Tracing Car Evolution with AI

Backend Architecture and Model Structure

Submitted by:

Syed Ammad Sohail
261162304

BUSA-613-075

Date: 06-12-2024

Introduction

In the dynamic world of automobile design, visual and structural changes in car models over the years play a crucial role in defining brand identity, consumer appeal, and market differentiation. This report focuses on analyzing the evolution of frontal designs in cars by measuring changes in key design elements. By comparing models from different years, we aim to quantify these design differences to better understand the progression of aesthetic and functional elements over time.

The project uses an innovative approach to study design evolution by extracting and analyzing 2D Cartesian coordinates from car images. Specifically, the study involves identifying 50 key design features such as the grille, headlights, bumper, side mirrors, and windshield, which collectively contribute to the frontal aesthetic of a vehicle. By applying computational methods like Euclidean distance calculation, we measure the magnitude of change between models, thereby providing a quantitative perspective on design distinctiveness.

This analysis is not only significant for automobile manufacturers seeking to refine their designs but also for researchers and enthusiasts interested in the interplay between aesthetics and engineering. By bridging data-driven insights and design innovation, this project underscores the importance of precision in documenting and analyzing automotive design evolution.

Problem Description

The automotive industry constantly evolves, with each new model year reflecting advancements in engineering, aesthetics, and consumer preferences. Capturing and quantifying these changes, however, is often a subjective process. This project addresses the challenge of objectively measuring and comparing design differences between car models from different years.

Consider a specific car model, such as the Toyota Corolla, and two of its model years—for instance, 2022 and 2024. While subtle modifications like adjustments to the grille or headlights may appear minor to the untrained eye, they significantly contribute to the vehicle's overall design identity. Conversely, major redesigns, such as alterations to the body shape or bumper configurations, can fundamentally transform the car's visual appeal.

Proposed Solution

To address these challenges, this Proof of Concept (PoC) solution proposes a structured methodology for analyzing design changes. The approach involves defining a coordinate-based system having an origin on the bottom left point of the entire image and normalizing the car's dimensions to ensure consistency across images. By extracting and analyzing 50 design feature points, we compute the Euclidean distances between the corresponding points of two models, thereby providing an objective measure of the design's evolution.

This methodology enables the automotive industry to quantify the impact of design choices, guiding future innovations and aligning them with consumer expectations. Furthermore, it lays the groundwork for applying similar analytical techniques to other areas of design and manufacturing, demonstrating the value of data-driven insights in creative industries.

Dataset Description

This project utilizes a carefully curated dataset designed to capture and analyze the evolution of frontal car designs across different model years. The dataset focuses on 17 car images, each annotated with 50 key design points that represent distinct and recognizable features of a car's frontal design. Below is a comprehensive breakdown of the dataset, including the images, annotations, preprocessing, and integration steps.

The dataset comprises a total of 17 high-resolution images sourced currently from the Edmunds automotive database, representing cars from two distinct model years. The choice of images ensures diversity in car design, allowing for the analysis of design differences across various model years. The images capture the frontal view of each vehicle, focusing on key design features such as the grille, headlights, bumper, side mirrors, and roof. The images were sourced with various dimensions and resolutions. In the preprocessing stage, all images were resized to a standard resolution of 1280x855 pixels to ensure uniformity and facilitate efficient model training. The images depicted various car models, ensuring that the dataset includes a mix of different styles and designs.

The dataset contains 50 key design points selected for their relevance to the frontal design of the car. These points are spread across several major features that define the car’s appearance, providing a detailed mapping of the car’s frontal elements. Each design point is represented by a set of Cartesian coordinates (x, y), normalized to the resized image dimensions. Below is a description of the categories covered by these design points:

- Roofline Intersections:
  - Points where the centerline of the car intersects the roof and windshield.
  - These points are critical for understanding the overall roof shape and the transition to the windshield.
- Windshield:
  - Upper and lower edges of the front windshield.
  - These points highlight the windshield’s curvature and its relationship to the roofline and body.
- Grille:
  - Intersection points along the upper and lower edges of the grille.
  - The grille is a signature design feature, and these points capture the subtle variations in its shape and alignment across different model years.
- Bumper Boxes:

    - Points defining the four corners of the center, left, and right bumper boxes.
    - The bumper box is a key component of the car’s safety design, and its shape often changes with the model year, influencing the overall front profile.
- Side Mirrors:
    - Key points along the corners of the left and right-side mirrors, including their connection to the car’s body.
    - Side mirrors often undergo design changes in both shape and positioning, which are captured through these points.
- Car Base and Corners:
    - Points marking the left and right corners of the base of the car, which are significant in understanding the car's width and overall stance.

