National College of Ireland

Advancing Autonomous Driving: CNNs vs. Transformers for Semantic Image Segmentation

MSc Research Project
Data Analytics

Ayush Kumar Shrivas
Student ID: x23331666

School of Computing
National College of Ireland

Supervisor: Yalemisew Abgaz

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Ayush Kumar Shrivas
Student ID:	x23331666
Programme:	Data Analytics
Year:	2024-2025
Module:	MSc Research Project
Supervisor:	Yalemisew Abgaz
Submission Due Date:	11/08/2025
Project Title:	Advancing Autonomous Driving: CNNs vs. Transformers for Semantic Image Segmentation
Word Count:	4732
Page Count:	19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Ayush Kumar Shrivas
Date:	11th August 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Advancing Autonomous Driving: CNNs vs. Transformers for Semantic Image Segmentation

Ayush Kumar Shrivas
x23331666

Abstract

This study evaluates CNN-based and hybrid transformer-based models for semantic image segmentation in autonomous vehicles, which addressing the need for accurate, real-time environmental understanding. Using the Cityscapes dataset, five models—UNet, SegNet, PSPNet (CNN-based), SegFormer, and UNETR (transformer-based) were compared under a unified pipeline. Results show that the transformer-based SegFormer model achieved the highest performance (87% pixel accuracy, 75% mIoU, 86% Dice Score), excelling in complex urban scenarios, while CNNs were more efficient, they were less precise in challenging scenes. The research demonstrates the superior segmentation capabilities of transformer-based models for autonomous driving but notes their higher computational demands, highlighting the importance of further optimization and broader datasets for real-world deployment.

1 Introduction

At present the world is about to experience a major pivot in the evolution of transportation, one that reverberates the significant shift to electric drivetrains from conventional gasoline powered engines. Just like the adaptation of electrification in vehicles transformed the path of energy and mobility, Autonomous vehicles are set to redefine navigation as we know it. AVs provide a real technological response to the global challenges like urban congestion, transportation emissions, and alarmingly high road fatalities. It brings the potential of safer roads, reduced pollution and saving lives besides convenience alone. In order to successfully traverse through the real world and share with people while maintaining harmony, these vehicles have to truly understand their environment. This forms their fundamental property of not just detecting objects in the path, but semantically interpret their entire surrounding with pixel-level precision. And this begins with semantic segmentation, a critical task that allows these vehicles to distinguish safe passage between pedestrians, other vehicles, traffic signs, sidewalks and background clutter.

This research tries to explore the differences in approach of two major techniques of semantic segmentation, Conventional CNN based models and hybrid transformer-based models when applied to real-world data from Cityscapes Dataset [1]

[1] Cityscapes Dataset .

1.1 Research Background

Automobile industry has come a long way since the invention of gasoline-powered engines in 1800s. From mechanical breakthroughs in the early year to AI powered feature integration, this industry has continuously evolved and pushed the boundaries of what vehicles can do. As per a report by World Health Organisation [2] more than 1.3 million people die in road crashes and almost 90% of which are accounted for human error. In recent decades, the demand has shifted from raw engine power to intelligent systems that can perceive the scenes around the vehicle and aid the safety and comfort of passengers and others around the vehicle. This has led to the integration of various visual sensors into the vehicle which have allowed for features like lane-assist, adaptive cruise control, adaptive suspension systems to exist in todays automobiles. Semantic segmentation forms the basis of most of these features, helping vehicles to build an understanding of their surrounding in real time by assigning class label to each pixel in captured images.

Until recently, Convolutional Neural Networks (CNN) have dominated the semantic segmentation tasks. Model architectures such as U-Net, PSPNet, DeepLabv3+ have become the standard due to their strong performance resulting from their hierarchical structure and efficiency in feature extraction. CNN architectures are naturally good at capturing local features such as edges. However, the very advantage forms the basis of their limitation as they lack global context which is of utmost importance as they might miss essential information in cluttered and complex real world street scenario.

Transformer-based models, which were originally built to address the same issue in the world of Natural Language Processing, are providing a solution to this problem in image segmentation. Termed as Vision Transformers (ViTs), they have brought in their ability to capture global context by leveraging self-attention mechanisms. This helps to tie up a distant traffic signal to the lane of the road currently being utilised. However, while these architectures are extremely powerful, they require a significantly larger datasets, which in turn increases the training time making them computationally expensive. Hybrid models such as SegFormer tries to provide a balance between both approaches, i.e., local feature extraction from CNN based architectures and Global context reasoning from transformer-based architectures.

1.2 Research Motivation

Autonomous vehicles are quickly becoming a reality of this world, from automated drone delivery, to smart traffic systems, it is just a matter of time for the possibility of full-fledged driverless vehicles eliminating the human error aspect in driving entirely, while also making traffic flow more laminar and reducing effect on climate. For this to become a safe reality, the systems' understanding of their surrounding must have the same nuance and context awareness that humans have.

CNN-based architectures are well established, reliable and well understood making them the go-to models for semantic segmentation in current times. However, their lack of global context limits their efficiency in complex, messy and cluttered real-world scenarios. Hybrid transformer-based architectures seem to cover for this limitations of CNN-based models, while also bringing in better performance on relatively small object classes. However, this improvement brings it's own challenges. The major one being the complex architecture of these models which increases the training cycles that in result increases

[2] WHO .

the computational cost of deploying them.

The motivation of this research is to evaluate the practicality of hybrid transformer-based models over conventional CNN-based models for semantic segmentation in Autonomous vehicles operating under complex real-world driving conditions.

1.3 Research Objective

Research Question: How do CNN-based and hybrid transformer-based architectures perform in semantic image segmentation in terms of computational efficiency, accuracy, resilience under complex and diverse real-world driving scenarios and their suitability for integration into production-level autonomous vehicles.

2 Related Work

The process of classifying every pixel in an image into predefined classes or categories is the platform which enables autonomous vehicles to understand and interpret their surroundings to navigate in a safe manner. Recent researches have focussed on improving the overall precision with which the segmentation models perform under varying and complex environmental conditions which present themselves during real-world driving.

Data selection process generally reflects the various practical challenges which are addressed through these researches. Zhou et al. (2020) uses real world dataset to capture year round variability in environment, while Chen et al. (2019) focuses on urban dataset emphasizing on the critical classes such as pedestrians. Ha et al. (2017) & Wang et al. (2022) utilises a custom Thermal dataset and a custom GAN generated synthetic images to address nighttime low-visibility. Fan et al. (2024) on the other hand utilises a combination of real-world data and dataset generated through controlled experiments to balance speed and accuracy.

Preprocessing of data before feeding it to the models forms a crucial part in the defining the performance of the architectures. Zhou et al. (2020) Uses LiDAR data to automatically generate labels for road segmentation while Chen et al. (2019) makes use of a hierarchical object classes, assigning weights to each category based on their significance in driving context. Ha et al. (2017) & Wang et al. (2022) aimed at making night-time low-light visibility dataset more visible by infusing thermal images and incorporating a light enhancement network with GAN respectively. Fan et al. (2024) on the other hand utilised a hybrid transformer-CNN based architecture to enhance feature extraction.

