The efficacy of the predictive models in accurately estimating repair costs is quantified using specific metrics that capture various aspects of performance. For this study, the chosen metrics are Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared ( $R^2$ ), each offering distinct insights into the models' accuracy, error magnitude, and explanatory power concerning the variance in repair costs. These metrics have been previously detailed in Section 4.3.4 of this thesis.

- **MAE** provides a straightforward interpretation by averaging the absolute errors, thereby reflecting the typical magnitude of errors in the predictions.
- **RMSE**, by squaring the errors before averaging, gives higher weight to larger errors, offering insight into the variability of the predictions.
- $R^2$  indicates the proportion of the variance in the dependent variable that is predictable from the independent variables, serving as an indicator of the goodness of fit for the predictive models.

The comparison of these scores across the different models should identify the most promising approach for cost estimation in the context of automatic car damage assessment.

### **6.1.6 Training Process and Performance Evaluation**

#### **Models training**

To ensure that the evaluation of the models is unbiased and to mitigate the risk of overfitting, the dataset was split into training and test sets. Of the 31,454 rows in the dataset, 75% (or 23,590 entries) were allocated to the training set, while the remaining 25% (or 7,864 entries) were used as the test set.

With an aim to maintain consistency across evaluations, the same training/test split was applied to all models. The following outlines the salient configurations and hyperparameters selected for each model during the training phase:

**Linear Regression** For the Linear Regression model, the training was straightforward given the model’s reliance on the inherent data characteristics rather than intricate configurations. No hyperparameters needed tuning, and the focus was on ensuring the data’s suitability for linear assumptions.

**Feedforward Neural Networks (FFN)** The FFN model was configured with two hidden layers, comprising 128 and 64 neurons, respectively, to provide sufficient complexity for capturing underlying patterns within the data. The network utilized the Adam optimizer with a learning rate of 0.001, and training occurred for 100 epochs with a batch size of 32.

**Random Forest** The Random Forest model was set up with 100 decision trees, with the maximum depth of each tree limited to 30 to prevent the model from becoming excessively complex and overfitting the data. The minimum number of samples required to split an internal node was set at 2, and the minimum number of samples required to be at a leaf node was set at 1, providing a balance between model complexity and training data fit.

**XGBoost** For the XGBoost model, the learning rate was established at 0.1, optimizing the balance between speed and efficiency of learning. The model was configured to run for 500 boosting rounds with early stopping enabled, observing the validation loss to prevent unnecessary computations if the loss failed to improve for 20 consecutive rounds. The maximum depth of the trees was set at 6, allowing for sufficient interaction between variables, and the subsample ratio was fixed at 0.8 to introduce randomness into the training procedure.

### Performance Evaluation

The XGBoost model demonstrated remarkable performance, indicative of its robustness in handling complex non-linear relationships amidst high-dimensional data. It achieved an RMSE of 162, indicating a high level of accuracy considering the standard deviation of repair costs in the dataset is 525. Furthermore, the model exhibited an  $R^2$  of 0.882, suggesting that approximately 88% of the variance in repair costs was accounted for by the model. The Table 6.3 encapsulates the comparative performance metrics.

Model	MAE	RMSE	$R^2$
Linear Regression	221	334	0.694
FFN (Feedforward Neural Network)	155	215	0.806
Random Forest	130	178	0.851
XGBoost	118	162	0.882

Table 6.3: Comparative performance metrics of the regression models

As depicted, Linear Regression, albeit its simplicity and interpretability, could not compete with the predictive power of more complex algorithms, reflected in its higher RMSE and lower  $R^2$  value. The FFN model, given its capacity for modeling non-linear relationships, showed improvement over Linear Regression but still fell short compared to the ensemble methods. Random Forest performed admirably,

benefiting from its ensemble nature, but was slightly outperformed by XGBoost, which stood out for its sophisticated boosting technique.

An analysis of the residuals from the XGBoost model, as shown in Figure 6.4, reveals critical insights into the prediction errors. Residuals, calculated as the difference between the actual and predicted values ( $y - \hat{y}$ ), provide a measure of model accuracy. The distribution's mean slightly above zero at 11.5 suggests a minimal underestimation bias across predictions. While the model manifests high accuracy overall, it is notable that there are instances of considerable prediction errors, with outliers exhibiting an absolute delta exceeding 1000 euros. These substantial deviations, albeit sparse, emphasize the necessity for further refinement of the model, particularly in its capacity to generalize and mitigate large residuals, thereby enhancing prediction reliability in scenarios of higher complexity.

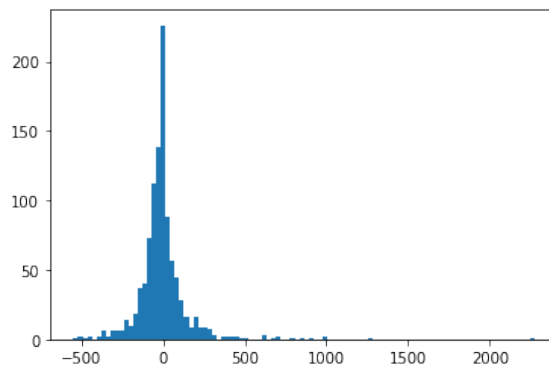


Figure 6.4: Distribution of residuals for the XGBoost model, highlighting the predominance of minor prediction errors and the presence of outliers with significant deviations.

The promising performance of the XGBoost model can be attributed to its gradient boosting framework, which is particularly adept at minimizing errors through systematic model adjustments with each iteration. However, despite its relative superiority, there is room for enhancements, especially when it comes to reducing large prediction errors, as evidenced by the analysis of residuals.

### 6.1.7 Results and Discussion

The evaluation metrics underscore the differential capabilities of the models in predicting vehicle repair costs. Notably, the sophisticated algorithms—Random Forest and XGBoost—outshone simpler models, underscoring the importance of complexity and adaptability in handling the multifaceted nature of the dataset (see Table 6.3).

One notable direction for future improvement is the integration of visual data into the prediction pipeline. Incorporating features extracted from damage photographs, as demonstrated in studies [5, 111, 114], could significantly enrich the prediction model's input, potentially enhancing the accuracy of cost estimations. This approach, however, necessitates the collection of a comprehensive dataset where each repair case is documented with corresponding high-quality images, establishing a foundation for more nuanced and context-aware predictions.

