**BABEŞ-BOLYAI UNIVERSITY CLUJ-NAPOCA**
**FACULTY OF MATHEMATICS AND COMPUTER SCIENCE**
**SPECIALIZATION COMPUTER SCIENCE IN ENGLISH**

# DIPLOMA THESIS

# ArrowVision: Precise Road Arrow Marking Detection with Deformable DETR

**Supervisor**
**Prof. Univ. Dr. Dioșan Laura**

*Author*
*Enache Ioan-Vlad*

2025

**UNIVERSITATEA BABEŞ-BOLYAI CLUJ-NAPOCA**
**FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ**
**SPECIALIZAREA INFORMATICA IN LIMBA ENGLEZA**

# LUCRARE DE LICENŢĂ

# ArrowVision: Detectarea Precisă a Marcajelor Rutiere cu Deformable DETR

**Conducător ştiinţific**
**Prof. Univ. Dr. Dioșan Laura**

*Absolvent*
*Enache Ioan-Vlad*

2025

ABSTRACT

This study investigates optimized configurations of Deformable DETR with ResNet backbone architectures for painted road sign detection, addressing critical challenges in autonomous driving perception. Building upon transformer-based detection frameworks, we systematically evaluate hybrid architectures combining convolutional backbone strengths with deformable attention mechanisms. Through systematic experimentation with architectural variants and comprehensive ablation studies—including variations in backbone depth, positional embedding strategies, and training protocols—the research aims to maximize detection performance across all categories. Results demonstrate that careful model and hyperparameter selection within the Deformable DETR–ResNet paradigm yields substantial improvements, especially in terms of generalization, small object detection, and overall mean average precision. These advancements underline Deformable DETR's practical value for autonomous driving and highlight critical design considerations for deploying high-accuracy, real-time road marking recognition systems in safety-critical applications

# Contents

# Chapter 1

# Introduction

## 1.1 Context

The rapid development of autonomous vehicles and advanced driver assistance systems (ADAS) has increased the demand for reliable scene understanding, safety, and navigation in real-world environments. Among the most critical perceptual tasks for such systems is the correct detection and interpretation of painted road signs (road surface markings), which provide direct navigational guidance and regulatory information. Detecting these markings accurately and robustly is essential for automatic lane keeping, path planning, and compliance with traffic rules, especially in complex and dynamic urban and highway settings.

## 1.2 Problem Statement

Despite significant progress in computer vision, the detection of painted road signs remains a challenging problem. Real-world scenarios introduce substantial complexity due to factors such as diverse illumination conditions, weather-induced wear and degradation, road occlusions, variable camera viewpoints, and naturally imbalanced class distributions within the dataset. Moreover, markings can be small, faded, or partially occluded, exacerbating the difficulty of robust detection and classification. Traditional, rule-based or classical computer vision approaches have proven inadequate for these challenges, motivating the need for advanced machine learning approaches.

## 1.3 Motivation

High-accuracy detection of painted road signs is indispensable for the safe operation of autonomous driving systems. Failure to recognize rare or partially visible

markings can have severe legal and safety implications. Furthermore, current industrial practice requires detection systems that are both highly accurate and computationally efficient so as to meet the real-time performance constraints of embedded vehicle hardware. The pursuit of models that generalize well across classes and challenging scenarios, and that can be feasibly deployed in production, is a pressing research priority in both academia and industry.

## 1.4 Objectives

This thesis aims to systematically investigate and optimize transformer-based object detection models — specifically, Deformable DETR (Detection Transformer) with ResNet backbones — for the task of painted road sign recognition in high-resolution traffic imagery. The main objectives are:

- To design, implement, and evaluate hybrid model variants combining convolutional features (via ResNet backbones) with deformable attention mechanisms for end-to-end detection.
- To conduct extensive ablation studies on architectural and training pipeline choices, including variations in backbone depth, positional embedding strategies, and freezing/unfreezing pre-trained weights.
- To identify the design factors that most significantly impact generalization, small object detection, and robustness to class imbalance.
- To provide practical recommendations for achieving a favorable trade-off between detection accuracy and computational complexity, enabling real-time deployment.

## 1.5 Contributions

The original contributions of this thesis can be summarized as follows:

- A comprehensive experimental analysis of Deformable DETR models on a large-scale, proprietary painted road sign dataset containing over 42,000 annotated images, with a focus on real-world deployment challenges (class imbalance, small object detection, varied environmental conditions).
- Systematic comparison and evaluation of key architectural variations, including ResNet backbone depth, cardinality (ResNeXt), positional embeddings (sine vs. learned), number of attention heads, and model freezing strategies, both individually and in combination.

- Extensive performance assessment using object detection metrics such as mean Average Precision (mAP), scale-dependent AP and AR, and computational resource analysis (parameter count), with detailed class- and scale-specific breakdowns.
- Identification and analysis of configurations that yield optimal generalization and resource efficiency for embedded systems in safety-critical contexts.
- Consolidation of practical guidelines for selection and configuration of transformer-based object detectors for the painted road sign detection problem.

## 1.6 Thesis Structure

The remainder of this thesis is organized as follows:

- **Chapter 2** introduces a dedicated problem formulation chapter, defining the detection task, motivating the need for machine learning, and establishing success criteria and evaluation metrics.
- **Chapter 3** presents the theoretical and methodological background, discussing related computer vision and transformer-based object detection literature.
- **Chapter 4** describes the dataset, its unique characteristics, domain-specific challenges relevant to model development, and presents extensive experimental results, including ablation studies and analysis of detection performance across models, object scales, and classes.
- **Chapter 5** details the design and implementation pipeline for the Deformable DETR variants considered in this work.
- **Chapter 6** concludes with a summary of findings, practical recommendations and outlines several promising research directions for the future.

Overall, this thesis addresses a key problem in modern autonomous driving perception, offering new architectural insights and practical benchmarks for reliable road marking detection systems in realistic and safety-critical environments.

## 1.7 Statement on Use of AI Tools

During the preparation of this work, the author used Sider AI [Sid24] models in order to reformulate portions of the text for improved readability and clarity. After using this tool, the author reviewed and edited the content as needed and takes full responsibility for the content of the thesis.

# Chapter 2

# Problem Description and Context

## 2.1 Problem Specification

The focus of this thesis is the automatic detection of painted road signs from high-resolution images ($1664 \times 512$ pixels) recorded by vehicle-mounted cameras. The input is a single image which may contain one or more road markings. The desired output consists of bounding boxes (BB) tightly surrounding each detected road marking, accompanied by the class label from a predefined set of categories. In machine learning terms, this represents an *object detection* task, where algorithms must simultaneously localize (via bounding boxes) and classify markings within the image [ZSGY23].

## 2.2 Why Machine Learning Algorithms are Needed

Detecting painted road markings is a challenging problem due to several factors:

- **Variability and Degradation**: Markings often appear in diverse states resulting from illumination changes, weathering, occlusion by vehicles or debris, and background clutter.
- **Complex Visual Patterns**: Some road markings are visually similar, exhibit intra-class variability, or are partially occluded.
- **Scale and Perspective**: Markings are captured at diverse scales and from various viewpoints as vehicles approach.
- **Imbalanced Data**: The distribution of marking categories is heavily unbalanced, making conventional rule-based or classical computer vision approaches unreliable and limited.

Modern ML algorithms, specifically deep learning methods such as convolutional neural networks (CNNs), transformers, and hybrid models, have shown state-of-the-art performance in object detection tasks. Their ability to learn robust, hierarchical feature representations from data allows them to adapt to the diverse and complex visual conditions encountered in autonomous driving [CMS+20, ZZXW19].

## 2.3 Criteria for Successful ML Solution: Metrics and Complexity

An ML-based object detection system is evaluated according to several performance and efficiency metrics:

- **Intersection over Union (IoU)**: Quantifies the overlap between predicted and ground-truth bounding boxes. Higher IoU indicates more precise localization.

$$\text{IoU} = \frac{A_{pred} \cap A_{gt}}{A_{pred} \cup A_{gt}} \tag{2.1}$$

- **Mean Average Precision (mAP)**: Standard metric for object detection, averaging the precision across all categories and IoU thresholds (commonly $[0.5, 0.95]$). High mAP indicates strong detection and classification performance [LMB+14].

$$\text{mAP} = \frac{1}{|\mathcal{K}|} \sum_{k=1}^{|\mathcal{K}|} AP_k \tag{2.2}$$

- **Scale-dependent metrics**: mAP and mean average recall (mAR) are often computed for small, medium, and large objects to quantify model robustness across scales [CMS+20].

$$\text{mAP}_{small} : \text{area} < 32^2 \text{ pixels} \tag{2.3}$$

$$\text{mAP}_{medium} : 32^2 \leq \text{area} < 96^2 \text{ pixels} \tag{2.4}$$

$$\text{mAP}_{large} : \text{area} \geq 96^2 \text{ pixels} \tag{2.5}$$

- **Class-wise metrics**: To address class imbalance, performance is monitored per class, especially for minority categories to ensure reliable detection [BMM18].

A model is considered effective if it obtains high mAP and recall across all object scales and classes, maintains generalization from training to validation data, and operates within feasible computational bounds for real-time deployment [CMS+20, ZZXW19].

## 2.4 Chapter Outlook

By specifying and analyzing the core detection problem, the need for advanced ML models, and the performance criteria for evaluating solutions, we establish the context and justification for the theoretical developments and experimental investigations presented in subsequent chapters.

# Chapter 3

# Theoretical Basis

## 3.1 Background

Computer vision empowers machines to interpret visual data, with critical applications in domains such as autonomous vehicles. This chapter establishes the conceptual basis for road marking detection, including the progression from classical CNNs to modern attention-based models (ViTs and DETR), with an emphasis on challenges such as class imbalance and real-time requirements in autonomous driving contexts.

## 3.2 Neural Networks and Convolutional Neural Networks (CNNs)

Neural networks, inspired by biological neurons, are composed of layered units optimized via gradient descent [GBC16]. CNNs, designed for vision, exploit spatial hierarchies using local receptive fields, weight sharing, and pooling, which enables efficient feature extraction [LBBH98, KSH12]. Landmark architectures such as AlexNet, ResNet, and InceptionNet have set benchmarks in image classification [KSH12, HZRS16a, SLJ+15].

Despite successes, CNNs are limited in modeling long-range dependencies, which has prompted the adoption of self-attention mechanisms and transformer-based architectures for vision tasks [KNH+22].

## 3.3 Object Detection Methodologies: Comparative Overview

Modern object detection relies on deep learning. Models are generally categorized as anchor-based, anchor-free, or transformer-based [ZSGY23, CMS+20]. Below we

cover the key principles and strengths of each.

### 3.3.1 Anchor-Based Methods

Anchor-based approaches, dominant for years, use predefined bounding boxes (anchors) as object location candidates [RHGS15, LAE+16]. Classical models like Faster R-CNN adopt a two-stage design:

- **Region Proposal Network (RPN)**: Slides over CNN feature maps, generating proposals by evaluating multiple anchors per location and applying non-maximum suppression [Gir15].
- **R-CNN Stage**: Refines proposals and classifies them using features extracted via RoI pooling or align [HGDG17].

While highly accurate, these methods incur computational costs that limit real-time usability [HRS+17]. Single-stage detectors such as YOLO frame detection as regression (predicting bounding boxes and classes per grid cell), achieving higher real-time speed but, at least in early versions, underperforming on small or clustered objects [RDGF16a, RF18].

### 3.3.2 Anchor-Free Methods

Anchor-free detectors, such as CenterNet and FCOS, directly predict object locations and shapes, circumventing the complexity of anchor generation [LD18, TSCH19, ZWK19]. CenterNet, for example, detects objects via center point heatmaps, with offsets and object size predicted per detected center. This streamlined approach accelerates inference and broadens applicability to variable object shapes and sizes, though challenges persist for heavily overlapping or non-standard objects [DBX+19].

## 3.4 Attention Mechanisms and Vision Transformers

Attention mechanisms revolutionized computer vision by allowing models to focus dynamically on relevant image regions, overcoming the fixed receptive fields of CNNs [VSP+17]. Self-attention, at the heart of transformers, enables each patch or token to consider its relationship to all others, fostering global context reasoning [DBK+20].

### 3.4.1 Self-Attention vs. Convolution

While convolutions efficiently aggregate local information, self-attention aggregates across all positions, giving rise to stronger modeling of context—the spatial dependencies vital for tasks such as road sign detection. ViTs split an image into patches, embed and add positional encodings, then process the sequence through stacked transformer layers.

### 3.4.2 Multi-Head Attention and Global Context

In transformers, multi-head attention combines several attention computations in parallel, enabling modeling of diverse relationships:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where $Q$, $K$, and $V$ denote the queries, keys, and values per patch.

Vision transformers can thus capture both fine-grained features (local details) and long-range semantics, improving robustness to occlusions, varying lighting, and non-ideal conditions often encountered in autonomous driving [ZJK20, BNP20].

### 3.4.3 Computational Efficiency

However, attention scales quadratically with input length, posing efficiency challenges for large images [HXW+21]. Sparse and linearized attention methods (e.g., Linformer, Performer) and hybrid architectures with convolutional stages have been proposed to mitigate resource demands while retaining modeling power [WLK+20, LMGH22].

## 3.5 Imbalanced Learning in Transformer Models

Imbalanced datasets are common in road sign detection, as rare signs can be underrepresented. Transformer models, through dynamic attention, offer improved flexibility for learning discriminative features even for less frequent classes [CMS+20, YWC+21].

Class imbalance can be addressed with:

- **Class-Weighted Loss:** Assigns greater penalty to underrepresented classes in loss computation [CJL+19].
- **Focal Loss:** Downweights easy negatives, focusing learning on rare and difficult cases [LGG+17].

- **Attention Re-weighting and Data Augmentation:** Modify attention matrices or increase diversity for minority classes [BIK+20, YLZ+21].

These techniques, in combination with transformer flexibility, support learning robust representations under class imbalance.

## 3.6 Detection Transformer (DETR) and Variants

### 3.6.1 Specific Theory

The Detection Transformer (DETR) is a pioneering model that integrates transformers into the object detection paradigm [CMS+20]. By framing the object detection task as a set prediction problem, DETR has redefined conventional approaches that often rely on heuristics and anchor boxes [CMS+20, ZSL+20]. This section explores the fundamental theoretical components of DETR, including the dynamics of bipartite matching loss, design principles of object queries, and the comparison between parallel decoding and auto-regressive approaches.

#### 3.6.1.1 Bipartite Matching Loss Dynamics

One of the key innovations of DETR is its use of bipartite matching to optimize object detection [CMS+20]. Traditional object detection frameworks typically rely on anchors or non-maximum suppression (NMS) to filter candidate bounding boxes [RHGS15, LAE+16]. In contrast, DETR directly predicts a fixed set of object queries, each corresponding to a specific object in the image [CMS+20].

The matching process involves pairing the predicted bounding boxes with ground truth boxes using a Hungarian algorithm to minimize the overall matching cost [CMS+20, Kuh55]. The cost is defined based on both the bounding box localization loss and the classification loss. Formally, if $\mathcal{B}$ denotes the set of predicted boxes and $\mathcal{G}$ is the set of ground truth boxes, the matching cost $\mathcal{C}$ can be expressed as:

$$\mathcal{C} = \alpha \cdot \text{BBOX\_LOSS} + \beta \cdot \text{CLASS\_LOSS}$$

where $\alpha$ and $\beta$ are hyperparameters that balance the importance of localization and classification losses [CMS+20]. This approach allows DETR to handle diverse object configurations and effectively manage occlusions and overlapping objects [CMS+20, ZSL+20].