Zhou et al. (2020) utilised an already existing and widely accepted models and benchmarked it while introducing it to various conditions. Chen et al. (2019) also utilised already existing models, mainly focusing on fine-tuning the hyperparameters to increase accuracy in categories with higher weights. Ha et al. (2017), Fan et al. (2024) and Wang et al. (2022) on the other hand proposed new architecture – MFNet, SegTransConv and SFNet – N, which are tailored for best performance utilising thermal fused data, real-time performance and night driving respectively.

mIoU is the widely accepted evaluation technique for image segmentation tasks across various fields as it takes into account results spanning all categories of classification. In addition to this metric, Zhou et al. (2020) introduces a robustness metric to separately asses the performance of the model across various conditions it was exposed to. Fan et al. (2024) on the other hand proposes a composite metric which encapsulates accuracy, speed

and computational cost. Muhammad et al. (2022) emphasises on the need to accept an unified benchmarking standard in order to maintain a level comparison field.

2.1 CNN-Based Architectures

CNN-based architectures have made image segmentation a reality. The most successful models of this architecture – Unet, SegNet & PSPNet each tackles a different challenge presented n front of them. They all utilise distinct architectural strengths to suit various real-world constraints. U-Net excels in capturing the spatial accuracy with significantly less training data. While SegNet focusses on memory efficient tasks tacking hardware limitation, PSPNet on the other hand leverages its pyramid pooling layer to retain global context to an extent, making them more efficient in complex, multi-scale scenarios.

2.1.1 U-Net

Anwar et al. (2024) implements a UNet architecture optimised for low computational cost over KITTI and custom UHA dataset. This study found this architecture to provide robust results across various lighting and weather conditions. Giurgi et al. (2022) and Patri et al. (2024) performs comparative study between multiple CNN based models when applied over datasets from Cityscapes and CamVid. Both the studies found the UNet to perform exceptionally well in boundary detection specially when the size of the data utilised is small. It was also noted that the strength of UNet lies in it's skip connection layer as it helps preserving feature details. It also requires minimal data preprocessing. The main limitation of this architecture lies in the lack of global context, which makes it less efficient in complex scenes.

2.1.2 SegNet

Gad et al. (2020) utilised cityscapes dataset to explore SEGNet architecture in urban multi-object scenarios. The performance of which is then evaluated against UNet & FCNs across metrics including computation time, pixel accuracy etc. Jebamikyous and Kashef (2021) adds in a weightage to categories called "Importance-Aware Loss" to prioritize critical categories in real world driving scenarios such as pedestrians. Both these studies finds that the SEGNet excels over other models in memory and processing time, therefore tackling hardware constraints. However, it lags behind the same models in feature preservation and accuracy.

2.1.3 PSPNet

PSPNet models makes use of a pyramid pooling layer to capture global context to an extent. Özen, Kaya, Semiz and Çelebi (2023) implements PSPNet in combination with post-processing for a multi-lane detection system over TuSimple dataset. Netrakar et al. (2024) and Pham (2021) on other hand provides a comparative study on performance of PSPNet against FCNs, UNet, SegNet and FPNet. Preprocessing of the data in these studies is pretty minimal and often includes just normalization of labelled data. The pyramid-pooling module in PSPNet provides exceptional results in complex environments when measured across various metrics which includes F1-Score IoU, etc. However, the main drawback of this models is the computational complexity it brings to the table.

2.2 Transformer-Based Architectures

Transformer-based models were originally developed for NLP tasks. However, they're now very much made their way into the image segmentation tasks. These architectures leverage their attention mechanism which enables them to maintain the global context while selecting features and therefore particularly excelling in complex urban scenarios with significant amount of noise. Models such as Vision Transformers (ViT), SegFormer consistently outperforms the CNN-based architectures in mIoU metric and efficiency.

Liu et al. (2023) and Cui and Lei (2024) utilises real world datasets such as cityscapes and Pascal VOC to carry out their research while Bavirisetti et al. (2023) and Guo et al. (2024) add on synthetic datasets to pretrain the models for scenarios where real world data is scarce to acquire. General steps of preprocessing which involves image normalization, resizing are carried out before special preprocessing steps of targeted image enhancement is carried out to handle low-light conditions, and make them more manageable by improving visibility in night-time images Guo et al. (2024). In order to effectively handle the global context through self-attention mechanisms, authors of Bavirisetti et al. (2023) and Dong et al. (2021) adopted another step in preprocessing to divide images into small patches.

These architectures such as SegFormer, Vision Transformer (ViT) show significant promise providing substantial improvement over mIoU metric and pedestrian intention detection when compared to conventional CNN-based models such as DeepLabV3+ Liu et al. (2023), Bavirisetti et al. (2023), Guo et al. (2024) and Cui and Lei (2024). These architectures are capable of performing parallel or multi-task segmentation, significantly improving the computational times over single task alternatives Bavirisetti et al. (2023). The Evaluation of these models are usually employ mIoU metrics, F1 Score etc. providing a solid level base for assessment.

Despite these strengths, these architectures pose significant challenges, specially in computational aspect as they require substantial resources for training making them exponentially complex for real-time deployment and scaling Cui and Lei (2024). Also, it's dependency on large annotated dataset remains the single largest weakness as these are resource intensive Liu et al. (2023), Bavirisetti et al. (2023).

3 Methodology

This study follows the Knowledge Discovery in Databases (KDD) methodology to utilise its structured process of analysing and extracting knowledge from large volume of raw data making it suitable for the objective of this research. This methodology performs best with a focussed pipeline dedicated on pattern discovery & interpretation which forms the backbone of semantic segmentation approaches.

3.1 Data Selection

The dataset utilised for the purpose of this study is obtained from Kaggle [3], which in itself is a subset of a larger pool available on Cityscapes [4]. Both of these sources are publicly available for research use. The dataset consists of a total of 3475 labelled image files taken from videos of vehicles driving in Germany. The left half of these images are

[3]Kaggle Dataset .

[4]Cityscapes Dataset .

the actual stills from the videos while the right half contains the semantic segmentation labels. The data is annotated with 30 classes, 19 of which are used for evaluation purposes, covering categories such as pedestrians, road, other vehicles and buildings. In order to accommodate this study, the dataset is divided into 2975 images for training, 250 for validation and 250 for testing.

