MULTI-TASK VISION TRANSFORMER

A PROJECT REPORT

Submitted by

MOHAMMED ABDULLAH

(2023176026)

A report for the phase-I of the project
submitted to the Faculty of

INFORMATION AND COMMUNICATION ENGINEERING

in partial fulfillment
for the award of the degree
of

MASTER OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

SPECIALIZATION IN AI & DS

DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY
COLLEGE OF ENGINEERING, GUINDY
ANNA UNIVERSITY
CHENNAI 600 025

JANUARY 2025

ANNA UNIVERSITY
CHENNAI - 600 025
BONA FIDE CERTIFICATE

Certified that this project report titled MULTI-TASK VISION TRANSFORMER is the bona fide work of MOHAMMED ABDULLAH (2023176026) who carried out project work under my supervision. Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on this or any other candidate.

PLACE:	**DR. G GEETHA**
DATE:	**ASSOCIATE PROFESSOR**
	PROJECT GUIDE
	DEPARTMENT OF IST, CEG
	ANNA UNIVERSITY
	CHENNAI 600025

COUNTERSIGNED

DR. S. SWAMYNATHAN
HEAD OF THE DEPARTMENT
DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY
COLLEGE OF ENGINEERING, GUINDY
ANNA UNIVERSITY
CHENNAI 600025

ABSTRACT

Segmentation and depth estimation for autonomous navigation in adverse environmental conditions such as fog, mist, and rain are challenging due to visibility degradation and loss of fine details. Typical state-of-the-art computer vision models for these tasks typically have high computational complexity and parameter count, resulting in slow runtimes.

This project proposes a multi-task Vision Transformer architecture (ViT) with shared layers to simultaneously learn the depth and segmentation parameters, thereby optimizing performance in low-visibility scenarios. The ViT backbone extracts features that are processed concurrently for segmentation and depth estimation. Multi-Scale Feature Fusion enhances depth prediction, while a Depth-Guided Segmentation Decoder integrates depth information into segmentation process, while refining object boundaries and improving spatial coherence.

Experimental results on the Cityscapes dataset (Phase 1) indicate that the proposed approach achieves better metrics for both segmentation and monocular depth estimation compared to single-task methods, with marginal performance gains. Phase 2 will focus on achieving superior segmentation accuracy and depth estimation in complex scenes with varying haze and blur levels.

ஆய்வுச் சுருக்கம்

மூடுபனி மற்றும் மழை போன்ற பாதகமான சுற்றுச்சூழல் நிலைமைகளில் தன்னியக்க வழிசெலுத்தலுக்கான பிரிவு மற்றும் ஆழமான மதிப்பீடு ஆகியவை தெரிவுநிலைச் சிதைவு மற்றும் நுண்ணிய விவரங்களின் இழப்பு காரணமாக சவாலாக உள்ளன. இந்த பணிகளுக்கான வழக்கமான அதிநவீன கணினி பார்வை மாதிரிகள் பொதுவாக அதிக கணக்கீட்டு சிக்கலான தன்மை மற்றும் அளவுரு எண்ணிக்கையைக் கொண்டுள்ளன, இதன் விளைவாக மெதுவான இயக்க நேரங்கள் ஏற்படும்.

ஒரே நேரத்தில் ஆழம் மற்றும் பிரிவு அளவுருக்களைக் கற்றுக்கொள்வதற்காக பகிரப்பட்ட அடுக்குகளுடன் கூடிய பல-பணி விஷன் டிரான்ஸ்ஃபார்மர் (ViT) கட்டமைப்பை இந்த திட்டம் முன்மொழிகிறது, இதன் மூலம் குறைந்த தெரிவுநிலை சூழ்நிலைகளில் செயல்திறனை மேம்படுத்துகிறது. பிரிவு மற்றும் ஆழம் மதிப்பீட்டிற்காக ஒரே நேரத்தில் செயலாக்கப்படும் அம்சங்களை ViT முதுகெலும்பு பிரித்தெடுக்கிறது. மல்டி-ஸ்கேல் ஃபீச்சர் ஃப்யூஷன் ஆழமான முன்கணிப்பை மேம்படுத்துகிறது, அதே சமயம் ஆழமான-வழிகாட்டப்பட்ட பிரிவு குறிவிலக்கியானது ஆழமான தகவலைப் பிரிப்புச் செயல்பாட்டில் ஒருங்கிணைக்கிறது, அதே நேரத்தில் பொருள் எல்லைகளைச் செம்மைப்படுத்துகிறது மற்றும் இடஞ்சார்ந்த ஒத்திசைவை மேம்படுத்துகிறது.

சிட்டிஸ்கேப்ஸ் தரவுத்தொகுப்பில் (கட்டம் 1) சோதனை முடிவுகள், முன்மொழியப்பட்ட அணுகுமுறை சிறந்த அளவுரு திறன் மற்றும் ஒற்றை-பணி முறைகளுடன் ஒப்பிடும்போது வேகமான அனுமான நேரத்தை அடைகிறது என்பதைக் குறிக்கிறது. கட்டம் 2, மாறுபட்ட மூடுபனி மற்றும் மங்கலான நிலைகளுடன் சிக்கலான காட்சிகளில் சிறந்த பிரிவின் துல்லியம் மற்றும் ஆழமான மதிப்பீட்டை அடைவதில் கவனம் செலுத்தும்.

ACKNOWLEDGEMENT

It is my privilege to express my deepest sense of gratitude and sincere thanks to Dr. G. Geetha, Associate Professor, Project Guide, Department of Information Science and Technology, College of Engineering, Guindy, Anna University, for her constant supervision, encouragement, and support in my project work. I greatly appreciate the constructive advice and motivation that was given to help me advance my project in the right direction.

I would also wish to express my deepest sense of gratitude to the Members of the Project Review Committee: Dr. G. Geetha, Associate Professor, Dr. S. Sridhar, Professor, Dr. D. Narashiman, Teaching Fellow, Department of Information Science and Technology, Anna University, Chennai, for their guidance and useful suggestions that were beneficial in helping me improve my project.

I am grateful to Dr. S.Swamynathan, Professor and Head, Department of Information Science and Technology, College of Engineering Guindy, Anna University, for providing me with the opportunity and necessary resources to do this project.

MOHAMMED ABDULLAH

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ViT	Vision Tranformer
LSTM	Long Term Short Term Memory
MLT	Multi-Task Learning
MLVT	Multi-Task Vision Transformer
mIoU	Mean Intersection over Union
mAcc	Mean Accuracy
aAcc	Accuracy
AbsRel	Absolute Relative Difference
SqRel	Squared Relative Difference
RMSE	Root Mean Squared Error
RMSElog	Root Mean Squared Logarithmic Error
AP	Average Precision
δ	Threshold Accuracy

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

With the rise of autonomous vehicles like Tesla and Waymo, there has been a surge in competitive research focused on segmentation and depth estimation tasks. These tasks are extensively studied and implemented using neural network.

Autonomous vehicles operate in dynamic and complex environments where understanding the surroundings is critical for safe navigation. Semantic segmentation enables vehicle to classify every pixel in an image into meaningful categories such as roads, pedestrians, vehicles, and obstacles. This fine-grained understanding allows a vehicle to make informed decisions, such as lane keeping, obstacle avoidance, and pedestrian safety. For example, detecting crosswalks or differentiating between road surfaces and curbs is essential for precise path planning.

