Depth Estimation from A Single Image Using Global Structure and Local Scene Information

（大域構造と局所シーン情報を用いた単眼深度推定）

July, 2023

Doctor of Philosophy (Engineering)

ANDI HENDRA

（アンディ・ヘンドラ）

Toyhohashi University of Technology

Abstract

We propose a novel framework for accurately estimating depth information from a single image. Despite its simplicity and compactness, our framework demonstrates consistent and reliable performance by effectively integrating global and local image features.

To address the challenge, we employ two distinct deep neural network architectures. We adopt an encoder-decoder model with a two-stage strategy in the first architecture. This approach leverages multi-task loss optimization and incorporates adaptive learning rate adjustment based on the loss behaviour. The second architecture expands the conditional GAN (cGAN) model, introducing a three-player GAN (TP-GAN) framework. To enhance the reliability of the depth estimation, we include the structural similarity measure (SSIM) loss as part of this architecture. By utilizing this architecture, we aim to optimize the depth estimation performance.

Our proposed architectures utilize 1×1 convolution to reduce the dimensionality of the feature maps, thereby enabling the model to focus on capturing high-level semantics and global context. Conversely, local features are extracted through stacks of convolution with smaller kernels relative to the input size that can help in capturing local context and details that might be overlooked by global features alone. Combining global and local features enables the model to leverage the overall scene understanding and fine-grained local details to enhance the accuracy of depth prediction.

To evaluate the effectiveness of our approaches, we conducted comprehensive quantitative and qualitative comparisons with several state-of-the-art methods in the field. Our experiments were conducted on two well-known publicly depth datasets, the indoor NYU Depth v2 and outdoor KITTI datasets. The results consistently demonstrate that our proposed method outperforms numerous previous related monocular depth strategies and delivers reliable performance when compared to the transformer-based model, further demonstrating its efficacy.

Furthermore, we investigated the generalization capabilities of our model to other datasets. To assess cross-dataset adaptation, we trained our model on one dataset and tested it on another and vice versa. Our model exhibits reliable generalization by effectively learning scene variations across indoor and outdoor datasets. Notably, when trained on indoor data and tested on the outdoor range dataset, our model achieved consistent performance of SSIM scores, with some values close to one.

In addition, we conducted an in-depth analysis to assess the robustness of our depth estimation model under different contrast levels. To evaluate our model's performance, we generated visualizations of estimated depth and calculated the (SSIM) score using images captured under different contrast conditions. Specifically, we evaluated six random KITTI data samples containing scenes with normal, lower, and higher contrast levels. The results demonstrated that our model outperformed other methods, indicating that the SSIM metrics consistently showed superior performance across the dataset.

In future research, it is imperative to further advance the single image depth estimation field by focusing on developing models that exhibit enhanced generalization capabilities across diverse datasets. This could be achieved by designing an adaptive model that effectively discriminates between ground truth and generated depth and accurately classifies whether the input image belongs to an indoor or outdoor dataset. Such advancements would significantly contribute to the robustness and versatility of depth estimation methods.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Research Background

Depth estimation encompasses measuring the distance between the camera and each pixel in an image. The extraction of geometric information from a scene is an essential procedure that has significant implications in various fields, including but not limited to robotics, autonomous driving, object recognition, scene understanding, 3D modelling and animation, augmented reality, industrial control, and medical diagnosis. The process of estimating depth requires the ability to infer the environment's semantic context and understand its fundamental geometric arrangement. The research within the domain of computer vision has extensively focused on depth estimation due to its inherent complexity.

In general, depth is extracted from two prevalent methodologies: multi-view (stereo) strategies that leverage epipolar geometry by utilizing multiple viewpoints or single-view (monocular) methods that analyze depth cues within the RGB image input. Traditional stereo matching was the most explored due to its strong connection to the human binocular system. It works by finding corresponding points in the two views and computing the horizontal displacement between them. On the one hand, the problem of establishing the geometric relation between disparity and 3D position is deterministic and well-defined. On the other hand, determining pixel correspondence between two images is quite challenging. Stereo vision exhibits limited performance in regions with low texture or repetitive patterns and when objects appear differently to both views or are partly occluded. Moreover, the resolution of the cameras and the distance between them, also known as the baseline, affect the effective range of accurate depth estimation. Therefore, stereo depth maps are typically post-processed to correct wrong estimates.

Having worked on depth estimation and its application to autonomous driving in particular, it is challenging for several reasons, including occlusion, a dynamic object in the scene, and imperfect stereo correspondence. Reflective, transparent, and mirror surfaces pose the greatest difficulty for stereo-matching algorithms to resolve. Consequently, many companies rely on utilizing Lidar to enhance depth measurements' accuracy, reliability, and robustness. Currently, the prevailing tendency in autonomous driving perception stacks leans towards sensor fusion, given that the extracted features of each sensor possess unique strengths. However, ever since the emergence of deep learning, this study area has received considerable interest and achieved remarkable results. As a result, numerous studies have been carried out to address these concerns.

Nowadays, significant advancements have been achieved in capturing depth data

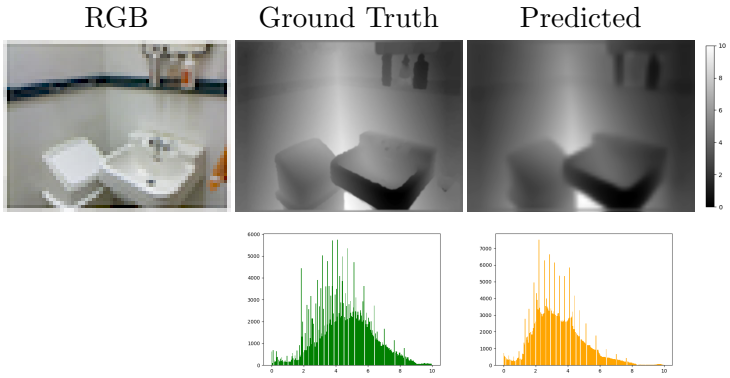
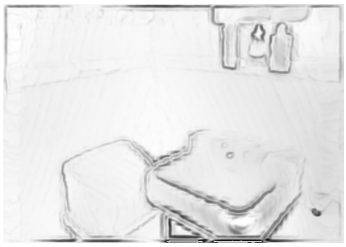


Figure 1.1: Depth Prediction on NYU Depth v2. Top: from left to right: input RGB image, ground truth depth, predicted depth by our model, and depth colorbar distance in meters. Bottom: from left to right: Histogram of depth value of the ground truth and histogram of depth value of the predicted depth.



Structural Similarity Index Measure (SSIM): 0.91

Figure 1.2: The SSIM difference in error when compared against the ground truth.

information from a single image. In contrast to inferring depth from stereo images, this technique only relies on one camera. Given that a single perspective is considered a priori, there are no performance limitations imposed by the appearance of objects in the field of view or their disposition in the scene. Hence, the estimation process relying on a single image should not have an issue with objects in close range, at a considerable distance, or when they are partially obscured. As monodepth estimation from a single still image is less amenable to mathematical analysis than stereo vision, mono estimators use learning strategies to infer depth from images. The feature extraction process for depth prediction is accomplished by minimizing error on a training set. Consequently, it cannot be assured that the model will exhibit effective generalization in the operational setting, particularly in cases where a significant difference exists between the operational and training environments.

Various depth estimation algorithms have been proposed to address the issue of single view image, which remains a challenging task in computer vision. The conventional approach for inferring depth from a single image has generally relied on simplifying assumptions about the geometric structure of the scene or incorporating some external knowledge about the scene, such as semantic labels. Recent approaches remove the necessity for assumptions as mentioned above and instead employ supervised learning, exclusively relying on cues that can be inferred from the input image.

It is important to incorporate global and local information from the scene to establish a relationship between monocular depth cues. Related to this, Eigen *et al.* [2]

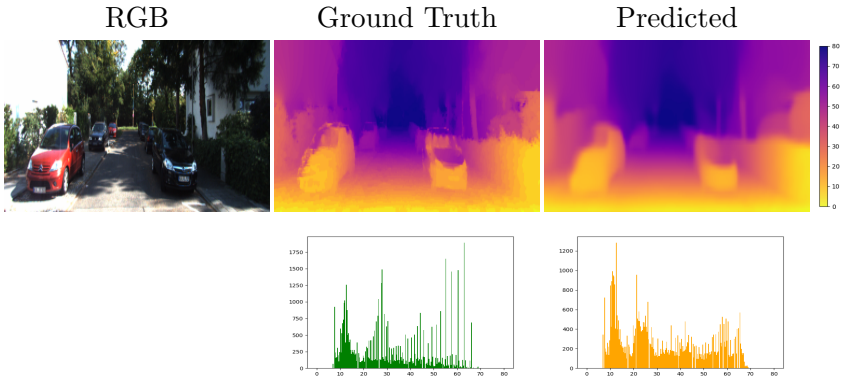


Figure 1.3: Depth Prediction on KITTI data. Top: from left to right: input RGB image, ground truth depth, depth predicted by our model, and depth colorbar distance in meters. Bottom: from left to right: Histogram of depth value of the ground truth and histogram of depth value of the predicted depth.

Structural Similarity Index Measure (SSIM): 0.92

Figure 1.4: The SSIM difference in error when compared against the ground truth.

proposed a method for estimating depth from a single image using two independent deep neural networks. The first network generates an initial global prediction, while the second network refines this prediction locally. Their method performance reported remarkable results and motivated research toward using the deep neural network for depth estimation.

1.2 Research Objectives

The first objective is to develop and implement a simple encoder-decoder based model for single image depth estimation. This model will implement a two-stage deep neural network, utilize multi-losses, and perform a training monitor for optimal learning rate adjustment parameters.

The second objective is to propose and evaluate an innovative depth estimation model utilizing a generative-based model and the structural similarity index measurement (SSIM) loss. This model expands a conditional GAN to include a refiner sub-model responsible as the third player. Here, the generator sub-model will extract global depth features and then integrate them with local feature information from additional sub-model to enhance the network's ability to capture depth cues and improve depth estimation accuracy.

The overall research objective is to contribute to the advancement of single image depth estimation by comprehensively investigating and studying the performance of

the two Convolutional Neural Network (CNN) architectures. Specifically, the aim is to explore how these proposed architectures can effectively enhance the accuracy and robustness of depth estimation from a single image utilizing deep learning by integrating global and local feature information. Additionally, this research attempts to introduce an innovative approach, conduct experiments to demonstrate its effectiveness, and subsequently assess the results of the experiments for the two most popular datasets, NYU depth v2 and KITTI. We show sample image reconstruction compared with their corresponding ground truth depth and histogram depth value for NYU and KITTI data, respectively in Figure 1.1 and 1.3. In Addition, we provide sample structural similarity index measurement (SSIM) differences between ground truth and reconstruction depth images for evaluation in Figure 1.2 and 1.4.

1.3 Related Works

This section will review some of the previous related works that utilized either deep learning or non-deep learning strategies. In the following, we provide brief descriptions of various similarly related methodologies for extracting depth information that have been developed in recent years. The related works are addressed in the following paragraphs.

Initially, extracted depth information from the image used a non-deep learning method that relied on the stereo vision approach, such as structure from motion [11, 12], structured light [13], and de-focusing method [14] which are utilizing image pairs from the same scenes. On the other hand, various approaches have been demonstrated for predicting the depth map from a single RGB image. Following the completion of the work in [15] for creating a 3D model from a single photograph by composing the geometric structure of the image region of outdoor scenes, Saxena *et al.* [16] had successfully estimated depth from a single image based on learning Markov Random Field (MRF) model. The achievement of their method later extended to develop a 3-D reconstruction of scene modeling [17]. However, their approach relies on strong assumptions about scene geometry, which works only in such a scenario.

In recent years, convolutional Neural Networks (CNN) have been utilized to extract depth information from a single input RGB image. To mention a few, Eigen *et al.* [2] estimated depth information from a monocular image using a multi-scale structure that stage-wisely refines the estimated depth map from low spatial resolution to high spatial resolution via independent networks. Liu *et al.* [18] discover the unary and pairwise potential of continuous Conditional Random Field (CRF) and train with a CNN network. In addition, Laina *et al.* [19] developed a fully convolutional architecture to learn feature map up-sampling in order to generate higher resolution output dense maps. Cao *et al.*[20] proposed a distinctive approach to estimate depth from a single image. They utilized fully convolutional residual networks as a classification task instead of a standard regression procedure. Godard *et al.*[10] further studied unsupervised learning with a deep CNN network for a monocular image depth estimation. Their works generated disparity images employing a left-right consistency image reconstruction loss.

Later on, Chen *et al.* [21] presented a residual pyramid decoder (RPD) that takes into account the underlying picture structure at many scales. Yin *et al.* [22] introduced a framework that consists of two primary modules; a depth prediction and a point cloud module, to improve the structure of point clouds derived from depth maps in order to recover more accurate 3D shape from a single image. Gur *et al.* [23] proposed

a deep learning-based method to estimate depth from a single image based on depth focus cues. In their method, the model requires at least one focused image of the same scene from the same viewpoint. Next, Bian *et al.* [6] proposed enhancing unsupervised depth estimation by removing relative rotational motions using an Auto-Rectify network and their innovative loss functions. Eventually, Ye *et al.* [24] introduced a transformer framework for multitask dense prediction. They used an inverted pyramid multitask transformer (InvPT) to learn long-range interaction in both spatial and all-task contexts in a unified architecture. Subsequently, studies on enhancing the quality of depth information using deep learning have been readily conducted.

Meanwhile, generative adversarial network (GAN) [25], also known as two players deep learning network, has received much attention in the research community in solving a variety of different image generating applications, including single image reconstruction [26, 27, 28]. The standard GAN, however, has no control over the generated image representations. In response, GAN is expanded to a conditional GAN (cGAN) [29], in which the generator and discriminator are conditioned on some extra information.

This research presents a simple and reliable image depth estimation task by integrating global image features and local structure information. We address the problem using two deep neural network architectures. The first architecture utilizes a standard encoder-decoder based model by employing two-stage global and local stage networks. Our global network base model is a minor remodel of the original ResNet-50 architecture [30], consisting of only thirty-eight convolution layers in the residual block followed by pair of two up-sampling layers. In contrast, the second stage network is a stack of five convolution layers that accepts the initial depth to be refined as the final output depth. During training, we monitor the loss behaviour and adjust the learning rate hyperparameter. Furthermore, our model evaluates loss based on a combination of three losses; pixel-wise, gradient-direction, and structure similarity.

In the second architecture, We expand the generative adversarial network into a three-player (TP-GAN) in order to capture the global scene layout and then combine it with the local structure information to align the detail of the captured depth. We adopted the residual networks (ResNet) proposed by He *et al.* [30] as the base model in our first sub-model, called generator (G). Our second sub-model, discriminator (D), is implemented as a patch GAN model [31] that effectively only penalizes structure at the scale of local image patches in the N×N output vector as opposed to outputting a single value indicating whether an image is fake or real. Our third sub-model, refiner (R), stacks of six convolutional layers followed by a linear activation to capture the depth local feature. This sub-model is later referred to as the third player in our TP-GAN proposed network.

Further, we implement a conditional adversarial network to assist the generator and refiner in mapping RGB images to their appropriate depths. In our practice, the third player model learns to improve depth structure by incorporating updated weight from the generator with local scene information and expressing feedback from the discriminator throughout each mini-batch training session. In addition, the SSIM loss will further evaluate the structural feature similarity rather than pixel-by-pixel between two images, which is a more effective strategy for image reconstructing tasks, including image depth estimation. The main idea behind this proposed strategy is to complement the refiner to compete with the discriminator.

1.4 Thesis Organization

This thesis is organized as follows: We first describe the research background, objectives and some previous related works in Chapter 1. Chapter 2 describes an introduction about depth estimation then explain two common approach for depth estimation; stereo vision and monocular image. Chapter 3 contains descriptions of the implementation of the convolutional neural networks in the context of monocular depth estimation. Describe the deep learning method for monocular depth estimation according its architectural design and functionality. In Chapter 4, we describe the detailed of our proposed single image depth estimation using the global structure and local depth features information. We explain our detailed experiment and show the effectiveness of our model performance in Chapter 5. Futher, we provide supplementary qualitative results to offer visual insights into the model prediction. We discuss the performance of our research in Chapter 6. Finally, our research work conclusion and future work are described in chapter 7.

Chapter 2

Depth Estimation

2.1 Introduction

Depth estimation can be defined as the process of determining the depth information of a 2D image. In computer vision, depth estimation is essential for enabling machines to perceive the depth and 3D structure of a scene, resulting in enhanced understanding, recognition, and interaction with the environment in various applications. In computer vision, computing depth is one the challenging task and a wide area of research. It has been applied in many vision applications such as 3-D modeling [32], robotics [33], and autonomous driving [34], as well as potentially leading to improve related studies in pedestrian detection tasks [35, 36, 37].