3.2 Data Pre-Processing & Transformation

To load and prepare the data for each model, a single uniform pipeline was built to a ensure level playground. A custom data generator class was built to handle this process for both frameworks of keras and pytorch. The steps of which are shown in Figure 1 as follows:

1. Image Loading: Each image is read from the storage using OpenCV library.
2. Resizing: Every image is resized to a fixed dimension of 256x512 pixels to ensure uniform input dimensions to neural network.
3. Splitting: The resized image is then split in half vertically at its midpoint to form 2 images of 256x256 pixels each. The left half of the image is treated as input image while the right half is treated as the segmentation mask which the model is trying to learn and predict.
4. Normalisation: The pixel values of both the images are then scaled down to fit into a range between 0 and 1, efficiently stabilizing the process and accelerating training process by converging faster.

All of these steps are encapsulated in custom data loaders which efficiently load and preprocess data while also feeding it to GPU in batches. This also avoids any unnecessary slowdown from disc reading.

3.3 Modelling Experimentation

This study explores the CNN based architectures and more recent transformer-based architectures to evaluate their effectiveness in semantic image segmentation. The diversity in the models selected offer a balanced comparison between encoder-decoder frameworks against attention-based methods. After carefully considering the researches mentioned in literature review, the following models are selected for the purpose of this study:

(A) CNN-Based Architecture

  (a) UNet
  (b) SegNet
  (c) PSPNet

(B) Transformer-Based Architecture

  (a) SegFormer
  (b) UNETr

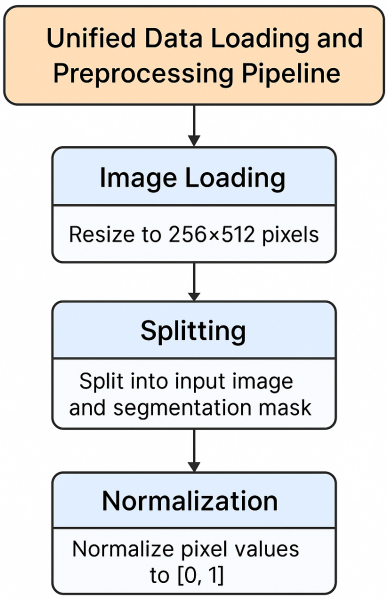

Figure 1: Process of Data Preparation

3.4 Evaluation

The quantitative performance of each model is assessed using three standard evaluation metrics for semantic segmentation tasks.

1. Pixel Accuracy: Measure the ratio of accurately classified pixels to total pixels.

$$\text{Pixel Accuracy} = \frac{\text{Number of Correctly Predicted Pixels}}{\text{Total Number of Pixels}}$$

2. Mean Intersection over Union (mIoU) : measuring the overlap of predicted segmentation mask and the actual truth. It is the average of intersection over Union for each class.

$$\text{IoU for class } c = \frac{\text{Number of True Positive pixels for class } c}{\left(\begin{array}{l}\text{Number of True Positive pixels for class } c \\ +\text{Number of False Positive pixels for class } c \\ +\text{Number of False Negative pixels for class } c\end{array}\right)}$$

$$\text{Mean IoU} = \frac{\text{IoU for class 1} + \text{IoU for class 2} + \cdots + \text{IoU for class N}}{N}$$

3. Dice Score (F1-Score): Measure the harmonic mean of precision and recall.

$$\text{Dice Score for class } c = \frac{2 \times \text{Number of True Positive pixels for class } c}{\left(\begin{array}{l}2 \times \text{Number of True Positive pixels for class } c \\ +\text{Number of False Positive pixels for class } c \\ +\text{Number of False Negative pixels for class } c\end{array}\right)}$$

4 Design Specification

The workflow used to carry out this study has been illustrated in Figure 2. The complete process can be classified into 4 distinct phases, Setup & data preparation, Model definition and configuration, Training & Evaluation and Inference & Visualization.

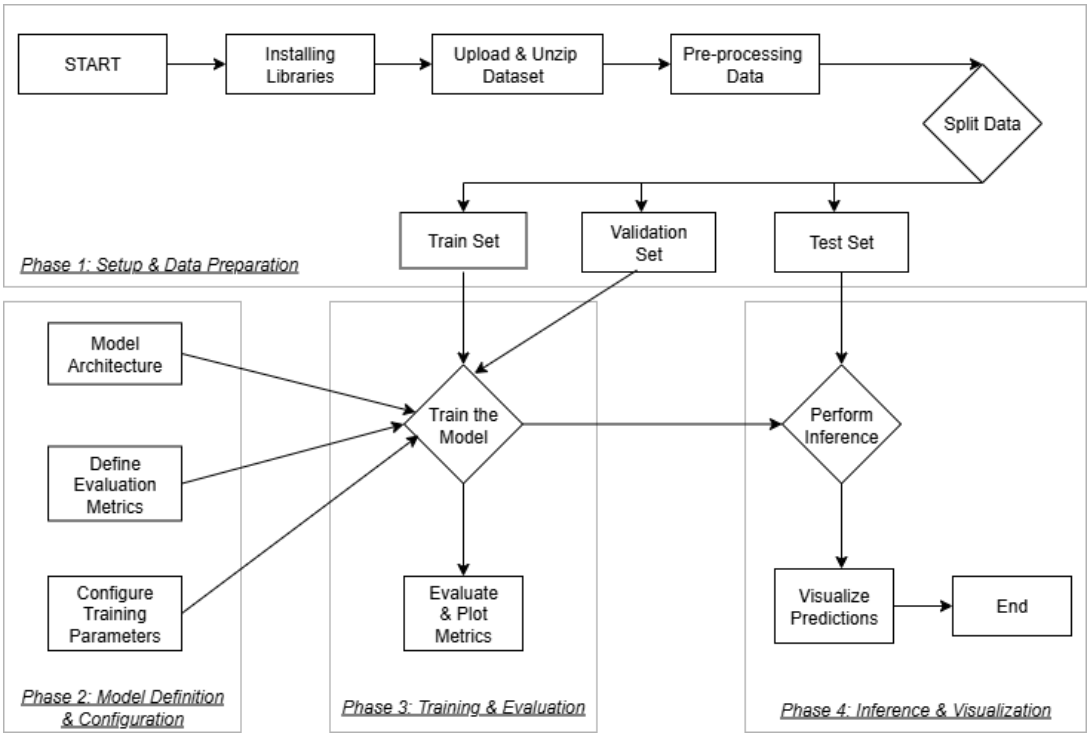

Figure 2: Workflow of this project

Setup & Data preparation ensures a solid base over which the research would be build upon. Step one in this stage would be to import all the required libraries such as Tensorflow, Keras, matplotlib etc. These will set up the environment and will hold the necessary tools for model building, visualisation and data handling. Once data is uploaded, it will be subject to preprocessing which involve resizing, splitting & normalization. After preprocessing, the dataset will be split into three subsets – training, validation and test. These subsets will be used to train, tune hyperparameters and evaluate the performance of the models.

Once the environment is setup, suitable architectures of CNN-based and Transformer-based will be defined. Alongside this, metrics required for the evaluation of the performance of these models will be defined. Next training parameters such as loss function, epochs, learning rate, optimizer type and batch size will be configured. These set models are then subject to training dataset by minimising loss function, and the validation dataset is used to monitor the general performance of the model. Upon completion of the training, performance plots will be utilised to understand and diagnose issues. Finally, the tuned model is then used to make predictions on the test set and the results are visualised to gain an insight of it's performance.