Furthermore, leveraging additional information from the extensive codebook database presents another promising avenue for refinement. The inclusion of more detailed technical descriptors of vehicles, such as horsepower, torque, drivetrain characteristics, and advanced safety features, could potentially sharpen the model’s predictive precision. These variables, often indicative of a vehicle’s performance and build complexity, might correlate with repair costs, particularly in cases involving high-performance or luxury vehicles where parts and specialized labor are typically more expensive.

Additionally, while the models used are robust, exploring more advanced machine learning algorithms or deep learning could unearth patterns less apparent to the current models, especially in instances where data is more nuanced or less structured.

In conclusion, the results signify a strong starting point for repair cost prediction using machine learning. However, they also highlight critical pathways for advancement, particularly concerning data enhancement and model sophistication.

## 6.2 End-to-End System Evaluation

This section outlines a comprehensive evaluation of the integrated system, designed to simulate real-world application scenarios. Distinct from the methodologies adopted in the studies [16, 5, 17, 111, 114], who evaluated each module in isolation, this evaluation presents an attempt to assess the system’s performance in a cohesive, end-to-end manner. The approach accentuates the practical implications and operational readiness of the system, highlighting areas for potential improvement.

### 6.2.1 Test Setup and Dataset

The end-to-end test leverages a collection of 200 documents, each corresponding to a vehicle acquired and repaired by the company, thus representing actual cases. These documents are enriched with images and technical meta data, including full version names, registration dates, gearbox specifications, among others. For each document, there is a generic view image intended for the recognition of the vehicle’s make, model, and year. Accompanying this generic view are pairs of photographs capturing the condition of specific vehicle components, be they damaged, very slightly damaged, or undamaged.

The images hold central importance in the evaluation process. The generic view image allows the system to extract the vehicle make, model, and year. This requires initial checks for exterior classification, followed by vehicle localization and subsequent cropping. On the other hand, the paired component-specific images are essential for evaluating the system’s capability to deduce the vehicle’s pose, pinpoint the pertinent component, and determine the presence or absence of damage.

For evaluation, the same generic view is used alongside each pair of component images, forming groups of three: one generic view and two component-specific images. A selection of images from this dataset is showcased in Figure 6.5, highlighting the range of vehicle conditions and photography techniques.

The test pipeline replicates the sequence of tasks performed in an actual application scenario, beginning with the validation of exterior photography, followed by a series of classification and estimation tasks, and culminating in the calculation of

## CHAPTER 6. VEHICLE DAMAGE REPAIR COSTS ESTIMATION AND SYSTEM EVALUATION

























Generic view	Components view		Details
			MMY: VLK / Golf 5d / 2012 Component: Door Damage present: Yes Repair cost: 125€
			MMY: TOY / Verso / 2014 Component: Fender Damage present: Yes Repair cost: 580€
			MMY: TOY / Avensis / 2015 Component: Fender Damage present: Yes Repair cost: 30€
			MMY: BMW / Serie 1 5d / 2018 Component: Tailgate Damage present: No Repair cost: -
			MMY: PEU / 508 sw / 2015 Component: Tail Light Damage present: Yes Repair cost: 60€
			MMY: MIN / Mini 3d / 2015 Component: Sill Damage present: Yes Repair cost: 430€
			MMY: CIT / C4 / 2015 Component: Door Damage present: No Repair cost: -
			MMY: FIA / 500X / 2015 Component: Bumper Damage present: Yes Repair cost: 85€

Figure 6.5: Illustrative examples from the end-to-end evaluation dataset. Each example, arranged in rows, comprises a generic exterior view of a vehicle (left column), followed by two distinct component-specific views (middle column). The rightmost column provides corresponding ground truth metadata, detailing the vehicle’s make, model, and year, the identified component type, the presence of damage (indicated as “Yes” or “No”), and the incurred repair cost.

predicted repair costs. These steps collectively contribute to a comprehensive evaluation, challenging the system’s capacity to deliver accurate and reliable predictions in a cohesive, real-world context.

## 6.2.2 Evaluation Protocol

The evaluation protocol, as outlined in Figure 6.6, serves to mimic real-world situations where the system will be deployed, offering a structured approach to gauge its effectiveness across various modules in a cohesive manner.

The process initiates with *Vehicle Identification*, starting with discerning the nature of the given image. If the image shows the interior of a vehicle or does not depict a vehicle at all, the process terminates for that specific image. However, if the image presents an exterior view of a vehicle, the process continues to *Vehicle Localization*. At this juncture, the system attempts to pinpoint and isolate the vehicle within the image. In cases where the vehicle is not localized successfully, the process for that specific image terminates. However, upon successful localization, a cropped image, limited to the vehicle’s bounding box, is generated. This cropped image is then subjected to the *Make/Model/Year classification*, where the system, in tandem with a codebook, identifies the exact make, model, and production year of the vehicle.

Simultaneously, in the *Damage Detection* phase, the image undergoes a secondary verification to confirm its exterior nature. If this confirmation fails, the image is skipped, otherwise, it transitions to *Pose Estimation*. This step estimates the vehicle’s azimuth, offering more context to the subsequent *Component Classification*. Using both component’s image and the estimated azimuth, the system is then able to classify the specific vehicle component depicted in the image. Thereafter, the system determines the presence of any damage. If no damage is detected, the process ends for that image. However, if damage is identified, it advances to the *Repair Cost Estimation* phase.

The *Damage Repair Cost Estimation* commences by gathering essential data points for each component’s image, encompassing the vehicle’s make, model, year, dimensions, category, listing price, and the classified component. This data is then funneled into the *Repair Cost Regressor* that estimates the repair cost for the recognized damage. If both input images show the damage, their repair cost predictions are averaged to produce a single repair cost estimate.

For the integrity of the evaluation, special scenarios wherein the system encounters non-standard situations, such as the absence of damage or recognition errors, have specific handling protocols. Assigning a repair cost of zero in relevant situations maintains consistency in the assessment process. For instance, when the system fails to recognize the exterior nature of the generic view photograph, it is assumed the repair cost is zero. This might lead to underestimation of costs for the validation, yet it remains the most consistent strategy given the current model capabilities.

To further augment the validation set and evaluate the system’s resilience to noise and potential real-world aberrations, an additional 40 documents were included. Out of these 40 documents, 20 contain images with either no vehicles or no exterior parts of the vehicle, and the remaining 20 feature documents with unrecognizable car components. The breakdown of the final validation dataset is shown in the Table 6.4.