There are different techniques for depth estimation, but stereo vision and monocular depth estimation are two prominent approaches. The use of two or more cameras to capture multiple images of the same scene from different points is required for stereo vision. Monocular depth estimation, in contrast, relies only on the information present in a single image, making it challenging to extract accurate depth information. Overall, stereo vision and monocular depth estimation have advantages and limitations, and the choice between the two depends on the application's specific requirements. For example, stereo vision reported more reliance on providing accurate and dense depth maps but required multiple cameras and careful calibration. In contrast, monocular depth estimation is a more adaptable approach that can be performed using a single camera. However, it could lead to less accurate and sparse depth information. Until recently, researchers have reduced this limitation by incorporating external data about the scene or by making simplified assumptions about the structure of the scene.

2.2 Stereo Vision

Stereo vision is a fundamental task for depth estimation, which involves using multiple cameras to capture a scene from multiple perspectives. The stereo vision is widely considered to be the most accurate representation of the natural depth perception process, given that it involves the integration of visual cues from both eyes. The traditional approach to stereo vision involves the computation of depth information through the identification of corresponding points in both the left and right images, followed by the utilization of the disparity values between them. In recent years, there has been extensive research on stereo vision for depth estimation, leading to the development of numerous new algorithms and techniques.

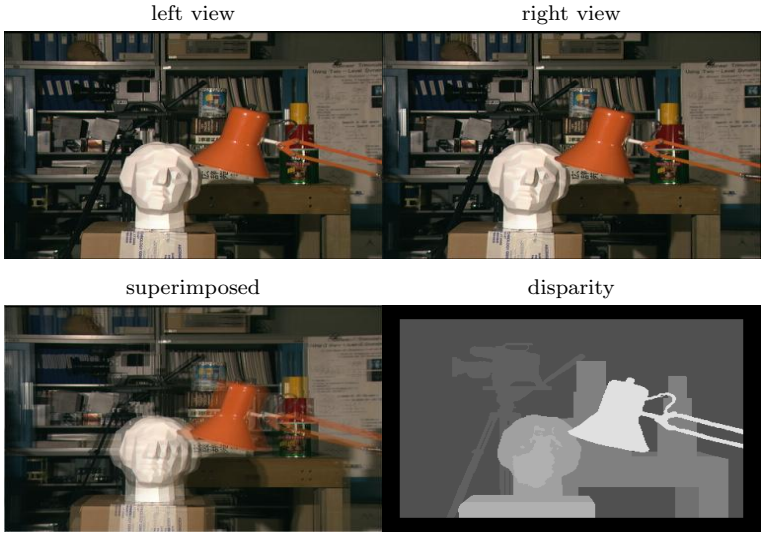


Figure 2.1: The Tsukuba stereo image pairs [1]

Initially, stereo-based depth estimation techniques relied on matching pixels across several images obtained through precisely calibrated cameras. The given scenario comprises a pair of cameras that possess established intrinsic and extrinsic properties. Given the predicted disparity, simple geometry is then used to reconstruct the missing depth dimension during image capture.

The stereo-matching technique is the conventional approach for determining the disparity map for a pair of rectified stereo images. This technique computes the correspondence between the pixels of the left and right images by comparing their pixel-neighbourhood information for both images. The disparity map is constructed using a stereo pair. There are a variety of stereo-matching algorithms, including local methods such as SIFT [38], SURF [39], and ORB [40] and global methods such as semi-global matching and graph cuts.

Stereo-matching algorithms aim to estimate a scene's depth given two images taken from different points of view. The Tsukuba stereo image pairs [1] as shown in Fig 2.1 have been widely used as a benchmark dataset to evaluate the performance of such algorithms. These image pairs consist of a left and a right image, captured from slightly different viewpoints, representing a scene with objects at different depths.

Although these techniques can achieve good results, they still need to be improved in many aspects. For instance, these methods are not well-suited for addressing occlusions, featureless regions, or highly textured regions with repetitive patterns. Nevertheless, it is noteworthy that humans can effectively address such ill-posed inverse problems by leveraging prior knowledge. We can easily infer the approximate sizes of objects, their relative locations, and even their approximate relative distance to our eyes.

The second generation of stereo methods tries to leverage this prior knowledge by formulating the problem as a learning task. In contrast to traditional stereo matching algorithms, which rely on manually crafted features and matching costs, deep learning-based stereo methods learn feature representations and matching functions through extensive training on large datasets. This stereo method enables the model to effectively apprehend complex interconnections and leverage advanced contextual data to enhance the accuracy of disparity estimation.

One common method used in deep stereo is matching cost convolutional neural networks (MC-CNN)proposed by Zbontar et al. [41]. They learn deep features by extracting depth information from a rectified image pair. One notable aspect of MC-CNN is its use of a siamese architecture. This architecture encourages the network to learn shared representations and similarities between the images, improving the depth estimation accuracy.

The authors evaluate MC-CNN on various benchmarks, including the Middlebury stereo dataset. The results demonstrate that their relatively simple convolutional neural network achieves competitive performance in terms of error rates. Hence, those relatively simple CNN methods demonstrate the high potential of modern deep learning approaches to solve the classic stereo vision computer vision problem, and there is still a great deal of space for improvement in this area.

2.3 Monocular Image

Single image depth estimation, also known as monocular image, is a challenging problem in computer vision because depth perception typically requires multiple viewpoints or depth cues, which are inherently missing in a single image. The objective of single image depth estimation is to estimate the depth map of a scene from a single RGB image without requiring additional information about the scene, such as stereo pairs or motion sensors.

Single image depth estimation is a fundamental problem in computer vision, with applications in robotics, autonomous driving, and augmented reality. Accurate depth information can enable robots and autonomous vehicles to navigate their environments, whereas augmented reality applications can use depth information to place virtual objects in the real world correctly.

Before the development of deep learning, researchers estimated depth from a single image using traditional computer vision techniques. These methods leveraged various cues and assumptions, such as perspective, texture gradients, shading, and semantic priors. However, these techniques often need to be improved in handling complex scenes, dealing with noise and ambiguities, and capturing global context. Deep learning-based approaches have shown significant improvements in overcoming these limitations and achieving more accurate depth estimation results.

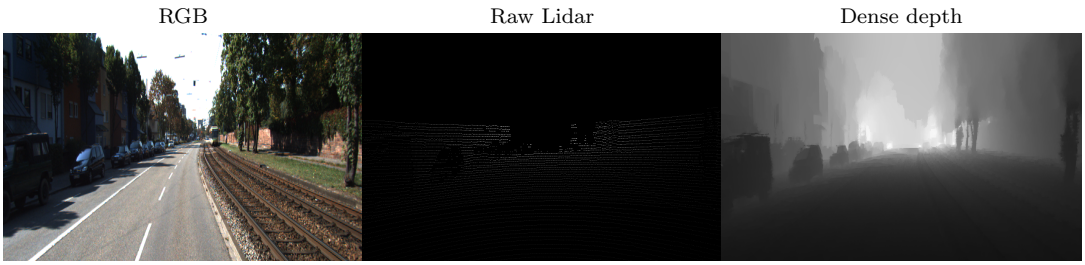


Figure 2.2: Monocular depth of KITTI

Single image depth estimation using deep learning can be achieved using various architectural models,such as encoder-decoder, generative adversarial networks, and transformer attention based model. Each of these architectural paradigms provides unique advantages and can be tailored to the specific requirements and challenges of

single image depth estimation tasks. These models have shown great performance on various datasets, including NYU Depth v2 [42] , and KITTI data [43].

However, using deep learning, estimating depth from a single image faces some challenges. The accuracy of depth estimation heavily relies on the quality and diversity of the training data. Handling occlusions, textureless regions, and depth ambiguities can also be challenging. Furthermore, the estimated depth may be subject to scale ambiguity because absolute depth values cannot be determined directly from a single image.

The outdoor KITTI and indoor NYUare benchmark datasets used for depth estimation research. The KITTI dataset focuses on autonomous driving scenarios and provides stereo image pairs along with accurate depth information. The NYU depth dataset, on the other hand, consists of RGB-D images captured from indoor scenes. It includes synchronized RGB and depth information obtained from depth sensors with different lighting conditions, object layouts, and room dimensions. These datasets serve as important benchmarks for evaluating and comparing depth estimation algorithms.

This study focuses on deep learning-based methods, which have been shown to effectively address any challenges associated with single image depth estimation tasks. In addition, we will discuss some limitations and prospective research directions for single-image depth estimation.

Chapter 3

Convolution Neural Networks in the Context of Monocular Depth Estimation

Estimating depth from a single image is a fundamentally challenging task and a significant area of computer vision research. Convolutional neural networks (CNN) have enabled significant improvements in acquiring depth information from a single image.

Artificial intelligence has grown tremendously in bridging the gap between human and computer capabilities. In order to achieve outstanding results, many researchers are focusing on multiple aspects of the field. Computer vision is only one of many such fields.

The objective of this field is to enable machines to perceive and comprehend the world similarly to humans and to leverage this comprehension for a diverse range of applications such as image and video recognition, image analysis and classification, media recreation, recommendation systems, and natural language processing. Convolutional neural networks (CNN) are the primary method used to build and refine breakthroughs in computer vision using Deep Learning.

CNN has been widely used for monocular depth estimation. These models discover how to predict depth maps directly from input images by minimizing a depth loss function. The basic idea behind CNN for depth estimation is to train a neural network to predict the depth of a scene from a single image by learning the relationship between image features and depth. Various approaches have accomplished remarkable improvements in extracting depth information using deep Neural Networks. It can be categorized according to its architectural design and functionality. Three common categories for CNN networks include encoder-decoder, generative, and attention models.

3.1 Encoder-Decoder Based Models

Encoder-decoder architecture is widely used for single image depth estimation. As shown in Fig 3.1, the basic structure of this model consists of two major components: an encoder and a decoder. The encoder processes the input image by progressively decreasing its spatial dimensions and extracting high-level features. Using these features, the decoder reconstructs the original image size and generates a depth map as the output. In addition, researchers have studied various architectural and training

techniques to enhance the performance of these models.

The seminal work of estimating depth using deep learning was introduced by Eigen *et al.* [2]. Their work estimated depth information from a monocular image using a multi-scale structure that stage-wisely refines the estimated depth map from low to high spatial resolution via independent networks. Building on this foundation, Liu *et al.* [18] discover the unary and pairwise potential of continuous Conditional Random Field (CRF) and train it using a CNN. Laina *et al.* [19] proposed a fully convolutional architecture to learn feature map up-sampling to generate higher resolution output dense maps. Godard *et al.* [10] proposed remarkable single image depth estimation Considering unsupervised learning for monocular depth estimation by constructing disparity images using a left-right consistency image reconstruction loss.

Later then, several notable works have proposed to enhance the quality of depth information using deep learning. Chen *et al.* [21] introduced Pyramid decoder (RPD) that takes into account the underlying image structure at many scales for estimating depth. Yin *et al.* [22] introduced a framework that consists of two primary modules; a depth prediction and a point cloud module, to improve the structure of point clouds derived from depth map. Gur *et al.* [23] proposed a deep learning-based method to estimate depth from a single image based on depth focus cues. Bian *et al.* [6] proposed an Auto-Rectify network to enhance unsupervised depth estimation by removing relative rotational motions in addition to their innovative loss functions.

They have demonstrated that deep neural networks can capture complex depth structures and provide accurate depth predictions from a single input image. Subsequently, numerous studies on improving the quality of in-depth data through deep learning have been conducted.

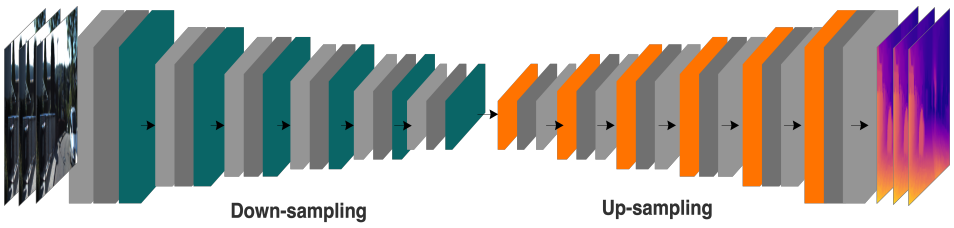


Figure 3.1: Basic structure of encoder-decoder network.

3.2 Adversarial Based Models

The objective of adversarial models is to discover the input data's underlying distribution and generate new samples similar to the training data. Generative adversarial networks (GAN) [25], also known as two-player deep learning networks, have been investigated for depth estimation. As depicted in Fig 3.2, GAN consists of a generator network that learns to generate a realistic depth and a discriminator network that attempts to distinguish between real and generated depth.

To mention a few, Aleotti *et al.* [44] introduced monocular depth estimation based on the intrinsic ability of GAN to detect inconsistencies in images. In their research, the generator network learns to estimate depth from the reference image to generate a warped target image. Simultaneously, the discriminator learns to differentiate between generated depth and target ground truth.

Several more studies are then presented to improve depth estimation based on the GAN-based feature-level consistency. Zheng *et al.* [45] proposed a two-module domain adaptive network with a generative adversarial loss to map real and synthetic images to the real domain. Kumar *et al.* [46] presented an adversarial network-based model in which their generator network consists of depth and relative object pose in addition to their adjustable loss functions. Pilzer *et al.* [47] and Kwak *et al.* [26] explored unsupervised deep learning depth generation based on a cycled generative adversarial network.

Furthermore, Zhao *et al.* [48] developed a Masked GAN framework comprised of two separate networks for monocular depth estimation and ego-motion utilizing their scale-consistency loss.

Integrating GAN into depth estimation frameworks has opened up new avenues for exploring generative models in this field.

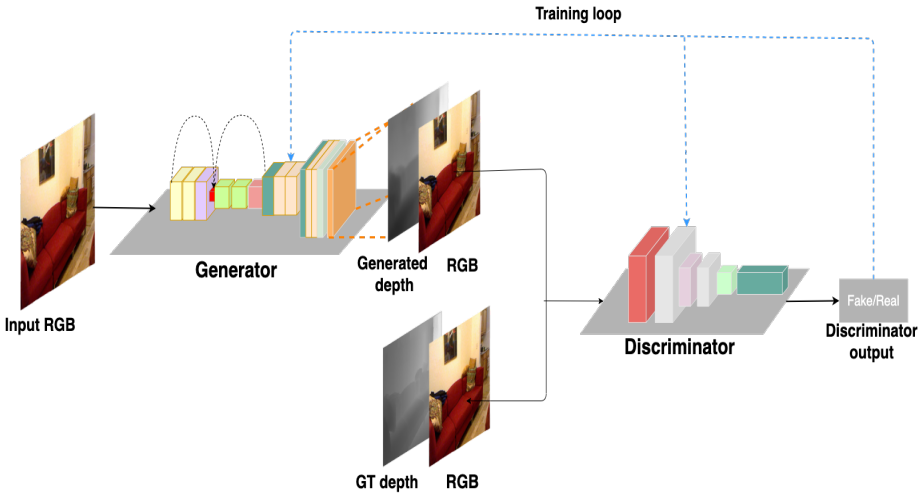


Figure 3.2: Basic structure of generative adversarial network (GAN)

3.3 Attention Based Models

Attention models focus on learning to selectively attend to specific regions or features in an input image. These models assign weights or probabilities to different image regions, enabling them to dynamically allocate attention to informative regions during processing. Attention mechanisms allow the network to adaptively focus on relevant areas, enhancing the network's ability to extract meaningful information and improve performance on various vision tasks. Transformer is a common attention-based model used for image generation tasks, such as image depth estimation.

In Fig 3.3, we show the fundamental structure of the transformer model. This model uses the self-attention mechanism to capture both global and local features. Multiple studies have been proposed to enhance the depth prediction of a single image using the transformer model.

One of the early work utilizing transformer for a single image depth estimation is the dense prediction transformer (DPT) proposed by Ranftl et al [49]. Their architecture, a transformer backbone inside the encoder-decoder design for fine-grained output, has been trained on a large-scale mixed depth dataset that covers a wide reange of scenes.

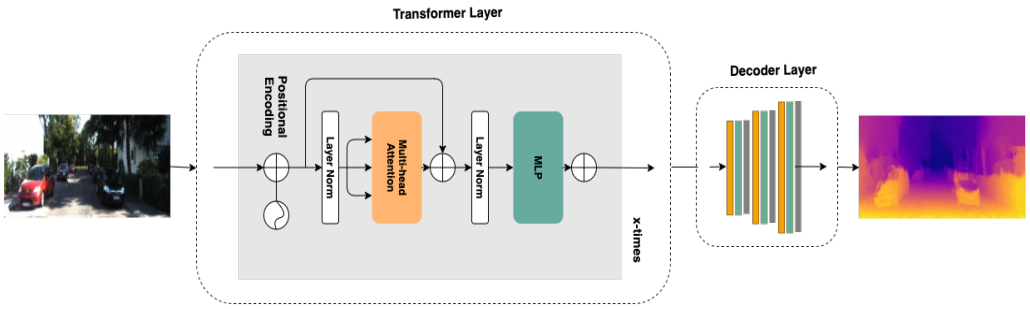


Figure 3.3: Basic structure of transformer network.

Another approach is the inverted pyramid transformer proposed by Ye et al. [24]. an Up-transformer block is presented to allow network to learn fine-grained dense prediction maps at higher resolution, for multi-task dense scene understanding.

In more recent work, Manimaran et al. [50] proposed Focal-WNet, a convolutional architecture along with a transformer layers to improve monocular depth prediction. Their architecture consists of two separate encoders and a single decoder.