Depth estimation is crucial for perceiving three-dimensional structure of the environment from 2D images, enabling the vehicle to judge distances to various objects. Monocular depth estimation, which derives depth information from a single camera image, is particularly valuable as it reduces the reliance on expensive hardware like LiDAR. Accurate depth information is critical for tasks such as collision avoidance, maintaining safe following distances, and navigating around static and dynamic obstacles. Monocular depth estimation is cost-effective and complements the perception stack by leveraging widely available camera sensors.

The integration of segmentation and depth estimation enhances the robustness of perception systems. While segmentation provides object-level understanding, depth estimation situates these objects in a 3D space, enabling the vehicle to react appropriately in real-time scenarios. This synergy is essential for achieving full autonomy in challenging conditions such as low visibility, cluttered urban areas, or high-speed highways.

1.2 PROBLEM STATEMENT

Existing segmentation and monocular depth estimation have high number of parameters and hence require more memory.

Current models often lack adaptability to perform both tasks seamlessly from a single input image, leading to increased computational demand, and hence high inference time.

In addition, the challenge is magnified in low visibility conditions like fog and rain, where blurry automobile images further degrade segmentation and depth estimation performance.

Thus, these limitations emphasize the need for a more robust, efficient, and adaptable ViT based solution to handle complex multitask scenarios in unfavorable conditions.

1.3 OBJECTIVE

The objective of this project is to develop a robust Vision Transformer (ViT) based model capable of simultaneously performing semantic segmentation and monocular depth estimation, specifically designed to address the challenges of low visibility environments such as fog and rain.

1.4 SCOPE OF THE PROJECT

The scope of the project includes training on datasets with images of adverse environmental conditions such as fog, haze, and rain using both real-world and synthetic datasets like Cityscapes. This will help assess the limitations of existing models and test the robustness of the proposed Vision Transformer (ViT) in handling low-visibility automobile images. Additionally, advanced preprocessing techniques will be developed, tailored specifically to enhance image clarity in these low-visibility environments. These haze-aware methods will mitigate the effects of fog and blur, improving the quality of input images for more efficient processing by the ViT model.

Furthermore, the scope extends to enhancement of Vision Transformer architecture by fine-tuning the backbone to increase robustness against haze and blur. This will ensure that the model maintains high accuracy in both segmentation and depth estimation tasks, even when the input images suffer from environmental degradation. To provide more comprehensive understanding of the scene, the project will also integrate data from multiple sensors and modalities, such as RGB images and LiDAR. This multi-modal approach will enable model to accurately interpret complex environments under poor visibility.

The project will place a strong focus on improving the simultaneous capabilities of semantic segmentation and monocular depth estimation, ensuring precise recognition and localization of vehicles and objects even in adverse conditions. Lastly, a key area of application for this model will be autonomous driving, where the ability to perform reliably in poor weather conditions is critical. The project aims to deliver a solution that can handle real-time data efficiently, ensuring both the safety and effectiveness in the complex driving environments.

1.5 SEGMENTATION AND MONOCULAR DEPTH ESTIMATION

Segmentation can be defined as the problem of per-pixel classification in learning-based approaches.[1] If this is done based on classes, then it is Semantic Segmentation, if it each object is given an unique identifier, then it is instance segmentation. Effective segmentation often faces challenges like variations in lighting, occlusions, and cluttered backgrounds. In real-world scenarios, the presence of adverse weather conditions such as fog or rain can further complicate accurate segmentation, necessitating advanced models and robust preprocessing techniques to achieve reliable results. Segmentation serves as critical building block for more advanced perception tasks, enabling machines to understand and interact with their surroundings intelligently.

Monocular depth is defined as the process of estimating the distance of objects in a scene or from a single image, or the camera viewpoint. The task is inherently ill-posed due to the lack of direct geometric information in a single image. Factors such as lighting variations, textureless surfaces, occlusions, and adverse weather conditions can degrade the accuracy of depth predictions. To address these challenges, modern methods leverage deep learning techniques, using large-scale annotated datasets and neural networks to infer depth from visual cues effectively. Monocular depth estimation bridges the gap between 2D image data and 3D scene understanding, playing a pivotal role in enabling intelligent systems to perceive and navigate their environments.

1.6 NEED FOR SEGMENTATION AND MONOCULAR DEPTH

For automobile navigation, the segmentation and monocular depth estimation provide information of the objects around the vehicle and relative distance between them. This is particularly useful, especially in the adverse environments where visibility is low, On other hand segmentation and monocular depth estimation is widely utilized in the applications like mixed reality, robot navigation, and autonomous driving, due to its cost-effectiveness. Traditional methods, such as vanishing point techniques (using geometric structures) and focus/defocus methods (analyzing image clarity), struggle with complex outdoor environments. The deep learning-based approaches have advanced the field by leveraging deep neural networks (DNNs) to establish regression relationships between images and depth [2]. However, real-time performance challenges arise due to the high computational complexity of these models, which require significant parameters to infer depth with precision.

CHAPTER 2

LITERATURE SURVEY

2.1 OVERVIEW

This literature survey explores the advancements in the segmentation and the depth estimation, emphasizing the methodologies that address adverse environmental conditions using the Vision Transformers and related architectures. The accuracy and robustness of these tasks are significantly influenced by the environmental conditions like haze, fog, and rain, which degrade image quality by reducing visibility, contrast, and clarity. So various models which are trained on datasets which simulate such environment are also explored.

2.2 MULTI-TASK MODELS

The study of Multi-Task Learning (MTL) is emerging in recent times. These models utilize shared information from multiple related tasks to enhance the generalization performance across all tasks [3]. For instance, a Multi-Task Vision Transformer (MTVT) had been proposed for a joint segmentation and monocular depth estimation task, mainly for autonomous vehicle applications [4]. Their model had employed a Hierarchical Transformer Encoder for feature extraction, along with task-specific decoders to address the distinct requirements of segmentation and depth estimation tasks. The design had aimed to enhance the accuracy of both tasks simultaneously. A shared vision transformer encoder had been utilized to process the input, while task-specific decoders had been employed to generate outputs for segmentation and depth estimation. The model had achieved a competitive mean Intersection over Union (mIoU) of 76.53 and a Root Mean Square Error (RMSE) of 6.289 for monocular depth estimation.

During multi-task training, their performance metrics had been slightly shifted, with an RMSE of 6.317 and an mIoU of 75.95. This approach had demonstrated the significant advancements in accuracy and the efficiency, thereby highlighting the effectiveness of shared encoders and task-specific decoders in multi-task learning scenarios for autonomous systems. Another study had attempted to implement the semantic segmentation, the instance segmentation and panoptic segmentation in a single model by using a similar shared encoder.[5].

2.3 METHODS OF SEGMENTATION

Segmentation is a computationally intensive task, and there are many research efforts which focus on developing lightweight segmentation methods to enhance efficiency and scalability. Recent advancements have demonstrated the potential of Transformer-based architectures for this task. One such model is SegFormer [6], which is a state-of-the-art framework. It had introduced a novel approach by integrating Transformers with a lightweight multilayer perceptron (MLP) decoders, striking a balance between simplicity, efficiency, and performance. It had featured a novel hierarchically structured Transformer encoder that produced multiscale features without relying much on positional encoding, which often required interpolation and led to performance degradation when the testing resolution differed from the training resolution. Another variant of Segformer [7] tried a different approach.Instead of using complex decoders, it employed a lightweight MLP decoder that effectively aggregated features from different layers, combining local and global attention to create robust representations. In the findings on the Cityscapes dataset, It had demonstrated remarkable performance, achieving 84.0% mIoU on the validation set with its largest model, SegFormer-B5, and showcasing excellent zero-shot robustness on Cityscapes. These results highlighted SegFormer's ability to deliver both high accuracy and efficiency, making it a standout framework in the segmentation. Another study had presented a similar architecture but with reduced number of tokens [8] to avoid the computational complexity for attention calculation in vision transformer encoder. Another study proposed PolyTransform, which introduced deep architecture integrating strengths of conventional segmentation methods and modern polygon-based approaches. It also employed segmentation network to generate instance masks, which were converted into polygons as the initialization, followed by a deforming network that refined these polygons to align with object boundaries. When evaluated on the Cityscapes dataset and its own novel dataset, PolyTransform demonstrated a precise, geometry-preserving instance segmentation, surpassing the backbone model with a validation mIoU of 46.6 when pretrained with SegFormer [9].