# CHAPTER 6. VEHICLE DAMAGE REPAIR COSTS ESTIMATION AND SYSTEM EVALUATION

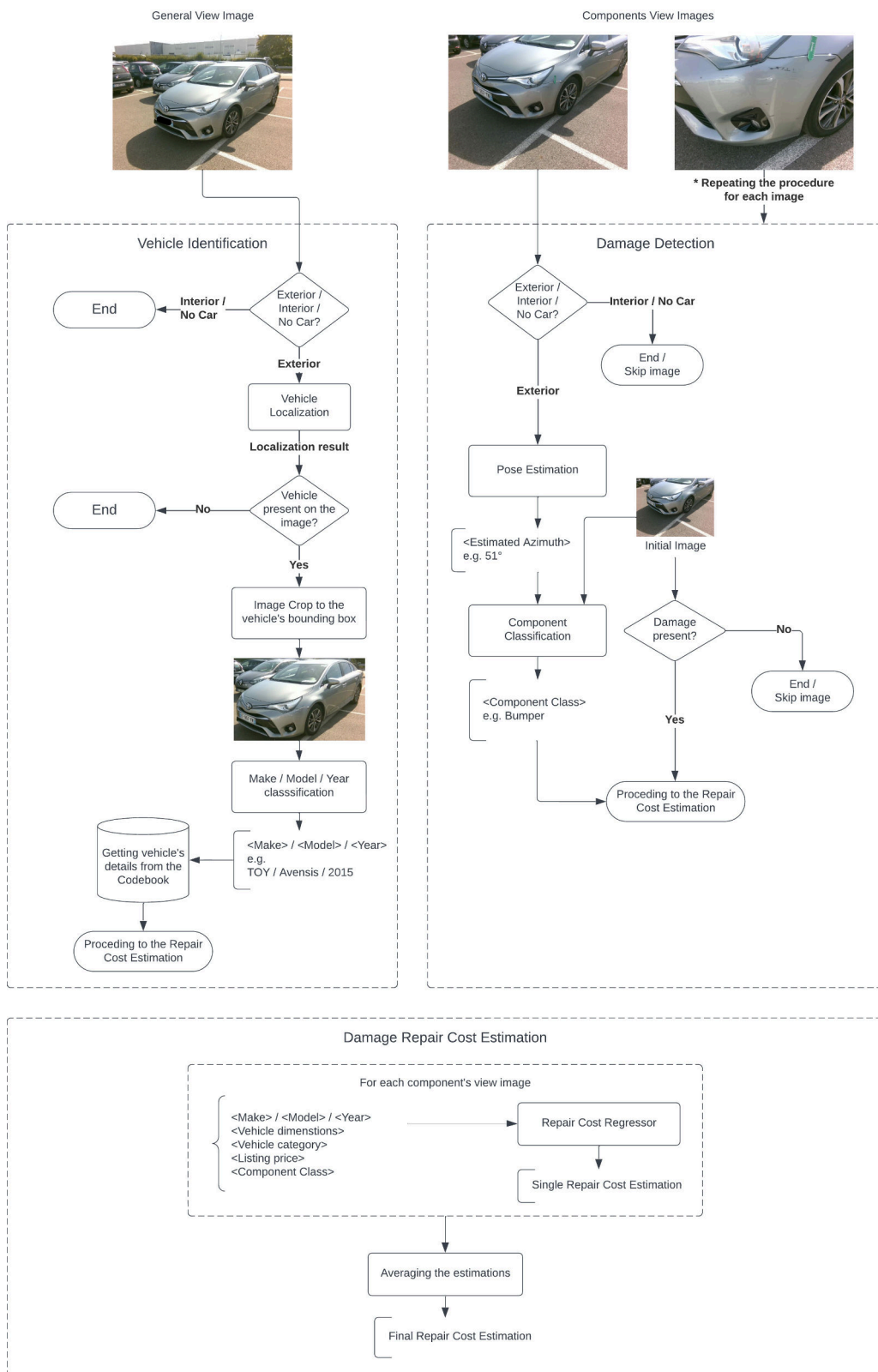


Figure 6.6: Flowchart illustrating the end-to-end process for vehicle damage detection and repair cost estimation, encompassing steps from vehicle identification to final repair cost computation.

Type of document in the validation	Number of documents
No external parts of the vehicle	20
Unknown / Unrecognisable components	20
Known components, No damages	71
Known components, With damages	129

Table 6.4: Composition of the validation set for the end-to-end test.

It should be noted that the ground truth for repair costs for the documents under “No external parts of the vehicle”, “Unknown / Unrecognisable components”, and “Known components, No damages” categories is assigned to 0.

The culmination of this protocol is the computation of the system’s prediction accuracy regarding the repair costs, utilizing standard regression metrics: MAE, RMSE, and  $R^2$ . Each of these metrics offers distinct insights into the system’s performance nuances, thereby providing a comprehensive understanding of its operational efficacy.

### 6.2.3 Results Evaluation

The evaluation of the end-to-end system performance was undertaken with an emphasis on three key areas: component prediction, damage presence classification, and repair cost regression.

#### Component Prediction

The overall accuracy for final component prediction, encompassing exterior recognition and vehicle localization, stands at 0.859. A confusion matrix detailed in Figure 6.7 provides deeper insights, notably incorporating the class “Internals / No Vehicle” to capture those instances where the image did not depict an external part of a vehicle, signifying the all-encompassing nature of the end-to-end test.

#### Damage Classification

The system’s performance in classifying images as “Damage” or “No Damage” is delineated in Table 6.5. An accuracy of 0.677 was achieved, highlighting the intricacies and challenges tied to damage detection. Particularly, this underscores the difficulties in pinpointing damages in images where they might be subtle or nuanced.