These studies demonstrate the potential of transformer networks for capturing long-range dependencies and managing global context in single image depth estimation tasks.

Chapter 4

Proposed Methods

The proposed approach aims to develop a simple and reliable single image depth estimation method by integrating global and local image features. This integration leverages the strengths of both types of features to improve the accuracy and robustness of depth estimation.

4.1 Image Features

Image features are fundamental components in computer vision that facilitate various tasks, including object recognition, image segmentation, and scene understanding. These features serve as meaningful representations of visual information, enabling efficient analysis and interpretation of images. Image features refer to distinctive patterns or characteristics present in an image that capture essential visual information. These features play a crucial role in depth estimation tasks as they provide cues and clues about the spatial structure and relative distances between objects in the scene.

In depth estimation, image features can be categorized into two main types: local and global.

4.1.1 Global Features

Global features in depth estimation using deep learning refer to information extracted from the entire image or larger spatial regions. These features capture the global context and scene-level cues allowing for an understanding of the image's overall structure and relationships. Typically, convolutional layers extract global features and encapsulate an image's global representation. They can include scene architecture, object arrangement, and semantic comprehension, essential for accurately inferring depth. Global features are useful for capturing long-range dependencies and gaining a global understanding of the scene.

The proposed architectures utilize 1×1 convolution to decrease the dimensionality of the feature maps, thereby enabling the model to concentrate on capturing high-level semantics and global context. The aforementioned layers perform a linear transformation on the input feature maps, gather information from the entire image, and offer a comprehensive comprehension of the scene that incorporates global context.

4.1.2 Local Features

Conversely, local features capture information from smaller spatial regions or local patches within an image. These characteristics enable capturing local characteristics, such as fine-grained texture details, edges, and geometric structures, contributing to accurate depth estimation. Local features are frequently extracted from the shallower layers or intermediate representations. They help in capturing local context and details that might be missed by global features alone.

Incorporating contextual information from various spatial scales can be achieved by stacking convolutional layers in the model. Each layer in the structure is taught to extract and encode distinct local feature types. The early layers work on low-level features such as edges and textures, whereas the deeper layers capture more complex and higher-order features specific to the depth estimation task. These features can include depth discontinuities, depth gradients, or geometric structures indicative of the scene's depth variations.

Global and local information are then combined within the network to predict the depth values for each pixel in the image. The combination of global and local features enables the model to leverage both the overall scene understanding and fine-grained local details to make accurate depth predictions.

4.2 Encoder-Decoder Based Model

The first architecture uses an encoder-decoder-based model, employing a two-stage architecture, multi-task loss optimization, and loss monitoring for optimal learning rate adjustment based on loss behaviour [51].

4.2.1 Problem Formulation

We formulate our model depth estimation as a two-stage deep neural network. In the first stage, we employ a residual network as our base model for computing feature depth maps. Here, we modified the original form of ResNet, which in this work consists of only thirty-eight convolution layers wrapped in six residual blocks. This new network is a remodelled ResNet-50 to avoid ambiguity with the original form. In addition, we provide a 1×1 convolution to reduce the dimensionality of the feature maps, enabling the model to focus on capturing global context and high-level semantics. Then, we incorporate an up-sampling layer followed by two blocks of convolution layer. Finally, we repeat this setting to handle the desired output map size, which has been proven effective for regression problems.

The second stage of our architecture guides the network to enhancing its output through a stack of five convolution layers. One of the main differences compared with [2] is the number of convolution layers before concatenating the two inputs (RGB image and initial output depth). We implement two convolution blocks with stride-2 rather than a convolution stride-2 followed by a max-pooling layer. In our second stage network, we downscale the input RGB two times using the stride-2 convolution to match the initial generated depth size. These settings will enable our network to reduce the blur result of our reconstruction depth. Further, we reduce the kernel size from 9×9 to 7×7 to ease the process time computation.

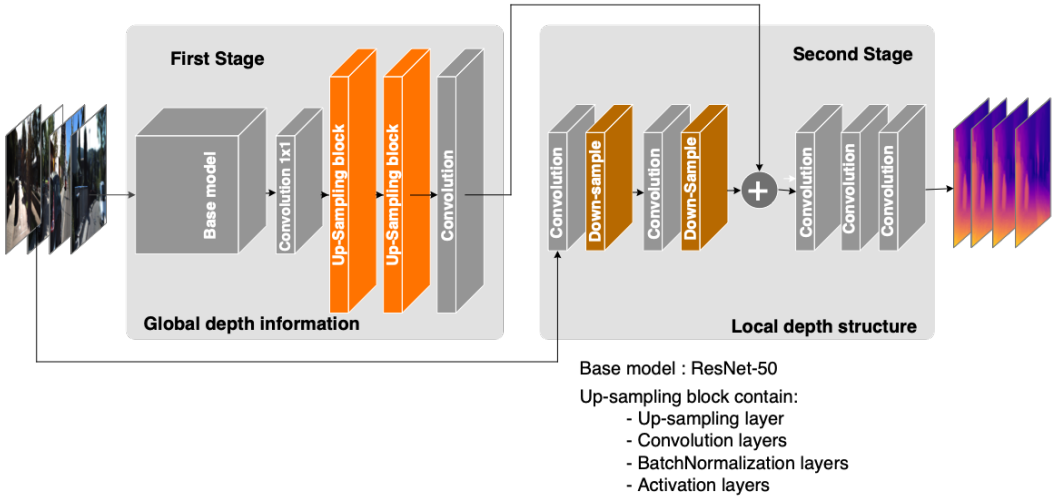


Figure 4.1: Outline of our proposed encoder-decoder depth method.

4.2.2 Network Architecture

The network architecture of our depth estimation model is illustrated in Fig. 4.1. The first part of the network is a smaller remodelled of ResNet-50 and initialized with random weights. Thirty-eight convolution layers are employed in the residual block to encode the RGB input image into a feature vector. A max-pooling layer, together with the first convolution layer with stride-2, is utilized to downscale the input image by a factor of 4. Then we pass this output into a residual block and repeat the operation six times to downscale the previous output by a factor of 2. Two stack pairs of up-sampling and convolution layers are then provided before fusing the residual network's output to the second stage network. The output size of the last residual layer will be 1/4 compared to the input RGB image.

In the second stage, we stack five blocks of the convolution layer. A stride-2 convolution is provided in the first and second block to downscale the input by a factor of 4 to match the output size from the previous stage. We use 7×7 filter size in the first and second blocks and 5×5 in the remaining three blocks. The rectified linear unit (ReLU) activation function is utilized to the output of each convolution layer but in the last layer. The convolution with the filter number of one is applied in the last layer, followed by a linear activation function to refine the feature depth maps into dense depth.

4.2.3 Multi-loss Function

The standard loss function for depth regression problems considers the difference between the ground truth depth map y_t and the depth regression network y_p prediction. The most common loss function used in deep learning for regression tasks is a mean square error or L_2 loss. These loss functions measure the average of the squared difference between prediction and actual observation. However, L_2 loss alone has insufficiency to predict data far from their real values.

In our work, we implement a combination of multi-task loss functions to improve the performance of the model. This idea was encouraged by the practice from work in [52] to compute loss based on pixel-wise, gradient-based, and structured simi-

larity. These combination losses have proven beneficial for neural network learning optimization and demonstrated adequate depth performance.

However, differing from their work, we utilized the Reversed Huber in Eq. (4.1), which is more flexible than MAE loss that can behave either as a mean absolute error (MAE) or mean squared error (MSE) depends on the prediction error result.

$$L_h = \begin{cases} |y_t - y_p|, & |y_t - y_p| \leq c, \\ \frac{(y_t - y_p)^2 + c^2}{2c}, & |y_t - y_p| > c, \end{cases} \tag{4.1}$$

where $c = 0.2 \max(|y_t - y_p|)$.

Equation (4.2) presents a loss function which estimate prediction error considering its gradient direction. The gradient of ground truth depth and the depth prediction is computed along x, and y direction then calculate its average.

$$L_g = \frac{1}{N} \sum_{p}^{n} |g_x(y_t, y_p)| + |g_y(y_t, y_p)|, \tag{4.2}$$

where g_x and g_y are the gradient along x and y, respectively.

The third loss function we used in this research is a structure similarity index (SSIM) loss, as presented in Eq. (4.3). While L_h and L_g compute error for each pixel between ground truth and generated depth, L_s measure similarity within pixels in the score range $[-1, 1]$. Eventually, the SSIM loss will compute the perceptual difference based on the visible structure of the ground truth and predicted image.

$$L_s = 0.5 - \frac{\text{ssim}(y_t, y_p)}{2}, \tag{4.3}$$

where $\text{ssim}(y_t, y_p) = \frac{(2\mu_{y_t}\mu_{y_p} + C_1)(2\sigma_{y_t y_p} + C_2)}{(\mu_{y_t}^2 + \mu_{y_p}^2 + C_1)(\sigma_{y_t}^2 + \sigma_{y_p}^2 + C_2)}$.

With:

μ_{y_t} and μ_{y_p} are the average of y_t and y_p, respectively.

$\sigma_{y_t y_p}$ is the covariance of y_t and y_p.

$\sigma_{y_t}^2$ and $\sigma_{y_p}^2$ are the variance of y_t and y_p, respectively.

We define the loss function for training our model as the weighted sum of the three loss functions defined in Eqs. (4.1), (4.2), and (4.3) as:

$$L_M = w_1 L_h + w_2 L_g + w_3 L_s, \tag{4.4}$$

where value of $w_1 = w_2 = 1$ and $w_3 = 0.1$ were arbitrarily initiate by experimentation.

4.2.4 Learning Rate Adjustment

One of the most challenging tasks of training deep learning neural networks comprises carefully deciding a reasonable learning rate. The learning rate is one of the critical hyper-parameters in the model, manages the speed of the neural network model to update their weights. On that account, it is necessary to provide a decent learning rate value to improve the performance while the model is training the data.

Unfortunately, the optimal learning rate cannot determine theoretically. This parameter must be observed by experimenting. Typically, a big learning rate enables the model to learn faster and converge to its local minima solution. In contrast, an extremely small learning rate may raise a lengthy training process, or the training

process may get stuck. In this works, we originate our initial learning rate value to 10^{-3} then reduce our learning rate depend on the loss behavior.

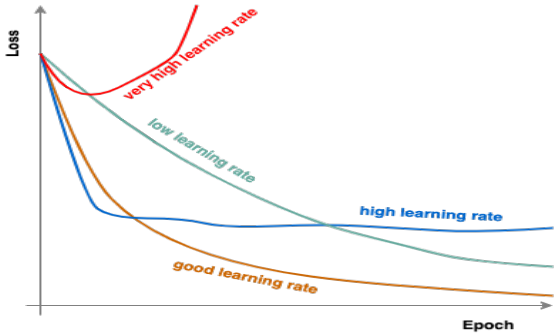


Figure 4.2: Learning rates hyper-parameter

4.2.5 Training Details

We implement our proposed approach using the Keras framework [53] with the Tensorflow backend. Training is done on Ubuntu 16.04 with an NVIDIA GeForce GTX 1080 GPU with 8 GB memory. The network is shown in Fig. 4.1 has been trained using initialized random weight with approximately 12 million trainable parameters. The training is performed using 16 mini-batches and load images and their depth using an online generator for GPU memory performance.

We train our model using an adaptive moment estimation (Adam) optimizer and an initial learning rate of 10^{-3}. During training, we monitor the model's performance and adjust the learning rate hyperparameter depending on the loss behavior. Specifically, we degraded our learning rate by order of magnitude divided by ten on epoch 45, 60, and 75, respectively. For some other details, we trained our proposed architecture on the multi-task loss function, as described in previous section.

4.3 Adversarial Based Model

The generative adversarial network (GAN) has significantly improved the learning of mapping high-dimensional data distributions. It has been demonstrated that a generative adversarial network is highly effective at capturing the global structure of a scene and producing realistic images. In the adversarial network, the generator model (G) is responsible for reconstructing newly synthesized images. At the same time, the discriminator (D) evaluates the probability that a given input image is either derived from training data or is synthetically generated.

We propose a three-player GAN (TP-GAN) that, when combined with the structural similarity measure (SSIM) loss, improves the depth estimation performance of a single image [54]. Our study examines the advantages of including an additional sub-model to the cGAN architecture. The broad idea here is to employ a refiner to improve depth prediction by incorporating generator output and discriminator feedback. Our strategy concurrently integrates global scene structure and local scene information to enhance the performance of the adversarial network for a single image depth estimation.

4.3.1 Network Architecture

In this research, we utilize adversarial learning advantages to formulate the problem of learning depth from monocular inputs as an image translation problem. While the discriminator discovers how to distinguish between ground truth and synthetic depth maps, the generator learns how to create more realistic depth maps. In fact, the generator continuously seeks the output that appears plausible to the discriminator.

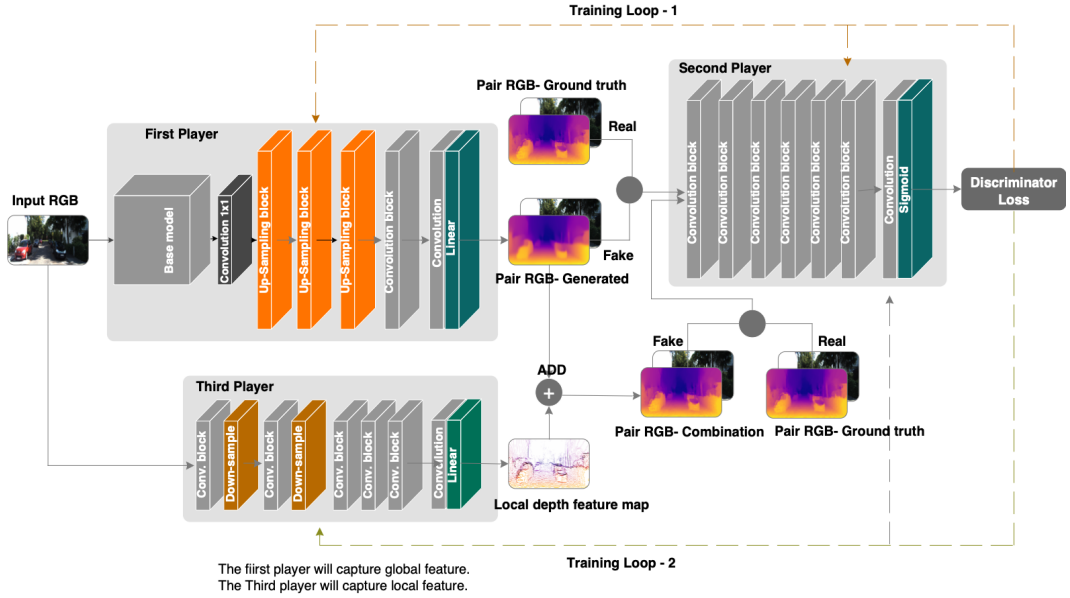


Figure 4.3: Outline of our proposed adversarial based model.

Our proposed adversarial model is a conditional generative adversarial neural network (cGAN) to assist the generator and refiner in mapping input images to their respective depth representation. This network consists of three sub-models: a generator as the first player, a discriminator as the second player, and an additional refiner sub-model that we refer to as the third player. The refiner will be responsible for fine-tuning the locally generated depth prediction with the global scene information.

The proposed technique updates the generator weight by back-propagating through the discriminator during adversarial training. Meanwhile, the refiner combines the updated weight of the generator and then forwards it to the discriminator model for each mini-batch training. Further details are discussed in the succeeding subsections.

The 1st Player: Generator

We reconfigured the residual network (ResNet) [30] structure in the generator as our backbone model, which has been demonstrated effective in improving the accuracy of depth prediction from a single image [6, 19, 51]. Then, we stacked some block layers to receive input from the previous layer; the first block is a convolution layer with 1×1 convolution kernels to enable feature transformation and dimensionality reduction while preserving spatial information. The remaining blocks consist of transpose-convolution (up-conv-activation), followed by regular convolution block (conv-batch-activation) with {1024, 512, 256, 128}, and 64 filters, respectively. We utilized bilinear interpolation for our up-sampling, while Leaky ReLU activation was

employed to minimize the vanishing gradient. The final depth extraction output layer has a linear activation function.

The 2nd Player: Discriminator

This discriminator model is encouraged by the work of Isola *et al.* [31], implemented as a patch GAN, which looks at the structure of local image patches and classifies each patch in an image as real or fake in the $N \times N$ output vector rather than the entire image level. Since the generator output is conditioned on the input, it is important to maintain the discriminator input image in the mix. Our discriminator, a conditional adversarial model, comprises pair of images as input: the RGB image and its ground truth depth and the RGB image and its corresponding generated image depth. Each of which is size 48×64 for NYU and 40×128 for KITTI data.

We concatenate the RGB with its depth before fusing them into the network. We modify parameter values using 4×4 kernel size and strides-2 except in the last two layers with {64, 128, 256, 512, 512, 1} filters, respectively. Batch-normalization is applied in all layers but in the first and last layers. At that, in the last layer, the convolution is utilized to map to a one-dimensional output with a size of 3×4 pixels, followed by a sigmoid activation function. The model output will be a probability of classifying whether the input patch images come from training or generated data.