2.4 METHODS OF MONOCULAR DEPTH ESTIMATION

Monocular depth estimation superseded the stereo depth estimation when neural networks were used for such tasks. For instance, a study proposed Global- Local Path Networks (GLPN) that utilized multi-scale feature fusion to enhance monocular depth estimation, with a particular focus on vertical-depth predictions. The vertical cut depth was used to augment images such that the final input combined the original image and a depth mask. This approach had significantly improved the depth accuracy, particularly in scenarios where traditional depth estimation methods were less effective. However, the model required further optimization to achieve faster inference speeds, particularly for real-time applications in autonomous driving. The approach was evaluated on the NYU Depth V2 dataset, achieving threshold accuracies of 0.915, 0.988, and 0.997 for 1.25, 1.25^2, and 1.25^3 respectively, with an absolute relative error of 0.098 and an RMSE of 0.344 [10]. Another study introduced a scale-aware, self-supervised method integrating visual-inertial data for depth estimation and odometry. This approach addressed the scale ambiguity issue which is inherent in self-supervised monocular setups by utilizing the inertial measurement unit values for scale information and introducing a pre-integration loss function that compared predicted and IMU-integrated ego-motion. The method involved two networks [11]: a depth network with a U-Net structure using ResNet18 as the encoder and an odometry network comprising visual and inertial encoders, feature fusion, and decoders. The visual encoder processed consecutive images, while the inertial encoder employed bidirectional LSTMs for temporal IMU data. Layer-normalized feature fusion combined visual and inertial features, which were decoded into relative pose, gravity direction, and IMU bias. This approach achieved metric-scaled depth and pose estimations, surpassing state-of-the-art methods on the KITTI dataset with an RMSE of 5.435, and demonstrated robustness in additional indoor driving experiments. On the other hand, there were studies which focused on using light weight models. One such model is MBUDepthNet, which had focused on the unsupervised learning for real-time monocular depth estimation in outdoor scenes. This approach had combined the multi-branch architectures with lightweight designs to achieve similar RMSE scores suitable for real-time applications [12]. However, the model had required further optimization for the complex outdoor environments, especially under the challenging lighting and weather conditions. ResNet-18 was used as the backbone for the pose estimation network, taking advantage of its effectiveness. They had proposed MDE-Lite Module, which was inspired by MobileNetV2's inverted residuals, enabled efficient feature extraction. The depth estimation network consisted of an encoder employing MDE-Lite Module for multi-scale feature extraction and a decoder using deconvolution layers for upsampling. Skip connections were included to integrate encoder and decoder features at corresponding levels. This architecture enabled fine-to-coarse feature extraction and precise depth map generation, achieving threshold accuracies of 0.853, 0.949, and 0.977 for δ_1,δ_2,δ_3 of 1.25, 1.25^2, and 1.25^3 respectively, with an RMSE of 5.144. Another study tried to achieve this with much lower ground truth by using synthetic data [13]. Depth estimation task was also implemented using a segmentation prior model to enhance the segmentation [14].

2.5 MULTI-MODAL DEPTH COMPLETION WITH ATTENTION MECHANISMS

Depth estimation is a challenging task in computer vision,particularly for autonomous systems and robotics, where accurate 3D understanding of the environment is essential. One of the primary challenges in depth completion is generating dense depth maps from sparse depth information, often combined with RGB images. The SemAttNet model addresses this challenge by integrating multi-modal data and attention mechanisms to enhance depth completion, like in noisy and low-light environments. The SemAttNet model had employed attention mechanisms to enhance semantic-guided depth completion, generating dense depth maps and addressing challenges in low-light and noisy environments [15]. Their approach had utilized a three-branch backbone to recover dense depth maps from sparse maps and RGB images. The architecture had included a color-guided branch for processing sparse depth maps and RGB images to capture object boundaries, a semantic-guided branch combining outputs from the color-guided branch with semantic images and sparse depth maps to estimate semantic depth, and a depth-guided branch for synthesizing sparse, color, and semantic depths to produce the final dense depth map. These outputs had been adaptively fused using confidence-weighted fusion to ensure robust results. To refine feature maps, their model had integrated a Semantic-Aware Multi-Modal Attention-Based Fusion Block (SAMMAFB), which fused RGB, semantic, and depth features using channel-wise and spatial-wise attention mechanisms to emphasize critical modalities while suppressing redundant information. For further refinement, CSPN++ with Atrous convolutions had been employed. On the KITTI depth completion benchmark, their method had achieved an RMSE of 2.03, while demonstrating the effectiveness of leveraging multi-modal data for overcoming the limitations of earlier approaches. Another framework had been designed to address adverse weather conditions by integrating image quality enhancement, weather classification, and object detection into a unified system

[16]. Their method had employed Super-Resolution Generative Adversarial Networks (SRGAN) and a modified YOLOv5 to enhance detection capabilities, particularly in scenarios like sandy weather, which are often underrepresented in autonomous driving research. The model had been structured into two key components: a Quality Block that had used the BRISQUE method to assess and enhance low-quality images via SRGAN, and a Classify and Detect Block that had performed weather classification and object detection on enhanced images using YOLOv5. Moreover, by using the augmented DAWN dataset, [17] which had been expanded from 1,027 to 2,046 images through data augmentation to capture severe weather types, the model had robustly detected six object classes (car, cyclist, pedestrian, motorcycle, bus, truck) and achieved a mean average precision (mAP) of 74.6%. By simultaneously addressing image enhancement and object detection, the framework had demonstrated significant performance improvements for autonomous vehicles in adverse conditions.

2.6 SUMMARY OF LITERATURE SURVEY

2.6.1 Depth Estimation Methods

Early monocular depth estimation methods heavily relied on the use of Convolutional Neural Networks (CNNs), such as ResNet and EfficientNet. These architectures effectively process the local pixel relationships through the convolutions to extract the meaningful features present in it, while achieving state-of-the-art performance in tasks such as the image classification and object detection [18]. While CNNs perform well for depth estimation, their focus on local features limits their ability to capture global context, which is critical for more accurate depth predictions.

2.6.2 Segmentation Approaches

For the segmentation tasks, transformers have emerged as a strong alternative. They were originally developed for natural language processing. Models like the Vision Transformer (ViT) [19] and Swin Transformer leverage multi-self-attention mechanisms to extract the global features, which are the complemented by MLPs for local feature representation. These architectures have demonstrated superior performance in segmentation and object detection. However, their reliance on large datasets and high computational requirements poses significant challenges for practical applications.

2.6.3 Multimodal Hybrid Approaches

Recent advancements integrate CNNs and transformers to combine their strengths for depth estimation and segmentation. Hybrid models, such as those merging ResNet with ViT, show notable improvements in supervised monocular depth estimation[20]. Advanced architectures, like Swin Transformer [21], incorporate hierarchical encoder-decoder structures and multiscale fusion attention mechanisms for enhanced depth and segmentation performance. As a result, ViT-based architectures have become highly promising for achieving superior accuracy across multimodal tasks [4].