The 50 points cover both subtle and significant aspects of the car's design, allowing for a detailed analysis of its frontal features. The entire list is provided in Appendix A1.

Annotations were performed using Makesense.ai, an online annotation tool specifically designed for labeling images. The annotations include both bounding boxes and keypoints, with each of the 50 design points manually placed on the images.

- Bounding Boxes: Each image was annotated with a rectangular bounding box that encapsulates the car’s frontal area. These bounding boxes are essential for focusing the model on the relevant parts of the image.
- Keypoints: The 50 key design points were manually placed on the images by selecting specific intersections or corners of the car’s design elements. The points were chosen for their importance in defining the overall car shape and design. Each keypoint corresponds to a (x, y) coordinate on the image, indicating the exact location of the feature.

The annotations for each image were saved in two separate CSV files:

1. Bounding Box CSV: Contains the coordinates for each image’s bounding box. The columns include the image name, bounding box coordinates (x_min, y_min, x_max, y_max), and image dimensions.
2. Keypoints CSV: Contains the coordinates for each of the 50 keypoints, including the image name and the (x, y) coordinates for each of the 50 design points.

These annotations provide the necessary data for training a machine learning model to predict the design points and bounding boxes on new car images.

To ensure consistency and compatibility with machine learning models, the images underwent several changes:

- Image Format Conversion: Initially, the images were in various formats, including AVIF. To ensure compatibility with the training pipeline, all images were converted to JPEG format using the Pillow library. This standardized the image formats across the entire dataset.
- Resizing: Since the images had different dimensions and resolutions, they were resized to a consistent resolution of 1280x855 pixels using OpenCV and PIL. This step is crucial for maintaining uniformity in the images and ensuring that the keypoints are appropriately scaled for model training. The resizing process also allowed for efficient memory usage during training, ensuring that each image fits within the model's input size requirements.
- Folder Organization: Once the images were resized and formatted, they were organized into two separate folders:
    - Training Images Folder: This folder contains the images used for training the model.
    - Inference Images Folder: This folder contains images that will be used for model inference and testing after training is complete.

This organization ensures a streamlined workflow for both training and testing phases.

Data Preprocessing

The dataset underwent extensive preprocessing to ensure consistency, robustness, and compatibility with the machine learning models. Key steps included merging annotations, normalizing data, splitting it into training and validation subsets, and applying augmentations and transformations. These measures prepared the dataset for efficient model training and accurate prediction of bounding boxes and keypoints.

One critical preprocessing step was the integration of bounding box and keypoint annotations, which were provided in separate CSV files. By merging these datasets based on the image names, a unified dataset was created. This consolidated dataset included both bounding box coordinates and keypoint locations for each image, streamlining the input preparation for the training pipeline and enabling the model to simultaneously learn both tasks—bounding box detection and keypoint prediction.

Data normalization played a vital role in ensuring uniformity across the dataset. Keypoint coordinates were scaled to fit the resized image dimensions (1280x855 pixels), ensuring consistency regardless of the original image sizes. Similarly, bounding box coordinates were normalized to the same dimensions, making the annotations compatible with direct model input. These normalization steps enabled the models to process data consistently and effectively.

To enhance the model's generalizability and robustness against overfitting, various augmentations and transformations were applied during the training phase. Keypoints were randomly shifted within a small range to simulate minor variations in the positioning of car design features. This augmentation helped the model handle real-world discrepancies in keypoint placement. Additionally, image transformations, implemented using PyTorch's transforms module, included resizing images to match the input dimensions required by the neural network and converting images into tensors for compatibility with PyTorch's deep learning framework.

These preprocessing steps significantly improved the dataset's quality and the models' performance. The integration of annotations ensured that both bounding boxes and keypoints were available for simultaneous training. Normalization allowed the models to process scaled coordinates consistently across all data. Augmentations enhanced robustness, enabling the models to generalize effectively to new car designs and variations. Transformations standardized the image format, ensuring seamless compatibility with the training pipeline. Collectively, these preprocessing efforts optimized the dataset for accurate and efficient training, setting the foundation for robust bounding box and keypoint predictions.