The entire flow is implemented using Python hosted on GPU-accelerated platforms such as Google Colab, which significantly reduce the computational time required by the resource intensive models selected for the purpose of this research. This pipeline ensures

clarity and set a designed path for reproducibility of the performance of models.

5 Implementation

To implement this study, python is utilised as the programming language for it provides a plethora of libraries for every task. The jupyter notebook in Google Colab environment is utlised to execute the complete process as it significantly reduced the time required to run the models. Libraries such as Keras and Pytorch were utilised to prepare the data into acceptable and consistent formats for each models. All models were trained for exactly 20 Epochs with a consistent batch size of 8 as shown in Figure 3.

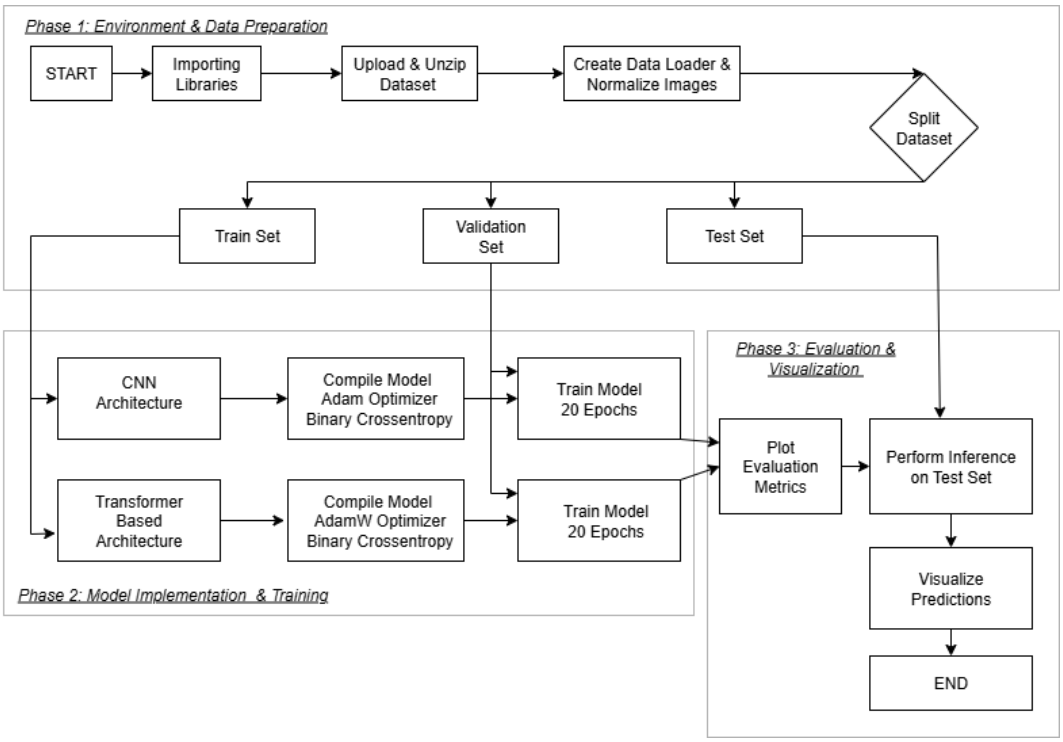

Figure 3: Framework of implemented study

5.1 Deep Learning Models

The CNN-based architectures follows the classic two-part encoder-decoder structure. Encoder being the contracting path which gradually reduces the spatial resolution of input to capture the semantic features and Decoder being the expansive path which stitches back the features to reconstruct the complete picture with segmentation.

5.1.1 UNet

UNet maintains a symmetric 5 layer encoder-decoder structure with "skip connections", which are essentially connection between encoder and decoder residing at same levels. The encoder follows a standard design and gradually reduces the spatial resolution using MaxPooling2D based on abstract features. The decoder uses Conv2DTranspose to stich

up the layers. Skip connection between encoder and decoder allows this architecture to generate highly precise boundaries between classes.

5.1.2 SEGNet

SegNet follows a lightweight encoder-decoder structure with 4 levels. The main difference between SegNet and Unet is in the decoder design where instead of using skip connections, SegNet utilised stored max-pooling indices to reconstruct. By omitting skip layer and learnable upsampling, this architecture significantly reduces the number of trainable parameters while maintaining the effectiveness in reconstructing the image with segmentation.

5.1.3 PSPNet

PSPNet consists of a shallow CNN encoder followed by Pyramid Pooling Module which performs average pooling at multiple grid scales capturing information from different portion of the image. Each pooled feature map is reduced in channel size through a 1x1 convolution then upsampled to match the original spatial resolution before being concatenated with he encoders final feature map. This context rich representation is then passed through the decoder consisting of upsampling and convolutional layers. The ability to feed context aware feature map into the final prediction map of the architecture, the model maintains a very high level of pixel-level accuracy while limiting the number of trainable parameter.

5.2 Transformer-based Models

The introduction of transformer-based architectures have shifted the paradigm in image segmentation tasks. The addition of self-attention mechanism gives these architectures the capability to evaluate the relevance of each element in the sequence of processing. This Makes these architectures extremely effective at capturing global context which the conventional models can't comprehend.

5.2.1 SEGFormer

SEGFormer uses a hierarchical transformer encoder which similar to feature pyramids in CNN, produces multi-scale features at resolution of 1/4, 1/8, 1/16 and 1/32 of input size. Instead of using rigid positional encodings like Vision Transformer, SegFormer employs overlapping patch embeddings and convolution-enhanced feed-forward layers to implicitly learn spatial relationships. A simple MLP-based decoder then unifies and upsamples these features to generate the final segmentation map. This study uses the 'nvidia/segformer-b0-finetuned-ade-512-512' checkpoint from HuggingFace, fine-tuned for a 3-class RGB segmentation task with 'ignore_mismatched_sizes=True' to adapt the classification head. In order to implement SEGFormer, the required dependencies were imported. Preprocessing was carried out before splitting the dataset into train, validate and test. Pre-trained SEGFormer model is then imported. Adam optimiser was utilised for this architecture with learning rate of 1e-4. Loss function utilised for this model was cross entropy.

5.2.2 UNETr

UNETR (U-Net with Transformers) is a hybrid semantic segmentation architecture that merges the global context modeling of Vision Transformers with the spatial precision of U-Net decoding. It uses a Mix Transformer (MiT-B2) encoder pretrained on ImageNet to extract long-range features by processing non-overlapping patches through self-attention layers. The U-shaped decoder reconstructs the segmentation map using a series of upsampling blocks and skip connections. This brings together the best of both worlds, the semantic depth of transformer models and boundary accuracy from UNet architecture. This model is configured for 3 channel input and 3 channel output. In order to implement UNETr, the required dependencies were imported. Preprocessing was carried out before splitting the dataset into train, validate and test. A UNET models is then initialized with 'mit-b2' encoder pre-trained on 'imagenet'. AdamW optimiser was utilised for this architecture with learning rate of 1e-4. Loss function utilised for this model was Cross Entropy. Table 1 and Table 2 provides the architectural summary and compiling summary of all models implemented in this research.