The 3rd Player: Refiner

The refiner model in our architecture is a sequence of six block layers. The first five blocks are a stack of convolution, batch normalization, ReLU activation, and dropout regularization (conv-batch-activation-dropout) to handle the common over-fitting problem with 64 filters. We use 7×7 kernel size and strides-2 to down-sample our input in the first and second blocks, while the following three blocks use 5×5 kernel size and stride-1. With a small enough kernel size relative to the input, the extracted feature will not depend on the value of the whole pixel in the input image. Since the receptive field is smaller than the size of the input image, extracted features will only depend on the local pixels. The last block is a convolution layer with a filter number of one and 5×5 kernel size, followed by a linear activation to capture the depth of local features.

4.3.2 Depth Reconstruction Loss

The discriminator is trained to maximize the predicted probability of real images and the inverted probability of deceptive images throughout training. The generator, on the other hand, works to maximize the log of the predicted probability of discriminator for counterfeit images. In addition, the refiner utilizes to improve the generator result as feedback from the discriminator. We set our depth reconstruction loss in Eq. (4.5).

$$\begin{aligned} \min_{G,R} \max_{D} (G, R, D) = \mathbb{E}_{x,y}[\log D(x, y)] \\ + \mathbb{E}_x[\log(1 - D(x, G(x)))] \\ + \mathbb{E}_x[\log(1 - D(x, R(x, G(x))))], \end{aligned} \tag{4.5}$$

where $D(x, y)$ is the discriminator from the input RGB image x with conditional target depth image y. $G(x)$ is the generator output when given input data x, and $R(x, G(x))$ is the refiner output that comes from the generator and real data x.

4.3.3 Structural Similarity Index Measurement (SSIM) Loss

In general, the Mean Squared Error (MSE) or Mean Absolute Error (MAE) is taken as the standard loss for regression tasks to calculate the discrepancies between prediction and target outputs. Similar to MAE, MSE computes the error between two images by comparing pixel by pixel as defined in Eq. (4.6). On the other hand, the Structural Similarity Index (SSIM) measurement analyzes the structural difference between two images. This structural information signifies the idea that neighboring pixels have strong inter-dependencies with one another, which is a more effective strategy for image reconstruction tasks.

$$\mathrm{MSE}(y_t, y_p) = \frac{1}{N} \sum_{y_p \in |N|} |y_p - y_t|^2 \tag{4.6}$$

The SSIM formula, as expressed in Eq. (4.7), was introduced by [55], which comprises three parameter comparison measurements: luminance, contrast, and structure.

$$\mathrm{SSIM}(y_t, y_p) = \frac{(2\mu_{y_t}\mu_{y_p} + c_1)(2\sigma_{y_t y_p} + c_2)}{(\mu_{y_t}^2 + \mu_{y_p}^2 + c_1)(\sigma_{y_t}^2 + \sigma_{y_p}^2 + c_2)} \tag{4.7}$$

In contrast to the MSE or MAE, the SSIM score range from -1 and 1, with 1 indicating perfect similarity. We define our SSIM loss (L_s) as expressed in Eq. (4.8) for our generator and refiner while training our adversarial network. Eventually, the SSIM loss will compute the perceptual difference based on the visible structure of the ground truth and predicted image.

$$L_s = 0.5 - \frac{\mathrm{SSIM}(y_t, y_p)}{2}, \tag{4.8}$$

With:
μ_{y_t} and μ_{y_p} are the mean of y_t and y_p, respectively.
$\sigma_{y_t}^2$ and $\sigma_{y_p}^2$ are the variance of y_t and y_p, respectively.
$\sigma_{y_t y_p}$ is the covariance of y_t and y_p.
c_1 and c_2 are constants represented by $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$, respectively.
L is the data range of the input image.
$k_1 = 0.01$ and $k_2 = 0.03$ following [55].

4.3.4 Training Details

We implement our adversarial depth estimation network based on deep learning Tensorflow [56] and Keras framework [53]. The training is done on Ubuntu 16.04 and an NVIDIA GeForce GTX 1080 GPU with 8 GB memory. The network architecture shown in Fig. 4.3 has been trained using initialized random weight. The training is performed using 16 mini-batches and load images and their corresponding depth using an online generator for GPU memory performance.

In our approach, we train our model for 50 epochs using an adaptive moment estimation (Adam) optimizer with the exception of the discriminator, which uses Stochastic Gradient Descent (SGD) as motivated in the works [57]. We started with a learning rate of 2×10^{-4} for the generator and refiner, while for the discriminator, we initialized from 4×10^{-4} and periodically adjusted as the training progressed using an exponential rate decay of 0.5 and 0.999 for 1^{st} and 2^{nd} momentum, respectively.

Since our strategy is an adversarial model, the generator model was not trained independently and instead had its weight updated by the loss of the discriminator. On the other hand, the refiner model is updated by the previous generator weight as well as the discriminator feedback for every input batch.

Chapter 5

Experiments

In this chapter, we describe the implementation details of the depth estimation model. The first section covers the NYU and KITTI datasets that we used for training and testing. In this section we also describes how we processed data and how we used data augmentation to enrich the feature of the dataset. In the next section, we describe the evaluation metrics to evaluate performance of our approaches. The last section present our experimentation results including our model performance generalizes to other datasets. We examined the cross-dataset adaptation capabilities by training on one dataset and testing on another, and vice versa.

5.1 RGBD Datasets

This section describes RGB-D datasets that are commonly used to benchmark the performance of a system for depth estimation. We train our model using two popular publicly available depth datasets, indoor NYU Depth v2 [42] and outdoor KITTI data [43], commonly used in the area of depth estimation.

KITTI data contains outdoor scenes with images resolution of roughly 376×1241 captured by cameras and depth sensors in a driving car. It contains over 93K depth maps from 56 scenes with corresponding raw LiDAR scans and RGB images. We train our method on 25K images from the random scene and set depth upper bound of 80 meters. We test our model on the 697 images which are not included in the training, following the split by the work of Eigen *et al.* [2].

Figure 5.1: Sample images on KITTI dataset from random scenes

Figure 5.2: Sample input KITTI data and ground truth.

NYU Depth v2 is an indoor dataset gathered using a Microsoft Kinect camera with a resolution of 640×480. It contains about 120K raw RGB images and their corresponding depth from 464 different scenes. Only 50K images from random scenes are used for training our network. To validate the performance of our method, following the works of Eigen *et al.* [2], we test on 654 from 1449 available densely labeled pairs of aligned RGB and ground truth depth images in the maximum depth of 10 meters.

Figure 5.3: Sample images on NYU depth v2 dataset from random scenes

Figure 5.4: Sample input NYU data and ground truth.

5.2 Data Augmentations

Data augmentation helps minimize overfitting by diversifying the training samples and discouraging the model from relying on particular image features. Using data augmentation, single-image depth estimation models can learn from a more diverse and

representative set of training samples. Hence, this results in improved generalization, enhanced robustness, and increased accuracy in depth estimation tasks.

We practice on-the-fly data augmentation procedures to enrich the features of our inputs during training as shown in Fig. 5.5.

Randomizing channels, the purpose of randomizing channels is to augment the training data by introducing color variations that the model should learn to be invariant to. This helps the model generalize better to images with different color representations.we randomize the channels of the input RGB images using a ratio of 0.5.

Poisson noise, adding Poisson noise to the input images can improve the model's resilience to noise and improve generalization. It simulates the variability and noise present in real-world depth sensors or imaging conditions. We apply a 0.25 ratio to our input RGB images to implement Poisson noise addition.

Horizontal flip, This augmentation involves flipping the image horizontally along the vertical axis. It helps create additional training samples by generating mirror reflections of the original images. Horizontal flip is particularly useful when the scene or objects in the depth images exhibit left-right symmetry. A horizontal flipping strategy is applied at a probability of 0.25 for both RGB images and depths.

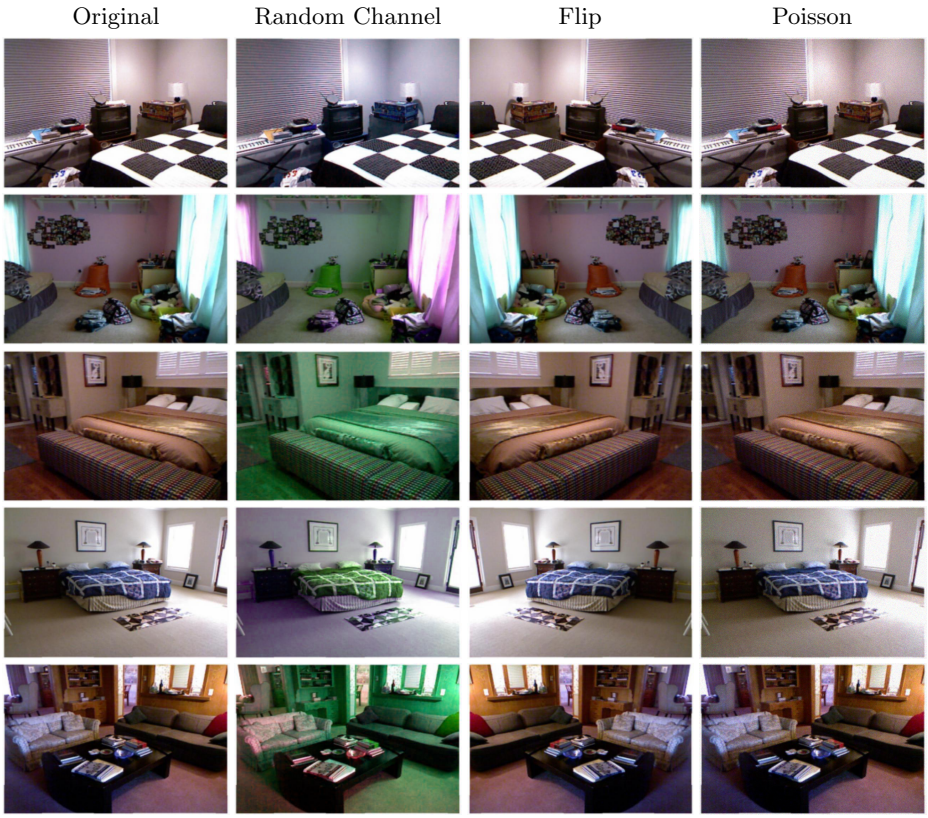


Figure 5.5: Sample image augmentation on NYU depth v2 data from left to right: RGB, randomize channel, horizontal flip, and poisson noise.

5.3 Evaluation Metrics

We validate the performance of our proposed depth estimation method on publicly available RGB-D NYU Depth v2 and KITTI datasets by evaluating our model compared with several relevant studies. In order to objectively assess the efficacy of our depth prediction model, we employ the following error rate and accuracy evaluation metrics, which have been widely employed in prior research. In addition, we implement Structural Similarity Index (SSIM) as an additional metric to provide valuable insights into the quality and perceptual accuracy of the estimated depth maps.

5.3.1 Error rate and Accuracy Performance

Specifically, we assess our method using metrics based on its error rate and accuracy in Eqs. (5.1), (5.2), (5.3), (5.4), (5.5), and (5.6).

- Root mean squared error (RMS):
  The standard deviation of the prediction errors to measure the difference between predicted (y_p) and the ground truth data (y_t).

$$\sqrt{\frac{1}{N}\sum_{y_p \in |N|} |y_p - y_t|^2}. \tag{5.1}$$

- Average $\log_{10}$ error (LOG10):
  The average of the absolute error of the log-transformed predicted (y_p) and log-transformed ground truth values (y_t).

$$\frac{1}{N}\sum_{y_p \in |N|} |\log_{10}(y_p) - \log_{10}(y_t)|. \tag{5.2}$$

- Average relative error (REL):
  The ratio of the absolute error of the predicted (y_p) to the ground truth (y_t).

$$\frac{1}{N}\sum_{y_p \in |N|} \frac{|y_p - y_t|}{y_t}. \tag{5.3}$$

- Root mean squared log error (RMS LOG):
  The Root Mean Squared Error of the log-transformed predicted (y_p) and log-transformed ground truth values (y_t).

$$\sqrt{\frac{1}{N}\sum_{y_p \in |N|} |\log y_p - \log y_t|^2}. \tag{5.4}$$

- Squared relative error (SQ REL):
  The ratio of the squared error of the predicted (y_p) to the ground truth (y_t).

$$\frac{1}{N}\sum_{y_p \in |N|} \frac{|y_p - y_t|^2}{y_t}. \tag{5.5}$$

- Accuracy with threshold (P_{th}): percentage (%) of y_p to $\max(\frac{y_t}{y_p}, \frac{y_p}{y_t}) = \delta < P_{\text{th}}$, where:

$$P_{\text{th}} \in \left\{1.25, 1.25^2, 1.25^3\right\}. \tag{5.6}$$

Here, y_p and y_t are the predicted and ground-truth depth, respectively, and N is the total number of pixels. With the exception of the accuracy with threshold, lower numbers indicate higher performance for all metrics.

5.3.2 Structural Similarity Index (SSIM) Metrics

While the previously mentioned depth-specific metrics are are more commonly used to evaluate depth estimation tasks, SSIM offers a different perspective that can be beneficial in understanding the performance of the depth estimation model.

In recent years, researchers have recognized the significance of incorporating perceptual quality assessment into the evaluation of depth estimation models. In this context, the SSIM has emerged as a valuable supplementary metric that complements depth-specific measures. SSIM is widely employed as an image quality assessment metric and has demonstrated a strong correlation with human perception of image similarity.

By implementing SSIM as an additional evaluation metric for single image depth estimation, we gain valuable insights into the perceptual quality of the generated depth maps. The SSIM formula, as expressed in Eq. (4.7), measures the structural similarity, luminance, and contrast information between the predicted and ground truth depth maps, offering a comprehensive evaluation of their visual resemblance.

Incorporating SSIM as an additional metric for single image depth estimation provides a more comprehensive evaluation aligned with human perception. It enables us to identify regions in the estimated depth maps where perceptual differences may exist, thus facilitating a deeper understanding of the model's performance. Moreover, it highlights the model's capability to capture intricate details, edges, and textures, which are important for achieving visually convincing depth maps.

5.4 Ablation Studies

The ablation study comprehensively analyzes the performance of models trained with different tasks. Our research presents the quantitative evaluation metrics for different tasks and discusses their implications.

We conducted comprehensive ablation studies to assess the impact of utilizing a combination of loss functions on our encoder-decoder model's performance, specifically incorporating three distinct loss functions: Huber loss, gradient loss, and Structural Similarity Index (SSIM) loss.

Our study assessed the model's performance when trained with each loss function individually and with the combination (multi) of Huber, gradient, and SSIM losses. We retrained our encoder-decoder model for 40 epochs, maintaining the same parameters except for the alteration in the loss function setting. We compute accuracy every epoch for each task and report the accuracy performance for the best epoch. This comprehensive approach enables us to assess whether the multi-loss functions result in the best model performance compared to using any one of them individually.

To offer a more comprehensive understanding, we have prepared an interactive visualization report accessible at the following URL: `https://tinyurl.com/mr4ddtfp`.

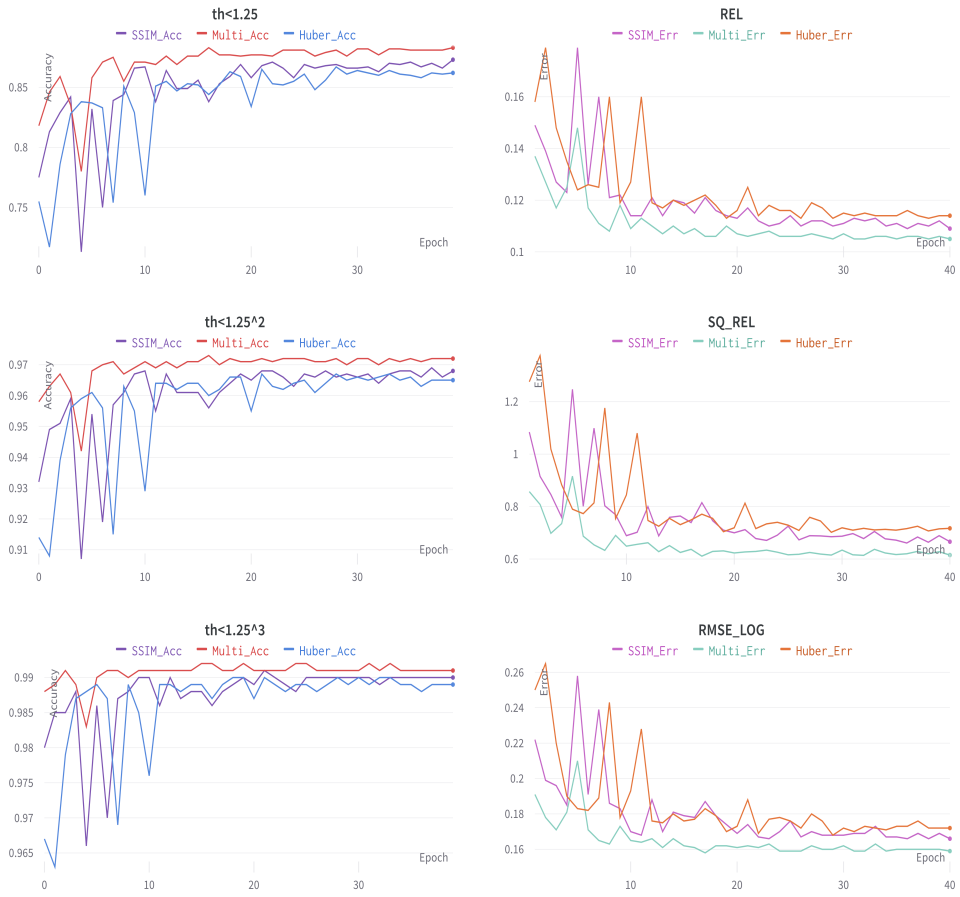


Figure 5.6: Training accuracy (1st column) and error performance(2nd column) comparison to evaluate the impact of utilizing a multi-loss functions.