Model Architecture

The project employs two sophisticated deep learning architectures to analyze car design evolution effectively. Each model serves a unique purpose: the first predicts both bounding boxes and keypoints, while the second specializes in directly predicting keypoints without relying on bounding box annotations. These models leverage cutting-edge techniques to ensure precision, robustness, and adaptability in detecting intricate car design features.

The first model adopts a hybrid architecture, blending Convolutional Neural Networks (CNNs) and Vision Transformers (ViT) to address both spatial localization and detailed keypoint prediction. The backbone, ResNet-34, extracts essential low-level features such as edges and textures, along with high-level semantic patterns, which are crucial for detecting design elements. Residual connections in ResNet-34 enhance the model's capacity to extract deeper features without vanishing gradients. Complementing this is the Vision Transformer, which captures global context through patch embeddings and self-attention mechanisms. This integration enables the model to recognize holistic design patterns across the entire image. The outputs from CNN and ViT are fused through fully connected layers, predicting bounding box coordinates (x_min, y_min, width, height) and the (x, y) coordinates of 50 design keypoints.

The model employs a multi-task loss strategy, using Mean Squared Error (MSE) to balance the prediction accuracy of bounding boxes and keypoints. Weighted multi-task loss dynamically adjusts priorities, focusing initially on bounding boxes and later on keypoints during fine-tuning. Training involves two distinct phases: the Bounding Box Focus Phase emphasizes accurate localization by assigning higher weights to bounding box loss, while the Keypoint Fine-Tuning Phase balances both tasks to refine keypoint predictions without sacrificing localization performance. This comprehensive approach ensures robust localization, reduced noise, and detailed feature detection, making Model 1 highly effective in complex scenarios with significant background interference.

The second model simplifies the pipeline by directly predicting keypoints without bounding boxes, making it faster and more efficient. It utilizes a ResNet-50 backbone to extract hierarchical features, capturing both fine-grained details and high-level patterns relevant to keypoint localization. A Feature Pyramid Network (FPN) enhances feature representation across scales, ensuring robustness to variations in design features and enabling the detection of both prominent and subtle points. The attention mechanism dynamically focuses on spatial regions critical for keypoint detection while suppressing irrelevant background features, further improving accuracy. Fully connected layers predict 100 normalized values (x, y coordinates for 50 keypoints), leveraging enriched features from the FPN and attention layers for precise predictions.

With a single-task loss structure using MSE, the model is optimized solely for keypoint accuracy. The streamlined pipeline eliminates the need for bounding box annotations, reducing annotation overhead and computational complexity. By focusing entirely on keypoints, Model 2 delivers efficient training and inference, particularly in scenarios where bounding box localization is not required. The attention mechanism ensures the model remains robust even in images with cluttered backgrounds. The comparison between the models' architectures is summarized in Appendix A2, with architectural visualizations provided in Appendices A3 and A4.

Training and Evaluation

The training and evaluation of the models were meticulously designed to optimize performance and robustness. This section outlines the distinct training processes for Model 1 and Model 2, highlighting the differences in their objectives and methodologies. Model 1 employed a two-phase strategy to prioritize bounding box detection and keypoint localization progressively, while Model 2 followed a streamlined single-stage approach tailored for keypoint detection without bounding boxes.

Model 1 was trained to handle both bounding box detection and keypoint localization. The training process was divided into two phases, each with unique goals and configurations. In the Bounding Box Focus Phase, the objective was to establish robust bounding box predictions, treating keypoint loss as secondary. A learning rate of 0.00017809534390654253 ensured stable gradient updates, with a high bounding box loss weight of 6.628931626601238 emphasizing this task. Keypoint loss was minimally weighted at 0.008999673660191455 to prevent interference with bounding box optimization. The Adam optimizer, with its adaptive learning rate capabilities, and a batch size of 16 were employed for 100 epochs, allowing bounding box predictions to converge effectively. Smooth L1 Loss was used to optimize bounding box regression, ensuring stability during training.