2.7 CHALLENGES AND FUTURE DIRECTIONS

The following are the challenges identified in literature survey of recent publications.

- High Number of Parameters: One major challenge in deep learning architectures for depth and segmentation is achieving a lightweight model with fewer parameters. Reducing the model size is essential for runtime efficiency, particularly for the real-world applications where the memory and computational resources are limited.

- Domain Generalization: Another significant challenge is addressing the lack of domain generalization. This arises due to class imbalances or insufficient data for specific classes, leading to poor generalization across diverse scenarios. Ensuring a balanced dataset or introducing regularization techniques is vital to improve model adaptability.

- Multi-task Learning: Multi-task learning has emerged as a promising future direction. By enabling shared feature learning across tasks such as the depth estimation and segmentation, this approach can significantly reduce the model parameters while improving overall performance. Multi-task models are expected to address challenges related to both memory efficiency and generalization by leveraging the synergies between multiple tasks. Similarly by learning those features from a similar task using multi-task model has a scope to improve domain generalization.

CHAPTER 3

SYSTEM DESIGN

3.1 OVERVIEW

The proposed model is built to predict two things at once from an image: how far the objects are (depth) and classifying each pixle in the image (segmentation). Each part of this model works together to understand surroundings in detail. The common vision transformer encoder is used for both in segmentation decoder and depth decoder. Segmentation decoder is obtained from the SegFormer Architecture. The monocular depth is obtained from the GLP depth decoder. This chapter explains how each module works in simpler terms.

3.2 DATASET

The dataset contains 2048 x 800 sized images, with 19 classes for semantic segmentation labels with addition to one unlabbled class and dense depth map . It contains 5,000 images with 2,975 images for training, 500 images for validation. The test dataset size is of 1,525 images. For instance segmentation, the instance masks are provided with unique mask values for each object instead of each class.

The architecture shown in figure 3.1 takes an image of dimensions *HxWxC*, with height H, width W and C channels. The output is in the form of segmentation and depth maps. These are calculated in parallel to each other at the decoder site. Thus the single encoder is used by two decoders performing multi-task operations in a single model

3.3 ARCHITECTURE

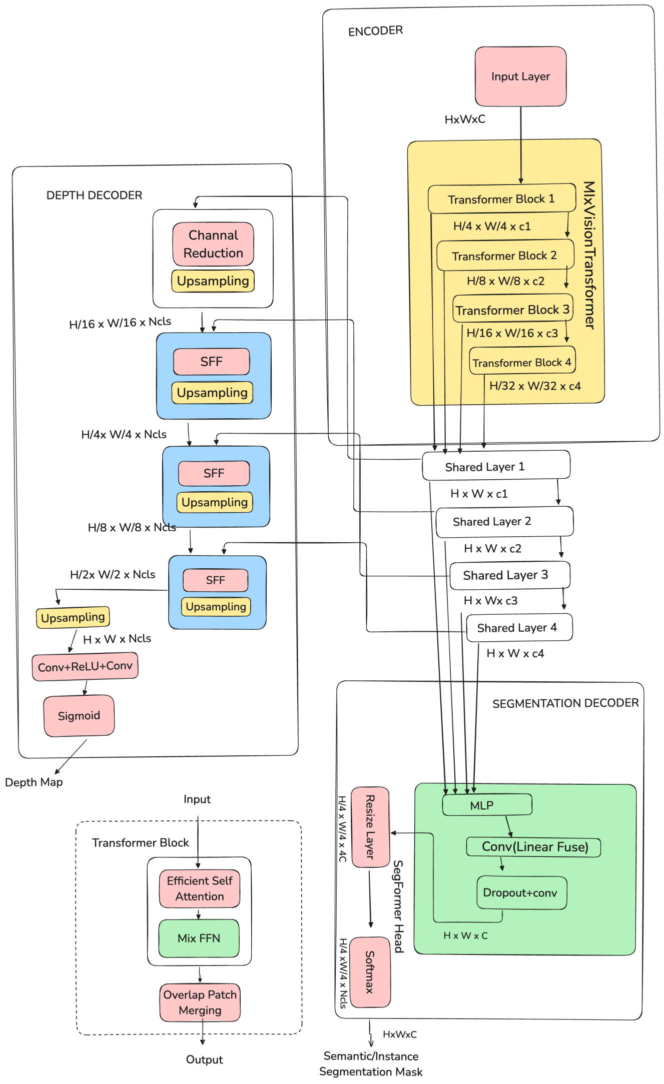

Figure 3.1: Architecture

3.4 DEPTH PREDICTION WITH GLPDEPTH MODULE

The GLPDepth module is the core part of the model that predicts depth from an image. It uses a powerful encoder called mitb4, a type of Vision Transformer (ViT) that captures important features from the image at different scales. This encoder breaks down the image into several layers, creating feature maps (conv1, conv2, conv3, and conv4) that each hold unique details.

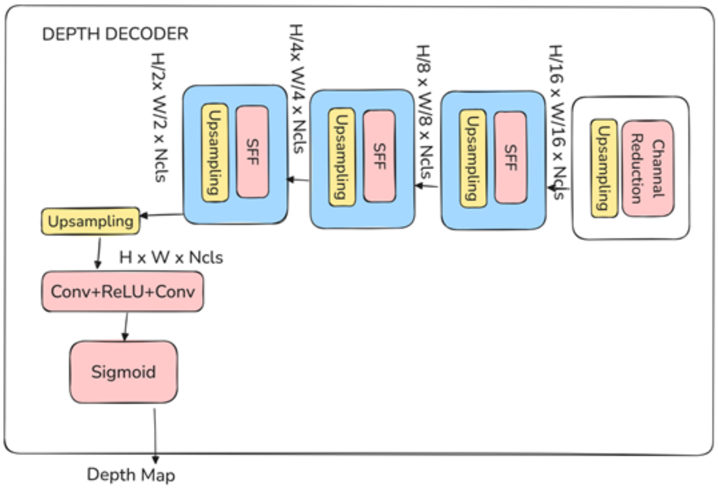

Figure 3.2: Depth Decoder Module

The depth decoder then takes these feature maps and processes them to predict how far objects are from the camera as shown in figure 5.1. This part uses upsampling to make the features larger and combines information from the different layers improving the depth prediction. A special module called Selective Feature Fusion helps by blending details from nearby and faraway objects, using attention to focus on the most important parts of the image. This process creates a refined depth map where each pixel represents distance.

The depth map is finally adjusted with a sigmoid function to keep values within a set range (limited by max depth). This makes sure that the depth predictions stay accurate and realistic.

3.5 SEGMENTATION WITH SEGMENTATION DECODER

The Segmentation Decoder is an additional module inspired by a model called SegFormer,which is very good at identifying and separating objects in an image. This decoder also uses the encoder's feature maps to find and label different parts of the scene, like roads, cars, and pedestrians. figure 5.2 shows the linear fuse layer responsible for aggregating all the learned parameters.

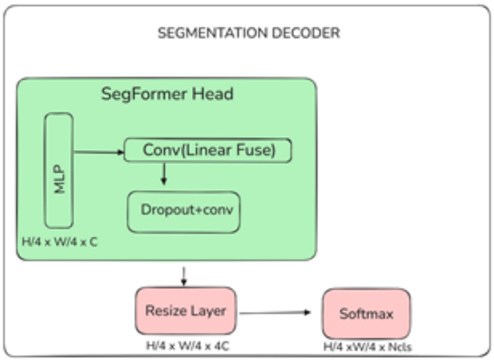

Figure 3.3: Segmentation Decoder Module

This decoder has several upsampling layers to make the feature maps larger and more detailed. The final layer is a segmentation head that produces pixel-level labels for each object, creating a full segmentation map. This allows the model to understand and label each part of the scene accurately, making it useful for tasks such as autonomous driving and scene analysis.