Table 1: Architectural summary of all models

Model	**Backbone/Core Principle**	**Framework**	**Trainable Parameters**	**Avg. Epoch Time (s)**
U-Net	Symmetric Encoder-Decoder with Skip Connections	TensorFlow/Keras	31M	˜21
SegNet	Encoder-Decoder with Pooling Indices (Simplified)	TensorFlow/Keras	2.7M	˜12
PSPNet	Pyramid Pooling Module for Context Aggregation	TensorFlow/Keras	1.9M	˜18
SegFormer	Hierarchical Transformer Encoder + MLP Decoder	PyTorch	˜3.7M (B0)	˜25
UNETR	Transformer Encoder (MiT-B2) + U-Net-style Decoder	PyTorch	˜8.7M (MiT-B2)	˜28

Table 2: Compiling Summary of all models

Model	**Activation**	**Optimiser**	**Learning Rate**	**Loss Function**
U-Net	Relu	Adam	1e-4	Cross Entropy
SegNet	Relu	Adam	1e-4	Cross Entropy
PSPNet	Relu	Adam	1e-4	Cross Entropy
SegFormer	Relu	Adam	1e-4	Cross Entropy
UNETR	Relu	AdamW	1e-4	Cross Entropy

6 Evaluation

This section provides a detail evaluation of performance for each model implemented which is then critically assessed in relation to the research objectives using relevant metrics such as Pixel Accuracy, mean Intersection over Union (mIoU), and Dice Score.

6.1 UNet

UNet model performance remains suboptimal as shown in Figure 4. Its pixel accuracy remains significantly low, indicating class imbalance. Moderate improvement in training mIoU and fluctualtion in validation mIoU indicates instability in class-wise overlap. Dice Score on the other hand improves gradually suggesting that the model finds learning the classes in foreground better with time. Figure 5 presents side-by-side comparison of the actual image and it's segmentation map created by UNet model.

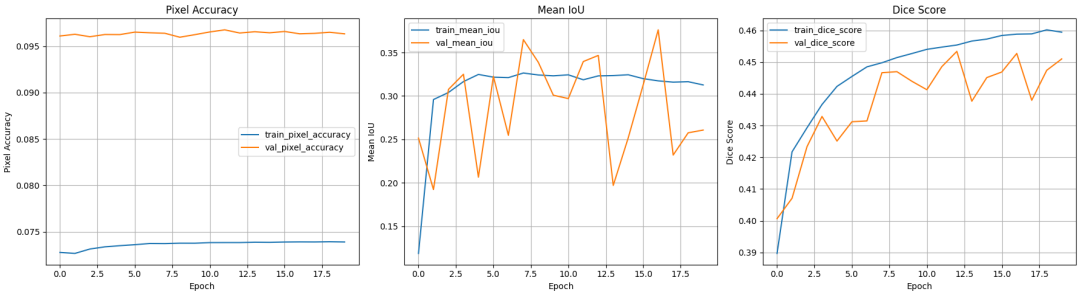

Figure 4: Evaluation results for UNet Model

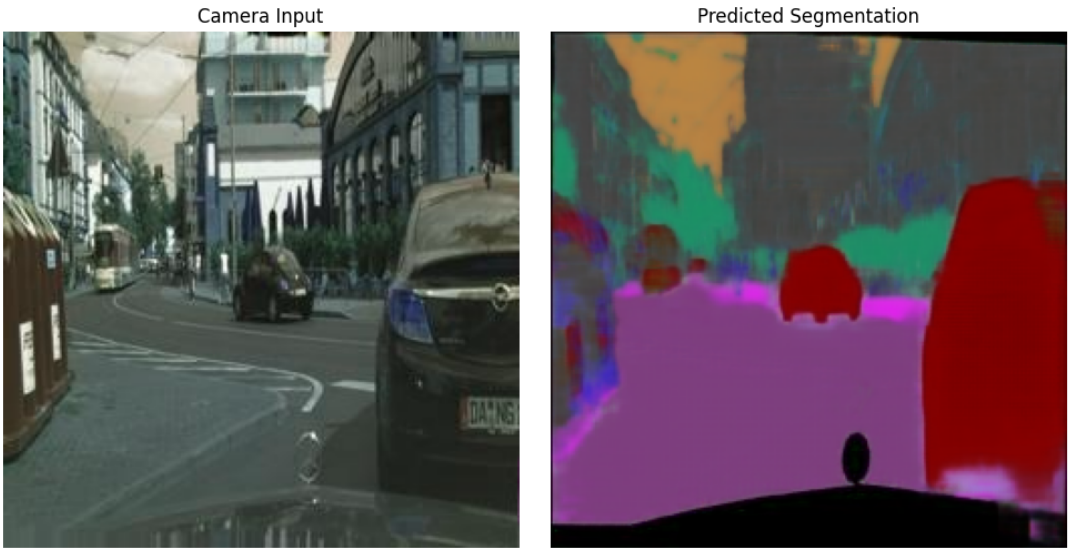

Figure 5: Real world Image vs. Segmentation Map of UNet Model

6.2 SegNet

The performance curve of the Segnet model shows upward growth across all metrics, though the results remain underwhelming. The pixel accuracy rises to 86% in training with validation closely trailing behind indicates a good generalisation of dataset. As depicted in Figure 6 Dice-score increases gradually and consistently suggesting that the model learns to differentiate between foreground and background efficiently. While this model is efficient, lightweight and require the lowest computational time as shown in Table, its encoder-only pooling for upsampling restricts its capability to capture detailed features, making it suboptimal for complex tasks. Figure 7 presents side-by-side comparison of the actual image and it's segmentation map created by SegNet model.