Classes	Precision	Recall	F1-score
No Damage	0.61	0.59	0.6
Damage	0.72	0.74	0.73

Table 6.5: Performance Metrics for Damage Classification

#### Repair Cost Regression

Turning attention to the system’s regression capabilities, the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) amounted to 186 and 278 euros, respectively. These figures provide a monetary context to the system’s predictions,

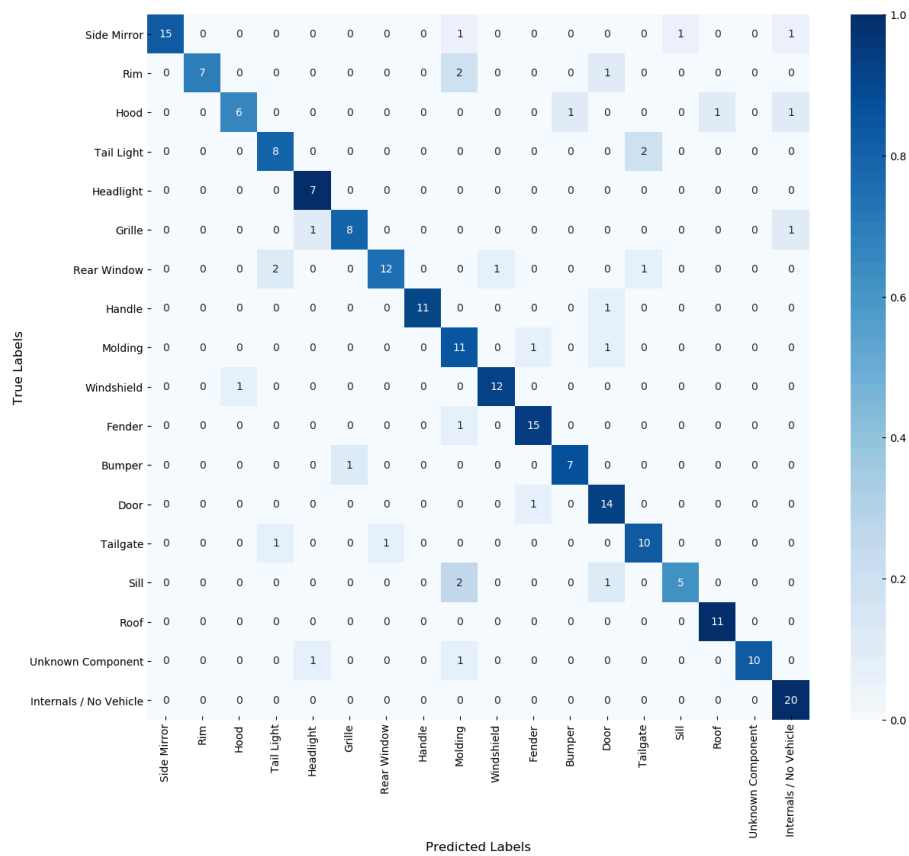


Figure 6.7: Confusion matrix for component prediction in the end-to-end system evaluation.

elucidating the potential financial implications of errors. Furthermore, the coefficient of determination,  $R^2$ , was computed to be 0.743, suggesting that 74% of the variance in the actual repair costs was accounted for by the model. A comprehensive presentation of these metrics can be found in Table 6.6, while the distribution of residuals is depicted in Figure 6.8.

Metric	Value (in euros, where applicable)
MAE	186
RMSE	278
$R^2$	0.743

Table 6.6: Regression capabilities of the end-to-end system.

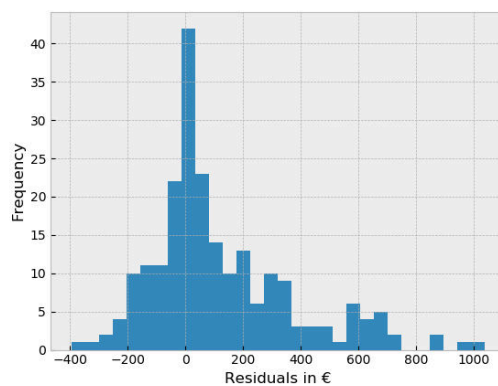


Figure 6.8: Distribution of residuals for the end-to-end system’s repair cost regression.

### Error Analysis

An examination of incorrect predictions identified several root causes:

1. Incorrectly identifying the type of photograph, constituting less than 2% of all predictions.
2. Erroneous component recognition, responsible for 15% of all predictions.
3. Misclassifications in discerning “Damage” from “No Damage”, forming 32% of all predictions.

To distill the findings, the end-to-end system showcased commendable performance in component prediction, evidenced by an accuracy of over 85%. However, the challenges faced in damage classification are evident, necessitating further scrutiny. The repair cost regression metrics highlight the model’s ability to make economically significant predictions, but also underscore the scope for improvement.

## 6.3 Conclusion

The results of the regression models for repair cost estimation underscore the efficacy of sophisticated machine learning models in handling the multifaceted dataset at hand. While Random Forest and XGBoost displayed marked superiority, the exploration emphasized the potential benefits of integrating visual data into the prediction pipeline. The incorporation of such features could greatly enhance the prediction model's input, leading to more accurate cost estimations. Additionally, the potential to leverage an extensive codebook database hints at a future where more detailed technical descriptors could refine the model's predictions. While the current models manifest robustness, the realm of advanced algorithms or even deep learning beckons, promising the potential to unearth intricate patterns in nuanced datasets.

The system evaluation section of this chapter provided insights into three principal areas: component prediction, damage classification, and repair cost regression. The system demonstrated impressive capabilities in component prediction with an accuracy nearing 86%. Nonetheless, the challenges tied to damage classification surfaced, revealing the intricate nature of damage detection in various contexts. The repair cost regression metrics, encapsulating MAE, RMSE, and the  $R^2$  value, elucidated the model's proficiency and the associated economic ramifications.

In summation, this chapter presents a robust framework for vehicle damage repair cost estimation and the broader system's end-to-end functionality. While the current accomplishments are notable, the insights gleaned underscore several avenues for further enhancement and refinement.

# Chapter 7

## Conclusions

The contemporary automotive industry has witnessed an evolution, not solely in vehicle advancements, but also in innovative sales and purchase methodologies. Among these methodologies, online car dealerships, exemplified by brumbrum, have come to the forefront, marrying traditional practices with modern technological advancements. A primary challenge confronting such businesses is the accurate assessment of car damages, a procedure deeply interwoven with the financial and operational stability of such establishments.

The research has meticulously delineated an end-to-end approach to address the intricacies and complexity of car damage assessment from photographs. It began by curating a dataset tailored to the unique demands of the Italian car market, emphasizing the quality and relevance of data. The subsequent modules on vehicle identification enriched the system with context. Distinguishing between interior and exterior photographs, precisely locating the vehicle, and identifying the make, model, and production year are foundational for the successive modules. These steps ensured the system was poised with all the necessary information before the rigorous processes of pose detection and damage assessment.