An intriguing observation emerged from our experiments, highlighting the inadequacy of utilizing gradient loss as a sole loss function for single image depth estimation. While gradient loss was able to capture local structure information effectively, it lacks the ability to address perceptual aspects important for depth estimation accuracy comprehensively. However, our findings demonstrated that when incorporated as an additional loss function alongside other metrics like Huber loss and SSIM loss, gradient loss substantially enhances the model's ability to produce depth maps that are not only geometrically accurate but also perceptually meaningful.

As shown in Fig. 5.6 and Tab. 5.1, our findings emphasize the significance of a multi-loss approach, revealing that the integration of Huber, gradient, and SSIM losses

Methods	δ_1	δ_2	δ_3
Encoder-Decoder_Huber	0.867	0.967	0.990
Encoder-Decoder_SSIM	0.873	0.968	0.990
Encoder-Decoder_Multi	0.883	0.973	0.992

Table 5.1: Accuracy performance of different task utilizing Huber, SSIM and Multi-loss function.

offers a more comprehensive evaluation strategy for single image depth estimation models. This approach bridges the gap between numerical accuracy and perceptual quality, ultimately leading to improved depth estimation algorithms that align with human perception.

We also perform an ablation study to examine the proposed three-player adversarial with a non-adversarial model counterpart to discover the effectiveness of our proposed approach. We report the quantitative result in terms of accuracy in Tab. 5.2 on the outdoor KITTI dataset. We observe that the presence of the third sub-model improves the depth performance of the standard GAN model. Further improvement is found by utilizing Stochastic Gradient Descent (SGD) optimizer in the discriminator compared using adaptive moment estimation (Adam) to all sub-models. Finally, the TP-GAN model achieves greater improvement by utilizing the Structural Similarity Index Measure (SSIM) loss rather than the standard Mean Squared Error (MSE). In particular, the TP-GAN-ADAM-SGD-SSIM improves the standard GAN-ADAM-MSE accuracy by 3%, 1%, and 0.5% for the threshold $\delta < 1.25$, $\delta < 1.25^2$, and $\delta < 1.25^3$, respectively.

Table 5.2: Ablation study on the outdoor KITTI data

	Optimizers*				**Accuracy Thresholds****		
	(G)	(D)	(R)	Loss	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Standard-GAN-ADAM-MSE	ADAM	ADAM	—	MSE	0.854	0.963	0.987
TP-GAN-ADAM-MSE	ADAM	ADAM	ADAM	MSE	0.869	0.969	0.990
TP-GAN-ADAM-SGD-MSE	ADAM	SGD	ADAM	MSE	0.880	0.971	0.991
TP-GAN-ADAM-SGD-SSIM	ADAM	SGD	ADAM	SSIM	**0.884**	**0.973**	**0.992**

*(G) generator, (D) discriminator, (R) refiner.
**The higher the better.

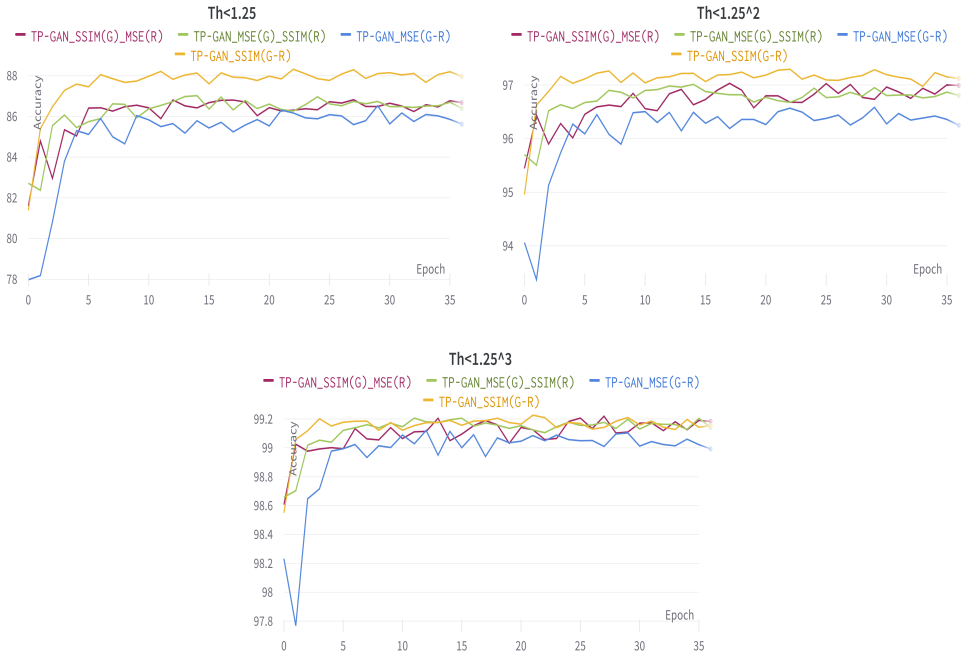


Figure 5.7: Training accuracy comparison to evaluate the influence of SSIM loss in our adversarial model.

Furthermore, we extended our study by conducting several additional ablation experiments. We re-trained our model over 35 epochs in these experiments to thoroughly investigate the implications of integrating the SSIM loss within the generator (G) and refiner (R) sub-models.

- TP-GAN_SSIM(G-R): Our model using SSIM loss in both (G) and (R).
- TP-GAN_SSIM(G)_MSE(R): our model using SSIM in (G) and MSE in (R).
- TP-GAN_MSE(G)_SSIM(R): our model using MSE in (G) and SSIM in (R).
- TP-GAN_MSE(G-R): our model using MSE loss in both (G) and (R).

The study reveal that employing SSIM loss for both the generator and refiner consistently yields improved accuracy, highlighting the compatibility of structural loss with the task and enhancing overall model performance. We have additionally assembled an interactive visualization chart that can be accessed through the designated URL: `https://tinyurl.com/42e3ph88`

Methods	δ_1	δ_2	δ_3	Best epoch
TP-GAN_SSIM(G-R)	0.88322	0.97295	0.99209	30
TP-GAN_SSIM(G)_MSE(R)	0.86821	0.97010	0.99219	28
TP-GAN_MSE(G)_SSIM(R)	0.86965	0.96941	0.99178	25
TP-GAN_MSE(G-R)	0.86526	0.96548	0.99103	30

Table 5.3: Accuracy performance on the best epoch for each task.

5.5 Experimental Results

The performance of our approach has been evaluated by conducting both quantitative and qualitative comparisons with several prior related methods on the publicly NYU and KITTI datasets.

5.5.1 Qualitative Performance

We present a comprehensive evaluation of the qualitative performance of our single model depth method, using RGB images taken directly from the original papers of related methods. The objective is to evaluate the algorithm's efficacy and emphasize its advantages over existing approaches.

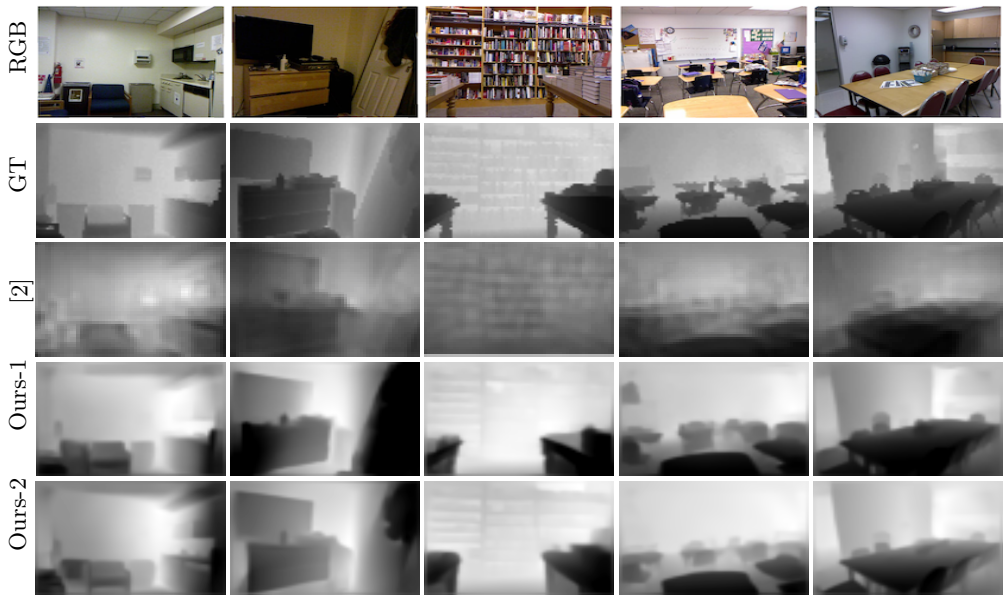


Figure 5.8: Depth Prediction on NYU Depth v2 from top to bottom: (a) RGB image, (b) ground truth, (c) Eigen *et al.* [2], (d) our encoder-decoder model, (e) our adversarial model.

Compared to [2] shown in Fig. 5.8 and 5.9, as well as [3] depicted in Fig. 5.10, our approach exhibits significant improvements in capturing fine details and subtle depth cues. Surface irregularities, object contours, and depth variations are more accurately represented in the depth maps generated by our algorithm, contributing to a more visually compelling and immersive representation of the scene's depth structure.

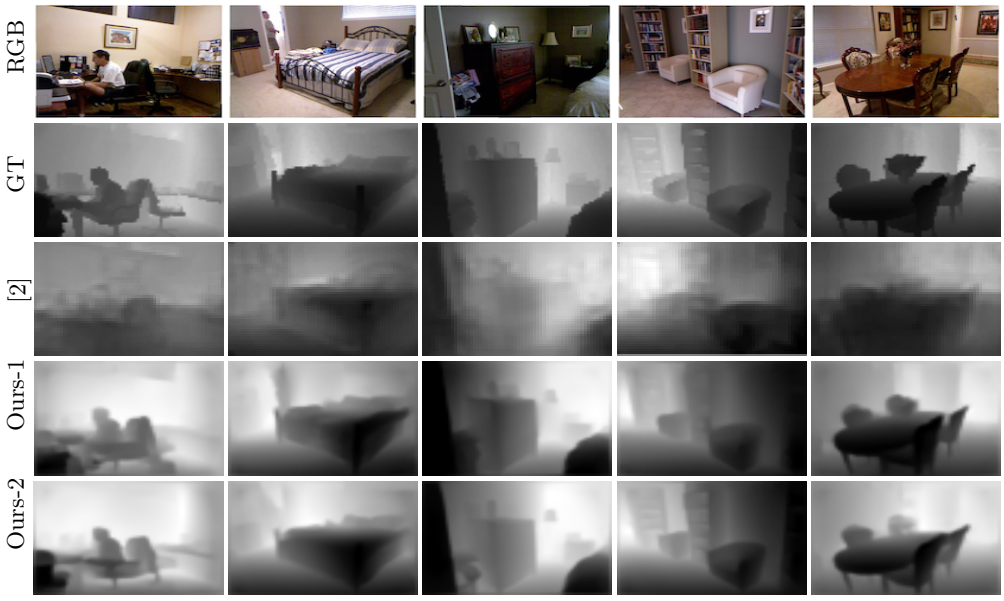


Figure 5.9: Depth Prediction on NYU Depth v2 from top to bottom: (a) RGB image, (b) ground truth, (c) Eigen *et al.* [2], (d) our encoder-decoder model, (e) our adversarial model (cont.)

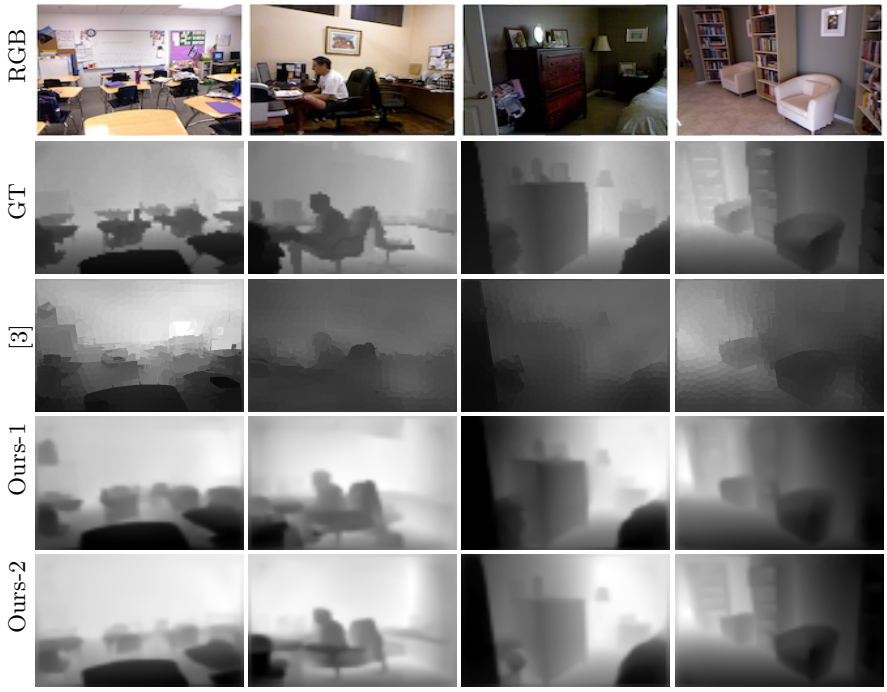


Figure 5.10: Additional qualitative result on NYU depth v2: (a) RGB image, (b) ground truth, (c) Liu *et al.* [3], (d) our encoder-decoder model, (e) our adversarial model.

In comparison to the works by [4, 5, 6] shown in Fig. 5.11, although our approach does not achieve the highest performance, it consistently exhibits reliable results. Our method produces depth maps that demonstrate improved consistency and smoother transitions, particularly in regions with gradual depth changes. Despite the unavailability of ground truth depth for direct comparison, the estimated depths align closely with the expected values, resulting in visually pleasing and perceptually accurate representations of the scene.

Figure 5.11: Additional qualitative result on NYU Depth v2 from left to right: RGB images, Godard *et al.* [4], Zhao *et al.* [5], Bian *et al.* [6], our encoder model, and our adversarial model.

Additionally, compared to the recent research works by [7, 8], our single model depth estimation method demonstrates a notable advantage in effectively handling challenging scenarios. It reliably infers depth information, resulting in more coherent depth maps with fewer artefacts or inconsistencies, as depicted in Fig. 5.12. Although our method may perform less than the compared approaches, it consistently delivers reliable results, showcasing its robustness and capability to handle challenging depth estimation scenarios.