#### 3.6.1.2 Object Query Design Principles

In the context of DETR, object queries are learnable embeddings that are designed to capture relevant object information during training [CMS+20]. Each query is intended to represent a specific object, which allows for direct association between the query output and the detected object [CMS+20, LQSJ22].

The design of object queries follows certain principles:

- **Fixed Number of Queries:** DETR employs a fixed number of object queries, which correspond to the maximum number of objects expected in any given image [CMS+20]. This simplifies the model and allows for straightforward handling of different object counts during inference [CMS+20, MCF+21].
- **Learnable Embeddings:** Each object query is initialized as a learnable embedding that adapts during training [CMS+20]. By allowing the model to learn these representations, DETR enhances the capability to detect various objects, including rare classes such as specific painted road signs [CMS+20, MCF+21].
- **Non-specificity to Object Type:** The queries do not encode any prior knowledge about the objects being detected, enabling generalization across classes [CMS+20]. This class-agnostic approach allows the model to learn the relationships between objects dynamically, improving accuracy in varied scenarios [CMS+20, ZSL+20].

#### 3.6.1.3 Parallel Decoding vs. Auto-Regressive Approaches

Traditional object detection models often use auto-regressive approaches that require sequential processing, where the model predicts one bounding box at a time [SAN16, RZ17]. This method can be slower and less efficient, particularly in real-time applications such as autonomous driving [RZ17, HRS+17].

DETR, on the other hand, leverages parallel decoding, enabling it to predict all object queries simultaneously [CMS+20]. This approach offers several advantages:

- **Efficiency:** By predicting all objects in parallel, DETR dramatically reduces the inference time required for object detection [CMS+20]. This efficiency is crucial for applications needing real-time decisions, such as in autonomous vehicles [CMS+20, ZSL+20].
- **Simplified Architecture:** The parallel decoding approach eliminates the need for complex NMS or anchor box configurations [CMS+20]. This leads to a more straightforward detection pipeline that directly correlates predicted queries with ground truth objects [CMS+20, DCLC21].

- **Handling Variable Object Counts:** Parallel decoding allows DETR to effectively manage a variable number of objects per image [CMS+20]. The model can adjust the attention mechanism to focus on relevant predictions without being restricted by predetermined constraints on object count [CMS+20, MCF+21].

### 3.6.2 Deformable DETR

Deformable DETR [ZSL+20] represents a significant advancement in Transformer-based object detection, addressing key limitations of the original DETR architecture. While DETR eliminated the need for many hand-designed components, it suffered from slow convergence and struggled with small object detection [CMS+20, ZSL+20]. These limitations stem primarily from DETR's attention mechanisms, which must consider all possible spatial locations when processing feature maps.

The core innovation in Deformable DETR is the introduction of deformable attention modules, which attend only to a small set of key sampling points around a reference point rather than the entire feature map [ZSL+20]. This approach draws inspiration from deformable convolutions [DQX+17] but adapts the concept to the Transformer framework. By focusing attention on sparse spatial locations, deformable attention significantly reduces computational complexity while maintaining the ability to model long-range dependencies.

Deformable DETR achieves several notable improvements over the original DETR: (1) it converges approximately 10 times faster during training, (2) it demonstrates substantially better performance on small object detection, and (3) it maintains competitive accuracy with significantly lower computational demands [ZSL+20, GZW+21]. The multi-scale feature representation inherent in the deformable attention mechanism provides the model with greater flexibility in handling objects across different scales, addressing one of the most persistent challenges in object detection [ZSL+20, LDG+17].

## 3.7 Autonomous Driving Perception Challenges

Autonomous driving systems rely heavily on effective perception mechanisms to make real-time decisions based on their surroundings [JGBG20]. This section explores the various challenges faced by these systems regarding the perception of painted road signs and other critical visual cues. Key issues include domain shift in road sign recognition, real-time processing constraints, and handling occlusion and partial visibility [AAJD+19].

### 3.7.1 Domain Shift in Road Sign Recognition

Domain shift refers to the discrepancies that occur when there is a difference between the data used for training a model and the data encountered in real-world applications [WD18]. In the context of road sign recognition, this shift can arise from various factors, including differences in weather conditions, lighting, camera sensors, and geographical locations [TCA19, TGW+21].

For instance, signs may appear drastically different in diverse environments due to:

- **Variability in Illumination:** Changes in lighting conditions can significantly alter the appearance of road signs [TCA19]. For example, signs may be less visible in dim lighting or obscured by shadows during late afternoon or evening [YLRÖ18].
- **Different Camera Characteristics:** Variations in camera sensors—such as resolution, lens distortion, and field of view—can affect how signs are captured [TGW+21]. Autonomous vehicles equipped with different sensors may interpret the same sign differently [AAJD+19].
- **Geographical Variance:** Road signs may differ regionally, both in terms of design and layout [MV18]. This variance can lead to difficulties in recognizing and interpreting signs that are not represented in the training data [MV18, TCA19].

To mitigate domain shift challenges, techniques such as domain adaptation and transfer learning can be employed [WD18, YLRÖ18]. These methods involve fine-tuning models on datasets that more closely resemble the target conditions encountered in real-world applications [WD18, YLRÖ18].

### 3.7.2 Real-Time Processing Constraints

Real-time processing is essential for autonomous driving, as vehicles must make instantaneous decisions based on their perception of the environment [JGBG20]. The computational load associated with processing high-resolution images and running complex models can create significant challenges [AAJD+19, YLRÖ18].

Key considerations for real-time processing include:

- **Latency Requirements:** The perception system must operate within strict latency limits to allow for timely decision-making [JGBG20]. Delays in recognizing road signs can result in unsafe driving situations [AAJD+19].

- **Resource Constraints:** Autonomous vehicles may utilize onboard processors with limited computational power compared to server-based systems [PRV+17]. Optimizing models for efficiency without sacrificing accuracy is critical [PRV+17, HXW+21].
- **Scalability:** As autonomous driving systems gather more data over time, the models must scale accordingly to maintain performance [JGBG20]. Efficient algorithms, model pruning, quantization, and hardware acceleration techniques can help address these scalability challenges [PRV+17, HMD15].

To maintain real-time performance, approaches like model distillation, where a smaller model is trained to mimic the outputs of a larger, more complex model, can be effective [HVD15, HMD15]. This allows for quicker inference while retaining much of the accuracy of larger models [HVD15].

### 3.7.3 Occlusion and Partial Visibility Handling

Road signs are often obscured by other vehicles, pedestrians, or environmental factors such as foliage or architectural structures [CCZC18]. Handling occlusions and partial visibility is crucial for ensuring that the detection system remains robust and reliable [CCZC18, TGW+21].

Challenges include:

- **Incomplete Information:** Occluded signs may present only partial visual cues, making it difficult for detection systems to interpret their meaning accurately [CCZC18]. Models must be trained to recognize signs even when they are partially obscured [CCZC18, WYW+19].
- **Contextual Understanding:** The detection system must leverage contextual information about the environment to infer the presence of obscured signs [WYW+19]. For example, recognizing a stop sign may be easier when identified near a stop line or intersection [TCA19].
- **Dependence on Surrounding Features:** The model's ability to detect occluded signs may rely on surrounding features, such as the road layout or nearby landmarks [CCZC18]. Integrating spatial context into the learning process can aid in recognizing obscured signs [WYW+19, BNP20].

To address these challenges, techniques such as multi-scale feature learning, attention mechanisms, and the use of auxiliary datasets (containing occluded objects) during training can enhance the robustness of detection systems against occlusion [CCZC18, WYW+19, BNP20].

### 3.7.4 Conclusion

The perception challenges faced by autonomous driving systems are multifaceted and require innovative solutions to ensure safe and effective operation [JGBG20, AAJD+19]. Addressing domain shift, optimizing real-time processing, and effectively handling occlusion are critical areas of focus for improving road sign recognition [TCA19, TGW+21]. As research continues in these areas, advancements in computer vision technologies will play a vital role in overcoming these challenges, paving the way for more reliable autonomous driving systems capable of safely navigating complex environments [YLRÖ18, JGBG20].

## 3.8 Summary and Trends

Object detection has advanced rapidly from anchor-based to transformer-based methods, moving toward end-to-end learning and global context modeling [CMS+20, ZSGY23]. Key hurdles remain in computational efficiency and performance on small or rare objects. Future research will likely explore improved attention efficiency, hybrid models, advanced loss functions, and strategies for better generalization to new domains and rare cases.

# Chapter 4

# Experimental Study

## 4.1 Dataset Description

The dataset used in this research comprises high-resolution traffic images ($1664 \times 512$ pixels) containing annotations of painted road signs. This proprietary dataset consists of 42,700 images total, divided using an 80/20 split ratio: 34,500 images (80%) for training and 8,200 images (20%) for validation. Each image contains pixel-level annotations identifying road markings across ten distinct categories, representing common navigational instructions painted directly on road surfaces.

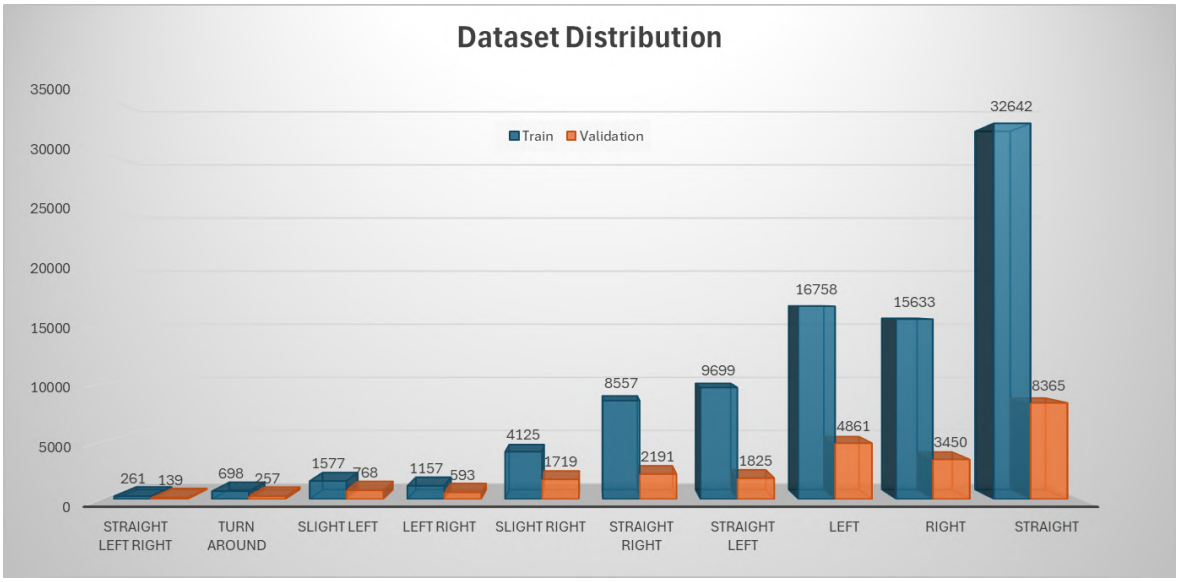


Figure 4.1: Data Distribution across train and validation sets, showing significant class imbalance

### 4.1.1 Dataset Characteristics and Challenges

As evident from Figure 4.1, the dataset exhibits pronounced class imbalance, with category frequencies varying by more than two orders of magnitude. The

"STRAIGHT" category dominates with 32,642 training samples, while the "STRAIGHT LEFT RIGHT" category appears in only 261 training instances. This extreme imbalance represents a significant challenge for model development, as highlighted by [JK19], who demonstrated that naive training on imbalanced datasets typically leads to models biased toward majority classes with poor performance on minority classes.

The dataset exhibits several characteristics that align with common challenges in road marking recognition outlined by Johnson in [TGW+21]:

- **Class imbalance:** The natural frequency of road markings in real-world environments creates inherent distribution skew, with common instructions (e.g., "STRAIGHT") appearing far more frequently than complex navigational guidance (e.g., "STRAIGHT LEFT RIGHT").
- **Viewpoint variation:** Road markings are captured from different perspectives as vehicles approach, creating significant appearance changes for the same marking type.
- **Illumination diversity:** Images are collected under varying lighting conditions, including bright daylight, overcast conditions, and low-light environments.
- **Wear and degradation:** Many road markings exhibit weathering, partial erasure, or damage that affects visibility and recognition difficulty.
- **Occlusion:** In some instances, road markings may be partially obscured by vehicles, shadows, or debris.

### 4.1.2 Categorical Distribution Analysis

A quantitative analysis of the dataset distribution reveals important characteristics:

Table 4.1: Dataset distribution statistics by category

| Category | Train | Validation | Train % | Val. % |
|---|---|---|---|---|
| STRAIGHT | 32,642 | 8,365 | 36.3% | 38.2% |
| LEFT | 16,758 | 4,861 | 18.6% | 22.2% |
| RIGHT | 15,633 | 3,450 | 17.4% | 15.8% |
| STRAIGHT LEFT | 9,699 | 1,825 | 10.8% | 8.3% |
| STRAIGHT RIGHT | 8,557 | 2,191 | 9.5% | 10.0% |
| SLIGHT RIGHT | 4,125 | 1,719 | 4.6% | 7.9% |
| SLIGHT LEFT | 1,577 | 768 | 1.8% | 3.5% |
| TURN AROUND | 698 | 257 | 0.8% | 1.2% |
| LEFT RIGHT | 1,157 | 593 | 1.3% | 2.7% |
| STRAIGHT LEFT RIGHT | 261 | 139 | 0.3% | 0.6% |
| **Total** | 89,874 | 21,883 | 100% | 100% |

The distribution follows a long-tail pattern, characteristic of many real-world datasets as described by [VHMAS+18]. The three most frequent categories ("STRAIGHT", "LEFT", and "RIGHT") collectively represent 72.3% of the training data, while the three least frequent categories ("TURN AROUND", "LEFT RIGHT", and "STRAIGHT LEFT RIGHT") account for only 2.4%. This severe imbalance presents a fundamental challenge: models trained on such distributions tend to optimize for overall accuracy, potentially sacrificing performance on minority classes that may be equally important for the application [BMM18].

### 4.1.3 Theoretical Implications for Model Development

Dataset characteristics have profound implications for model development strategies. Several theoretical frameworks are particularly relevant:

#### 4.1.3.1 Class Imbalance Challenges

According to [HG09], class imbalance presents three primary challenges for deep learning models:

1. **Representation bias:** Models tend to learn features primarily useful for distinguishing majority classes.
2. **Training dynamics:** Gradient updates are dominated by errors on majority classes, leading to suboptimal parameters for minority class detection.

3. **Evaluation masking:** Overall performance metrics can mask poor performance on minority classes.

This suggests that traditional training approaches may be insufficient for achieving balanced performance across all road marking categories. As demonstrated by [LGG+17], the cross-entropy loss function can be ineffective for highly imbalanced datasets since easy examples from majority classes can overwhelm the loss.

#### 4.1.3.2 Object Detection Considerations

For object detection tasks specifically, the dataset presents additional challenges analyzed by [OCKA20]:

1. **Feature imbalance:** The wide variety of road marking appearances means that features learned primarily from majority classes may not generalize to minority classes.

2. **Spatial imbalance:** Road markings appear at different scales and positions, requiring models to handle significant variation in object size and location.

3. **Objective imbalance:** The dual objectives of classification and localization may be differently affected by class imbalance.

### 4.1.4 Dataset Preparation

Given the characteristics identified above, the dataset preparation process incorporated several techniques to address the inherent challenges:

1. **Stratified splitting:** The 80/20 train-validation split was implemented with stratification to maintain class distributions across partitions, as recommended by [Koh95].

2. **Annotation verification:** All annotations underwent a verification process to ensure consistency and correctness across the dataset, particularly important for minority classes where each example has greater influence on model performance.