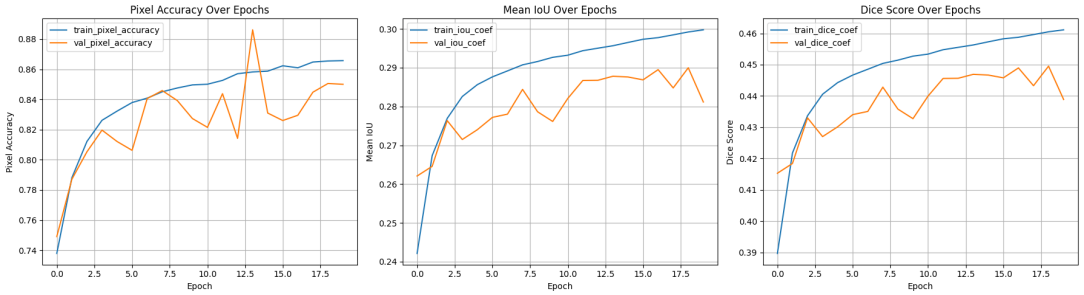

Figure 6: Evaluation results for SegNet Model

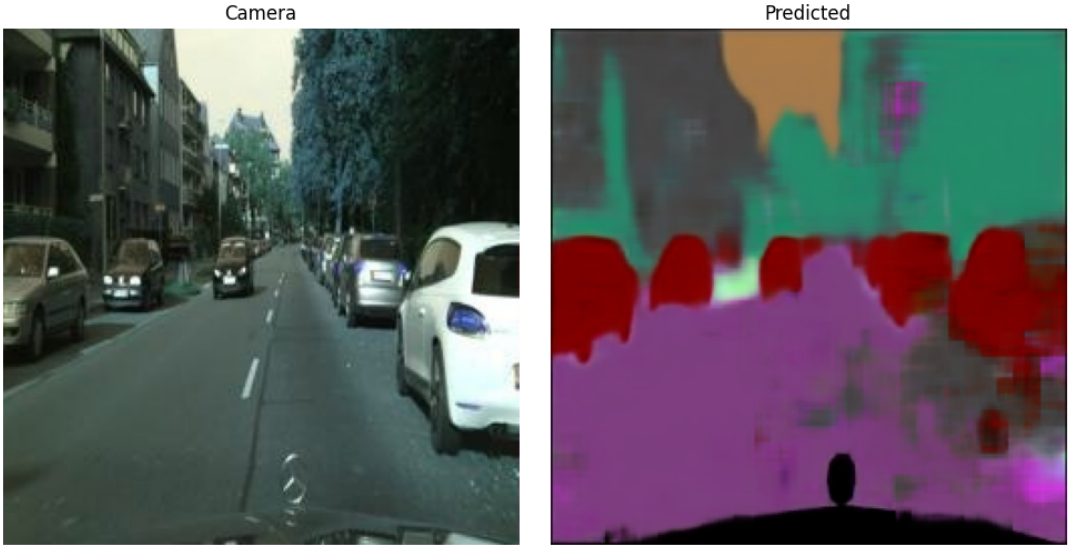

Figure 7: Real world Image vs. Segmentation Map of SegNet Model

6.3 PSPNet

While the training results for PSPNet model improves over performance of previous models with Pixel accuracy reaching to 86%, mIoU to 60% and Dice Score to 49%, the results for validation fluctuate significantly, indicating poor generalisation in dataset. The

results shown in Figure 8 clearly imply more diverse data to stabilize outcome from it's multi-scale context aggregation through pyramid pooling. Figure 9 presents side-by-side comparison of the actual image and it's segmentation map created by PSPNet model.

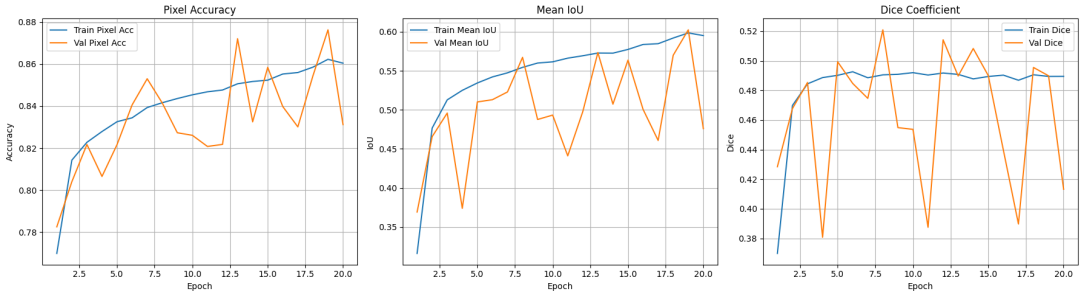

Figure 8: Evaluation results for PSPNet Model

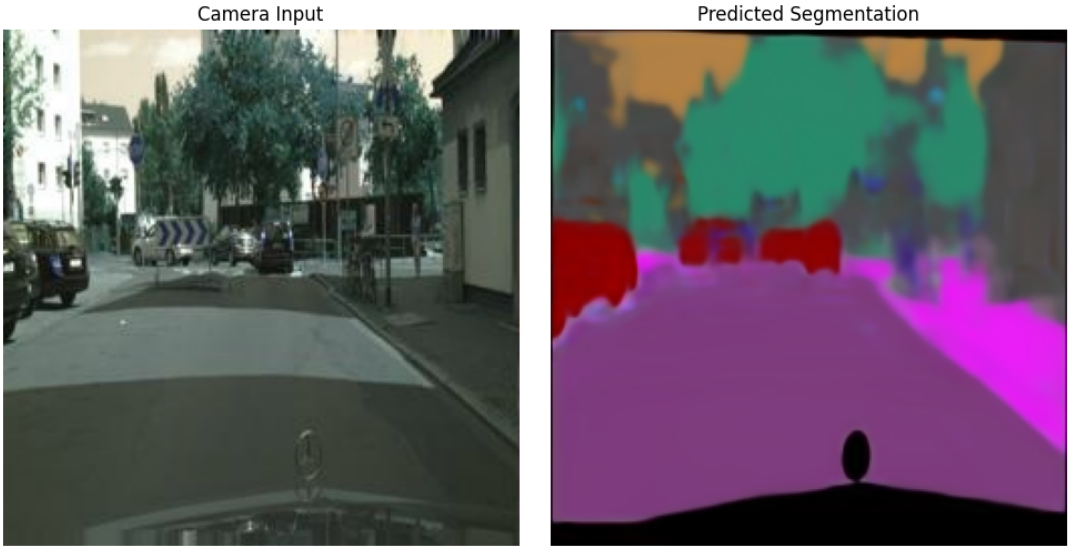

Figure 9: Real world Image vs. Segmentation Map of PSPNet Model

6.4 SegFormer

Segformer model provided the best result among all the models implemented in this study. It displayed excellent and consistent performance across all metrics as shown in Figure 10. The pixel accuracy increased steadily and smoothly to 87%, mIoU to 75% and Dice Score to 86%. Despite minor fluctuations, the validation curve closely follow the training curve, indicating strong generalisation and minimal overfitting. Figure 11 presents side-by-side comparison of the actual image and it's segmentation map created by SegFormer model.