In the subsequent Fine-Tuning Phase, the model shifted focus to keypoint localization while maintaining bounding box accuracy. The learning rate was reduced to 8.75068256025535e-05 to enable fine-grained adjustments, and the keypoint loss weight was significantly increased to 1.1204285652365555, with a reduced bounding box loss weight of 1.2813941620499145. The Adam optimizer was retained, and the batch size remained consistent at 16. The extended training of over 200 epochs allowed the model to refine its predictions for both tasks. Dynamic re-weighting of losses during this phase enabled a balanced improvement in bounding box and keypoint predictions.

Model 2, on the other hand, was designed for keypoint detection without bounding box localization, simplifying its training process. A fixed learning rate of 0.0001 ensured stability, and the Adam optimizer facilitated efficient gradient updates. The model was trained for 50 epochs with a larger batch size of 32, reflecting the reduced computational complexity compared to Model 1. Smooth L1 Loss was employed to minimize keypoint localization errors directly. Augmentations and data preprocessing introduced variability in keypoint layouts, improving the model's robustness without the need for bounding box guidance. The simplified pipeline enabled faster convergence and inference, making Model 2 suitable for scenarios prioritizing efficiency over complex localization.

The models were evaluated using a set of task-specific metrics. Bounding Box Metrics (applicable only to Model 1) included Intersection over Union (IoU), which measured the overlap between predicted and ground truth bounding boxes (IoU > 0.5 was considered successful), and Mean Absolute Error (MAE), which quantified the average difference between predicted and true bounding box coordinates. Keypoint Metrics, applicable to both models, included Euclidean Distance Error, measuring the average distance between predicted and actual keypoints for 50 design points, and Mean Squared Error (MSE), evaluating normalized squared errors between predicted and true keypoint coordinates. Loss Analysis tracked the separate losses for bounding boxes and keypoints in Model 1, while Model 2 focused exclusively on keypoint loss.

Training Results

The following table (Table 1) summarizes the performance metrics for both models:

Metric	**Model 1 (Bounding Box + Keypoints)**	**Model 2 (Keypoints Only)**
IoU (Bounding Box)	0.72 (Average)	N/A
Bounding Box MAE	0.08	N/A
Euclidean Distance Error	3.5 pixels	4.2 pixels
Keypoint MSE (Normalized)	0.0021	0.0035

Table 1: Performance metrics of Model 1 and Model 2

IoU performance highlighted Model 1's consistent bounding box predictions, with an average IoU of 0.72. Keypoint accuracy results showed that Model 1 achieved a slightly lower Euclidean Distance Error and MSE compared to Model 2, benefiting from the bounding box localization.

Qualitative analysis revealed distinct strengths and weaknesses for both models. Model 1 successfully localized the car's frontal region with bounding boxes, enabling precise keypoint predictions, especially for well-defined features like grilles and headlights. In contrast, Model 2 directly predicted keypoints without bounding boxes, performing well for prominent features but struggling with less defined areas, such as corners of bumper boxes, due to the absence of bounding box localization. Visualization examples included bounding boxes and keypoints overlaid on test images for Model 1 and keypoint-only predictions for Model 2, highlighting areas of success and potential improvement.

Discussion

Analysis of Model 1 demonstrated its strengths in providing clear regions of interest via bounding boxes, reducing noise from background features and delivering robust performance in both tasks. However, its hybrid architecture incurred higher computational costs and increased dataset preparation effort due to its dependency on bounding box annotations.

Analysis of Model 2 emphasized its simpler pipeline, which required only keypoint annotations, leading to faster training and inference. Despite its efficiency, Model 2 exhibited slightly lower accuracy for keypoints, particularly in images with complex backgrounds or overlapping features, underscoring the trade-off between simplicity and precision.