While the pose detection module accentuated the importance of understanding a vehicle's orientation for accurate damage assessment, the subsequent damage evaluation phase unraveled the intricacies involved in distinguishing damages from visual data. These challenges ranged from ambiguous reflections that can distort visual perceptions, the spatial proximity of damages to other vehicular components which can complicate differentiation, to inconsistent photograph qualities that can introduce variability in the data. Despite these obstacles, this research succeeded in elucidating the intricate process of converting visual inputs into actionable assessments.

Delving further into the component recognition task, it became evident that discerning specific vehicular parts is not straightforward due to ambiguities present in image data and the often subtle differences between vehicle components. Even though it was initially anticipated that the integration of pose estimation data would substantially boost performance, the actual enhancement was found to be marginal. This observation calls for a reassessment and potential development of a more tailored pose estimation technique that can cater specifically to the requirements of component recognition. The proximity of damages to adjacent components can also influence recognition accuracy, underscoring the need for more refined models.

Furthermore, the damage presence detection task spotlighted the complexities

inherent in identifying damages. Factors such as deceptive reflections, variations in photographic quality, and human biases in damage evaluation all converge to challenge the model's efficacy. Such intricacies necessitate the continual refinement of machine learning techniques and a deeper understanding of the visual data at hand. Reflecting on the instances where the model misclassified certain components or damages, it becomes clear that there is a pressing need for a robust, meticulously curated dataset. This dataset should be complemented by sophisticated feature recognition strategies to further enhance the system's precision and dependability.

Building upon the foundational insights established, the primary objective was to provide a robust estimate for vehicle damage repair costs. Drawing from factors such as the vehicle's make, model, production year, and specific damaged components identified, an endeavor was made to derive a reliable monetary figure. This transition from technical damage metrics to financial values stands paramount for service providers and consumers, bridging the gap between intricate data and real-world utility. In this context, the repair cost estimation was approached as a regression task.

The results highlighted the potency of advanced machine learning models in tackling the diverse dataset in question. Noteworthy models like Random Forest and XGBoost showcased their edge. However, a key revelation was the potential augmentation in prediction quality when integrating visual data. Incorporating features extracted from damage photographs, as demonstrated in studies [5, 111, 114], could significantly enrich the prediction model's input, potentially enhancing the accuracy of cost estimations. This approach, however, necessitates the collection of a comprehensive dataset where each repair case is documented with corresponding high-quality images, establishing a foundation for more nuanced and context-aware predictions. Another possible improvement could be harnessing an expansive codebook database. This suggests an avenue where nuanced technical descriptors could further refine and sharpen predictions.

Shifting focus to the system's holistic evaluation, several key areas stood out: component prediction, damage classification, and repair cost regression. The system showcased commendable skill in component prediction, achieving an accuracy close to 86%. Damage classification posed certain challenges, highlighting the intricacies involved in identifying damage across varied scenarios. In the context of repair cost regression, the model's mean absolute error during end-to-end validation was 186 euro. While this figure provides a foundation, it also underscores the need for further refinement to enhance predictive accuracy in real-world scenarios.

In culmination, the presented framework offers a sturdy foundation for vehicle damage repair cost estimation, seamlessly integrating into the system's broader functionality. While the present achievements deserve accolades, the insights garnered reveal numerous prospects for further augmentation and finesse. Emphasizing the essence of iterative development, it also stresses the importance of accommodating the fluid nature of real-world challenges and data nuances.



# Appendices

## Appendix A. Car Makes 3-character codes

Code	Car Make Name
ABA	Abarth
ALF	Alfa Romeo
AUD	Audi
BMW	BMW
CHC	Chevrolet
CHR	Chrysler
CIT	Citroën
CUP	Cupra
DAC	Dacia
DAE	Daewoo
DAI	Daihatsu
DOD	Dodge
DRM	DR Motor
DSA	DS Automobiles
FIA	Fiat
FOR	Ford
GRW	Great Wall
HON	Honda
HYU	Hyundai
INF	Infiniti
JAG	Jaguar
JEE	Jeep
KIA	Kia
LAD	Lada
LAN	Lancia
LEX	Lexus
LND	Land Rover
MAS	Maserati
MAZ	Mazda
MER	Mercedes-Benz
MIN	Mini
MIT	Mitsubishi
NIS	Nissan
OPE	Opel
PEU	Peugeot
POR	Porsche
REN	Renault
SAN	Ssangyong
SEA	Seat
SKO	Skoda
SMA	Smart
SUB	Subaru
SUZ	Suzuki
TAT	Tata
TES	Tesla
TOY	Toyota
VLK	Volkswagen
VOL	Volvo

Table 1: Translation of car make codes to names

# Bibliography

- [1] Bandi, H.; Joshi, S.; Bhagat, S.; Deshpande, A. Assessing car damage with convolutional neural networks. 2021 International Conference on Communication information and Computing Technology (ICCICT). 2021; pp 1–5.
- [2] De Deijn, J. Automatic car damage recognition using convolutional neural networks. 2018 Internship report MSc Business Analytics. 2018.
- [3] Li, P.; Shen, B.; Dong, W. An anti-fraud system for car insurance claim based on visual evidence. *arXiv preprint arXiv:1804.11207* **2018**,
- [4] Zhang, Q.; Chang, X.; Bian, S. B. Vehicle-damage-detection segmentation algorithm based on improved mask RCNN. *IEEE Access* **2020**, *8*, 6997–7004.
- [5] Mallios, D.; Xiaofei, L.; McLaughlin, N.; Del Rincon, J. M.; Galbraith, C.; Garland, R. Vehicle damage severity estimation for insurance operations using in-the-wild mobile images. *IEEE Access* **2023**,
- [6] van Ruitenbeek, R.; Bhulai, S. Convolutional Neural Networks for vehicle damage detection. *Machine Learning with Applications* **2022**, *9*, 100332.
- [7] Patil, K.; Kulkarni, M.; Sriraman, A.; Karande, S. Deep learning based car damage classification. 2017 16th IEEE international conference on machine learning and applications (ICMLA). 2017; pp 50–54.
- [8] Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. 2009 IEEE conference on computer vision and pattern recognition. 2009; pp 248–255.
- [9] Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016; pp 779–788.
- [10] Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C. L. Microsoft coco: Common objects in context. Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. 2014; pp 740–755.
- [11] HV, Y.; Karthik, V. Car Damage Detection and Analysis Using Deep Learning Algorithm For Automotive. **2019**,
- [12] Widjojo, D.; Setyati, E.; Kristian, Y. Integrated Deep Learning System for Car Damage Detection and Classification Using Deep Transfer Learning. 2022 IEEE 8th Information Technology International Seminar (ITIS). 2022; pp 21–26.