6.5 UNETr

The UNETr model shows unstable learning dynamics while validation performance remains high. There are frequent fluctuation across all metrics suggesting insufficient train-

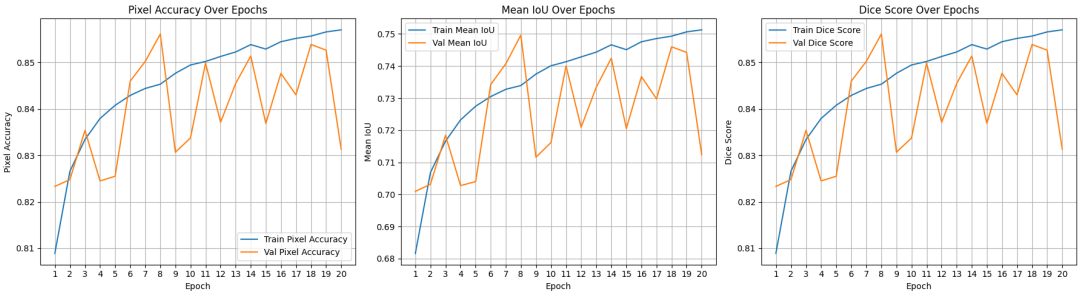

Figure 10: Evaluation results for SegFormer Model

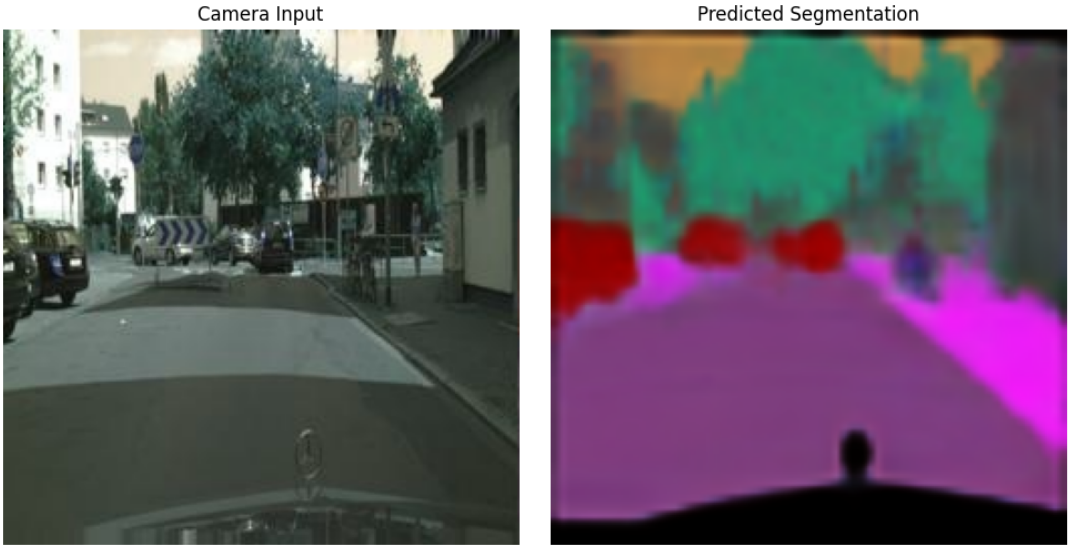

Figure 11: Real world Image vs. Segmentation Map of SegFormer Model

ing or unstable convergence. The flatness in the training curve as depicted in Figure 12 indicates a need for regularisation and data scaling for training. Figure 13 presents side-by-side comparison of the actual image and it's segmentation map created by UNETr model.

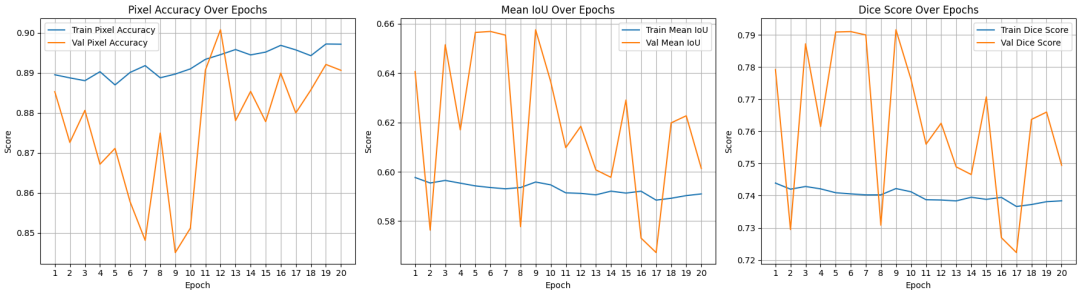

Figure 12: Evaluation results for UNETr Model

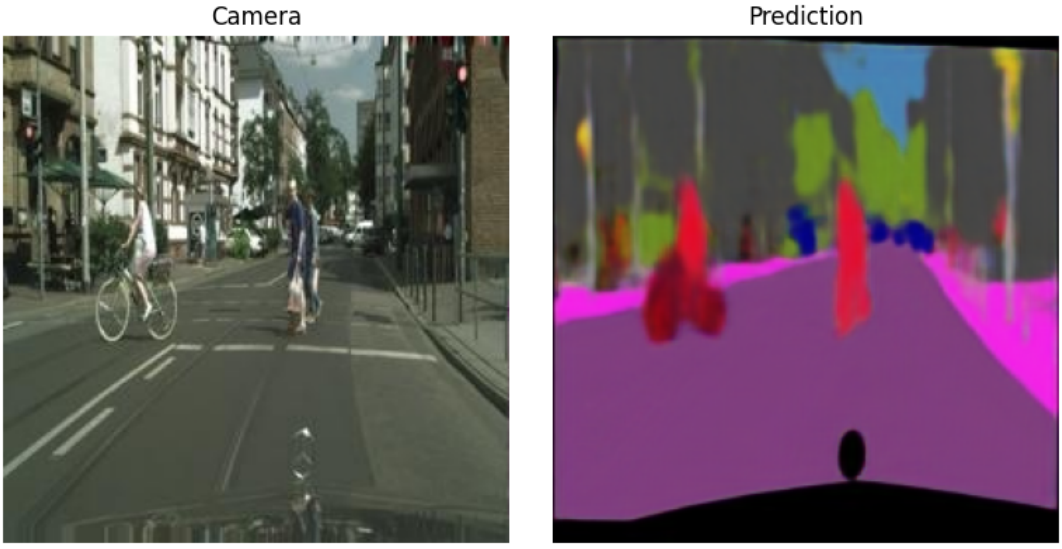

Figure 13: Real world Image vs. Segmentation Map of UNETr Model

7 Conclusion and Future Work

This study was sought for exploring and comparing the performance of conventional CNN-based models and hybrid transformer-based models in the domain of semantic image segmentation for autonomous driving systems. A comprehensive comparison among 5 leading architectures namely UNet, SegNet, PSPNet (CNN-based), and SegFormer, UNETR (Transformer-based) was performed using real world data from Cityscapes Dataset [5].