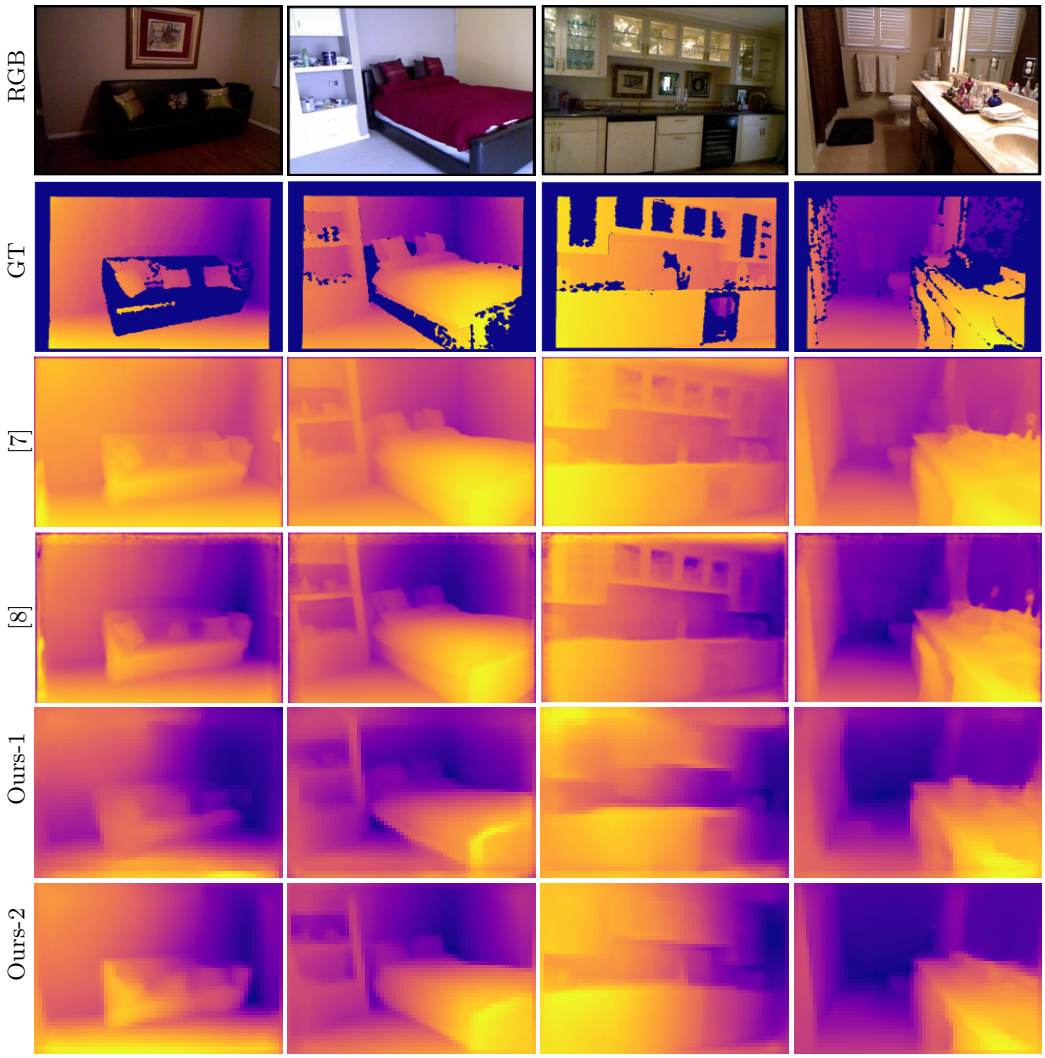


Figure 5.12: Additional qualitative result on the NYU dataset against Yuan et al. [7] and Agarwal et al. [8].

Similarly, we conduct a qualitative comparison on the KITTI dataset to showcase the effectiveness of our single model depth estimation approach in outdoor scenarios, as illustrated in Fig. 5.13 and 5.14.

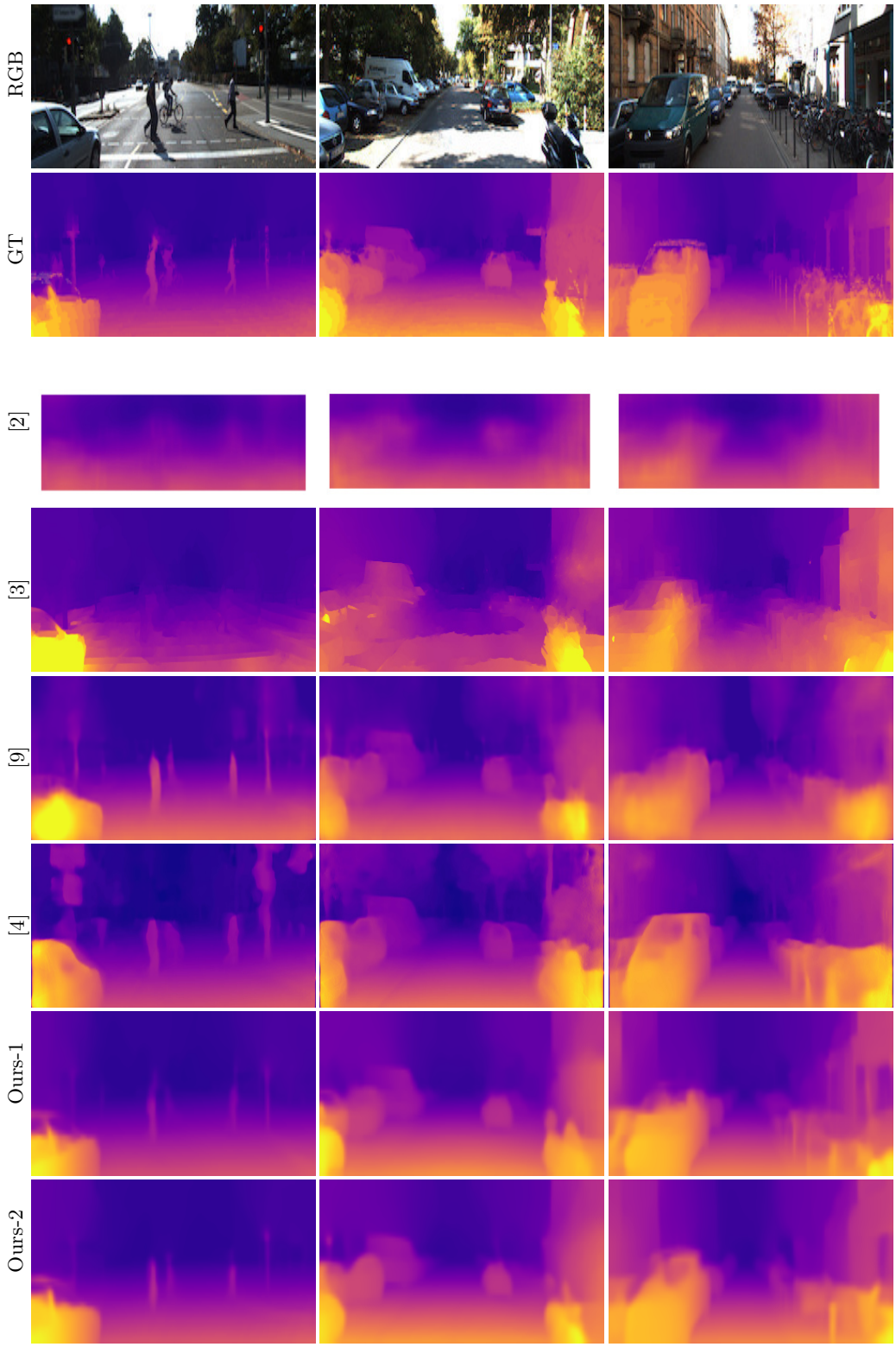


Figure 5.13: Qualitative comparison result on KITTI data. (a) RGB image, (b) ground truth, (c) Eigen *et al.* [2], (d) Liu *et al.* [3], (e) Kutznizov *et al.* [9], (f) Godard *et al.* [10], (g) our encoder-decoder model, (h) our adversarial model.

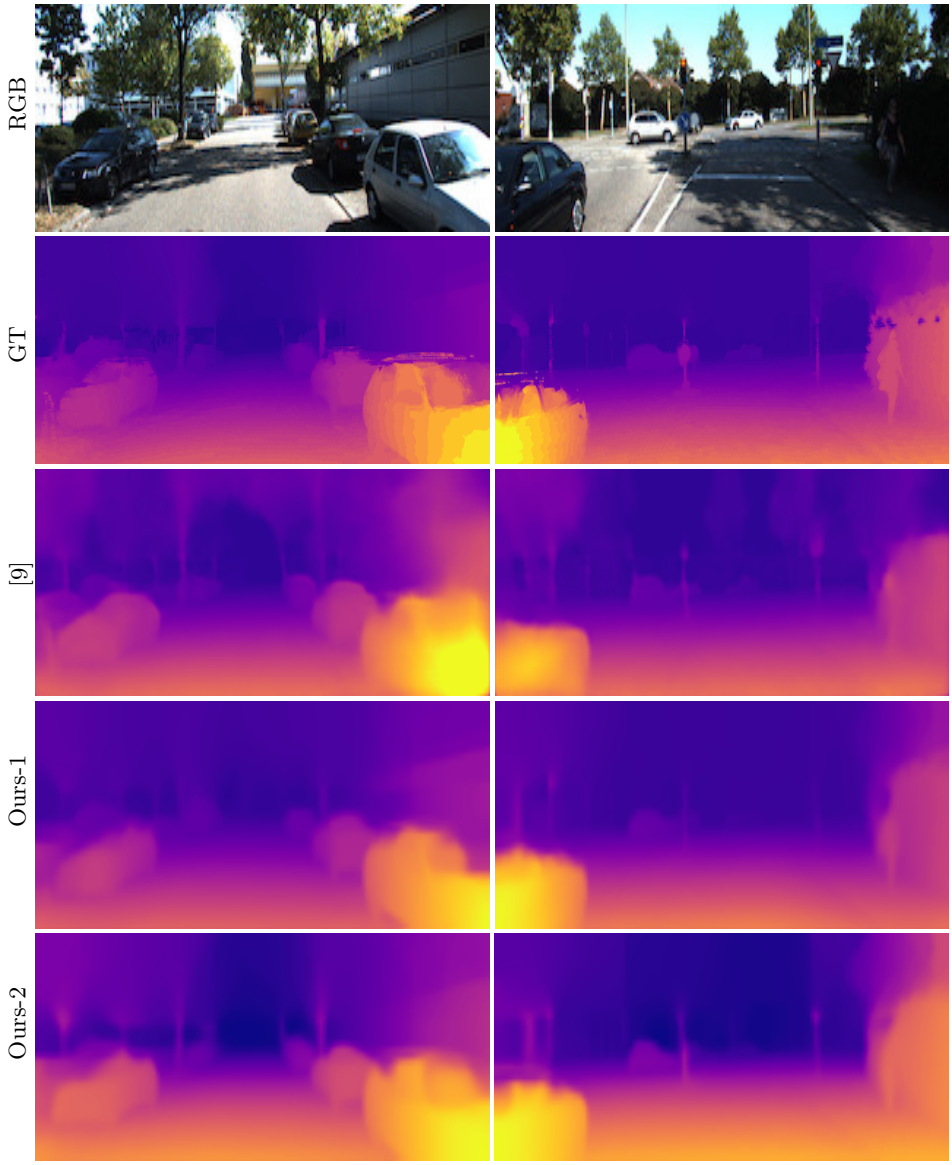


Figure 5.14: Additional qualitative result on KITTI data from top to bottom: (a) RGB image, (b) ground truth, (c) Kutznietsov *et al.* [9], (d) our encoder-decoder model, (e) our adversarial model.

When compared to [2, 3, 10] and [9] in Fig. 5.13 and 5.14 respectively, our approach consistently generates depth maps that exhibit adequate accuracy and more effective preservation of depth boundaries. It effectively captures the depth variations and occlusions present in outdoor scenes, yielding depth maps that closely resemble the ground truth. The estimated depths proficiently convey the relative distances between objects, road surfaces, and distant structures, thereby contributing to a visually compelling representation of the depth structure within outdoor scenes.

Our estimated depths align effectively with the ground truth, contributing to a more immersive and realistic representation of the outdoor scenes. This accuracy is essential for applications such as autonomous driving, where precise depth perception is crucial for making informed decisions in dynamic environments. Through this evaluation, it becomes evident that our algorithm performs better in various aspects,

including accuracy, detail preservation, and depth representation.

5.5.2 Quantitative Performance

We present a comprehensive analysis of the quantitative performance of our single depth estimation method in comparison to several state-of-the-art approaches on two challenging datasets: the indoor NYU Depth V2 dataset Tab. 5.4 and 5.5 and the outdoor KITTI dataset in Tab. 5.6 and 5.7. To evaluate the performance, we employed commonly used metrics as described in Sec. 5.3. The objective is to provide an in-depth assessment of the proposed methods accuracy and error performance in estimating depths from a single image.

Table 5.4: Accuracy comparison with previous works on NYU Depth v2.

		Accuracy*		
	range [m]	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Adversarial Methods				
Zheng *et al.* 2018 [45]	1–10	0.540	0.832	0.948
Kwak *et al.* 2020 [26]	—	**0.834**	0.941	0.976
Non-adversarial Methods				
Eigen *et al.* 2014 [2]	0–10	0.611	0.887	0.971
Eigen *et al.* 2015 [58]	0–10	0.769	0.950	0.988
Wang *et al.* 2015 [59]	—	0.605	0.890	0.970
Roy *et al.* 2016 [60]	0–10	—	—	—
Chakrabarti *et al.* 2016 [61]	—	0.806	0.958	0.987
Li *et al.* 2019 [62]	—	0.788	0.958	**0.991**
Zhao *et al.* 2020 [5]	—	0.701	0.912	0.987
Gur *et al.* 2020 [23]	0–10	0.772	0.942	0.984
Bian *et al.* 2021 [6]	0–10	<u>0.820</u>	<u>0.956</u>	<u>0.989</u>
Ye *et al.* 2022 [24]	—	—	—	—
Our Encoder Model	0-10	0.784	0.947	0.984
Our Generative Model	0-10	0.819	**0.960**	<u>0.989</u>

*the higher the better.

Upon careful examination of the quantitative results, it is evident that our single depth estimation method may achieve a lower performance compared to the state-of-the-art approaches. However, it demonstrates reliable and consistent performance on the indoor NYU Depth V2 and outdoor KITTI datasets.

Table 5.5: Error rate comparison with previous works on NYU Depth v2.

		Error Rates**		
	range [m]	RMS	LOG10	REL
Adversarial Methods				
Zheng *et al.* 2018 [45]	1–10	0.915	—	0.257
Kwak *et al.* 2020 [26]	—	0.652	—	—
Non-adversarial Methods				
Eigen *et al.* 2014 [2]	0–10	0.907	—	0.215
Eigen *et al.* 2015 [58]	0–10	0.641	—	0.158
Wang *et al.* 2015 [59]	—	0.824	—	0.220
Roy *et al.* 2016 [60]	0–10	0.774	—	0.187
Chakrabarti *et al.* 2016 [61]	—	0.620	—	0.149
Li *et al.* 2019 [62]	—	0.635	0.063	<u>0.143</u>
Zhao *et al.* 2020 [5]	—	0.686	0.079	0.189
Gur *et al.* 2020 [23]	0–10	0.546	0.063	0.149
Bian *et al.* 2021 [6]	0–10	0.532	**0.059**	**0.138**
Ye *et al.* 2022 [24]	—	<u>0.518</u>	—	—
Our encoder Model	0-10	0.547	0.065	0.153
Our adversarial Model	0-10	**0.509**	<u>0.060</u>	<u>0.143</u>

**the lower the better.

Table 5.6: Accuracy comparison with previous works on KITTI data.

		Accuracy*		
	range [m]	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Adversarial Methods				
Kumar *et al.* 2018 [46]	—	0.732	0.897	0.959
Pilzer *et al.* 2018 [47]	0–80	0.789	0.918	0.965
Aleotti *et al.* 2018 [44]	0–80	0.808	0.939	0.975
Zheng *et al.* 2018 [45]	1–50	0.867	0.960	0.986
Almalioglu *et al.* 2019 [63]	0–50	0.867	0.970	0.983
Li *et al.* 2019 [64]	0–80	0.823	0.936	0.974
Puscas *et al.* 2019 [65]	0–80	0.828	0.933	0.967
Groenendijk *et al.* 2020 [66]	—	0.847	0.945	0.975
Zhao *et al.* 2021 [48]	0–80	0.821	0.942	0.978
Non-adversarial Methods				
Eigen *et al.* 2014 [2]	0–80	0.692	0.899	0.967
Liu *et al.* 2015 [3]	—	0.647	0.882	0.961
Godard *et al.* 2017 [10]	0–50	0.861	0.949	0.976
Kutznietsov *et al.* 2017 [9]	0–80	0.862	0.960	0.986
Zhan *et al.* 2018 [67]	0–80	0.820	0.933	0.971
Our encoder Model	0–80	0.872	0.969	0.990
Our adversarial Model	0–80	<u>0.884</u>	<u>0.973</u>	<u>0.992</u>

*the higher the better.

The obtained results demonstrate that our method consistently achieves reliable

Table 5.7: Error rate comparison with previous works on KITTI data.

		Error Rate**		
	range [m]	ABS REL	SQ REL	RMSE LOG
Adversarial Methods				
Kumar *et al.* 2018 [46]	—	0.211	1.979	0.264
Pilzer *et al.* 2018 [47]	0–80	0.152	1.388	0.247
Aleotti *et al.* 2018 [44]	0–80	0.150	1.414	0.216
Zheng *et al.* 2018 [45]	1–50	0.114	<u>0.627</u>	<u>0.178</u>
Almalioglu *et al.* 2019 [63]	0–50	0.137	0.892	0.201
Li *et al.* 2019 [64]	0–80	0.150	1.127	0.229
Puscas *et al.* 2019 [65]	0–80	0.135	1.1815	0.235
Groenendijk *et al.* 2020 [66]	—	0.122	0.928	0.215
Zhao *et al.* 2021 [48]	0–80	0.139	1.034	0.214
Non-adversarial Methods				
Eigen *et al.* 2014 [2]	0–80	0.190	1.515	0.270
Liu *et al.* 2015 [3]	—	0.217	1.841	0.289
Godard *et al.* 2017 [10]	0–50	0.114	0.898	0.206
Kutznietsov *et al.* 2017 [9]	0–80	0.113	0.741	0.189
Zhan *et al.* 2018 [67]	0–80	0.135	1.132	0.229
Our encoder Model	0–80	0.110	0.673	0.166
Our adversarial Model	0–80	<u>0.103</u>	**0.624**	**0.156**

**the lower the better.

accuracy and precision in depth estimation compared to the several previous methods. The lower error values indicate a closer alignment between the estimated depths and the ground truth depths. These findings reinforce the algorithm's reliability and potential for applications such as 3D reconstruction, where even minor errors can significantly affect the overall quality of the reconstructed scene.