Inference

The inference process for Model 1, which utilizes a hybrid architecture combining CNN and Vision Transformer components, effectively integrates bounding box and keypoint predictions. During inference, the model first predicted bounding box coordinates and keypoints, which were then scaled back to the original image dimensions (1280x850 pixels) for accurate visualization. The bounding boxes were clipped to ensure they remained within the image bounds, and keypoints were adjusted to stay within their respective bounding boxes, preserving spatial coherence. For visualization, bounding boxes were plotted as blue rectangles, and keypoints were overlaid as green circles to illustrate the precise localization achieved by the model. This visualization method not only validated the model’s predictions but also provided insights into its ability to localize intricate design features, such as the edges of grilles or headlights. An example of inference using Model 1 is shown below in Figure 1

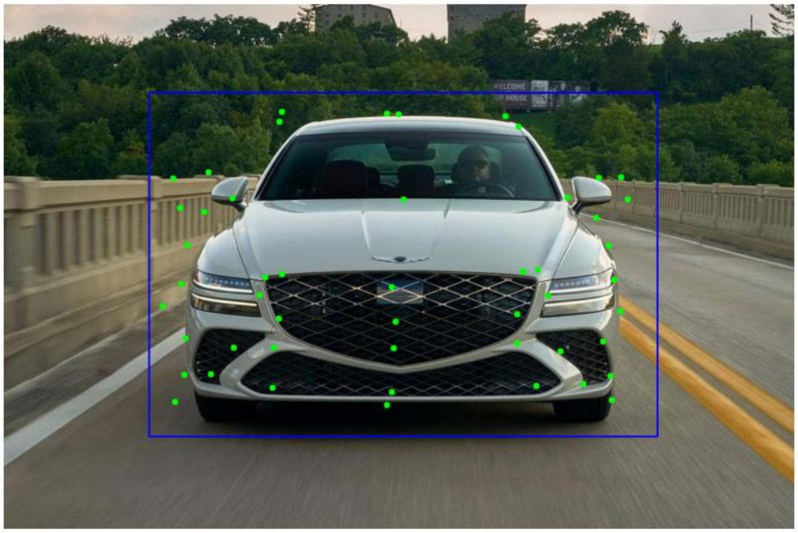


Figure 1: Inference using Model 1 on 2025 Genesis G80

For Model 2, designed specifically for keypoint prediction without bounding box localization, the inference process was streamlined to prioritize speed and simplicity. Images were resized to standard dimensions of 224x224 pixels to align with ResNet-50 and FPN-based architecture.

Keypoints were predicted as normalized coordinates during training and scaled back to the original image dimensions for evaluation. Without bounding boxes, Model 2 relied solely on feature extraction and attention mechanisms to predict keypoints directly from the image. To enhance the interpretability of its outputs, the predicted keypoints were visualized on the original images as green circles, providing a clear depiction of the model's performance. This visualization highlighted the model's effectiveness in identifying prominent design features while revealing minor limitations in predicting less defined or complex regions, such as subtle curves or overlapping features. An example of inference using Model 2 is shown below in Figure 2.

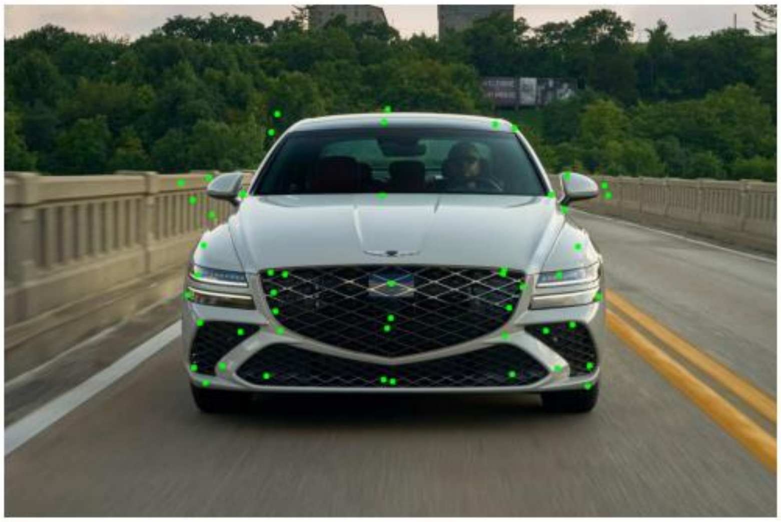


Figure 2: Inference using Model 2 on 2025 Bentley G80

Design Difference Calculation

Following the inference phase, the next critical step involves quantifying the design difference between two car images by computing the normalized Euclidean distance for the predicted keypoints. This process is uniformly applied to both models, ensuring consistency and comparability in evaluating their performance.