The experiment resulted in confirming the growing advantage of transformer-based architectures in particularly complex and cluttered urban scenarios. SegFormer consistently

[5]Cityscapes Dataset .

showed better performance among other models, achieving the highest metrics in pixel accuracy (87%), mean Intersection over Union (mIoU at 75%), and Dice Score (86%). This demonstrates its strong ability to capture both global context and fine-grained details with minimal overfitting. This makes it suitable for real-time segmentation in autonomous driving applications. CNN-based models however were easier to train and lightweight in terms of computational resources but showed several limitations. UNet and SegNet on one hand were effective with learning low-level features and offered competent inference speeds, but they struggled with scene complexity and fine-grained segmentation in cluttered environments. PSPNet, was supposed to work on this limitation however it came with its own instability in generalisation most likely due to data diversity constraints. Even though Hybrid transformer-based model UNETR showed promise due of its combination of global attention and U-Net-style skip connections, it under went inconsistent learning curves. This brought a need for deeper regularization strategies and use of a more extensive training data to reach optimal performance. The findings from these experiments validate the hypothesis: CNN-based and hybrid transformer-based architectures offer significant improvement in semantic image segmentation in terms of computational efficiency, accuracy, resilience under complex and diverse real-world driving scenarios and are suitable for integration into production-level autonomous vehicles. The results obtained through this study focus on the fundamental comparison in semantic segmentation model design between accuracy and efficiency. On one hand models like SegFormer offers better performance, but their high computational requirements can impose deployment challenges for resource-constrained environments such as embedded systems in autonomous vehicles. On the other hand, lightweight CNNs like SegNet offer faster inference and lesser computation but fall behind in segmentation precision under complex conditions. It is therefore worth exploring this domain over following areas of the research:

- Model Compression and Optimization: Implementing model compression techniques, like pruning, quantization, and knowledge distillation, mainly for transformer-based models may help reduce their computational load and ensure real-time deployment in embedded automotive systems without compromising accuracy.
- Dataset Diversity and Expansion: The current study uses a limited subset of the Cityscapes dataset. Experimenting with a more diverse and larger datasets including different lighting conditions, weather scenarios, and geographic locations may improve model robustness and generalization, especially for transformer models sensitive to training data variance.
- Semi-Supervised and Unsupervised Learning: The high cost of annotation in segmentation datasets probes the possibility to explore semi-supervised or self-supervised learning approaches for transformer-based architectures, in turn reducing dependency on large and manually annotated datasets.

In conclusion, although transformer-based architectures represent the future of semantic segmentation in autonomous driving, achieving a balance between performance, scalability, and efficiency remains a promising research area.

References

Anwar, H., Indrabayu and Areni, I. S. (2024). A modification of u-net decoder architecture improve performance of object detection for autonomous vehicles, *2024 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, pp. 454–459.

Bavirisetti, D. P., Martinsen, H. R., Kiss, G. H. and Lindseth, F. (2023). A multi-task vision transformer for segmentation and monocular depth estimation for autonomous vehicles, *IEEE Open Journal of Intelligent Transportation Systems* **4**: 909–928.

Chen, B., Gong, C. and Yang, J. (2019). Importance-aware semantic segmentation for autonomous vehicles, *IEEE Transactions on Intelligent Transportation Systems* **20**(1): 137–148.

Cui, H. and Lei, J. (2024). An algorithmic study of transformer-based road scene segmentation in autonomous driving, *World Electric Vehicle Journal* **15**(11).
URL: *https://www.mdpi.com/2032-6653/15/11/516*

Dong, J., Chen, S., Zong, S., Chen, T. and Labi, S. (2021). Image transformer for explainable autonomous driving system, *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pp. 2732–2737.

Fan, J., Gao, B., Ge, Q., Ran, Y., Zhang, J. and Chu, H. (2024). Segtransconv: Transformer and cnn hybrid method for real-time semantic segmentation of autonomous vehicles, *IEEE Transactions on Intelligent Transportation Systems* **25**(2): 1586–1601.

Gad, G. M., Annaby, A. M., Negied, N. K. and Darweesh, M. S. (2020). Real-time lane instance segmentation using segnet and image processing, *2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, pp. 253–258.

Giurgi, D.-V., Josso-Laurain, T., Devanne, M. and Lauffenburger, J.-P. (2022). Real-time road detection implementation of unet architecture for autonomous driving, *2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*, pp. 1–5.

Guo, Z., Perminov, S., Konenkov, M. and Tsetserukou, D. (2024). Hawkdrive: A transformer-driven visual perception system for autonomous driving in night scene, *2024 IEEE Intelligent Vehicles Symposium (IV)*, pp. 2598–2603.

Ha, Q., Watanabe, K., Karasawa, T., Ushiku, Y. and Harada, T. (2017). Mfnet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes, *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5108–5115.

Jebamikyous, H.-H. and Kashef, R. (2021). Deep learning-based semantic segmentation in autonomous driving, *2021 IEEE 23rd Int Conf on High Performance Computing Communications; 7th Int Conf on Data Science Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud Big Data Systems Application (HPCC/DSS/SmartCity/DependSys)*, pp. 1367–1373.

Liu, Z., Guo, S. and Xie, Z. (2023). A detection algorithm based on segformer for unmanned driving, *2023 IEEE International Conference on Image Processing and Computer Applications (ICIPCA)*, pp. 675–680.

Muhammad, K., Hussain, T., Ullah, H., Ser, J. D., Rezaei, M., Kumar, N., Hijji, M., Bellavista, P. and de Albuquerque, V. H. C. (2022). Vision-based semantic segmentation in scene understanding for autonomous driving: Recent achievements, challenges, and outlooks, *IEEE Transactions on Intelligent Transportation Systems* **23**(12): 22694–22715.

Netrakar, M., Upare, D., Tejas, C., Baddi, T. Y. and Benni, R. (2024). A comparative analysis of u-net, pspnet, and fpnet: Deep learning techniques for image segmentation, *2024 IEEE Conference on Engineering Informatics (ICEI)*, pp. 1–10.

Patri, H. V., Priya, M. B., Mothukuri, M. B., Kumar, D. M., Ratna Prabha, K. V. and Gandham, S. R. K. (2024). U-net advancements in semantic segmentation for autonomous vehicles, *2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Vol. 1, pp. 2292–2296.

Pham, T. (2021). Semantic road segmentation using deep learning, *2020 Applying New Technology in Green Buildings (ATiGB)*, pp. 45–48.

Wang, H., Chen, Y., Cai, Y., Chen, L., Li, Y., Sotelo, M. A. and Li, Z. (2022). Sfnet-n: An improved sfnet algorithm for semantic segmentation of low-light autonomous driving road scenes, *IEEE Transactions on Intelligent Transportation Systems* **23**(11): 21405–21417.

Zhou, W., Berrio, J. S., Worrall, S. and Nebot, E. (2020). Automated evaluation of semantic segmentation robustness for autonomous driving, *IEEE Transactions on Intelligent Transportation Systems* **21**(5): 1951–1963.
ΩÖzen et al.

Özen, S., Kaya, U., Semiz, A. and Çelebi, A. T. (2023). Multi-lane detection system based on segmentation model for autonomous vehicles, *2023 10th International Conference on Electrical and Electronics Engineering (ICEEE)*, pp. 18–23.