- [13] He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. Proceedings of the IEEE international conference on computer vision. 2017; pp 2961–2969.
- [14] Parhizkar, M.; Amirfakhrian, M. Car detection and damage segmentation in the real scene using a deep learning approach. *International Journal of Intelligent Robotics and Applications* **2022**, *6*, 231–245.
- [15] Imaam, F.; Subasinghe, A.; Kasthuriarachchi, H.; Fernando, S.; Haddela, P.; Pemadasa, N. Moderate automobile accident claim process automation using machine learning. 2021 International Conference on Computer Communication and Informatics (ICCCI). 2021; pp 1–6.
- [16] Fernando, N.; Kumarage, A.; Thiyaganathan, V.; Hillary, R.; Abeywardhana, L. Automated vehicle insurance claims processing using computer vision, natural language processing. 2022 22nd International Conference on Advances in ICT for Emerging Regions (ICTer). 2022; pp 124–129.
- [17] Poon, F.; Zhang, Y.; Roach, J.; Josephs, D.; Santerre, J. Modeling and Application of Neural Networks for Automotive Damage Appraisals. *SMU Data Science Review* **2021**, *5*, 3.
- [18] Pasupa, K.; Kittiworapanya, P.; Hongngern, N.; Woraratpanya, K. Evaluation of deep learning algorithms for semantic segmentation of car parts. *Complex & Intelligent Systems* **2022**, *8*, 3613–3625.
- [19] Kyu, P. M.; Woraratpanya, K. Car damage detection and classification. Proceedings of the 11th international conference on advances in information technology. 2020; pp 1–6.
- [20] Sruthy, C.; Kunjumon, S.; Nandakumar, R. Car damage identification and categorization using various transfer learning models. 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI). 2021; pp 1097–1101.
- [21] Naganandini, G.; Adarak, S.; Bagel, P. CNN-Based Car Damage Detection. Proceedings of the International Conference on Cognitive and Intelligent Computing: ICCIC 2021, Volume 1. 2022; pp 673–680.
- [22] Amirkhani, A.; Barshooi, A. H. DeepCar 5.0: vehicle make and model recognition under challenging conditions. *IEEE Transactions on Intelligent Transportation Systems* **2022**, *24*, 541–553.
- [23] Lu, L.; Wang, P.; Huang, H. A large-scale frontal vehicle image dataset for fine-grained vehicle categorization. *IEEE Transactions on Intelligent Transportation Systems* **2020**, *23*, 1818–1828.
- [24] Krause, J.; Stark, M.; Deng, J.; Fei-Fei, L. 3d object representations for fine-grained categorization. Proceedings of the IEEE international conference on computer vision workshops. 2013; pp 554–561.
- [25] Yu, F.; Seff, A.; Zhang, Y.; Song, S.; Funkhouser, T.; Xiao, J. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365* **2015**,

- [26] Kramberger, T.; Potočník, B. LSUN-Stanford car dataset: enhancing large-scale car image datasets using deep learning for usage in GAN training. *Applied Sciences* **2020**, *10*, 4913.
- [27] Buhrmester, M.; Kwang, T.; Gosling, S. D. Amazon’s Mechanical Turk: A new source of inexpensive, yet high-quality, data? *Perspectives on psychological science* **2011**, *6*, 3–5.
- [28] Peer, E.; Vosgerau, J.; Acquisti, A. Reputation as a sufficient condition for data quality on Amazon Mechanical Turk. *Behavior research methods* **2014**, *46*, 1023–1031.
- [29] Tafazzoli, F.; Frigui, H.; Nishiyama, K. A large and diverse dataset for improved vehicle make and model recognition. Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 2017; pp 1–8.
- [30] Yang, L.; Luo, P.; Change Loy, C.; Tang, X. A large-scale car dataset for fine-grained categorization and verification. Proceedings of the IEEE conference on computer vision and pattern recognition. 2015; pp 3973–3981.
- [31] Buzzelli, M.; Segantin, L. Revisiting the compcars dataset for hierarchical car classification: New annotations, experiments, and results. *Sensors* **2021**, *21*, 596.
- [32] Cheng, G.; Ma, C.; Zhou, P.; Yao, X.; Han, J. Scene classification of high resolution remote sensing images using convolutional neural networks. 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). 2016; pp 767–770.
- [33] Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition. 2015; pp 1–9.
- [34] Han, W.; Feng, R.; Wang, L.; Gao, L. Adaptive spatial-scale-aware deep convolutional neural network for high-resolution remote sensing imagery scene classification. IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. 2018; pp 4736–4739.
- [35] Kumar, B.; Gupta, H.; Ingale, S. P.; Vyas, O. Classification of Indoor–Outdoor Scene Using Deep Learning Techniques. Machine Learning, Image Processing, Network Security and Data Sciences: Select Proceedings of 3rd International Conference on MIND 2021. 2023; pp 517–535.
- [36] Xiao, J.; Hays, J.; Ehinger, K. A.; Oliva, A.; Torralba, A. Sun database: Large-scale scene recognition from abbey to zoo. 2010 IEEE computer society conference on computer vision and pattern recognition. 2010; pp 3485–3492.
- [37] López-Cifuentes, A.; Escudero-Vinolo, M.; Bescós, J.; García-Martín, Á. Semantic-aware scene recognition. *Pattern Recognition* **2020**, *102*, 107256.

- [38] Ye, O.; Li, Y.; Li, G.; Li, Z.; Gao, T.; Ma, T. Video scene classification with complex background algorithm based on improved CNNs. 2018 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC). 2018; pp 1–5.
- [39] Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. Proceedings of the IEEE conference on computer vision and pattern recognition. 2018; pp 4510–4520.
- [40] Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. International conference on machine learning. 2019; pp 6105–6114.
- [41] He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016; pp 770–778.
- [42] Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. Proceedings of the IEEE conference on computer vision and pattern recognition. 2018; pp 6848–6856.
- [43] Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* **2014**,
- [44] Canny, J. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence* **1986**, 679–698.
- [45] Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). 2005; pp 886–893.
- [46] Lowe, D. G. Distinctive image features from scale-invariant keypoints. *International journal of computer vision* **2004**, *60*, 91–110.
- [47] Bay, H.; Tuytelaars, T.; Van Gool, L. Surf: Speeded up robust features. Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7–13, 2006. Proceedings, Part I 9. 2006; pp 404–417.
- [48] LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **1998**, *86*, 2278–2324.
- [49] Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition. 2014; pp 580–587.
- [50] Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* **2015**, *28*.
- [51] Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A. C. Ssd: Single shot multibox detector. Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. 2016; pp 21–37.