3.6 SHARED ENCODER FOR DEPTH AND SEGMENTATION

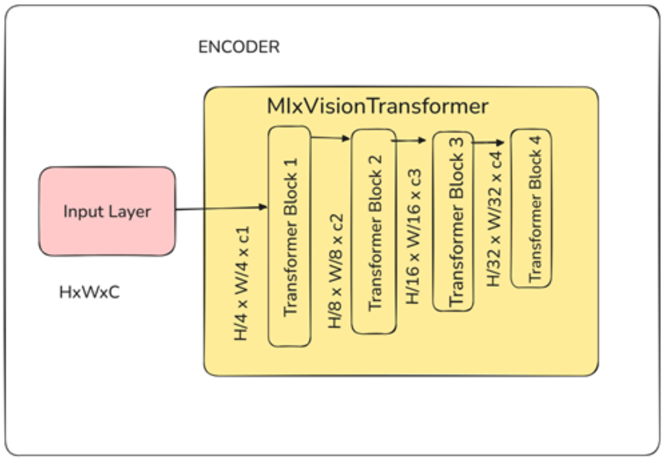

Figure 3.4: Shared Encoder Module

Both the depth and segmentation tasks use the same encoder to save time and improve results. This shared encoder, as shown in figure 3.4, lets the model extract useful information once and then apply it to both depth and segmentation. For example, knowing where objects are (segmentation) helps model understand how depth changes across image. Even when this model is used in automobiles, instead of two models taking the same images and processing individually; having a single model take one input and process it makes the model much more efficient, while decreasing the number of parameters required for the model.

During training, each task has its own loss function, depth estimation uses a loss for continuous values, while segmentation uses a classification loss for object labels. These loss functions are balanced so that both tasks learn effectively without disturbing each other.This approach helps the model perform well at both depth and segmentation tasks. These losses are combined together and the model is trained based on the combined losses. These losses are back propagated to update the losses.

CHAPTER 4

IMPLEMENTATION

4.1 OVERVIEW

This chapter discusses the flow from preprocessing of data to the inference of the results.

4.2 DATA PREPARATION

The model is designed to work with Cityscapes dataset, which is often used in autonomous driving research. The data pipeline loads pre-processed files for images and labels, so the model can learn in batches of 8 images. The following section elucidates the following sections elucidate the algorithms used in the preprocessing of the dataset.

4.3 DATA AUGMENTATION

This section describes the data augmentation steps employed in the proposed approach, as summarized in Algorithm 4.2. These transformations aim to enhance model generalization, efficiently utilize the dataset, and prepare the data for training, validation, and testing.

The input consists of the Cityscapes dataset with image dimensions OG_W and OG_H, resized dimensions W and H, being 2048x800 and other augmentation parameters. The output is a transformed dataset with the Telea

Inpainting performed over all the input images. This diffusion-based approach iteratively fills in the missing region (or mask) by propagating information from the boundary of the region inward. It uses both the known pixel values and their gradients to estimate the missing values. The output images are clipped version of input images. This is because the bottom portion consists of ego vehicle class which will confuse the model if it is given as input. The depth is calculated based on the focal length and baseline as shown in equation 4.1 and equation 4.2.

$$d = \frac{\text{float}(p) - 1.0}{256.0} \tag{4.1}$$

$$D = \frac{B \cdot f_x}{d} \tag{4.2}$$

The d, p, B, f_x, D described in the equation 4.1 and equation 4.2 are as follows:

- d is the true disparity of the image subracted by one to avoid divition by zero error,

- p is the pixel value of the disparity image in the range 0 to 255 integer value,

- B is the baseline, i.e., the distance between the two stereo cameras and

- f_x is the focal length in pixels.

- D represents the computed depth.

4.3.1 Depth Mask Generation

The dense depth map images were generated by computing disparity between the stereo images. The depth image has jitter and disconnected depth mask regions. The problem of jitter in the image is addressed by performing depth mask inpainting on the depth mask images. The algorithm 4.1 explains the steps involved in depth map computation from disparity using opencv libarry for depth map inpainting. The baseline and focal length are used to compute this. Baseline is the distance between the two cameras, and focal length is the distance between the point where the light meets inside the lens and the camera's sensor. For cityscapes dataset the base line distance in meters is 0.2093 and the focal length is 2262.52. The maximus depth was cut to 500 meters.

4.3.2 Image Resizing

The original image dimensions were resized using the fixed scaling factors to convert 2048 × 800 images to 256 × 256. Resized dimensions were calculated as follows:

$$W = \lfloor OG\_W/9.14 \rfloor,$$
$$H = \lfloor OG\_H/3.57 \rfloor.$$

This resizing ensures computational efficiency while maintaining relevant visual features. Also, it ensures compatibility with the vision transformer for feature

Algorithm 4.1 Disparity Images to Dense Depth Map

```
Input: Input disparity image disparity, image dimensions $w$, $h$, inpaint radius $r$, stereo baseline $b$, focal length $f$, maximum depth $D_{max}$
Output: Depth map depth
Crop noisy areas: remove top and side regions of the disparity image
disparity $\leftarrow$ resize(disparity$[50 : h \times 0.8, 100 :], (w, h)$)
if $r \neq$ None then
    Inpaint invalid disparities
    disparity $\leftarrow$ inpaint(disparity, mask, $r$)
end if
Apply median blur to reduce noise
disparity $\leftarrow$ median_blur(disparity, 5)
Scale disparity values to obtain true disparity
disparity[disparity $> 0$] $\leftarrow$ (disparity[disparity $> 0$] $- 1)/256$
Compute depth using stereo geometry
depth $\leftarrow b \cdot f/($disparity $+ 0.1)$
Clip depth values to the maximum allowed depth
depth $\leftarrow$ clip(depth, $0, D_{max}$)
return depth
```

extraction. The square dimensions generally train better than that of an image having irregular height and width.

4.3.3 Data Transformation Pipeline

To improve diversity of training data, several augmentation techniques were applied to the dataset:

- **Tensor Conversion**: Converts input images, masks, and annotations to tensors for compatibility with PyTorch models.
- **Rescale**: Resizes all the images and annotations to $H \times W$ dimensions.
- **Normalize**: Applies channel-wise normalization using mean values $[0.485, 0.456, 0.406]$ and standard deviation $[0.229, 0.224, 0.225]$ for RGB images.

Algorithm 4.2 Data Augmentation and Preprocessing

```
Input: Dataset dataset, original image dimensions $OG\_W$, $OG\_H$, resized dimensions $W$, $H$, augmentation parameters.
Output: Transformed dataset.
Compute resized dimensions:
    $$W \leftarrow \lfloor OG\_W/9.14 \rfloor, \quad H \leftarrow \lfloor OG\_H/3.57 \rfloor.$$
Define transformations for training:
    $\text{Pipeline}_{\text{train}} \leftarrow [\text{Tensor Conversion}, \text{Rescale}, \text{Normalize}, \text{RandomHorizontalFlip}].$
Define transformations for validation:
    $\text{Pipeline}_{\text{valid}} \leftarrow [\text{Tensor Conversion}, \text{Rescale}, \text{Normalize}].$
Define transformations for testing:
    $\text{Pipeline}_{\text{test}} \leftarrow [\text{Tensor Conversion}, \text{Normalize}].$
Define a custom collation function to handle:
    $\text{Images} \rightarrow \text{Stack as tensors},$
    $\text{Masks/Depths} \rightarrow \text{Stack as tensors},$
    $\text{Bounding Boxes} \rightarrow \text{List structure}.$
Apply transformations to datasets:
    $\text{Dataset}_{\text{train}} \leftarrow \text{Apply Pipeline}_{\text{train}} \text{ to training split},$
    $\text{Dataset}_{\text{valid}} \leftarrow \text{Apply Pipeline}_{\text{valid}} \text{ to validation split},$
    $\text{Dataset}_{\text{test}} \leftarrow \text{Apply Pipeline}_{\text{test}} \text{ to test split}.$
Define data loaders for batch-wise processing:
    $\text{Loader}_{\text{train}}, \text{Loader}_{\text{valid}}, \text{Loader}_{\text{test}}.$
return Transformed datasets and data loaders.
```

- **RandomHorizontalFlip**: Randomly flips images horizontally with a probability of 0.5 to introduce variability in spatial orientation.