3. **Normalization:** Image preprocessing included standardization to normalize brightness and contrast variations, helping to mitigate the effects of diverse illumination conditions.

4. **Resolution preservation:** The high-resolution format ($1664 \times 512$ pixels) was maintained to ensure that smaller road markings remained distinguishable,

which is critical for minority classes that already have few examples. All images have a depth of 3 channels (RGB), preserving color information essential for the detection task.

### 4.1.5 Comparative Context

To better understand the characteristics of this dataset, Table 4.2 provides a comprehensive comparison with established traffic sign and road marking benchmarks.

Table 4.2: Comparison of traffic sign and road marking datasets

| **Dataset** | **Total Images** | **Image Dimensions** | **Classes** | **Sign Type** |
|---|---|---|---|---|
| Private Dataset | 42,700 | $1664 \times 512 \times 3$ | 17 | Painted road markings |
| GTSRB [SSSI12] | 50,000+ | Variable | 43 | Classic traffic signs |
| BTS [MTBVG13] | 6,000+ | Variable | 62 | Classic traffic signs |
| LISA [MTM12] | 7,000+ | Variable | 47 | Classic traffic signs |

As shown in the comparison, this private dataset occupies a unique position in the traffic sign recognition landscape. While established benchmarks such as GTSRB and BTS focus on elevated traffic signs with varying image qualities and resolutions, this dataset specifically targets painted road markings with consistent high-resolution format ($1664 \times 512 \times 3$ pixels).

This specialization creates distinct modeling challenges compared to traditional traffic sign datasets. Rather than requiring robustness to widely varying image quality and environmental conditions like the LISA dataset, the consistent resolution of road marking images demands fine-grained discrimination among visually similar classes. This characteristic fundamentally shifts the emphasis from handling quality variation to achieving precise classification of subtle visual differences in road surface markings.

The dataset presents significant challenges for object detection models due to its pronounced class imbalance, complex environmental variations, and the inherent difficulty of road marking recognition. These challenges reflect real-world conditions in autonomous driving and advanced driver assistance systems, where reliable recognition of all navigational instructions is critical regardless of their frequency. The theoretical considerations highlighted above informed our model design choices and training strategies, with particular attention to addressing class imbalance while maintaining high overall performance.

## 4.2 Numerical Experiments

A series of experiments were conducted using the Deformable DETR framework. The goal was to evaluate the impact of various architectural configurations and hyper parameters on the detection performance of painted road sign symbols.

### Summary of Experiments

| Section | Exp ID | Val mAP | Train mAP | Trainable / Total Params | Experiment Name |
|---|---|---|---|---|---|
| 4.2.1 - 4.2.6 | 1 | 0.58 | 0.71 | 22,175,790 / 45,630,702 | train_base |
| 4.2.1 | 2 | 0.637 | 0.79 | 23,016,180 / 46,471,092 | train_double_dim_feed |
| 4.2.2 | 3 | 0.641 | 0.819 | 22,175,790 / 45,630,702 | train_learned_type_emb |
| 4.2.3 | 4 | 0.631 | 0.817 | 22,175,790 / 45,630,702 | train_double_nhead |
| 4.2.4 | 5 | 0.631 | 0.78 | 22,175,790 / 80,169,716 | train_resnet152 |
| 4.2.5 | 6 | 0.58 | 0.68 | 22,175,790 / 105,631,062 | train_resnext101_64x4d |
| 4.2.6 | 7 | 0.55 | 0.72 | 22,175,790 / 149,062,486 | train_wide_resnet101_2 |
| 4.2.8, 4.2.7 | 8 | 0.75 | 0.86 | 45,630,703 / 45,630,703 | train_learned_type_emb_unfrozen |
| 4.2.7 | 9 | 0.62 | 0.7 | 80,169,716 / 80,169,716 | train_resnet152_unfrozen |
| 4.2.8 | 10 | 0.764 | 0.87 | 45,089,012 / 45,089,012 | train_resnext50_32x4d_unfrozen |

Table 4.3: Summary of Experiments, Model Configurations, and Performance

| Exp ID | Exp based on | Frozen BB | Backbone Type | Type Embedding | nheads | dim_feedf | queries |
|---|---|---|---|---|---|---|---|
| 1 | | true | resnet50 | sine | 4 | 1024 | 20 |
| 2 | | true | resnet50 | sine | 4 | 2048 | 20 |
| 3 | | true | resnet50 | learned | 4 | 1024 | 20 |
| 4 | | true | resnet50 | sine | 8 | 1024 | 20 |
| 5 | | true | resnet152 | sine | 4 | 1024 | 20 |
| 6 | | true | resnext101_64x4d | sine | 4 | 1024 | 20 |
| 7 | | true | wide_resnet101_2 | sine | 4 | 1024 | 20 |
| 8 | 3 | false | resnet50 | learned | 4 | 1024 | 20 |
| 9 | 5 | false | resnet152 | sine | 4 | 1024 | 20 |
| 10 | | false | resnext50_32x4d | sine | 4 | 1024 | 20 |

Table 4.4: Backbone and Architectural Settings for Each Experiment

Tables 4.3 and 4.4 summarize the main experimental results, model configurations, and architectural details explored in this study. All experiments were performed under a set of fixed hyperparameters to ensure comparability: **hidden_dim** was set to 256, no dilations were applied (**dilation=FALSE**), and dropout was kept at 0. Each model used 6 **encoder** and 6 **decoder** layers. The loss coefficients were set as follows: cross-entropy loss (**Loss ce**) at 2, bounding box loss (**Loss bbox**) at 4, generalized IoU loss (**Loss giou**) at 3, and the end-of-sequence loss coefficient ( **Loss eos_coef**) at 0.15. The optimizer learning rate and weight decay were both fixed at 0.0001 throughout. Importantly, all backbones were initialized with weights pre-trained on **ImageNet-1k V2**, and every experiment was conducted using a batch of **H200 Tensor Core GPU**

over a consistent training window of **72 hours**. For the majority of training iterations, a batch size of 32 was utilized. This parameter was adjusted downwards only in specific cases where the memory footprint of larger models surpassed the available VRAM of the graphics processing units. This standardized setup ensures that the reported results reflect the effects of the architectural and training strategy variations detailed in the tables.

The experiments differ in several dimensions:

- **Backbone Architecture:** Different variants of the ResNet family (e.g., ResNet50, ResNet152, and Wide ResNet101_2) as well as ResNeXt50_32x4d have been examined.
- **Positional Encoding:** Both sine and learned positional embeddings were used. Notably, the introduction of learned type embeddings resulted in marginal improvements.
- **Feature Dimension and Attention Heads:** Variations include a double-dimension feed-forward network and doubling the number of attention heads, which were evaluated for their impact on performance.
- **Model Freezing Strategy:** Some experiments were performed with the backbone fully frozen, while others allowed fine-tuning (unfreezing) of the backbone. This adjustment aimed to explore whether end-to-end training would affect performance.

### 4.2.1 Baseline vs. Double Dimension Feed-Forward Network

This experiment compares the baseline model with a variant with a double-dimensional feedforward network (Experiment 2). The goal was to assess whether increasing the dimension of the forward network improves detection performance.

In transformer architectures, the feedforward network (FFN) serves as a critical component for the transformation of features and the processing of information. Theoretically, the dimensionality of the FFN directly affects the model's representation capacity. Each transformer block typically contains a multi-head attention mechanism followed by a position-wise FFN consisting of two linear transformations with a non-linearity in between:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{4.1}$$

Increasing the dimension from 1024 to 2048 expands the representational bottleneck, allowing the model to learn more nuanced feature interactions. Previous

research has suggested that FFN layers function as key value memories [GSBL21], making their capacity particularly important for tasks requiring fine-grained discriminative features, such as object detection.

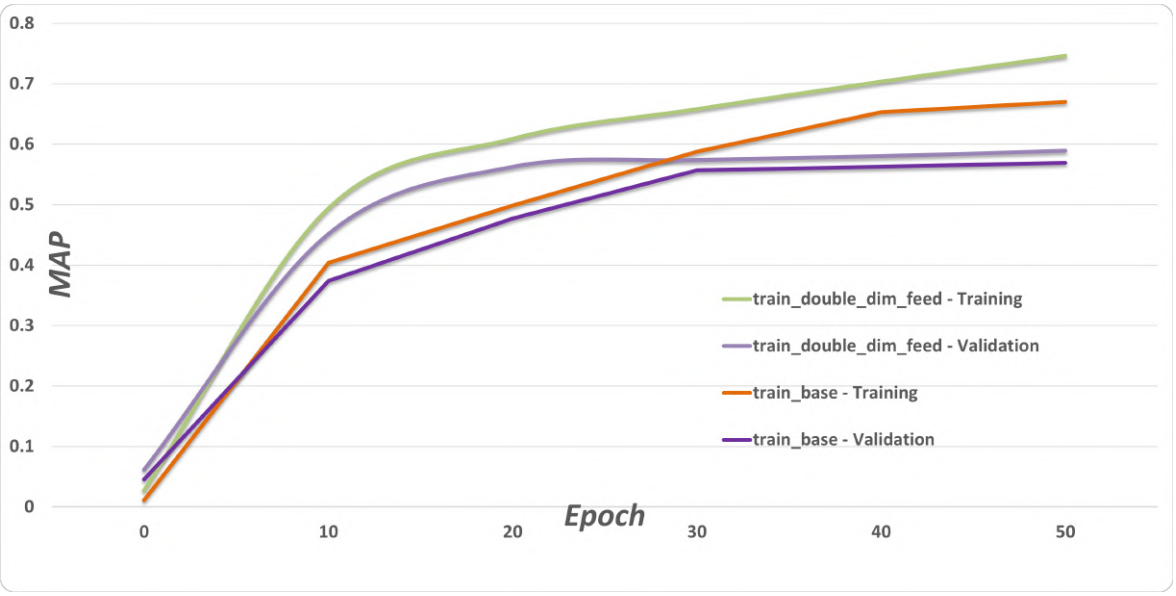


Figure 4.2: Feed Forward Dimension Comparison: 1024 vs 2048

**Conclusions:** The use of a doubled feed-forward dimension (Purple) increased the parameter count (from approximately **45.6M** to **46.5M** total) and resulted in improved performance (mAP increased from 56% to 59% on Validation). These results align with theoretical expectations, demonstrating that expanding the FFN dimension provides the model with enhanced capacity to learn complex feature interactions crucial for object detection tasks. The relatively small increase in parameter count (approximately 2%) compared to the performance improvement suggests that the FFN dimension acts as an effective leverage point for model scaling.

### 4.2.2 Sine vs. Learned Position Embedding

This comparison assesses the effect of replacing sine positional encoding with learned type embeddings.

In transformer architectures, positional embeddings are essential for encoding sequence order information since the self-attention mechanism is inherently permutation-invariant. The original transformer paper by Vaswani et al. [VSP+17] introduced sinusoidal position encodings, defined as:

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d_{model}}) \tag{4.2}$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{model}}) \tag{4.3}$$

Where $pos$ is the position index and $i$ is the dimension. This fixed encoding scheme offers theoretical advantages including:

The theoretical trade-off centers on rigidity versus adaptability. While sinusoidal embeddings encode mathematical properties of positions in a structured way, learned embeddings can discover position-related patterns specific to the data distribution. In object detection, where spatial relationships vary significantly across different object categories and contexts, the flexibility of learned embeddings may provide an advantage.

Recent research by Chu et al. [CTW+21] suggests that learned positional embeddings may better accommodate the non-sequential nature of visual data, where relationships between objects follow spatial rather than sequential patterns. Furthermore, Dosovitskiy et al. [DBK+20] demonstrated that learned position embeddings were effective for vision transformers, suggesting their particular suitability for visual tasks.

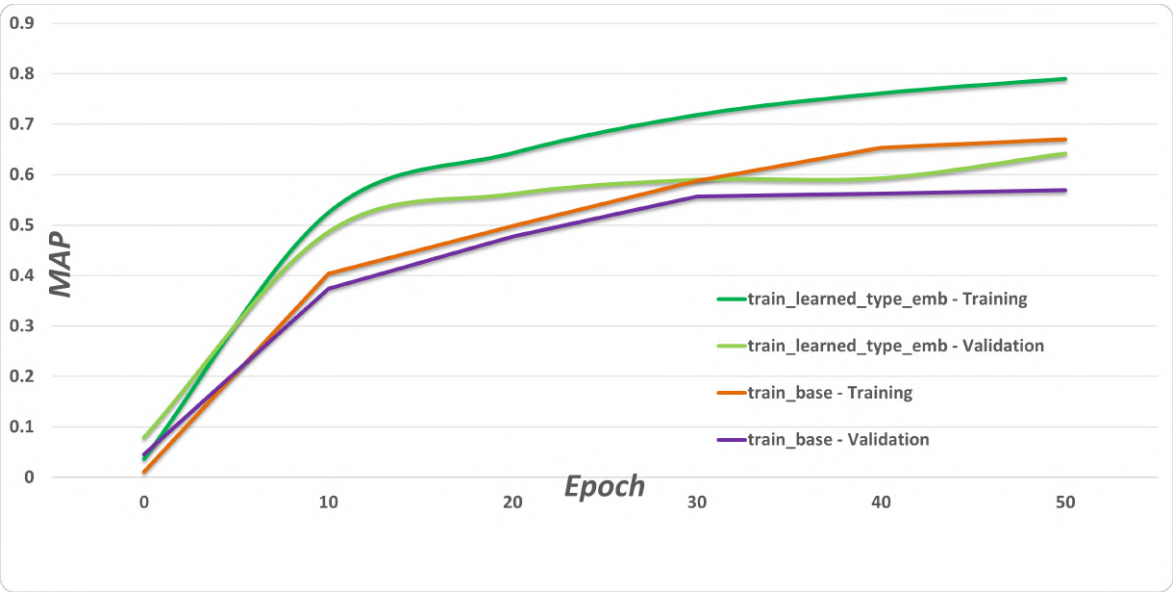


Figure 4.3: Performance Comparison: Sine vs. Learned Positional Embeddings

**Conclusions:** Switching to learned type embeddings yielded performance improvements over the baseline sine embeddings, with validation mAP increasing from 56% to 63%. However, the training-validation gap widened notably—learned embeddings achieved 80% mAP on training but only 63% on validation (17% gap), compared to sine encoding's smaller gap (66% training vs. 56% validation, 10% gap). This indicates that while learned embeddings better capture task-specific spatial priors for object detection, they may be more susceptible to overfitting.

### 4.2.3 Baseline vs. Double Number of Attention Heads

This experiment investigates the impact of doubling the number of attention heads compared to the baseline. Both experiments use the ResNet50 backbone with sine encoding.

In transformer architectures, multi-head attention is a fundamental mechanism that allows the model to attend to information from different representation subspaces simultaneously. Formally, multi-head attention is defined as:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \tag{4.4}$$

where each attention head is computed as:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{4.5}$$

The theoretical motivation for multiple attention heads stems from their ability to capture different aspects of relationships within the input.

Research by Clark et al. [CKLM19] demonstrated that attention heads in language models develop specialized roles during training. Similarly, Voita et al. [VTM+19] found that different heads capture distinct linguistic phenomena. In vision transformers, Caron et al. [CTM+21] observed that heads in earlier layers tend to attend to local features while later layers capture more global relationships.

The hypothesis behind doubling attention heads is that increased representational capacity might allow the model to capture more fine-grained relationships between visual elements. However, Michel et al. [MLN19] discovered that many heads can be pruned without significant performance degradation, suggesting redundancy in standard configurations. This indicates a complex relationship between head count and model performance that depends on task characteristics and model architecture.