- [52] Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *International journal of computer vision* **2010**, *88*, 303–338.
- [53] Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. 2012 IEEE conference on computer vision and pattern recognition. 2012; pp 3354–3361.
- [54] Tang, Y.; Zhang, C.; Gu, R.; Li, P.; Yang, B. Vehicle detection and recognition for intelligent traffic surveillance system. *Multimedia tools and applications* **2017**, *76*, 5817–5832.
- [55] Rameau, F.; Bailo, O.; Park, J.; Joo, K.; Kweon, I. S. Real-time multi-car localization and see-through system. *International Journal of Computer Vision* **2022**, *130*, 384–404.
- [56] Zhou, X.; Karpur, A.; Luo, L.; Huang, Q. Starmap for category-agnostic keypoint and viewpoint estimation. Proceedings of the European Conference on Computer Vision (ECCV). 2018; pp 318–334.
- [57] Kreiss, S.; Bertoni, L.; Alahi, A. Openpipaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Transactions on Intelligent Transportation Systems* **2021**, *23*, 13498–13511.
- [58] Chen, X.; Ma, H.; Wan, J.; Li, B.; Xia, T. Multi-view 3d object detection network for autonomous driving. Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2017; pp 1907–1915.
- [59] Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. Proceedings of the IEEE/CVF international conference on computer vision. 2019; pp 9627–9636.
- [60] Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. Proceedings of the IEEE international conference on computer vision. 2017; pp 2980–2988.
- [61] Wang, H.; Peng, J.; Zhao, Y.; Fu, X. Multi-path deep cnns for fine-grained car recognition. *IEEE Transactions on Vehicular Technology* **2020**, *69*, 10484–10493.
- [62] Wang, J.; Zheng, H.; Huang, Y.; Ding, X. Vehicle type recognition in surveillance images from labeled web-nature data using deep transfer learning. *IEEE Transactions on Intelligent Transportation Systems* **2017**, *19*, 2913–2922.
- [63] Hu, Q.; Wang, H.; Li, T.; Shen, C. Deep CNNs with spatially weighted pooling for fine-grained car recognition. *IEEE Transactions on Intelligent Transportation Systems* **2017**, *18*, 3147–3156.
- [64] Sochor, J.; Herout, A.; Havel, J. Boxcars: 3d boxes as cnn input for improved fine-grained vehicle recognition. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016; pp 3006–3015.

- [65] Lee, H. J.; Ullah, I.; Wan, W.; Gao, Y.; Fang, Z. Real-time vehicle make and model recognition with the residual SqueezeNet architecture. *Sensors* **2019**, *19*, 982.
- [66] Iandola, F. N.; Han, S.; Moskewicz, M. W.; Ashraf, K.; Dally, W. J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and 0.5 MB model size. *arXiv preprint arXiv:1602.07360* **2016**,
- [67] He, H.; Shao, Z.; Tan, J. Recognition of car makes and models from a single traffic-camera image. *IEEE Transactions on Intelligent Transportation Systems* **2015**, *16*, 3182–3192.
- [68] Liang, J.; Chen, X.; He, M.-l.; Chen, L.; Cai, T.; Zhu, N. Car detection and classification using cascade model. *IET Intelligent Transport Systems* **2018**, *12*, 1201–1209.
- [69] Boukerche, A.; Siddiqui, A. J.; Mammeri, A. Automated vehicle detection and classification: Models, methods, and techniques. *ACM Computing Surveys (CSUR)* **2017**, *50*, 1–39.
- [70] Zauner, C. Implementation and benchmarking of perceptual image hash functions. **2010**,
- [71] Orlov, I.; Buzzelli, M.; Schettini, R. Vehicle Pose Estimation: Exploring Angular Representations. Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. 2024.
- [72] Felzenszwalb, P. F.; Huttenlocher, D. P. Pictorial structures for object recognition. *International journal of computer vision* **2005**, *61*, 55–79.
- [73] LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *nature* **2015**, *521*, 436–444.
- [74] Cao, Z.; Simon, T.; Wei, S.-E.; Sheikh, Y. Realtime multi-person 2d pose estimation using part affinity fields. Proceedings of the IEEE conference on computer vision and pattern recognition. 2017; pp 7291–7299.
- [75] Güler, R. A.; Neverova, N.; Kokkinos, I. Densepose: Dense human pose estimation in the wild. Proceedings of the IEEE conference on computer vision and pattern recognition. 2018; pp 7297–7306.
- [76] David, L. Distinctive image features from scale-invariant keypoints. *International journal of computer vision* **2004**, *60*, 91–110.
- [77] Lepetit, V.; Moreno-Noguer, F.; Fua, P. EP n P: An accurate O (n) solution to the P n P problem. *International journal of computer vision* **2009**, *81*, 155–166.
- [78] Kendall, A.; Grimes, M.; Cipolla, R. PoseNet: A convolutional network for real-time 6-dof camera relocalization. Proceedings of the IEEE international conference on computer vision. 2015; pp 2938–2946.