These transformations were combined into a pipeline for the training dataset. The validation dataset and test dataset used fewer transformations to avoid altering the input data distribution:

- **Validation Transformations**: Includes **Tensor Conversion**, **Rescale**, and **Normalize**.
- **Test Transformations**: Includes **Tensor Conversion** and **Normalize**.

4.3.4 Custom Collation Function and Dataset Preparation

A custom collation function was defined to handle variable-sized annotations and facilitate efficient data loading. The function ensures that:

- Images, masks, and depth maps are stacked as tensors for batch processing.

Using this custom function, the datasets were transformed and then prepared and then, the data loaders were defined for each dataset to enable efficient batch-wise processing during model training, validation, and evaluation. Finally when all the transformations mentioned in algorithm 4.2 are applied, the resulting images are rescaled, normalized and have augmentation.

4.4 MODEL BUILDING

This section describes the model building process for single task and multi-task model. There were no changes made to the layers by itself, rather there was a separation of the modules in the architecture.

4.4.1 Single Task Model

The single task model consists of semantic or instance segmentation outputs only for segmentation task and depth mask output for monocular depth estimation task. This is achieved by having the encoder directly connected with decoder instead of having the encoder share its feature outputs to another decoder. Also, the loss is calculated using only the depth loss and no the combined loss.

4.4.2 Multi-Task Model

A shared encoder is used to give feature outputs to both segmentation and monocular depth estimation decoders. Also, instead of using their specific loss functions for their tasks, the combined losses are used for training the model. Thus while updating the gradients in back-propagation, the model will be able to learn the features of both depth and segmentation simultaneously.

4.5 MODEL TRAINING

For the training of the single-task model, the loss function used for segmentation is categorical cross entropy as defined in equation 4.3. This was used for both instance and semantic segmentation.

$$\mathrm{CCE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}) \tag{4.3}$$

.

For the training of the single-task model, the loss function used for monocular depth estimation is RMSLE, root mean square logarithmic error as defined in equation 4.4.

$$\text{RMSLE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\log(y_i+1)-\log(\hat{y}_i+1)\right)^2} \tag{4.4}$$

For the training of the multi-task model, the loss function used for segmentation and monocular depth estimation is the combined loss function of categorical cross entropy as defined in equation 4.3 and RMSLE, root mean square logarithmic error as defined in equation 4.4. The resulting equation is as shown in 4.5.

$$\text{Multi-Task loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}) + \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\log(y_i+1)-\log(\hat{y}_i+1)\right)^2} \tag{4.5}$$

This loss is used in the back-propagation to update the weights during training of the multi-task model for both instance segmentation with monocular depth and semantic segmentation with monocular depth.

CHAPTER 5

RESULTS AND ANALYSIS

This chapter presents the results of training and testing the proposed multi-task model against single task models. The objective of the experiment was to assess the impact of multitask training on performance and compare it against the single-task models. This was achieved by training the model to perform both segmentation and monocular depth estimation concurrently. There were two multi-task models created, one with semantic segmentation with monocular depth, and the other with instance segmentation with monocular depth task. For each of them two separate models were trained: one focused solely on segmentation, another on depth estimation. All models were evaluated using the Cityscapes dataset.

To study the impact of multi-task training for segmentation, Jaccard loss was calculated and compared with single-task model against multi-task model for both semantic segmentation and instance segmentation.

5.1 GENERATION OF MULTI-TASK OUTPUT

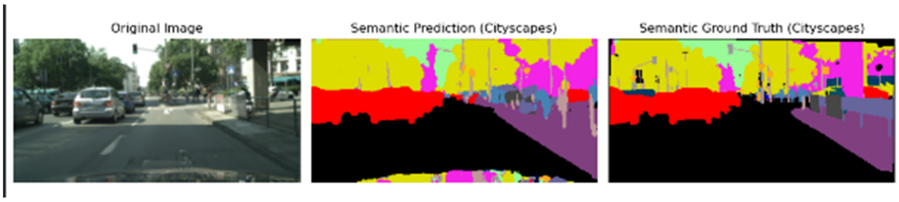
Original Image
Semantic Prediction (Cityscapes)
Semantic Ground Truth (Cityscapes)

Figure 5.1: Segmentation Output

Both, figure 5.1 and figure 5.2 illustrate the comparison with the ground truth and the generated segmentation and depth masks from the Multi Task model.

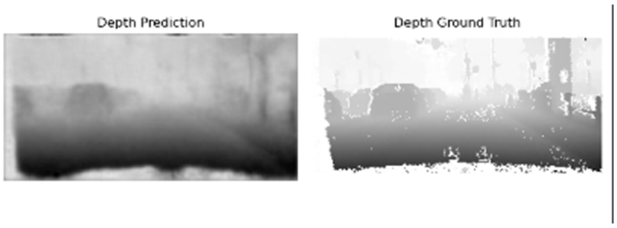

Figure 5.2: Depth Output

In figure 5.1, the rightmost image is the ground truth image, the middle image is the predicted image and the first image is the original image given as input. These images are resized and visualized in $256x128$ format using matplotlib library. Similarly the rightmost image is the instance segmentation image output, the middle image is the semantic output image for comparison and the first image is the original image given as input. These images are also resized and visualized in $256x128$ format using matplotlib library done previously.

The rightmost image in Figure 5.2 represents the ground truth depth mask created from the disparity image using the inpainting process, while the leftmost image shows the depth map generated by the prediction model.

5.1.1 Semantic and Instance Segmentation Comparison

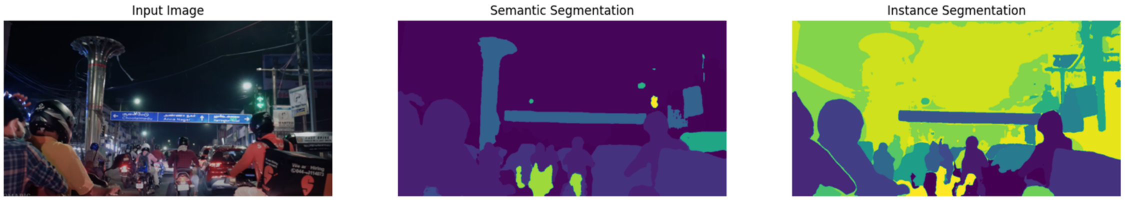

Figure 5.3: Instance Vs Semantic Segmentation

Since this model was not given class identifiers along with the train identifiers, semantic output is needed to classify them into classes. In figure 5.3, the rightmost image is the instance segmentation giving a unique color to each of the object recognized and segmented. The instance segmentation output displays each object as unique and thus they have unique color. While the middle image describes the semantic segmentation. The leftmost image is the input image which is given to the model.