5.5.3 Parameters and Hyper-parameters Comparison

We compare the conciseness of our single depth estimation model's parameters and hyperparameters with those of other related methods. Conciseness refers to the ability to achieve optimal performance while maintaining simplicity and efficiency in terms of the number of parameters and hyperparameters involved.

To ensure a fair comparison, we carefully selected a set of state-of-the-art methods known for their performance in depth estimation. We focused on evaluating the compactness and efficiency of our model's parameters and hyperparameters in relation to these methods. We compare the parameters on KITTI dataset and NYU Depth v2 in Tab. 5.8, 5.9 and 5.10, respectively.

Regarding the model parameters, we analyzed various factors, including the training and testing parameters and the number of training data and batch size used in one epoch. Our model is designed to strike a balance between model complexity and performance. We achieve conciseness without compromising accuracy by employing a streamlined network architecture with a reduced number of training and testing parameters and a smaller training data size. This reduction in parameter count enables efficient memory usage and computational efficiency regardless of our GPU device

during both the training and testing stages.

In addition to parameters, we investigated the model's hyperparameters, such as the learning rate. We aimed to strike a balance between fine-tuning the model's performance and keeping the hyperparameter space concise. Through careful optimization, we selected a set of hyperparameters that provide optimal results without unnecessary complexity.

Table 5.8: Compare Parameters and Hyper-parameters with previous related works on KITTI data.

Methods	GPU Device	Training Time	# Params Train/Test	Epochs Converge	Init LR	Batch Size	δ_1
Our Adversarial	Ge Force GTX 1080: 8GB	39.5 min/epoch	59M/520K	28	$2e-04$ $4e-04$	16	0.884
Our Encoder -decoder	Ge Force GTX 1080: 8GB	14.6 min/epoch	12M/12M	70	$1e-03$	16	0.872
Manimaran et al. 2022	2 Nvidia RTX 3090: 2×24GB	-	-	35	$1e-04$	16	0.926
Bian et al. 2021	Tesla V100: 16GB	44.4 min/epoch	-	50	$1e-04$	4	0.873
Zhao et al. 2020	-	-	-	50	$1e-04$	8	0.871
Ranjan et al. 2019	Tesla V100: 16GB	7 days for all iterations (depth, cam motion, opt flow, & segm)	-	-	$1e-04$	4	0.826
Godard et al. 2019	Titan X 12GB	36 min/epoch (12 hours)	-	20	$1e-04$	12	0.876
Zou et al. 2018	Testa K80: 12GB	-	-	-	$2e-04$	6	0.806
Zhan et al. 2018	-	-	-	-	$1e-03$	-	0.820
Kutznietzov et al. 2017	GTXTi 6GB	-	-	at least 15	-	5	0.862
Liu et al. 2015	GTX 780 6GB	-	20M/	60	$1e-04$	-	0.647
Eigen et al. 2014	-	-	-	-	$1e-03$	32	0.692
Zhao et al. 2021	RTX 8000	-	-	50	$2e-04$	-	0.821
Groenendijk et al. 2020	-	~31.6M	-	50	-	8	0.847

In conclusion, the evaluation of our model's parameters and hyperparameters highlights its conciseness when compared to other methods. The streamlined network architecture, reduced parameter count, and optimized hyperparameters allow our model to achieve competitive performance in depth estimation tasks while maintaining simplicity and efficiency. These findings reinforce the model's suitability for resource-constrained environments, where a balance between performance and conciseness is most importance.

Table 5.9: Compare Parameters and Hyper-parameters with previous related works on KITTI data (cont.)

Methods	GPU Device	Training Time	# Params Train/Test	Epochs Converge	Init LR	Batch Size	δ_1
Puscas et al. 2019	-	-	-	-	$1e-04$	8	0.828
Li et al. 2019	GTX 1080Ti	-	-	-	$1e-04$	4	0.823
Almalioglu et al 2019.	Titan V	-	-	-	$1e-01$	8	0.867
Zheng et al. 2018	-	-	-	-	$1e-04$	-	0.867
Aleotti et al. 2018	Titan X Pascal	-	39M/31M	50	-	-	0.808
Pilzer et al. 2018	Tesla K80 12GB	45 hours	-	50	$1e-05$	8	0.789
Kumar et al. 2018	-	-	-	-	-	32	0.732

Table 5.10: Compare Parameters and Hyper-parameters with previous related works on NYU depth v2.

Methods	GPU Device	Training Time	# Params Train/Test	Epochs Converge	Init LR	Batch Size	δ_1
Our Adversarial	Ge Force GTX 1080: 8GB	50 min/epoch	59M/520K	36	$2e-04$ $4e-04$	16	0.819
Our Enc-decoder	Ge Force GTX 1080: 8GB	24 min/epoch	12M/12M	65	$1e-03$	16	0.784
Zheng et al. 2018	-	-	-	-	$1e-04$	-	0.540
Kwak et al. 2020	Ge Force GTX 1080 Ti: 11GB	-	-	-	-	-	0.834
Eigen et al. 2014	-	-	-	-	$1e-03$	32	0.611
Eigen et al. 2015	-	-	-	>100	$1e-01$	32/16	0.769
Wang et al. 2015	Tesla K40: 12GB	4 days	-	-	-	-	0.605
Chakrabarti et al. 2016	Titan X: 12GB	-	-	14	$1e-02$	-	0.806
Li et al. 2019	Titan X: 12GB	50 hours	-	2-3	$1e-03$	16	0.806
Zhao et al. 2020	-	-	-	50	$1e-04$	8	0.701
Gur et al. 2020	Titan X Pascal: 12GB	-	-	-	$2e-05$	3	0.772
Bian et al. 2020	GeForce RTX 2080: 8GB	44 hours	-	50	$2e-04$	8	0.820

5.5.4 SSIM Reconstruction Error

We comprehensively evaluated the SSIM reconstruction error on a diverse range of datasets on sample 45 images from random scenes in the indoor NYU Depth V2 and outdoor KITTI datasets.

We utilize the SSIM score to assess how well the predicted depth maps align with the ground truth depth maps to demonstrate the effectiveness of our method. A higher SSIM score indicates a better similarity between the predicted and ground truth depth maps, indicating a more accurate depth estimation and supporting the reliability and effectiveness of our approach.

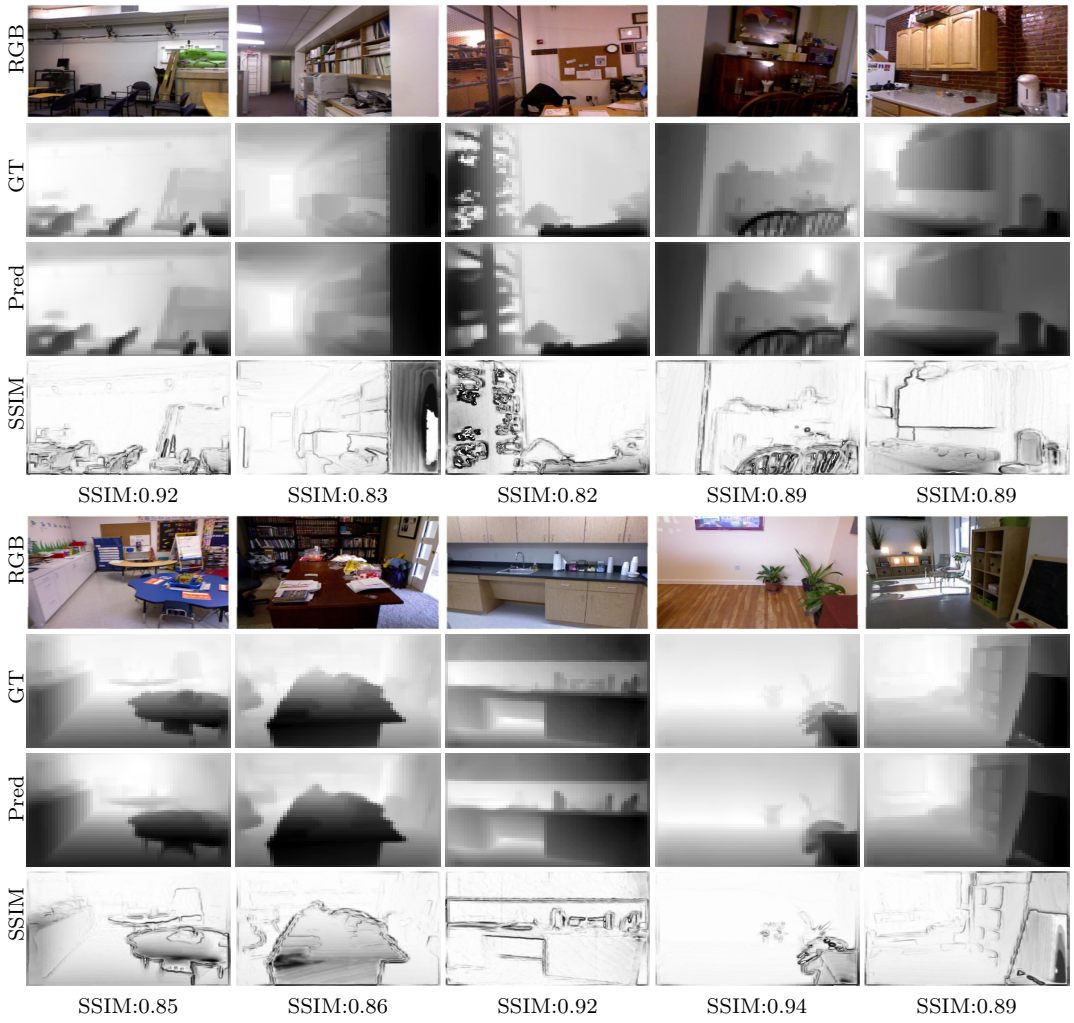


Figure 5.15: SSIM error compare with the ground truth depth on NYU Depth v2.

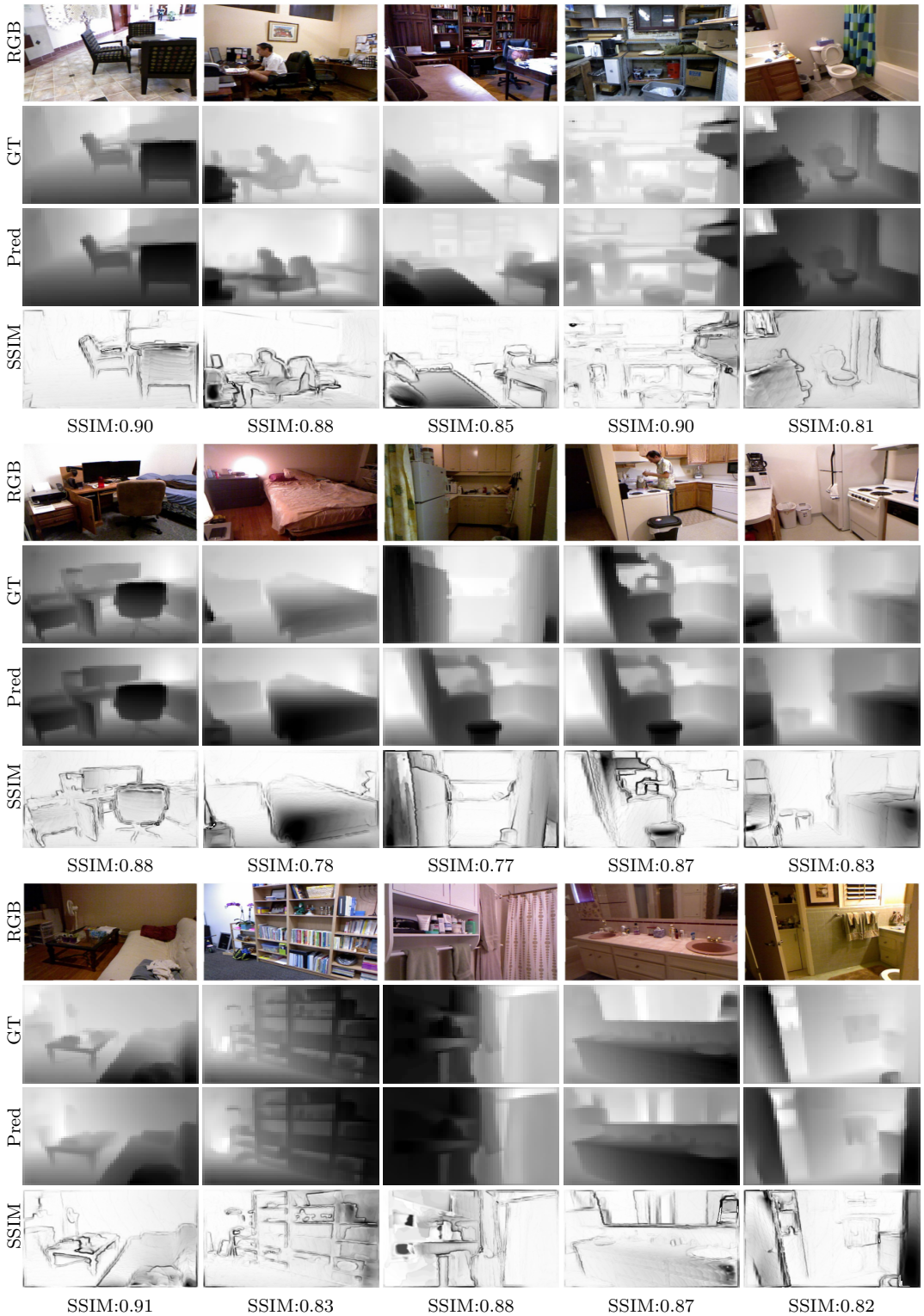


Figure 5.16: SSIM error compare with the ground truth depth on NYU Depth v2 (cont.)

Figure 5.17: SSIM error compare with the ground truth depth on NYU Depth v2 (cont.)

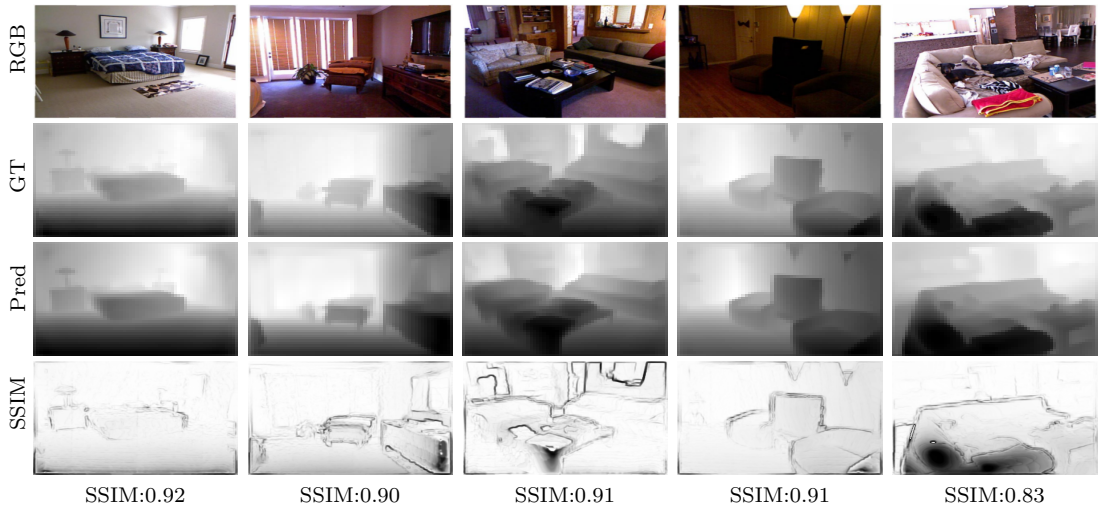


Figure 5.18: SSIM error compare with the ground truth depth on NYU Depth v2 (cont.)

On the indoor NYU Depth V2 dataset, depicted in Fig. 5.15, 5.16, 5.17, and 5.18, our method exhibited remarkable performance in terms of SSIM reconstruction error. The reconstructed depth maps closely resembled the ground truth, preserving fine details and accurately representing the structure of the scene. This results demonstrate the effectiveness of our method in capturing depth variations and generating visually pleasing depth maps.