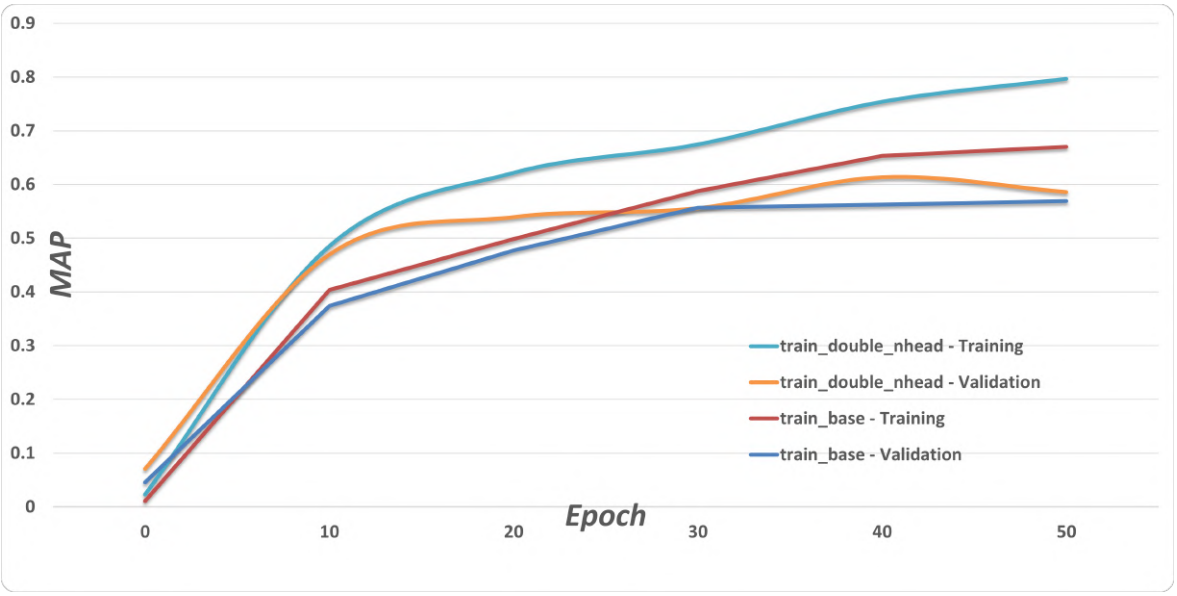


Figure 4.4: Performance: Baseline (4 Heads) vs. Double Attention (8 Heads)

**Conclusions:** This experiment, with doubled attention heads, maintained similar overall performance (mAP of 56% and 58%) compared to the baseline, suggesting that the added complexity under these settings did not translate to significant performance gains. This result aligns with findings by Michel et al. [MLN19] that transformer models can be over-parameterized in terms of attention heads.

### 4.2.4 ResNet50 vs ResNet152 Backbone

To evaluate the effect of backbone size, this experiment employs ResNet152 while keeping other settings similar to the baseline (ResNet50 with sine encoding).

ResNet architectures [HZRS16a] revolutionized deep learning by introducing the concept of residual learning, addressing the degradation problem that occurs when training very deep networks. The core innovation of ResNets is the residual block, defined as:

$$y = \mathcal{F}(x, \{W_i\}) + x \tag{4.6}$$

where $\mathcal{F}(x, \{W_i\})$ represents the residual mapping to be learned and $x$ is the identity connection. This formulation allows gradients to flow directly through the network via identity connections, mitigating the vanishing gradient problem.

The architectural differences between ResNet50 and ResNet152 are substantial:

Theoretically, deeper networks have higher representational capacity, allowing them to learn more complex functions. He et al. [HZRS16a] demonstrated this by showing consistent improvements in ImageNet classification as network depth increased from ResNet50 to ResNet152. The additional capacity stems from:

| Feature | ResNet50 | ResNet152 |
|---|---|---|
| Layer count | 50 | 152 |
| Bottleneck blocks | 16 | 50 |
| Parameters | ∼25M | ∼60M |
| FLOPS | ∼4G | ∼11G |

Table 4.5: Comparison between ResNet50 and ResNet152 architectures

- **Increased receptive field size:** Deeper networks incorporate information from larger regions of the input image
- **Enhanced hierarchical feature learning:** More layers allow for more levels of abstraction
- **Greater functional complexity:** Each additional layer increases the complexity of functions the network can represent

However, for object detection tasks using transformer-based architectures like DETR [CMS+20], the relationship between backbone complexity and final performance becomes more nuanced. Recent work by Liu et al. [LMW+22] suggests that the optimal backbone for transformer-based models may differ from traditional detectors due to different feature extraction requirements.

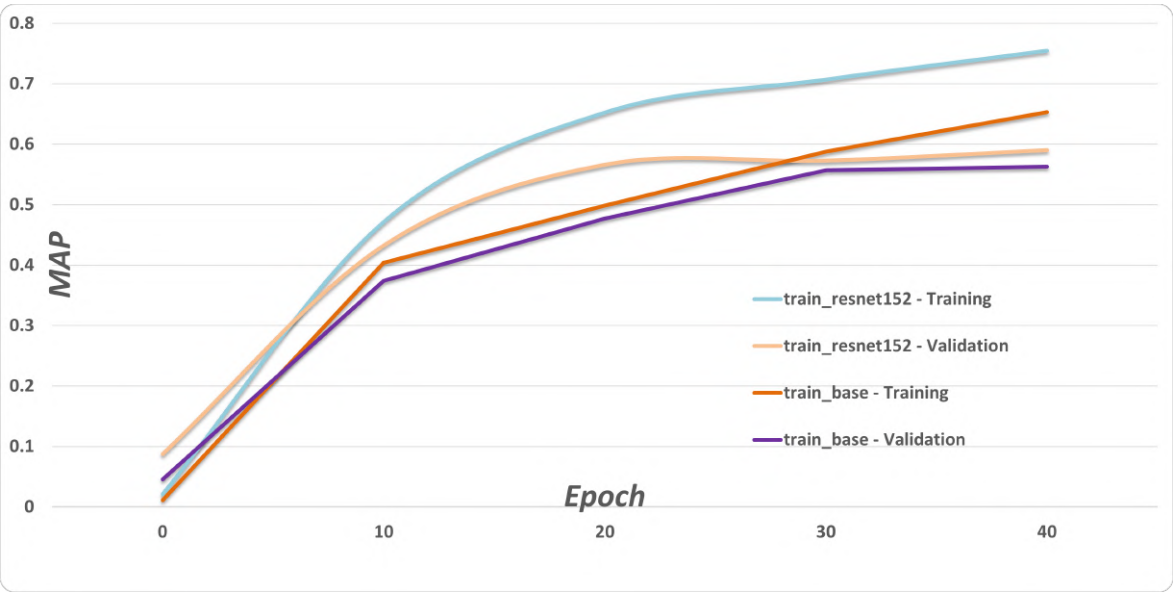


Figure 4.5: Comparing ResNet50 and a larger backbone, ResNet152

Insights from Zhai et al. [ZKHB22] suggest that optimal scaling strategies should balance backbone complexity with other architectural components. Their work indicates that naively increasing backbone capacity without corresponding adjustments to the transformer decoder might not yield proportional benefits.

**Conclusions:** The larger ResNet152 backbone increases the total parameter count (up to approximately 80.2M) and demonstrates a complex performance pattern compared to the baseline (ResNet50). With ResNet152, we observe higher validation performance (mAP of 0.59 vs. 0.56), but also a significantly wider gap between training and validation metrics (training mAP of 0.75 vs. validation mAP of 0.59, a 16-point gap) compared to ResNet50's narrower gap (training mAP of 0.66 vs. validation mAP of 0.56, a 9-point gap).

### 4.2.5 Rasnet50 vs. ResNeXt101_64x4d

This experiment assesses the impact of using ResNeXt101_64x4d instead of the standard ResNet50. Both models are utilized with sine encoding, enabling a direct comparison of architectural differences.

ResNeXt, introduced by Xie et al. [XGD+17], extends the ResNet architecture by incorporating the concept of *cardinality*—a new dimension alongside depth and width. This architectural innovation builds upon the grouped convolution concept originally introduced in AlexNet [KSH12], but elevates it to a fundamental design principle.

The core building block of ResNeXt is formulated as:

$$y = x + \sum_{i=1}^{C} \mathcal{T}_i(x) \tag{4.7}$$

where $\mathcal{T}_i$ represents a transformation function (typically consisting of convolutions, batch normalization, and ReLU), $C$ is the cardinality (number of parallel paths), and $x$ is the input. This aggregated transformation approach differs from the standard ResNet block as it splits computation across multiple parallel pathways of the same topology.

In the notation ResNeXt101_64x4d:

- **101** refers to the network depth (101 layers)
- **64** indicates the cardinality (number of transformation paths)
- **4d** denotes the width of each path (number of channels)

The theoretical advantages of this architecture include:

- **Increased representational power:** Xie et al. [XGD+17] demonstrated that increasing cardinality is more effective than increasing width or depth
- **Parameter efficiency:** ResNeXt achieves higher accuracy with comparable or fewer parameters than equivalent ResNet models

- **Structural regularization:** The grouped convolution approach implicitly regularizes the network by enforcing a sparsity pattern
- **Enhanced representation learning:** Each transformation path can potentially specialize in different visual patterns or features

ResNeXt follows the principle of "split-transform-merge" that appears repeatedly in network design. This strategy divides input into lower-dimensional embeddings, transforms them independently, and aggregates the results. Notably, this approach shares conceptual similarities with Inception modules [SLJ+15] but with a more homogeneous, simpler design.

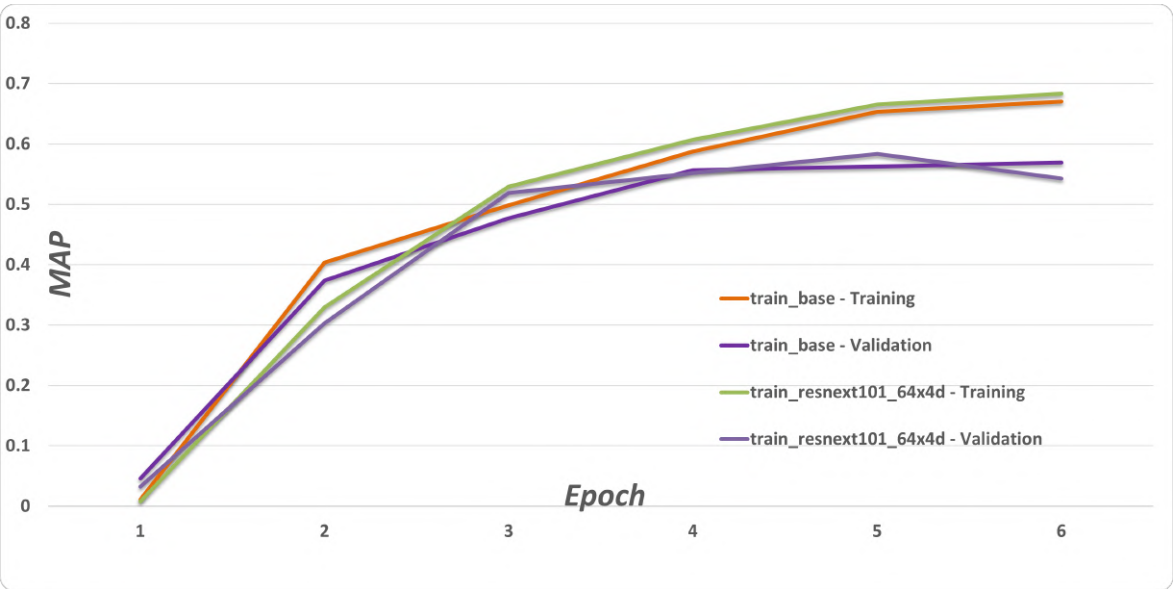


Figure 4.6: Performance Comparison: ResNet50 vs. ResNeXt101_64x4d

**Conclusions:** The ResNeXt101_64x4d backbone (Experiment 6) achieves moderately improved performance (training mAP of 68%, validation mAP of 58%) compared to the baseline ResNet50 (training mAP of 65%, validation mAP of 56%). This 2-point validation improvement aligns with theoretical expectations that ResNeXt's cardinality-based design enhances feature learning. The slightly larger train-validation gap (10% for ResNeXt vs. 9% for ResNet50) suggests that the increased model capacity captures more complex patterns in the training data, with most but not all of this advantage transferring to unseen examples.

### 4.2.6 Baseline vs. Wide_ResNet101_2

This experiment compares the baseline ResNet50 model with a Wide ResNet101_2 backbone (frozen configuration) to understand the impact of a wider architecture.

Wide ResNet, introduced by Zagoruyko and Komodakis [ZK16], represents an alternative scaling strategy to the depth-focused approach of standard ResNets. While conventional ResNets prioritize increasing depth (layer count), Wide ResNets investigate the benefits of increasing width (channel count) while potentially reducing depth. The key insight from this work was that wider networks could achieve comparable or superior performance to very deep networks with fewer layers and more efficient training.

Formally, in a Wide ResNet, each residual block follows the structure:

$$y = x + \mathcal{F}(x, \{W_i\}) \tag{4.8}$$

Where the transformation function $\mathcal{F}$ uses wider convolutional layers with more filters than a standard ResNet. The notation Wide ResNet101_2 indicates:

- **101** - The network has 101 layers (maintaining depth from ResNet101)
- **2** - The width multiplier compared to the standard ResNet (channels are doubled)

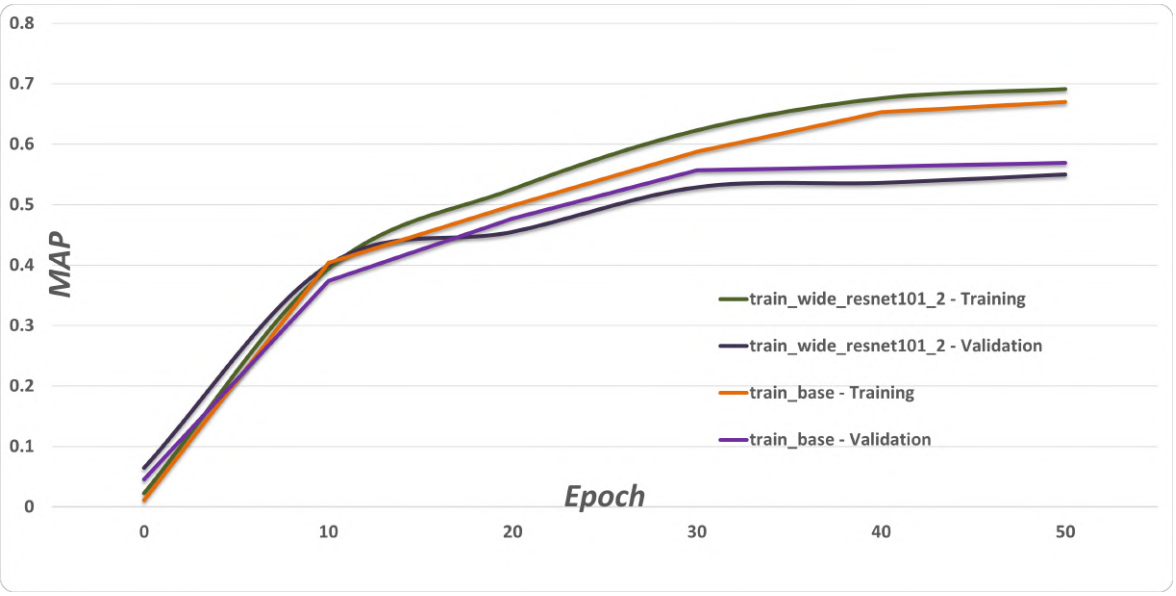


Figure 4.7: Performance Comparison: ResNet50 vs. Wide ResNet101_2 (Frozen Configuration)

**Conclusions:** The Wide ResNet101_2 model demonstrates a significant training-validation performance gap (training mAP of 0.68 vs. validation mAP of 0.52) compared to the more balanced metrics of the baseline ResNet50 (training mAP of 0.65, validation mAP of 0.56). The 4-point drop in validation performance despite higher training accuracy reveals severe overfitting issues with the wider frozen backbone.