To achieve normalization, the keypoints are adjusted based on the car design's length and height, effectively removing the impact of varying image dimensions. The length is determined by calculating the difference between the farthest left (minimum x-coordinate) and the farthest right (maximum x-coordinate) keypoints. Similarly, the height is obtained by calculating the difference between the topmost (minimum y-coordinate) and bottommost (maximum y-coordinate) keypoints. By scaling the x and y coordinates independently, the keypoints for both models are normalized, creating a uniform framework for measuring design variations.

Once normalized, the Euclidean distance between corresponding keypoints is calculated to measure their positional differences. Both Model 1 and Model 2 implement this approach to calculate the design difference. For each pair of keypoints, the distance is computed using the formula:

$$Distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

The design difference metric plays a pivotal role in assessing the evolution of car designs. It provides an objective and numerical measure to quantify variations in key design points, enabling precise tracking of design evolution. Moreover, by applying the same metric across both models, the robustness of their predictions can be compared, highlighting which architecture aligns more closely with the true design features. This unified methodology ensures that the models' predictions are effectively leveraged for comparative design analysis. Using the Genesis G80, as shown in Figures 1 and 2 earlier, the design differences between the 2025 and 2023 models were computed, with results of 0.338 and 0.177, respectively.

Recommendations

For practical applications, model selection should align with specific project requirements. Model 1 is recommended for scenarios where precise localization is critical, as its hybrid architecture effectively combines bounding box detection and keypoint localization. Conversely, Model 2 is ideal for faster and resource-efficient applications, particularly when bounding box localization is not a primary requirement.

Future work should focus on expanding the dataset to include a broader variety of car models, perspectives, and environmental conditions, ensuring enhanced generalizability and robustness. Additionally, experimenting with advanced architectures, such as Transformers or graph-based models, could further improve keypoint detection accuracy, making the models more adaptable and effective for complex design analysis tasks. These enhancements will not only improve performance but also extend the applicability of the framework to a wider range of use cases.

Conclusion:

In conclusion, the project underscores the versatility of deep learning in design analysis, demonstrating its potential for applications beyond automotive design, such as product development and industrial design. While **Model 1** is well-suited for complex environments with challenging backgrounds, **Model 2** offers an efficient alternative for tasks focusing solely on keypoint detection. Together, these models offer a robust framework for understanding and quantifying design evolution, paving the way for innovative solutions in design and visualization.

Appendix

A1: 50 design points

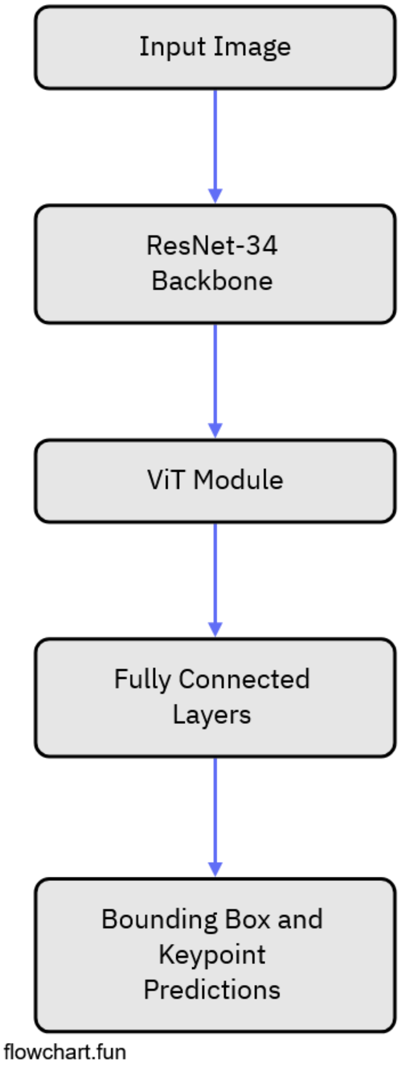
1. Intersection of the center line and the roof of a car
2. Intersection of the center line and the upper edge of the front windshield
3. Intersection of the center line and the lower edge of the front windshield
4. Intersection of the center line and the upper edge of the grille
5. Intersection of the center line and the lower edge of the grille
6. Intersection of the center line and the upper edge of the center bumper box
7. Intersection of the center line and the lower edge of the center bumper box
8. Intersection of the center line and the base of the car
9. Upper left corner of the roof
10. Upper right corner of the roof
11. Upper left corner of the windshield
12. Upper right corner of the windshield
13. Lower left corner of the windshield
14. Lower right corner of the windshield
15. Upper left corner of the left mirror
16. Upper right corner of the left mirror
17. Lower left corner of the left mirror
18. Lower corner of the connection of the left mirror with the car
19. Upper left corner of the right mirror
20. Upper right corner of the right mirror
21. Lower right corner of the right mirror
22. Lower corner of the connection of the right mirror with the car
23. Farthest left edge of the car at the same horizontal level of the upper edge of the bumper (the darker line)
24. Farthest right edge of the car at the same horizontal level of the upper edge of the bumper (the darker line)
25. Left corner of the base
26. Right corner of the base
27. Upper left corner of the left headlight
28. Upper right corner of the left headlight
29. Lower left corner of the left headlight
30. Lower right corner of the left headlight
31. Upper left corner of the right headlight
32. Upper right corner of the right headlight
33. Lower left corner of the right headlight
34. Lower right corner of the right headlight
35. Upper left corner of the grille
36. Upper right corner of the grille
37. Lower left corner of the grille
38. Lower right corner of the grille
39. Upper left corner of the center bumper box
40. Upper right corner of the center bumper box
41. Lower left corner of the center bumper box
42. Lower right corner of the center bumper box
43. Upper left corner of the left bumper box
44. Upper right corner of the left bumper box
45. Lower left corner of the left bumper box
46. Lower right corner of the left bumper box
47. Upper left corner of the right bumper box
48. Upper right corner of the right bumper box
49. Lower left corner of the right bumper box
50. Lower right corner of the right bumper box