- [79] Mousavian, A.; Anguelov, D.; Flynn, J.; Kosecka, J. 3d bounding box estimation using deep learning and geometry. Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2017; pp 7074–7082.
- [80] Pavlakos, G.; Zhou, X.; Chan, A.; Derpanis, K. G.; Daniilidis, K. 6-dof object pose from semantic keypoints. 2017 IEEE international conference on robotics and automation (ICRA). 2017; pp 2011–2018.
- [81] Qin, J. W., Zengyi; Lu, Y. Monogrnet: A geometric reasoning network for monocular 3d object localization. Proceedings of the AAAI Conference on Artificial Intelligence. 2019.
- [82] Xiao, Y.; Qiu, X.; Langlois, P.-A.; Aubry, M.; Marlet, R. Pose from shape: Deep pose estimation for arbitrary 3d objects. *arXiv preprint arXiv:1906.05105* **2019**,
- [83] Su, H.; Qi, C. R.; Li, Y.; Guibas, L. J. Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. Proceedings of the IEEE international conference on computer vision. 2015; pp 2686–2694.
- [84] Grabner, A.; Roth, P. M.; Lepetit, V. 3d pose estimation and 3d model retrieval for objects in the wild. Proceedings of the IEEE conference on computer vision and pattern recognition. 2018; pp 3022–3031.
- [85] Nie, W.-Z.; Jia, W.-W.; Li, W.-H.; Liu, A.-A.; Zhao, S.-C. 3d pose estimation based on reinforce learning for 2d image-based 3d model retrieval. *IEEE Transactions on Multimedia* **2020**, *23*, 1021–1034.
- [86] Mahendran, S.; Lu, M. Y.; Ali, H.; Vidal, R. Monocular object orientation estimation using riemannian regression and classification networks. *arXiv preprint arXiv:1807.07226* **2018**,
- [87] Geiger, P. L., Andreas; Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. IEEE conference on computer vision and pattern recognition. 2012.
- [88] Xiang, Y.; Mottaghi, R.; Savarese, S. Beyond pascal: A benchmark for 3d object detection in the wild. IEEE winter conference on applications of computer vision. 2014; pp 75–82.
- [89] Song, X.; Wang, P.; Zhou, D.; Zhu, R.; Guan, C.; Dai, Y.; Su, H.; Li, H.; Yang, R. ApolloCar3d: A large 3d car instance understanding benchmark for autonomous driving. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019; pp 5452–5462.
- [90] Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* **2014**, *15*, 1929–1958.
- [91] Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M., *et al.* Imagenet large scale visual recognition challenge. *International journal of computer vision* **2015**, *115*, 211–252.

- [92] Prokudin, S.; Gehler, P.; Nowozin, S. Deep directional statistics: Pose estimation with uncertainty quantification. Proceedings of the European conference on computer vision (ECCV). 2018; pp 534–551.
- [93] Dani, M.; Narain, K.; Hebbalaguppe, R. 3DPoseLite: A compact 3d pose estimation using node embeddings. Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2021; pp 1878–1887.
- [94] Klee, D. M.; Biza, O.; Platt, R.; Walters, R. Image to sphere: Learning equivariant features for efficient pose prediction. *arXiv preprint arXiv:2302.13926* **2023**,
- [95] Jurado-Rodríguez, D.; Jurado, J. M.; Pádua, L.; Neto, A.; Munoz-Salinas, R.; Sousa, J. J. Semantic segmentation of 3D car parts using UAV-based images. *Computers & Graphics* **2022**, *107*, 93–103.
- [96] Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. 2015; pp 234–241.
- [97] Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016; pp 2818–2826.
- [98] Chua, A. C.; Mercado, C. R. B.; Pin, J. P. R.; Tan, A. K. T.; Tinhay, J. B. L.; Dadios, E. P.; Billones, R. K. C. Damage Identification of Selected Car Parts Using Image Classification and Deep Learning. 2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM). 2021; pp 1–5.
- [99] Khanal, S. R.; Amorim, E. V.; Filipe, V. Classification of car parts using deep neural network. CONTROL 2020: Proceedings of the 14th APCA International Conference on Automatic Control and Soft Computing, July 1-3, 2020, Bragança, Portugal. 2021; pp 582–591.
- [100] Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* **2014**,
- [101] Dwivedi, M.; Malik, H. S.; Omkar, S.; Monis, E. B.; Khanna, B.; Samal, S. R.; Tiwari, A.; Rathi, A. Deep learning-based car damage classification and detection. Advances in Artificial Intelligence and Data Engineering: Select Proceedings of AIDE 2019. 2021; pp 207–221.
- [102] Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S., *et al.* An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* **2020**,
- [103] Gandhi, R. Deep Learning Based Car Damage Detection, Classification and Severity. *International Journal* **2021**, *10*.

- [104] Rio-Torto, I.; Campaniço, A. T.; Pereira, A.; Teixeira, L. F.; Filipe, V. Automatic quality inspection in the automotive industry: a hierarchical approach using simulated data. 2021 IEEE 8th International Conference on Industrial Engineering and Applications (ICIEA). 2021; pp 342–347.
- [105] Waqas, U.; Akram, N.; Kim, S.; Lee, D.; Jeon, J. Vehicle damage classification and fraudulent image detection including moiré effect using deep learning. 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE). 2020; pp 1–5.
- [106] Anwer, M. A.; Shareef, S. M.; Ali, A. M. Accident vehicle types classification: a comparative study between different deep learning models. *Indonesian Journal of Electrical Engineering and Computer Science* **2021**, *21*, 1474–1484.
- [107] Chaudhari, S. S. Deep Learning Networks for Detection, Classification and Analysis of Car Damage. Ph.D. thesis, Dublin, National College of Ireland, 2022.
- [108] Dhieb, N.; Ghazzai, H.; Besbes, H.; Massoud, Y. A very deep transfer learning model for vehicle damage detection and localization. 2019 31st International Conference on Microelectronics (ICM). 2019; pp 158–161.
- [109] Tian, X.; Han, H. Deep convolutional neural networks with transfer learning for automobile damage image classification. *Journal of Database Management (JDM)* **2022**, *33*, 1–16.
- [110] Mohammed, N. A.; Potrus, M. Y.; Ali, A. M. Deep Learning Based Car Damage Classification and Cost Estimation. *Zanco Journal of Pure and Applied Sciences* **2023**, *35*, 1–9.
- [111] Jameel, M.; Arif, M. u. I.; Hintsches, A.; Schmidt-Thieme, L. Automation of leasing vehicle return assessment using deep learning models. Machine Learning and Knowledge Discovery in Databases: Applied Data Science Track: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part IV. 2021; pp 259–274.
- [112] Breiman, L. Random forests. *Machine learning* **2001**, *45*, 5–32.
- [113] Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016; pp 785–794.
- [114] Ul Islam Arif, M.; Wieland, F.; Bianchin, C.; Hintsches, A.; Lange, K.; Schmidt-Thieme, L. Object Regression: Multi-Modal Data Enhanced Object Detection for Leasing Vehicle Return Assessment. 2022 International Conference on Digital Image Computing: Techniques and Applications (DICTA). 2022; pp 1–16.