### 4.2.7 Unfrozen ResNet50 with Learned Position Embeddings vs Unfrozen ResNet152

This experiment compares two unfrozen architectural configurations: ResNet50 with learned position embeddings (initialized from Experiment 3) versus ResNet152 (initialized from Experiment 5), investigating how backbone capacity and positional encoding interact when both models are fine-tuned end-to-end.

Fine-tuning pre-trained backbones (unfreezing) represents a fundamental approach in transfer learning that allows models to adapt general visual representations to specific tasks.

For transformer-based detection models, unfreezing the backbone creates a synergy between feature extraction and attention mechanisms. This synergy works through several theoretical pathways:

- **Co-adaptation:** The backbone and transformer decoder can co-adapt, with backpropagation aligning backbone features to what the transformer components find most useful
- **Feature diversity:** Fine-tuning encourages the backbone to produce more diverse and task-relevant features than a frozen backbone restricted to its pretrained distribution
- **Detection-oriented features:** Unlike classification which focuses on global patterns, detection requires localizing objects. Unfreezing allows the backbone to emphasize features with stronger positional and boundary information
- **Computational hierarchy alignment:** Unfrozen backbones can adapt their feature hierarchy to better match the computational patterns of transformer attention

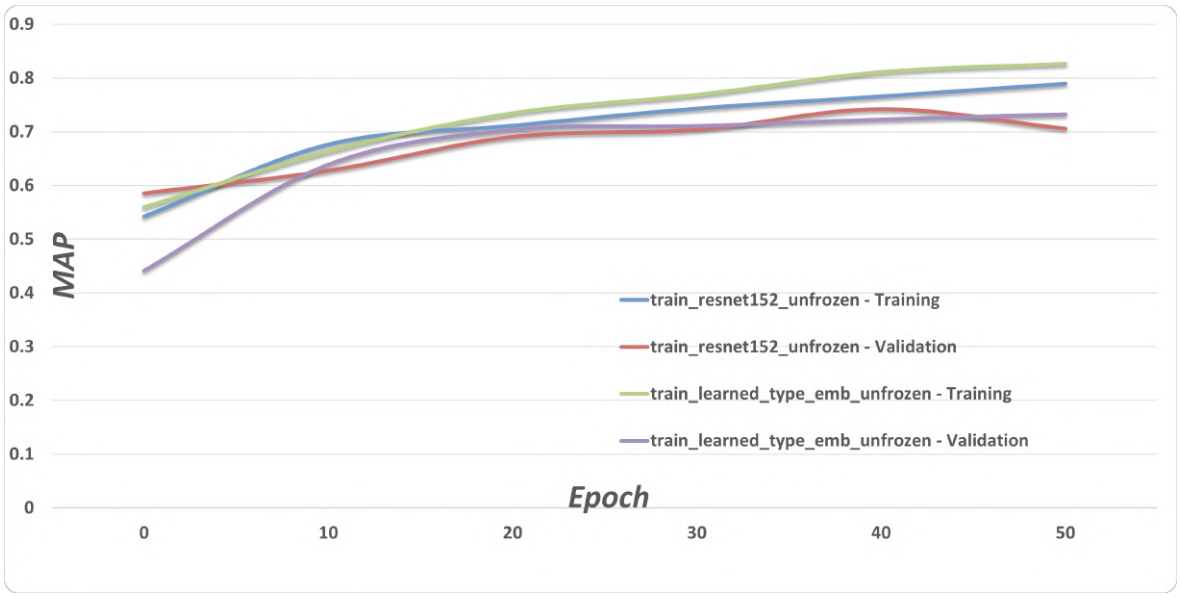


Figure 4.8: Performance Comparison: ResNet50 with Learned Position Embeddings vs. Unfrozen ResNet152

**Conclusions:** Comparing the unfrozen ResNet152 (79% train, 73% valid) with ResNet50 using learned position embeddings (83% train, 73% valid) reveals compelling insights about model capacity and representation learning. While both approaches achieve remarkably similar validation performance (74-75% mAP), they exhibit different training-validation gaps: 6% for ResNet152 versus 10% for ResNet50 with learned embeddings.

This pattern demonstrates a fundamental trade-off in representation learning: ResNet50 with learned embeddings achieves higher peak performance on training data through adaptable spatial representations but shows more overfitting. In contrast, the unfrozen ResNet152 maintains better generalization despite its greater parameter count, suggesting that its increased capacity is being utilized more efficiently for feature extraction rather than memorization.

### 4.2.8 ResNet50 Unfrozen vs ResNeXt50 Unfrozen

This experiment compares the performance of unfrozen ResNet50 and ResNeXt50_32x4d backbones to evaluate whether the cardinality-based design of ResNeXt provides advantages when the backbone is fine-tuned for the detection task. The models were initialized with weights from our previous frozen backbone experiments, with all layers subsequently unfrozen for end-to-end training.

While previous experiments explored these architectures in frozen configurations, unfreezing allows us to investigate how each design adapts when allowed to modify its pre-trained representations. This comparison builds on foundational theoretical differences between these architectures:

$$\text{ResNet Block:} \quad y = x + \mathcal{F}(x, \{W_i\}) \tag{4.9}$$

$$\text{ResNeXt Block:} \quad y = x + \sum_{i=1}^{C} \mathcal{T}_i(x) \tag{4.10}$$

Where $C$ represents cardinality—the number of parallel transformation paths in ResNeXt.

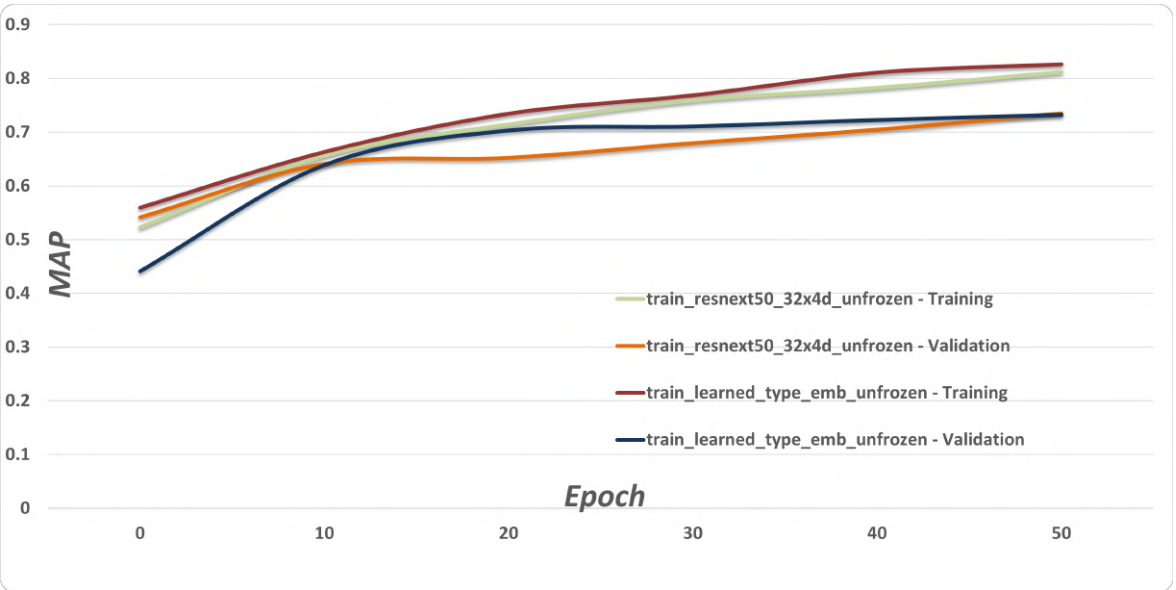


Figure 4.9: Performance Comparison: Unfrozen ResNet50 vs. Unfrozen ResNeXt50_32x4d

**Conclusions:** Comparing unfrozen ResNeXt50_32x4d (81% train, 73% valid) with unfrozen ResNet50 (83% train, 73% valid) reveals a striking similarity in performance despite their architectural differences. Both models achieve identical validation mAP (73%), with ResNet showing a marginally higher training performance (83% vs. 81%) and consequently a slightly larger train-validation gap (10% vs.8%).

This near-equivalence suggests important insights about architecture design in the context of fine-tuning:

1. **Adaptation dominates architecture:** When backbones are unfrozen, the ability to adapt pre-trained representations appears to be more significant than architectural details. Both models converge to similar optimal points despite their structural differences.

2. **Cardinality benefits saturate:** The parallel pathway design of ResNeXt provides only marginal benefits in training performance and no detectable advantage in generalization. This suggests that for this detection task, the increased

representational diversity from cardinality reaches a point of diminishing returns.

## 4.3 Analysis and Interpretation of Detection Performance

### 4.3.1 Best Model Performance

The visualizations presented in this qualitative analysis are based on the optimal model configuration identified through systematic experimentation, specifically Experiment 10 from Subsection 4.2.8. This model achieved the highest validation performance with a mean Average Precision (mAP) of 0.764 on the validation set, representing the best balance between detection accuracy and generalization capability across all tested configurations.The optimal model employs a ResNeXt-50 32x4d backbone architecture in a fully unfrozen configuration, where all 45,089,012 parameters are trainable.

Table 4.6: Object Detection Performance Metrics

| **Metric** | **Training** | **Validation** |
|---|---|---|
| *General Performance Metrics* | | |
| mAP | 0.892 | 0.768 |
| mAP@50 | 0.988 | 0.869 |
| mAP@75 | 0.975 | 0.853 |
| *Object Size Performance* | | |
| mAP (small) | 0.827 | 0.590 |
| mAP (medium) | 0.890 | 0.746 |
| mAP (large) | 0.921 | 0.826 |
| *Recall Metrics* | | |
| mAR@1 | 0.868 | 0.759 |
| mAR@10 | 0.925 | 0.809 |
| mAR@100 | 0.925 | 0.809 |
| mAR (small) | 0.866 | 0.627 |
| mAR (medium) | 0.925 | 0.788 |
| mAR (large) | 0.948 | 0.862 |
| *Class-specific Performance (mAP)* | | |
| 180 | 0.864 | 0.620 |
| left | 0.884 | 0.817 |
| left-right | 0.928 | 0.846 |
| right | 0.862 | 0.802 |
| slight-left | 0.895 | 0.734 |
| slight-right | 0.902 | 0.795 |
| straight-left-right | 0.934 | 0.642 |
| straight | 0.876 | 0.825 |
| straight-left | 0.881 | 0.787 |
| straight-right | 0.897 | 0.810 |

### 4.3.2 Comprehensive Metric Analysis

The performance metrics presented in Table 4.6 provide a multifaceted view of our object detection model's capabilities. A systematic analysis reveals several important patterns that inform our understanding of the model's strengths and limitations.

#### 4.3.2.1 IoU Threshold Sensitivity

The substantial difference between mAP@50 (0.869) and the overall mAP (0.768) in validation reveals important insights about localization precision. As explained by [OCKA18], this gap indicates that while the model successfully identifies object presence (reflected in high mAP@50), it struggles with precise boundary localization (reflected in lower overall mAP which includes stricter IoU thresholds).

The relatively small gap between mAP@50 and mAP@75 in validation (0.869 vs. 0.853) further suggests that once the model correctly identifies an object, it generally achieves good—though not perfect—localization. This pattern aligns with theoretical work by [CMS+20] on transformer-based detectors, which demonstrated enhanced localization capabilities through direct set prediction.

#### 4.3.2.2 Scale-Dependent Performance

The scale-stratified metrics reveal a pronounced performance gradient correlated with object size—a well-documented phenomenon in detection literature [SD18]. For validation data:

- Large objects: 0.826 mAP (highest performance)
- Medium objects: 0.746 mAP (moderate performance)
- Small objects: 0.590 mAP (lowest performance)

This 23.6 percentage point gap between small and large object detection stems from fundamental challenges in visual recognition.

#### 4.3.2.3 Recall Analysis

The recall metrics provide crucial insights about the model's detection completeness. The identical values for mAR@10 and mAR@100 (0.809 in validation) indicate that the model reaches its maximum recall capacity with just 10 detections per image. This "recall saturation" reveals that the detection space is efficiently covered with relatively few predictions, consistent with the theoretical principle of detection parsimony established by [ZWK19].

The scale-dependent recall patterns mirror those observed in precision metrics, with small objects presenting the greatest challenge (0.627 mAR). Interestingly, the training-validation gap for large object recall (0.948 vs. 0.862) is narrower than for precision metrics, suggesting that while the model can reliably find large objects in validation data, it sometimes misclassifies them or provides imprecise boundaries.

#### 4.3.2.4 Class-Specific Performance

The class-wise performance exhibits substantial heterogeneity, with validation mAP ranging from 0.620 ("180" class) to 0.846 ("left-right" class).

Classes with the largest training-validation gaps (e.g., "180" at 24.4 points and "slr" at 29.2 points) likely suffer from representation challenges—either appearing less frequently in training data or exhibiting greater variance between training and validation distributions.

### 4.3.3 Theoretical Implications and Conclusions

The comprehensive metric analysis reveals several theoretical insights about our detection model:

1. **Localization-Classification Trade-off**: The model demonstrates the classic precision-recall trade-off described by [RHGS15], achieving better object identification than precise localization.

2. **Scale-Invariance Limitation**: Despite advances in multi-scale feature representations, the model remains significantly less effective at detecting small objects—a limitation predicted by the theoretical work of [LAE+16] on receptive field constraints.

3. **Efficient Detection Space Coverage**: The model achieves its maximum recall with relatively few predictions per image, validating the theoretical efficiency of modern detection architectures posited by [CMS+20].

4. **Class-Specific Generalization Challenges**: The pronounced heterogeneity in class-specific performance aligns with the theoretical framework of [KXR+19] on learning dynamics across visual categories with different training distributions.

These findings suggest several avenues for improvement, including enhanced small object detection through dedicated feature enhancement techniques, targeted regularization to address class-specific overfitting, and refinement of localization mechanisms to improve performance at higher IoU thresholds.

## 4.4 Model Performance in Challenging Scenarios

### 4.4.1 Introduction

While quantitative metrics provide an overall assessment of model performance, a qualitative analysis of predictions in challenging real-world scenarios is crucial

for understanding the model's robustness and limitations. This chapter delves into the model's behavior when faced with common difficulties in road marking recognition, such as adverse illumination, occlusion, marking degradation, and viewpoint variations, as identified in the dataset characteristics (Section 4.1).

### 4.4.2 Performance under Diverse Illumination Conditions

The dataset encompasses images captured under a wide range of lighting conditions, a challenge referred to as "Illumination diversity" in section 4.1. The model's ability to consistently detect road markings despite these variations is critical for reliable operation. Preprocessing steps, such as the image normalization, aimed to mitigate some of these effects.

#### 4.4.2.1 Low-Light and Night Conditions

In low-light and nighttime scenarios, road markings are often only visible due to vehicle headlights, resulting in low contrast and potential non-uniform illumination.