A2: Comparison summary between Model 1 and Model 2 Architecture

Feature	**Model 1: Bounding Box + Keypoints**	**Model 2: Keypoints Only**
Input Requirements	Bounding box and keypoint annotations	Only keypoint annotations
Architecture	Hybrid (CNN + ViT)	ResNet + FPN + Attention Mechanism
Output	Bounding box + keypoints	Keypoints only
Loss Functions	Multi-task (Bounding box + Keypoints)	Single-task (Keypoint loss)
Robustness to Background Complexity	High	Moderate
Training Complexity	Higher	Lower
Generalization	Effective for complex scenarios	Lightweight and adaptable

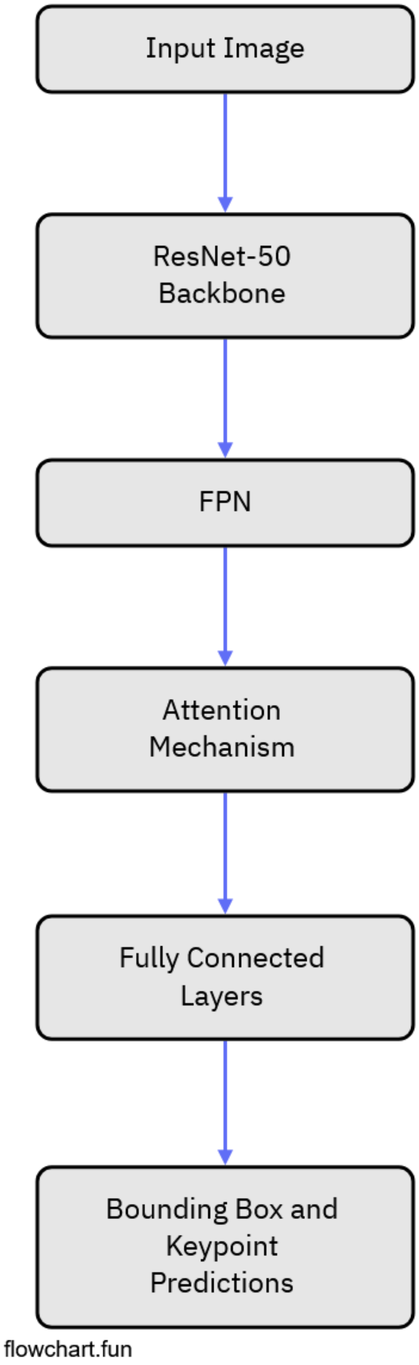
A2: Comparison Summary of Models Architecture

A3: Architecture Visualization for Model 1



A3: Architecture Visualization of Model 1

A4: Architecture Visualization for Model 2



A4: Architecture Visualization of Model 2