5.2 COMPARISON OF MULTITASK PERFORMANCE

The proposed multitask model demonstrates competitive performance in several metrics when compared to the ”Depth only” and ”Segmentation only” models. Below is the detailed comparison based on the tables:

5.2.1 Loss Metrics

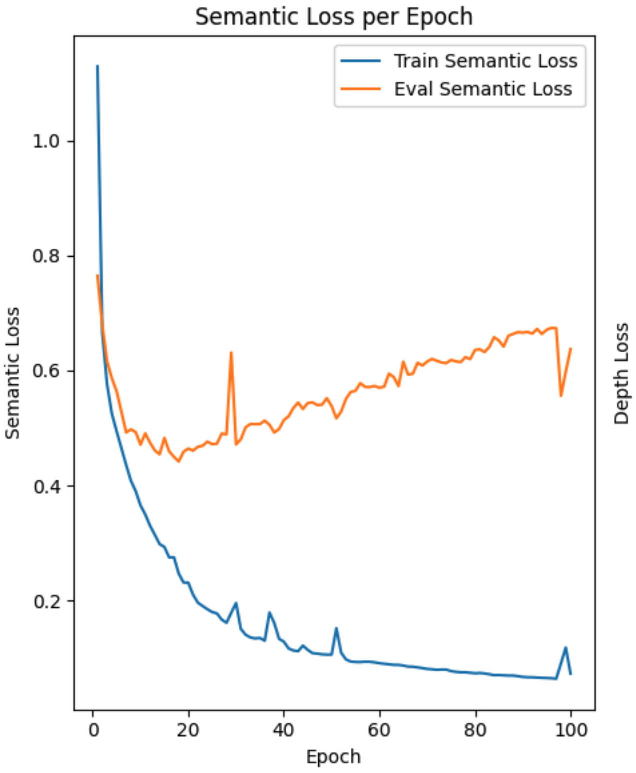

Figure 5.4: Segmentation Loss

The training was run for 100 epochs for instance segmentation with monocular depth tasks, and single task of instance segmentation, and single

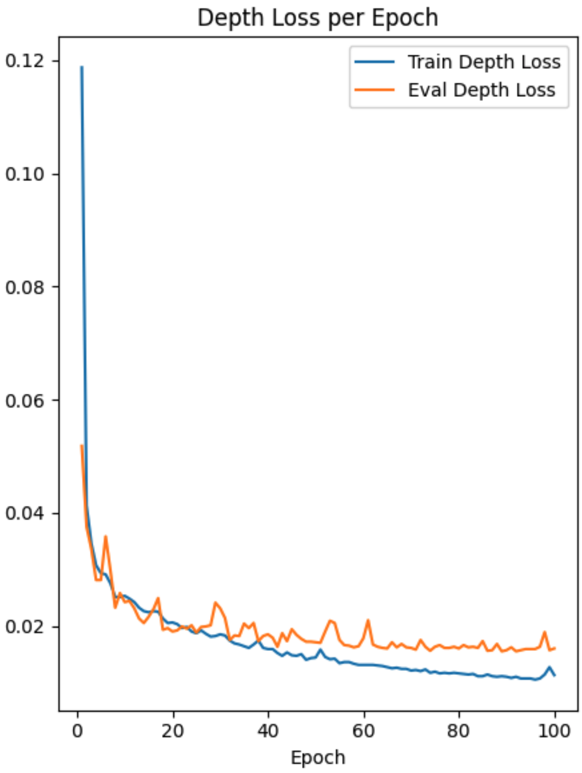

Figure 5.5: Depth Loss

task of monocular depth estimation and similarly for semantic segmentation with monocular depth estimation, and single task of semantic segmentation, and single task of monocular depth estimation . For reference Figure 5.4 shows instance segmentation loss which has lesser variance than that of depth loss shown in 5.5. The depth loss has lesser and lesser variance in validation loss as it tries to converge. The combined loss shows rather less variance and a smoother convergence as shown in figure 5.6

Since both the losses are combined the model tries to update the weights based on the loss of both tasks, segmentation and depth, which means that the model is able to learn the features of depth and segmentation at the same time. This shared loss leads to better identification of the edges and other features for their specific tasks.

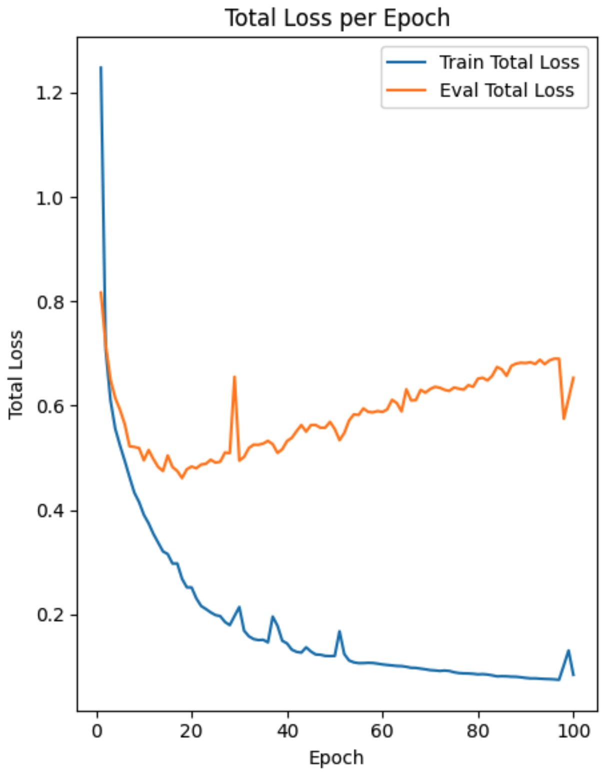

Figure 5.6: Combined Total Loss

5.2.2 Depth Metrics

The multitask model matches the ”Depth only” model in **RMSE** having (87.4768 vs. 79.2366 and 83.1817), but achieves identical performance in **RMSElog** with a value of **0.2901** and **0.1830**, **0.2763** for the depth only model and the two Multi-task models respectively as shown in table 5.1. It can also be seen that semantic segmentation loss helped more in reducing the depth loss compared to instance segmentation loss.

Table 5.1: Cityscapes: Depth Metrics

Model	**RMSE**	**RMSElog**
Depth only	87.4768	0.2901
Multitask (semantic)	79.2366	0.1830
Multitask (Instance)	83.1817	0.2763

5.2.3 Segmentation Metrics

Compared to the single-task model for semantic segmentation, the multitask model with semantic segmentation and monocular depth estimation achieves slightly higher but comparable performance in **Jaccard Index** (0.7524 vs. 0.7904) as shown in table 5.2).

Table 5.2: Cityscapes: Semantic Segmentation Metrics

Model	**Jaccard Index**
Semantic Segmentation only	0.7524
Multitask	0.7904

Table 5.3: Cityscapes: Instance Segmentation Metrics

Model	**Jaccard Index**
Instance Segmentation only	0.6424
Multitask	0.7124

On the other hand, the single-task model for instance segmentation, the multitask model with instance segmentation and monocular depth estimation achieves slightly higher but comparable performance in **Jaccard Index** (0.6424 vs. 0.7124) as shown in table 5.3). This is due to the fact that the model has to learn how to differentiate objects, and since the semantic segmentation has knowledge about the classes, it can expect what features to use to identify a particular class. Thus this task is challenging for instance segmentation as it doesn't know about the classes, thereby having lower Jaccard Index than that when compared to semantic segmentation.