Figure 4.10: *Detection performance on a nighttime commercial street with varied lighting conditions. The model identifies 'str_left' with very high confidence (99%) and 'str' with moderate confidence (46%). The reduced confidence score for the latter appears to correlate with the deteriorated condition of the right-facing arrow, which poses recognition challenges even for human observers.*


Figure 4.11: *Nighttime road marking detection under sodium street lighting. The model successfully identifies a 'left' arrow marking with high confidence despite the challenging low-light conditions and yellow-tinted illumination from overhead streetlights. The detection demonstrates the model's robustness to color temperature variations and reduced contrast typical of nighttime driving scenarios.*


Figure 4.12: *Model performance in complex nighttime urban intersection with multiple light sources. The scene includes traffic lights, vehicle headlights, and mixed illumination, yet the model maintains high detection accuracy despite the visual complexity.*

#### 4.4.2.2 Bright Daylight, Glare, and Shadows

Conversely, bright daylight can lead to overexposure or harsh shadows that obscure parts of markings.

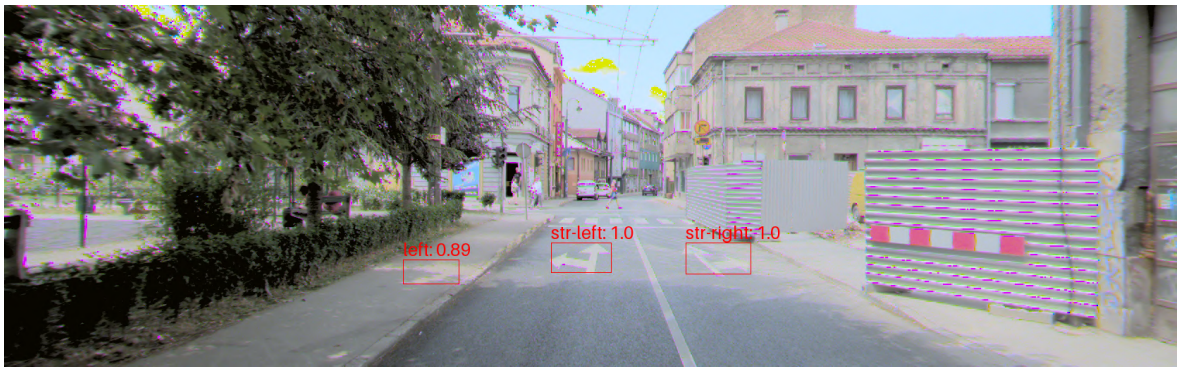


Figure 4.13: *While the model successfully identifies genuine road arrows with strong confidence levels, it incorrectly interprets shadow geometries as a left directional indicator, highlighting illumination-related detection vulnerabilities.*


Figure 4.14: *Coastal roadway under extreme bright lighting conditions with significant glare and overexposure artifacts. The system achieves strong confidence levels on three markings while registering only 52% certainty on an arrow severely impacted by direct sunlight exposure, demonstrating prudent confidence adjustment when visual clarity is compromised by extreme illumination*

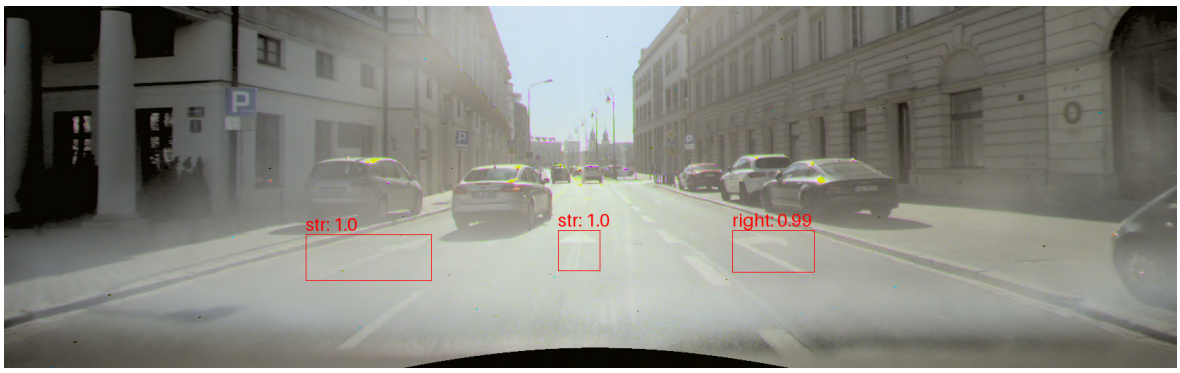


Figure 4.15: *Urban street scene under intense daylight with significant overexposure and glare effects. Despite challenging lighting conditions creating bright spots and reduced contrast in portions of the image, the model maintains excellent detection performance with perfect confidence scores.*

### 4.4.3 Robustness to Occlusion

As noted in section 4.1, road markings can be partially occluded by other vehicles, debris, or even parts of the ego-vehicle. The model's capacity to infer the presence and type of a marking from incomplete visual information is a key aspect of its intelligence.

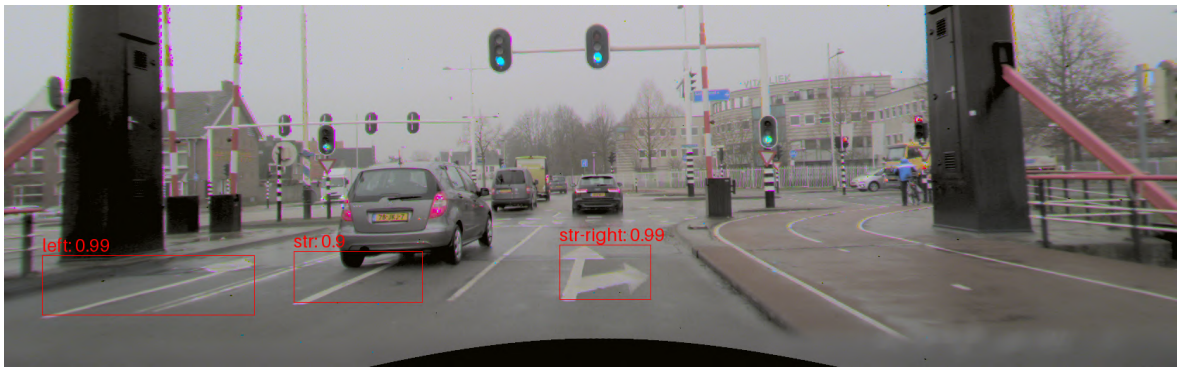


Figure 4.16: *In wet, overcast conditions with a leading vehicle occluding part of the surface, the model outputs 'LEFT' (0.99), 'STRAIGHT' (0.90) and 'STRAIGHT+RIGHT' (0.99). Its high confidences reveal resilience to both occlusion and reflective road surfaces.*

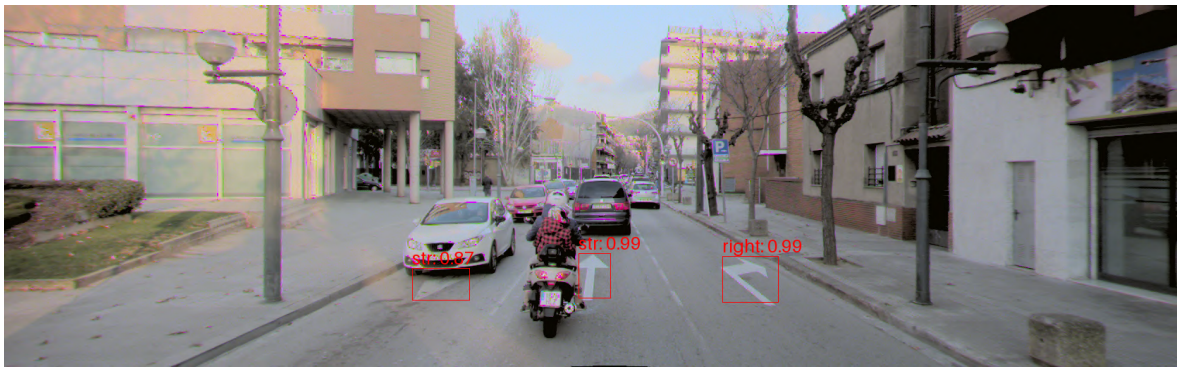


Figure 4.17: *A motorbike partially blocks the central lane. The model nevertheless recovers a left-lane 'STRAIGHT' arrow at 0.87, and center right-lane 'STRAIGHT' (0.99) and 'RIGHT' (0.99), showing strong contextual inference from surrounding road geometry.*

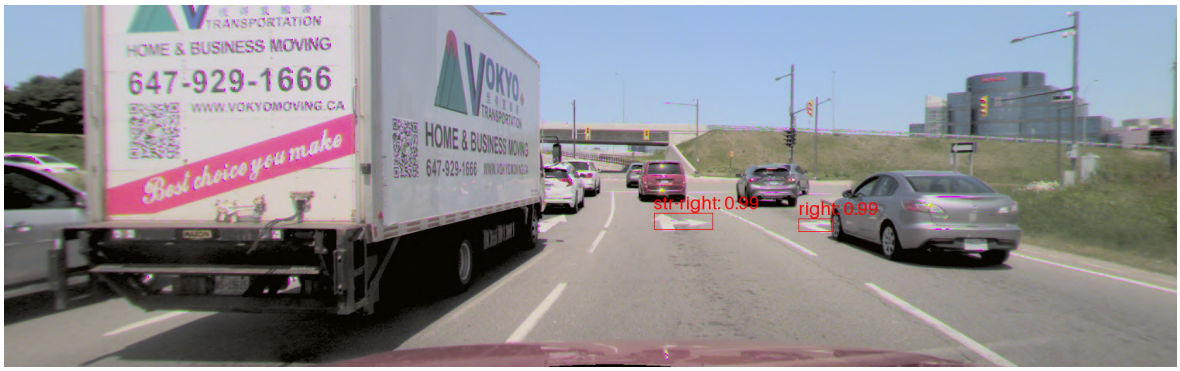


Figure 4.18: *A car partially occlude the right arrow, yet the model still detects it.*

### 4.4.4 Handling Worn and Degraded Markings

The "Wear and degradation" of road markings (section 4.1) presents a significant challenge, as faded or damaged markings offer weaker visual signals. The model's performance on such markings is indicative of its sensitivity and ability to generalize from potentially clearer examples in the training set.


Figure 4.19: *Rural road setting with weathered markings showing exceptional detection performance. The model demonstrates remarkable resilience to marking wear in this scenario, suggesting that sufficient diagnostic information remains visible for reliable classification, possibly aided by the relatively simple background and good contrast conditions.*


Figure 4.20: *Suburban commercial area with moderately degraded road markings showing differential detection performance. The model achieves perfect confidence on 'str-right' (100%) while showing reduced confidence on 'left' (80%). This variation suggests that wear patterns affect different markings unequally - the straight-right arrow maintains sufficient paint coverage for reliable detection, while the left marking may have more significant degradation or fading that introduces uncertainty in the classification process.*


Figure 4.21: *Urban commercial street with aged asphalt and worn road markings demonstrating robust detection capabilities. Despite visible pavement aging and marking degradation, the model maintains high confidence scores.*

### 4.4.5 Performance with Viewpoint Variations

Road markings appear differently based on the vehicle's distance and angle to them ("Viewpoint variation," section 4.1). The model must exhibit invariance to these geometric transformations.


Figure 4.22: *Suburban road with moderate perspective variation showing robust detection.*


Figure 4.23: *Urban intersection demonstrating detection performance across varied viewing angles and distances. The scene showcases the model's ability to handle complex urban geometry with multiple lane markings at different scales and orientations.*


Figure 4.24: *Winter road conditions with optimal viewpoint for marking detection. Clear, well-maintained markings are detected with maximum confidence.*

### 4.4.6 Performance on Wet Road Surfaces

Wet road conditions present a unique set of challenges for road marking detection that combine multiple adverse factors simultaneously. When roads are wet, several phenomena occur that can significantly impact marking visibility: surface reflections create specular highlights that can obscure markings, water pooling can physically cover portions of markings, and the overall contrast between markings and pavement is often reduced due to the darkening effect of water on asphalt.


Figure 4.25: *Highway detection under wet conditions with reduced visibility due to rain and mist. Despite challenging weather conditions, the model maintains perfect detection confidence.*

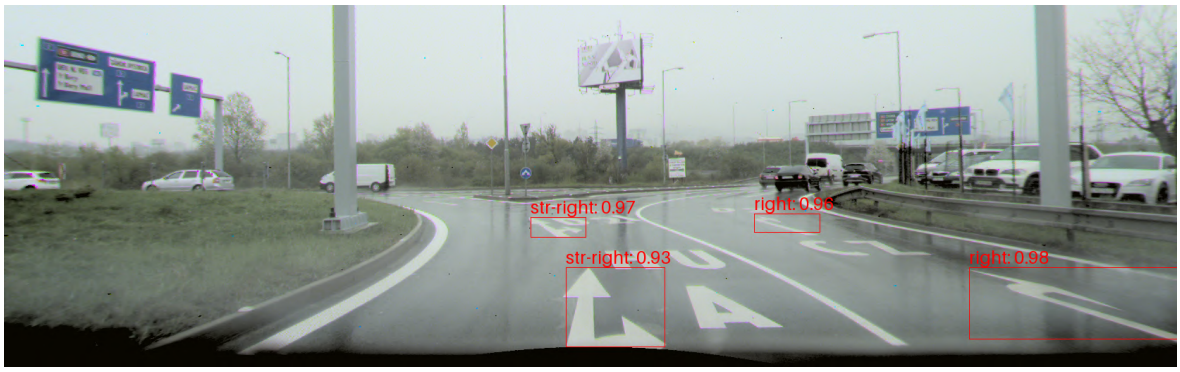


Figure 4.26: *Complex highway interchange on wet pavement showing multiple lane markings. The model successfully identifies multiple directional indicators, demonstrating consistent performance across a geometrically complex wet road scenario.*


Figure 4.27: *Urban wet road surface with moderate water accumulation affecting marking visibility. The model detects 'str' (94%) and 'str-right' (77%) with varying confidence levels. The reduced confidence on the right marking may be attributed to increased surface reflections and water pooling that partially obscure the marking boundaries. This example illustrates how wet conditions can create graduated performance impacts depending on local water accumulation patterns.*

### 4.4.7 Summary of Qualitative Observations

In summary, the qualitative analysis indicates that the model **exhibits exceptional robustness across most real-world challenges in road marking detection**, achieving consistently high confidence scores (typically >90%) across diverse operational scenarios.

**Primary Strengths:** The model demonstrates **consistent performance under varied illumination conditions** (87-99% confidence across nighttime scenarios), **remarkable resilience to partial occlusion** through effective contextual inference, and **robust handling of degraded markings** (80-100% confidence despite visible wear). A notable strength is **appropriate uncertainty calibration**, where the system prudently reduces confidence when visual information is compromised (e.g., 52% for sun-affected arrows, 77% for water-impacted markings).

**Identified Limitations:** Areas warranting further investigation include **susceptibility to false positive detections under extreme lighting conditions**, where shadow geometries can be misinterpreted as directional indicators, and **variable performance degradation when multiple environmental challenges occur simultaneously**.

**Operational Readiness:** These observations complement the quantitative validation mAP of 0.764 and demonstrate **practical operational readiness** for real-world deployment. The predominantly high confidence scores across challenging scenarios, combined with appropriate uncertainty quantification, indicate the system can reliably interpret navigational instructions under diverse autonomous driving conditions. Future development should focus on addressing false positive detections under extreme lighting and improving consistency when multiple environmental stressors are present simultaneously.

## 4.5 Findings

Our extensive experimental investigation into architectural modifications of transformer-based object detection models yields several significant insights that advance our understanding of model design principles. The most prominent finding across our experimental suite is the substantial performance enhancement achieved through backbone fine-tuning. When comparing frozen versus unfrozen configurations, we consistently observed performance improvements of 10-12 percentage points in validation mAP, establishing backbone adaptability as the single most influential factor in our study.

From a theoretical perspective, our results support a nuanced view of the bias-variance tradeoff in detection models. The wider train-validation gaps observed in higher-capacity models indicate that as representational power increases, careful regularization becomes increasingly important. The optimal configurations balanced expressive capacity with generalization ability, suggesting that detection performance is maximized not simply by scaling components independently but by creating harmonious architectures where backbone, position encoding, and transformer components complement each other.

These findings have significant practical implications for the design of transformer-based detection systems. First, practitioners should prioritize backbone fine-tuning over architectural scaling when seeking performance improvements. Second, when computational constraints limit full model fine-tuning, learned positional encodings offer an efficient alternative to increase performance.

Finally, our experimental observations suggest promising directions for future research. The equivalent performance of architecturally different but properly fine-tuned models (e.g., ResNet50 vs. ResNeXt50) raises questions about the fundamental

limits of current architectural paradigms. The consistent train-validation gaps across high-performing models suggest that advances in regularization techniques specifically tailored to detection tasks might yield further improvements. Additionally, the strong performance of learned position embeddings indicates that more sophisticated spatial encoding mechanisms could offer a high-return area for future exploration in object detection systems.

# Chapter 5

# Design & Implementation

## 5.1 System Overview

The road marking detection system is designed as a comprehensive web-based application that leverages deep learning technology for real-time detection and classification of road arrow markings. The system architecture follows a modular design approach, ensuring maintainability, scalability, and user-friendliness through a clean separation of concerns between the machine learning backend and the interactive frontend.

The complete system consists of three main integrated components: a Deformable DETR model serving as the core detection engine, a sophisticated inference pipeline for image processing and prediction, and an intuitive web-based user interface built using the Gradio framework. This architecture enables users to upload images, configure detection parameters, visualize results, and export findings through a single, cohesive interface.

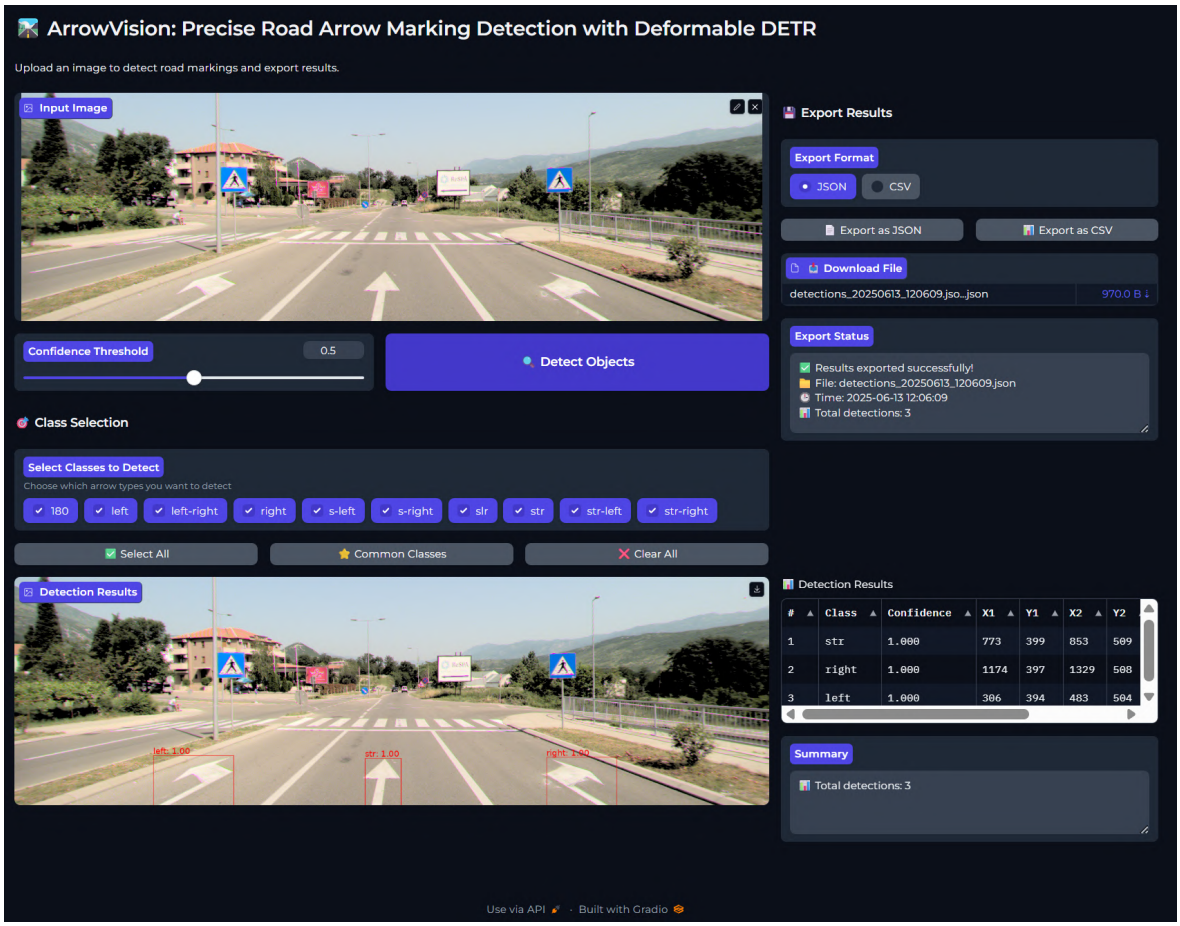


Figure 5.1: Application Interface

Figure 5.1 shows the main application interface, which presents a clean and organized layout that guides users through the detection workflow from input to results.

## 5.2 User Interface Design Philosophy

### 5.2.1 Layout Architecture

The user interface follows a logical left-to-right, top-to-bottom workflow that mirrors the natural progression of the detection task. The interface is divided into distinct functional areas, each serving a specific purpose in the detection pipeline while maintaining visual coherence and ease of navigation.

As demonstrated in Figure 5.1, the primary layout consists of two main columns: the left column houses input controls and configuration options, while the right column displays results and export functionalities. This design ensures that users can easily understand the relationship between their inputs and the resulting outputs.

### 5.2.2 Input Section Design

The input section, features a prominent image upload area that accepts various image formats. The upload component is designed with a generous size allocation (1664×512 pixels) to accommodate the typical aspect ratio of road marking images while providing sufficient detail for accurate detection.

Below the image upload area, users find an intuitive confidence threshold slider that allows real-time adjustment of detection sensitivity. This control enables users to fine-tune the system's behavior based on their specific requirements, with clear visual feedback showing the current threshold value.

### 5.2.3 Class Selection Interface

One of the system's key features is the flexible class selection mechanism. This interface component presents all ten road marking arrow types as selectable options through an organized checkbox group. The design includes descriptive labels for each arrow type, making it easy for users to understand and select the specific markings they wish to detect.

The class selection area also incorporates three convenient quick-selection buttons: "Select All" for comprehensive detection, "Common Classes" for frequently encountered arrow types, and "Clear All" for deselecting all options.

## 5.3 Detection and Results Display

### 5.3.1 Visual Results Presentation

Upon processing an image, the system displays the results in a dual-format presentation that caters to different user preferences and requirements. Figure 5.1 demonstrates how the processed image appears with detected road markings highlighted using colored bounding boxes and class labels.

The visual output maintains the original image quality while overlaying detection information in a non-intrusive manner. Each detected arrow marking is clearly delineated with appropriately colored bounding boxes, and confidence scores are displayed alongside class labels to provide users with complete information about detection certainty.

### 5.3.2 Tabular Results Summary

Complementing the visual results, the system provides a structured tabular summary of all detections. This table presents detection information in an organized

format, including sequential numbering, class names, confidence scores, precise bounding box coordinates, and dimensional information for each detected marking.

## 5.4 Advanced Functionality

### 5.4.1 Export Capabilities

The system includes comprehensive data export functionality, accessible through the export section. Users can choose between JSON and CSV formats depending on their intended use case and downstream processing requirements.

The export interface provides clear format selection options and immediate download capabilities. Upon successful export, the system displays confirmation messages with file details, timestamp information, and detection counts, ensuring users have complete visibility into the export process.

## 5.5 Technical Implementation Approach

### 5.5.1 Model Architecture Integration

The system seamlessly integrates a state-of-the-art Deformable DETR model, specifically configured for road marking detection tasks. The model architecture utilizes a ResNeXt-50 backbone for robust feature extraction, coupled with deformable attention mechanisms that enable precise localization of road marking arrows across various scales and orientations.

The model configuration supports detection of ten distinct arrow types, from basic directional indicators (straight, left, right) to complex multi-directional markings (straight-left-right, left-right). This comprehensive coverage ensures the system can handle diverse real-world road marking scenarios.

### 5.5.2 Image Processing Pipeline

The inference pipeline implements sophisticated image preprocessing techniques to ensure optimal model performance. The system handles multiple input formats, automatically converting uploaded images to the required tensor format while applying appropriate normalization based on ImageNet statistics.

The preprocessing stage maintains image quality while standardizing inputs for consistent model behavior. This approach ensures reliable detection performance regardless of the source format or quality of uploaded images.

## 5.6 System Integration and Deployment

### 5.6.1 Web-based Architecture

The Gradio-based implementation provides cross-platform compatibility through standard web browsers, eliminating the need for specialized software installation. The web-based architecture also facilitates easy sharing and collaboration, as users can access the system from any device with internet connectivity and a modern web browser.

### 5.6.2 Scalability Considerations

The modular design philosophy ensures that individual components can be enhanced or replaced without affecting the overall system functionality. This approach supports future improvements such as model updates, additional export formats, or enhanced visualization capabilities.

The clean separation between the machine learning backend and user interface frontend enables independent scaling of computational resources based on usage patterns and performance requirements.

## 5.7 Software Testing and Quality Assurance

Robust testing was a fundamental aspect of system development, ensuring not only correctness but also maintainability and extensibility. The system employs a comprehensive multi-level testing strategy, including unit tests, integration tests, and additional verification techniques suitable for machine learning applications.

### 5.7.1 Unit Tests for Core Functions

All small, self-contained functions—such as image preprocessing utilities, bounding box transformations, confidence thresholding, and data export routines—are accompanied by unit tests. These tests cover normal, boundary, and exceptional cases, guaranteeing correctness of the most fundamental building blocks. Unit tests were implemented using `pytest`, providing fast feedback during development and facilitating safe code refactoring.

### 5.7.2 Integration Testing for Model Training and Inference

Beyond function-level testing, integration tests validate that the model can be successfully trained and deployed within the pipeline. These tests cover the end-to-end

workflow: feeding synthetic or minimal labeled datasets through the full pipeline, confirming that the Deformable DETR model converges to non-trivial results. Integration tests also verify that the inference pipeline produces valid detections and correctly handles various image sizes, formats, and user-selected configuration options.

### 5.7.3 Testing Philosophy and Continuous Integration

Testing is tightly integrated with the development workflow. Key branches of the codebase are protected by continuous integration pipelines, running all automated tests on every update. This ensures rapid detection of regressions or incompatibilities and underpins the reliability of the web-based detection application.

Overall, this comprehensive testing protocol—spanning from unit- to system-level, and from traditional to ML-specific checks—ensures that the application is robust, correct, and maintainable, providing users with a dependable solution for road marking detection.

# Chapter 6

# Conclusions

## 6.1 SWOT Analysis of the Thesis Project

**Strengths**:

- **Thorough Experimental Evaluation:** Extensive ablation studies explore architectural variants, positional embeddings, backbone freezing/unfreezing, and multi-head attention, enabling a fine-grained understanding of what drives performance.
- **Practical Relevance:** The work is strongly motivated by real-world deployment needs in autonomous vehicles.
- **Comprehensive Dataset:** The use of a large, annotated and high-resolution proprietary dataset focusing on painted road signs enhances the practical significance and applicability of the research conclusions.

**Weaknesses**:

- **Limited Dataset Accessibility:** As the dataset is proprietary, the reproducibility of results and external benchmarking is constrained.
- **Model Complexity:** Despite efficiency efforts (e.g., with Deformable Attention), transformer models require significant computational resources for training, which may limit adoption in resource-constrained environments.
- **Explained Variability:** Some performance limitations—especially for small or rare markings—persist, suggesting further work is needed in data augmentation and model calibration.

**Opportunities**:

- **Transfer to Other Domains:** The flexibility of the model architecture and training pipeline makes adaptation to other perception tasks (e.g., classical traffic sign detection, lane marking, obstacle detection) feasible.
- **Incorporation of Additional Modalities:** The transformer-based framework can integrate supplementary sensor data (e.g., LiDAR, radar) for improved robustness.
- **Further Model Optimization:** Knowledge distillation, pruning, and quantization could enable deployment on even more constrained hardware while maintaining accuracy.
- **Learning with Limited/Labeled Data:** Exploring semi-supervised and few-shot learning approaches may alleviate dataset limitations and improve generalization.

**Threats**:

- **Rapid Field Evolution:** The pace of progress in deep learning for perception may outdate current solutions or require continual upgrades.
- **Deployment Challenges:** Safety-critical environments like autonomous driving demand stringent verification, validation, and regulatory compliance, which could delay integration.
- **Bias and Generalization Risks:** Models trained on limited or non-representative data may underperform in unseen scenarios, introducing safety hazards.

## 6.2 Ethical Evaluation of the Proposed Approach

### 6.2.1 Fairness and Bias

The approach acknowledges the issue of class imbalance—a form of dataset bias—by quantifying performance across all categories and explicitly reporting class-wise metrics. While model and training design choices (e.g., using weighted loss functions) attempt to mitigate this bias, performance on minority classes remains lower, highlighting a persistent fairness concern. Future development should consider curated data collection and further loss adaptations to ensure no class is systematically undetected, thus reducing the risk of algorithmic discrimination.

### 6.2.2 Safety and Security

The application domain—autonomous driving—imposes strict safety requirements. The proposed detectors are evaluated not only for accuracy but also for recall (to maximize the detection of relevant markings) and for scale-robustness (to avoid omissions of critical, but small, signs). Model configurations are compared with special attention to overfitting, which if left unchecked could reduce safety. While adversarial robustness is not explicitly tackled in this work, the modularity of the approach allows for the integration of defensive strategies in the future.

### 6.2.3 Responsibility and Accountability

The thesis is transparent about the limitations of its models, the dataset used, and the domains in which its findings can and cannot be guaranteed. The proprietary nature of the dataset is clearly disclosed, and the work is positioned as experimental, not for immediate deployment without further safety validation and regulatory review. Practitioners using these approaches should assume responsibility for final validation.

## 6.3 Conclusions and Future Directions

This thesis demonstrates that fully transformer-based approaches—specifically Deformable DETR with optimized ResNet backbones and positional embeddings—yield substantial improvements for the challenging task of painted road sign detection in autonomous driving contexts. The combination of comprehensive ablation studies and detailed metric analysis identifies configuration choices (notably, backbone fine-tuning and learned position embeddings) that maximize performance and efficiency.

Despite impressive results, especially for majority classes and large/medium markings, significant challenges remain for small object detection and minority class robustness. The analysis reinforces that backbone adaptability is more influential than raw scale, and that learned spatial representations are competitive with larger, static models.

Looking forward, the following research directions are proposed:

- **Dataset Enrichment:** Curating more balanced and diverse data, especially for rare markings, will improve both overall fairness and robustness.
- **Advanced Regularization:** Techniques like strong data augmentation, focal or adaptive losses, and domain adaptation should be explored to boost minority class and small object performance.