In summary, multitask model performs equally well in depth-related metrics (**Jaccard Index** and **RMSElog**). While it works better in segmentation metrics, the performance remains comparable across all tasks, making it a strong choice for scenarios requiring both depth and segmentation capabilities.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

This project implemented the application of Vision Transformers for visual prediction tasks in autonomous driving scenarios. The primary objective of designing multitask model capable of performing monocular depth estimation and segmentation simultaneously while maintaining similar metrics of single task models was achieved. A summarized review of the existing ViT based models was undertaken, leading to the selection of SegFormer and GLP-Depth as foundation for the proposed multitask framework. The performance of the multitask model was evaluated cityscape dataset and compared against the single task models for depth estimation and segmentation.

The results revealed that the multitask model delivered comparable accuracy to single-task models while significantly reducing combined inference time for both tasks. Semantic segmentation with monocular depth estimation model had higher Jaccard index than that of the Instance segmentation with monocular depth estimation model.Similarly, the RMSE scores had been greatly reduced due to multi-task training loss functions. Thus multi-task models had produced superior metrics than the single-task model making it a better choice.

6.2 FUTURE WORK

Future research could further expand this work in several directions. For instance, semi-supervised or self-supervised learning techniques could be explored to minimize the dependency on large amounts of labeled data, reducing training costs and effort. This approach may encourage the model to extract more universal features rather than dataset-specific ones.

This project has scope to address issue of visual tasks in unfavorable environments, by reducing the load of finding image representations by having the common encoder.

REFERENCES

[1] Wonjun Kim, Sanghoon Kim, Ryong Lee, Rae-Young Jang, and Myung-Seok Choi. Feature high-boosting for semantic segmentation. *IEEE Access*, 10:114749–114758, 2022.

[2] Juan Luis Gonzalez Bello, Jaeho Moon, and Munchurl Kim. Self-supervised monocular depth estimation with positional shift depth variance and adaptive disparity quantization. *Trans. Img. Proc.*, 33:2074–2089, March 2024.

[3] Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(12):5586–5609, 2022.

[4] Durga Prasad Bavirisetti, Herman Ryen Martinsen, Gabriel Hanssen Kiss, and Frank Lindseth. A multi-task vision transformer for segmentation and monocular depth estimation for autonomous vehicles. *IEEE Open Journal of Intelligent Transportation Systems*, 4:909–928, 2023.

[5] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. Oneformer: One transformer to rule universal image segmentation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2989–2998, 2023.

[6] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: simple and efficient design for semantic segmentation with transformers. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2024. Curran Associates Inc.

[7] Haoli Bai, Hongda Mao, and Dinesh Nair. Dynamically pruning segformer for efficient semantic segmentation. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3298–3302, 2022.

[8] Daniel Kienzle, Marco Kantonis, Robin Schön, and Rainer Lienhart. Segformer++: Efficient token-merging strategies for high-resolution semantic segmentation. In *2024 IEEE 7th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 75–81, 2024.

[9] Justin Liang, Namdar Homayounfar, Wei-Chiu Ma, Yuwen Xiong, Rui Hu, and Raquel Urtasun. Polytransform: Deep polygon transformer for instance segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9128–9137, 2019.

[10] Doyeon Kim, Woonghyun Ka, Pyunghwan Ahn, Donggyu Joo, Sewhan Chun, and Junmo Kim. Global-local path networks for monocular depth estimation with vertical cutdepth. *ArXiv*, abs/2201.07436, 2022.

[11] Chungkeun Lee, Changhyeon Kim, Pyojin Kim, Hyeonbeom Lee, and H. Jin Kim. Scale-aware visual-inertial depth estimation and odometry using monocular self-supervised learning. *IEEE Access*, 11:24087–24102, 2023.

[12] Zhekai Bian, Xia Wang, Qiwei Liu, Shuaijun Lv, and Ranfeng Wei. Mbudepthnet: Real-time unsupervised monocular depth estimation method for outdoor scenes. *IEEE Access*, 12:63598–63609, 2024.

[13] Amir Atapour-Abarghouei and Toby P. Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2800–2810, 2018.

[14] Amir Atapour-Abarghouei and Toby P. Breckon. Monocular segment-wise depth: Monocular depth estimation based on a semantic segmentation prior. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 4295–4299, 2019.

[15] Danish Nazir, Alain Pagani, Marcus Liwicki, Didier Stricker, and Muhammad Zeshan Afzal. Semattnet: Toward attention-based semantic aware guided depth completion. *IEEE Access*, PP:1–1, 01 2022.

[16] Nasser Aloufi, Abdulaziz Alnori, and Abdullah Basuhail. Enhancing autonomous vehicle perception in adverse weather: A multi objectives model for integrated weather classification and object detection. *Electronics*, 13(15), 2024.

[17] Mourad A. Kenk and M. Hassaballah. Dawn: Vehicle detection in adverse weather nature dataset. *ArXiv*, abs/2008.05402, 2020.

[18] Seung-Jun Hwang, Sung-Jun Park, Joong-Hwan Baek, and Byungkyu Kim. Self-supervised monocular depth estimation using hybrid transformer encoder. *IEEE Sensors Journal*, 22(19):18762–18770, 2022.

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020.

[20] Lorenzo Papa, Paolo Russo, and Irene Amerini. Meter: A mobile vision transformer architecture for monocular depth estimation. *IEEE Trans. Cir. and Sys. for Video Technol.*, 33(10):5882–5893, March 2023.

[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. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021.

[22] Christoph Hümmer, Manuel Schwonberg, Liangwei Zhou, Hu Cao, Alois Knoll, and Hanno Gottschalk. Strong but simple: A baseline for domain generalized dense perception by clip-based transfer learning, 2024.

[23] Marya Rasib, Muhammad Butt, Faisal Riaz, Adel Sulaiman, and Muhammad Akram. Pixel level segmentation based drivable road region detection and steering angle estimation method for autonomous driving on unstructured roads. *IEEE Access*, 9, 12 2021.

[24] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.

[25] Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. Vlt: Vision-language transformer and query generation for referring segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(6):7900–7916, June 2023.

[26] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1280–1289, 2022.

[27] 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. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing.

[28] Muhammad Atif Butt and Faisal Riaz. Carl-d: A vision benchmark suite and large scale dataset for vehicle detection and scene segmentation. *Image Commun.*, 104(C), May 2022.

[29] Salih Can Yurtkulu, Yusuf Hüseyin Şahin, and Gozde Unal. Semantic segmentation with extended deeplabv3 architecture. In *2019 27th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4, 2019.

[30] Jianyu Zhang, Hexuan Hu, Tianjin Yang, Qiang Hu, Yufeng Yu, and Qian Huang. Hr-aspp: An improved semantic segmentation model of cervical nucleus images with accurate spatial localization and better shape feature extraction based on deeplabv3+. In *Proceedings of the 15th International Conference on Digital Image Processing*, ICDIP '23, New York, NY, USA, 2023. Association for Computing Machinery.

[31] Yongfeng Su, Juhui Zhang, and Qiuyue Li. Daformer: A novel dimension-augmented transformer framework for multivariate time series forecasting. In *Advanced Intelligent Computing Technology and Applications: 20th International Conference, ICIC 2024, Tianjin, China, August 5–8, 2024, Proceedings, Part II*, page 175–187, Berlin, Heidelberg, 2024. Springer-Verlag.

[32] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 91–99, Cambridge, MA, USA, 2015. MIT Press.

[33] Guo-Ye Yang, George Kiyohiro Nakayama, Zikai Xiao, Tai-Jiang Mu, Xiaolei Huang, and Shi-Min Hu. Semantic-aware transformation-invariant roi align. In *AAAI Conference on Artificial Intelligence*, 2023.