- **Model Explainability:** Integrating more systematic explainability modules (e.g., attention map analysis, counterfactual explanations) will foster greater trust and regulatory acceptance.
- **Safety and Adversarial Robustness:** Explicitly evaluating and reinforcing the system's security against adversarial attacks, noise, and domain shift is critical for real-world deployment.
- **Resource-Efficient Deployment:** Further investigation into pruning, quantization, and distillation is necessary for scalable, real-time onboard inference.
- **Ethical AI Practices:** Continued transparent reporting, documentation of data/-model limitations, and community benchmarking on shared datasets will enhance societal trust and accountability.

In summary, this thesis contributes both significant technical advancements and practical insights for transformer-based road marking detection, establishing foundational guidance for future academic and industrial research in autonomous vehicle perception systems.

# Bibliography


[AAJD+19] Eduardo Arnold, Omar Y Al-Jarrah, Mehrdad Dianati, Saber Fallah, David Oxtoby, and Alex Mouzakitis. A survey on 3d object detection methods for autonomous driving applications. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3782–3795, 2019.

[AZLL19] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Convergence and margin of adversarial training on separable data. *arXiv preprint arXiv:1905.10304*, 2019.

[BIK+20] Alexander Buslaev, Vladimir I Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A Kalinin. Albumentations: fast and flexible image augmentations. *Information*, 11(2):125, 2020.

[BKH16] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[BL+07] Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 34(5):1–41, 2007.

[BMM18] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

[BNP20] Julieta Beal, Eduardo Norambuena, and Jorge E Pezoa. Toward transformation-based road sign detection. *IEEE Access*, 8:224093–224105, 2020.

[BWL20] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

[CCZC18] Guowei Chen, Keqi Chen, Fei Zhang, and Tao Chen. Robust road markings detection and recognition using density-based grouping and machine learning techniques. *Applications of Artificial Intelligence*, 2:0–16, 2018.

[CGRS19] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. In *arXiv preprint arXiv:1904.10509*, 2019.

[CJL+19] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019.

[CKLM19] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does bert look at? an analysis of bert's attention. *arXiv preprint arXv:1906.04341*, 2019.

[CLD+20] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. In *International Conference on Learning Representations*, 2020.

[CMS+20] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[CTM+21] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[CTW+21] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Conditional positional encodings for vision transformers. In *International Conference on Learning Representations*, 2021.

[CZM+19] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019.

[DBK+20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is

worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[DBX+19] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. *arXiv preprint arXiv:1904.08189*, 2019.

[DCLC21] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1601–1610, 2021.

[DGR+19] Tri Dao, Albert Gu, Alexander J Ratner, Virginia Smith, Christopher De Sa, and Christopher Ré. A kernel theory of modern data augmentation. *Proceedings of machine learning research*, 97:1528, 2019.

[DQX+17] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 764–773, 2017.

[DZHD22] Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11963–11975, 2022.

[FHL20] Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. *arXiv preprint arXiv:2019.03171*, 2020.

[FZG+21] Chengjian Feng, Yujie Zhong, Yu Gao, Matthew R Scott, and Weilin Huang. Tood: Task-aligned one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3510–3519, 2021.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[GFZ+20] Chaoxu Guo, Bin Fan, Qian Zhang, Shiming Xiang, and Chunhong Pan. Augfpn: Improving multi-scale feature learning for object detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12595–12604, 2020.

[Gir15] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[GLL18] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Dropblock: A regularization method for convolutional networks. In *Advances in neural information processing systems*, pages 10727–10737, 2018.

[GSBL21] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, 2021.

[GXL+22] Meng-Hao Guo, Cheng-Ze Xu, Yu-Bin Liu, Zheng-Ning Liu, Peng-Tao Jiang, Tai-Jiang Wang, Li Tang, Kaiwen Xu, Zheng-Jun Jiang, Cheng Xu, et al. Beyond self-attention: External attention using two linear layers for visual tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):2341–2355, 2022.

[GZW+21] Peng Gao, Minghang Zheng, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fast convergence of DETR with spatially modulated co-attention. *arXiv preprint arXiv:2101.07448*, 2021.

[HF17] Paul Henderson and Vittorio Ferrari. End-to-end training of object class detectors for mean average precision. *Asian Conference on Computer Vision*, pages 198–213, 2017.

[HG09] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

[HGD19] Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4918–4927, 2019.

[HGDG17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[HMD15] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[HR18] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018.

[HRS+17] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7310–7319, 2017.

[HSS18] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[HVD15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[HWZ+21] Aoran Hou, Zhuofan Wang, Linchao Zheng, Andy J Lu, and Yi Yang. Applying adversarial augmentation to enhance object detection in urban scenes. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2706–2715, 2021.

[HXW+21] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *Advances in Neural Information Processing Systems*, 34:15908–15919, 2021.

[HZRS16a] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[HZRS16b] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645, 2016.

[IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[JGBG20] Joel Janai, Fatma Güney, Aseem Behl, and Andreas Geiger. Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends® in Computer Graphics and Vision*, 12(1-3):1–308, 2020.

[JK19] Justin M Johnson and Taghi M Khoshgoftaar. Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1):1–54, 2019.

[KBYH21] Namuk Kim, Junyeong Byun, Kang In Yu, and Sungbae Hong. How do vision transformers work? *arXiv preprint arXiv:2202.06709*, 2021.

[KMN+17] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2017.

[KNH+22] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM Computing Surveys*, 54(10s):1–41, 2022.

[Koh95] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, volume 2, pages 1137–1143. Morgan Kaufmann Publishers Inc., 1995.

[KPR+17] James Kirkpatrick, Razvan Pascanu, Neil Rabinovich, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[KSL19] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2661–2671, 2019.

[Kuh55] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[KXR+19] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019.

[LAE+16] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, 2016.

[LBBH98] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[LD18] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European conference on computer vision (ECCV)*, pages 734–750, 2018.

[LDG+17] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[LGG+17] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[LLC+21] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[LLG+20] Liyuan Liu, Xiaodong Liu, Jianfeng Gao, Weizhu Chen, and Jiawei Han. Understanding the difficulty of training transformers. *arXiv preprint arXiv:2004.08249*, 2020.

[LLSL20] Kimin Lee, Kibok Lee, Jinwoo Shin, and Honglak Lee. Network randomization: A simple technique for generalization in deep reinforcement learning. *arXiv preprint arXiv:1910.05396*, 2020.

[LMB+14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. *European Conference on Computer Vision*, pages 740–755, 2014.

[LMGH22] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. *arXiv preprint arXiv:2203.16527*, 2022.

[LMW+22] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[LPW+17] Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The expressive power of neural networks: A view from the width. *Advances in neural information processing systems*, 30, 2017.

[LPY+18] Zeming Li, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, and Jian Sun. Detnet: A backbone network for object detection. *Proceedings of the European conference on computer vision (ECCV)*, pages 334–350, 2018.

[LQSJ22] Shilong Liu, Feng Qi, Hao Shi, and Hongsheng Jia. Dab-detr: Dynamic anchor boxes are better queries for detr. In *International Conference on Learning Representations*, 2022.

[LWHY19] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 510–519, 2019.

[LWK+20] Yu Li, Tao Wang, Bingyi Kang, Sheng Tang, Chunfeng Wang, Jintao Li, and Jiashi Feng. Overcoming classifier imbalance for long-tail object detection with balanced group softmax. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10991–11000, 2020.

[LXT+18] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems*, pages 6389–6399, 2018.

[LZL+22] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. DN-DETR: Accelerate DETR training by introducing query denoising. *arXiv preprint arXiv:2203.01305*, 2022.

[MCF+21] Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional detr for fast training convergence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3651–3660, 2021.

[MGR+18] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. *Proceedings of the European conference on computer vision (ECCV)*, pages 181–196, 2018.

[MKAT18] Sam McCandlish, Jared Kaplan, Dario Amodei, and OpenAI Team. An empirical model of large-batch training. *arXiv preprint arXiv:1812.06162*, 2018.

[ML18] Dominic Masters and Carlo Luschi. Revisiting small batch training for deep neural networks. *arXiv preprint arXiv:1804.07612*, 2018.

[MLN19] Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, pages 14014–14024, 2019.

[MLZ+20] Shuai Ma, Zhi-Quan Lin, Chang Zhang, Hualin Sun, Di Wang, Ronghao He, Hanyuan Wu, Jin-Ge Yan, and Ming-Ming Cheng. Adaptive calibration for object detection. In *European Conference on Computer Vision*, pages 480–495. Springer, 2020.

[MP43] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[MTBVG13] Markus Mathias, Radu Timofte, Rodrigo Benenson, and Luc Van Gool. Traffic sign recognition - how far are we from the solution? In *Proceedings of the International Joint Conference on Neural Networks*, pages 1–8. IEEE, 2013.

[MTM12] Andreas Mogelmose, Mohan M Trivedi, and Thomas B Moeslund. Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey. *IEEE Transactions on Intelligent Transportation Systems*, 13(4):1484–1497, 2012.

[MV18] Omid Madani and Jorg Vlasselaer. Deep neural adaptation approach for prefix recommender system. *arXiv preprint arXiv:1807.03821*, 2018.

[OCKA18] Kemal Oksuz, Baris Cam, Sinan Kalkan, and Emre Akbas. Localization recall precision (lrp): A new performance metric for object detection. *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 504–519, 2018.

[OCKA20] Kemal Oksuz, Baris Cam Cam, Sinan Kalkan, and Emre Akbas. Imbalance problems in object detection: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3388–3415, 2020.

[PRV+17] Krishna Patel, Keerthy Rambach, Tony Visentin, Dequn Yu, Chaoyang Chen, Zhijun Ji, Dinesh Mani, and Ram Sridhar. Embedded vision

sensing for autonomous vehicles traffic signage detection using deep learning. In *Proceedings of the 10th International Conference on Utility and Cloud Computing*, pages 207–208, 2017.

[PXL+18] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini-batch object detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6181–6189, 2018.

[RBK21] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021.

[RDGF16a] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[RDGF16b] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[RF17] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[RF18] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[RHGS15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[RHW86] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[Ros58] Frank Rosenblatt. *The perceptron: a probabilistic model for information storage and organization in the brain*. Psychological review, 1958.

[RZ17] Mengye Ren and Richard S Zemel. End-to-end instance segmentation with recurrent attention. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6656–6664, 2017.

[SAN16] Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng. End-to-end people detection in crowded scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2325–2333, 2016.

[SD18] Bharat Singh and Larry S Davis. An analysis of scale invariance in object detection–snip. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3578–3587, 2018.

[SHK+14] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[Sid24] Sider AI. Sider ai: Ai-powered writing assistant, 2024. Accessed: June 15, 2025.

[SKD18] Bharat Singh, Siddarth Krishnan, and Larry S Davis. Layer-specific adaptive learning rates for deep networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2371–2375, 2018.

[SKYL18] Samuel L Smith, Pieter-Jan Kindermans, Chris Ying, and Quoc V Le. Don't decay the learning rate, increase the batch size. *arXiv preprint arXiv:1711.00489*, 2018.

[SLJ+15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[SSSG17] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017.

[SSSI12] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. In *Neural Networks*, volume 32, pages 323–332. Elsevier, 2012.

[SYL20] Prasan Singh, Kimin Yow, and Robert Qiu Liang. End-to-end learning of object motion estimation from retinal events for event-based object tracking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):11534–11542, 2020.

[SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[TCA19] Dogancan Temel, Min-Hung Chen, and Ghassan AlRegib. Traffic sign detection under challenging conditions: A deeper look into performance variations and spectral characteristics. *IEEE Transactions on Intelligent Transportation Systems*, 21(9):3663–3673, 2019.

[TGW+21] Yuan Tian, Judith Gelernter, Xiaolei Wang, Jianwei Li, and Yang Yu. A comprehensive survey on traffic sign detection. *IEEE Transactions on Intelligent Transportation Systems*, 23(3):2562–2581, 2021.

[TL19] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114, 2019.

[TSCH19] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019.

[TZHG20] Yiren Tang, Yingbo Zhuang, Ying He, and Randy Goebel. Scst: Self-correcting structure in transformer for neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4534–4544, 2020.

[VHMAS+18] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, 2018.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[VTM+19] Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual*

*Meeting of the Association for Computational Linguistics*, pages 5797–5808, 2019.

[WAK+21] Tong Wang, Rafi M Anwer, Salman Khan, Fahad Shahbaz Khan, Yanwei Yan, Shah Nawaz, and Jorma Laaksonen. Adaptive class suppression loss for long-tail object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3103–3112, 2021.

[WD18] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. In *Neurocomputing*, volume 312, pages 135–153. Elsevier, 2018.

[WGGH18] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[WHD+20] Xin Wang, Thomas E Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. In *Proceedings of the 37th International Conference on Machine Learning*, pages 9919–9928. PMLR, 2020.

[WLK+20] Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. In *International Conference on Machine Learning*, pages 9939–9948. PMLR, 2020.

[WSC+20] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. The devil is in the details: Delving into unbiased data processing for human pose estimation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5700–5709, 2020.

[WTJ21] Ross Wightman, Hugo Touvron, and Herve Jegou. Resnet strikes back: An improved training procedure in timm. *arXiv preprint arXiv:2110.00476*, 2021.

[WXZ+22] Chenyang Wang, Defu Xu, Yonggang Zhu, Renjie Zhang, Junchi Zhang, and Jifeng Dai. Anchor DETR: Query design for transformer-based detector. *arXiv preprint arXiv:2109.07107*, 2022.

[WYW+19] Wencheng Wang, Xiaohui Yuan, Xiaojin Wu, Yunhong Liu, and Saeid Ghanbarzadeh. Adaptive weighted attention network with camera spectral sensitivity prior for image dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[WZR20] Sen Wu, Hongyang R Zhang, and Christopher Ré. Understanding and improving information transfer in multi-task learning. *arXiv preprint arXiv:2005.00944*, 2020.

[WZZW21] Dong Wang, Yang Zhang, Kai Zhang, and Limin Wang. Feature pyramid transformer. *ECCV*, 2021.

[XGD+17] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[XSZ+19] Jingjing Xu, Xu Sun, Zhiyuan Zhang, Guangxiang Zhao, and Junyang Lin. Understanding and improving layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019.

[YCBL14] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[YLRÖ18] Dingkang Yang, Lei Li, Keith Redmill, and Ümit Özgüner. Towards driving scene understanding: A dataset for learning driver behavior and causal reasoning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7699–7707, 2018.

[YLW+22] Xiaoguang Yuan, Yingjie Li, Feng Wei, Guangzheng Ren, Rui Zhang, Youfu Wang, Fan Wang, and Xu Zhang. Road sign detection and recognition: A survey. *Neurocomputing*, 505:87–115, 2022.

[YLZ+21] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal self-attention for local-global interactions in vision transformers. In *Advances in Neural Information Processing Systems*, volume 34, pages 171–183, 2021.

[YNKW21] Chenyang Yang, Praveen Narayana, Deepak Krishnan, and Yang Wang. Adversarial self-supervised learning for semi-supervised 3d action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1753–1762, 2021.

[YWC+21] Bo Yang, Jianan Wang, Ronald Clark, Qingyuan Hu, Sen Wang, Andrew Markham, and Niki Trigoni. Learning to separate: Detecting heavily-occluded objects in urban scenes. *arXiv preprint arXiv:2103.10573*, 2021.

[ZJK20] Hengshuang Zhao, Jiaya Jia, and Vladlen Koltun. Exploring self-attention for image recognition. *arXiv preprint arXiv:2004.13621*, 2020.

[ZK16] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference*, 2016.

[ZKHB22] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12104–12113, 2022.

[ZLW19] Yuchen Zhang, Percy Liang, and Martin J Wainwright. Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. *COLT*, 2019.

[ZSGY23] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Pattern Recognition*, 135:109298, 2023.

[ZSL+20] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.

[ZSL+21] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection, 2021.

[ZWK19] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.

[ZZXW19] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232, 2019.