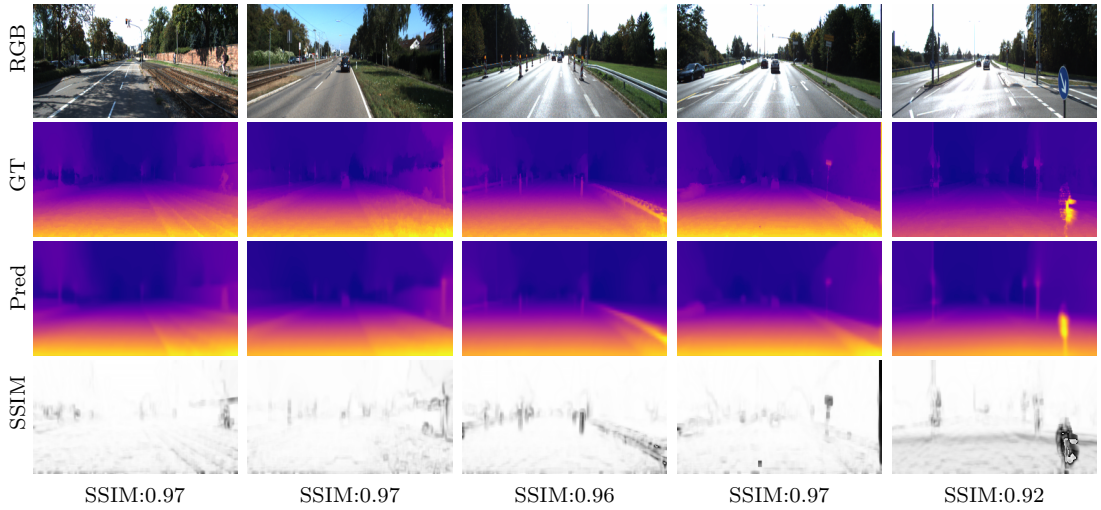


Figure 5.19: SSIM error compare with the ground truth depth on KITTI data.

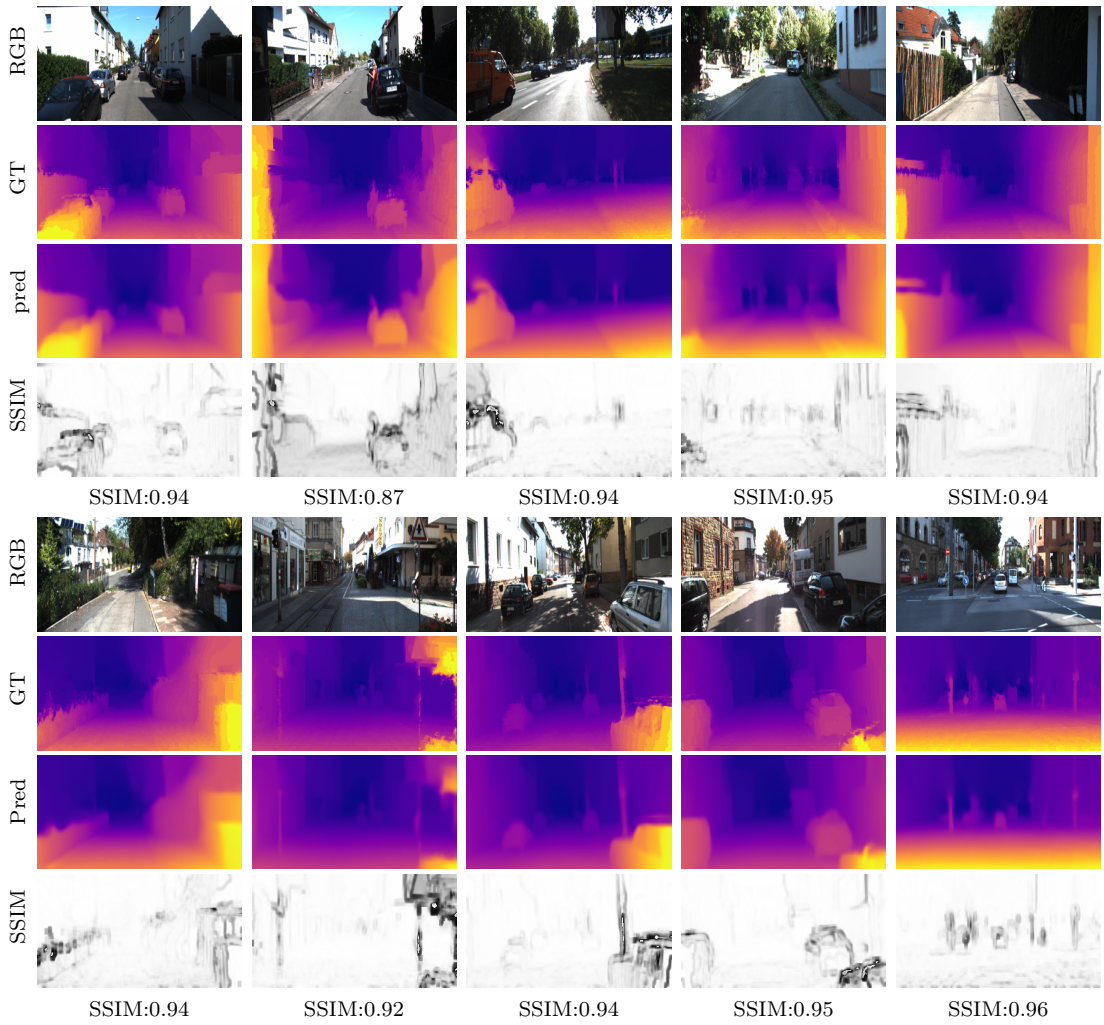


Figure 5.20: SSIM error compare with the ground truth depth on KITTI data (cont.)

Similarly, in Fig. 5.19, 5.20, 5.21, and 5.22, we present the strong performance of our methods in terms of SSIM reconstruction error on the outdoor KITTI dataset. Despite the challenges posed by dynamic objects, varying lighting conditions, and complex scenes, our method effectively reconstructed the depth maps with high fidelity. This highlights its robustness and ability to handle diverse outdoor scenarios.

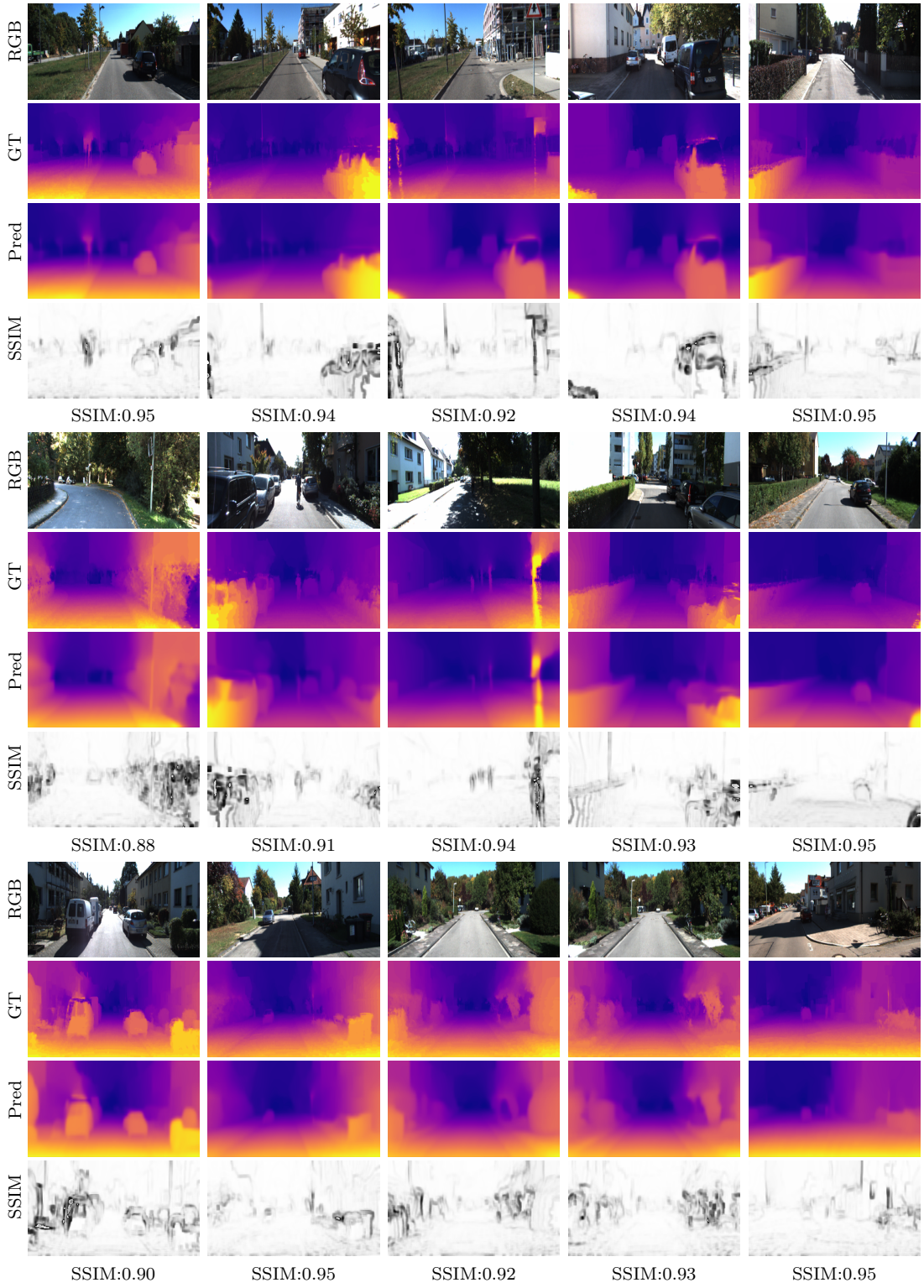


Figure 5.21: SSIM error compare with the ground truth depth on KITTI data (cont.)

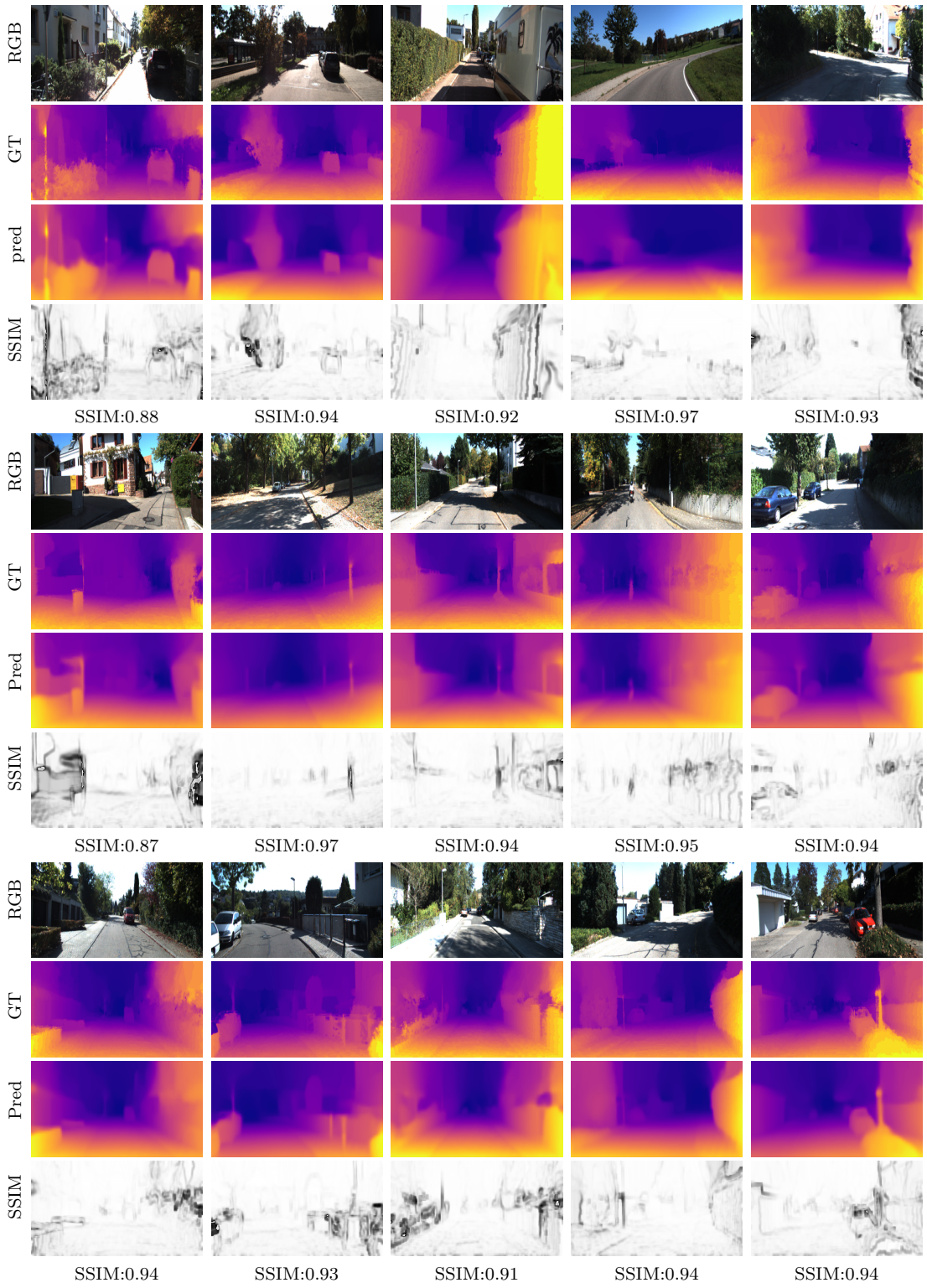


Figure 5.22: SSIM error compare with the ground truth depth on KITTI data (cont.)

The results on both the indoor NYU Depth V2 and outdoor KITTI datasets demonstrate the consistent performance and reliability of our method in producing accurate and visually appealing depth maps. These findings further support the suitability of our method for a wide range of applications, including 3D reconstruction,

augmented reality, and scene understanding in indoor and outdoor environments.

5.5.5 Depth Value Distribution

We present the experimental results of evaluating the depth value distribution histogram and comparing the histograms of the predicted depths with those of the ground truth. This analysis provides insights into the consistency and accuracy of our single depth estimation method in capturing the depth value distribution.

We conducted a thorough evaluation of both the indoor NYU Depth v2 and outdoor KITTI datasets to assess the performance of our method. Our primary objective was to compare the predicted depth value distribution histograms with the ground truth histograms, allowing us to examine the similarity and alignment between the two distributions.

A well-performing model should yield depth estimates that closely align with the true depths in the scene. We compute depth value histograms from 45 randomly selected images for each dataset to provide insights into the accuracy of the estimated depths.

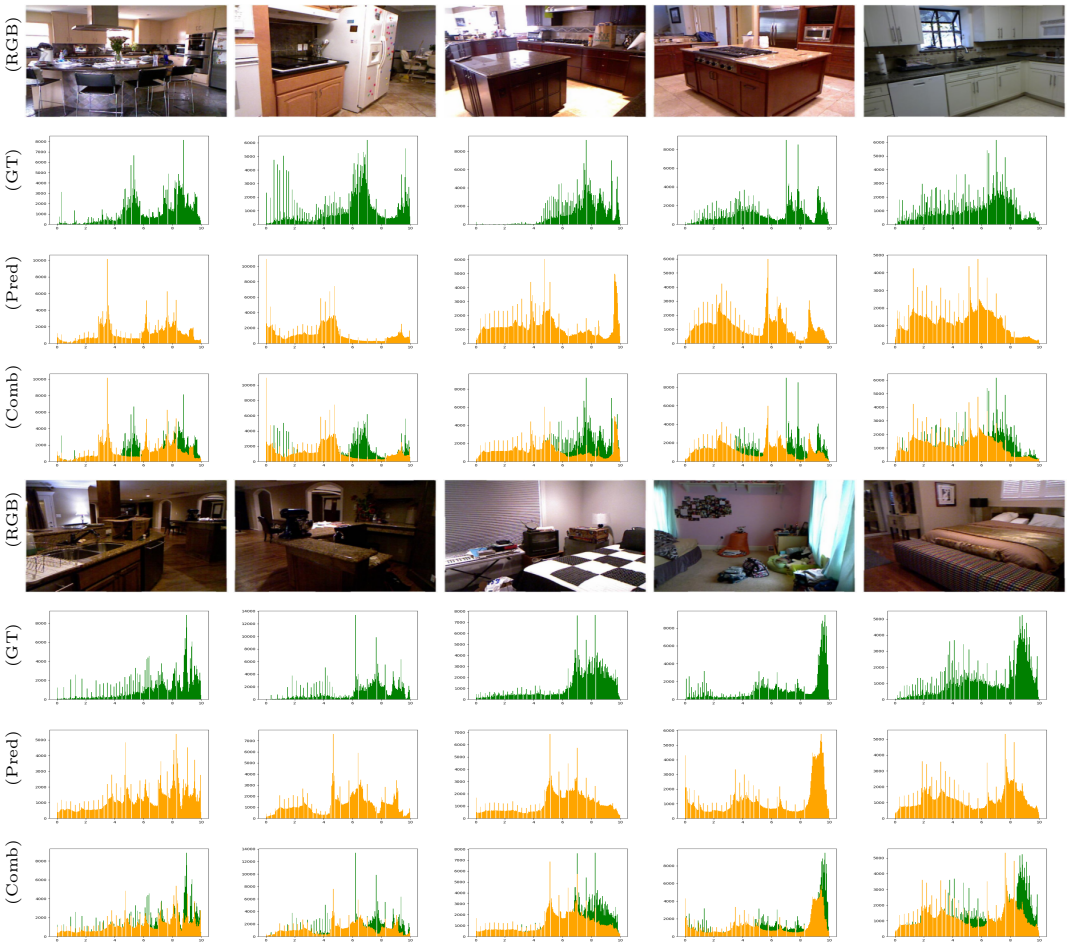


Figure 5.23: Depth value histogram from random images on NYU Depth v2.

Figure 5.24: Depth value histogram from random images on NYU Depth v2 (cont.)

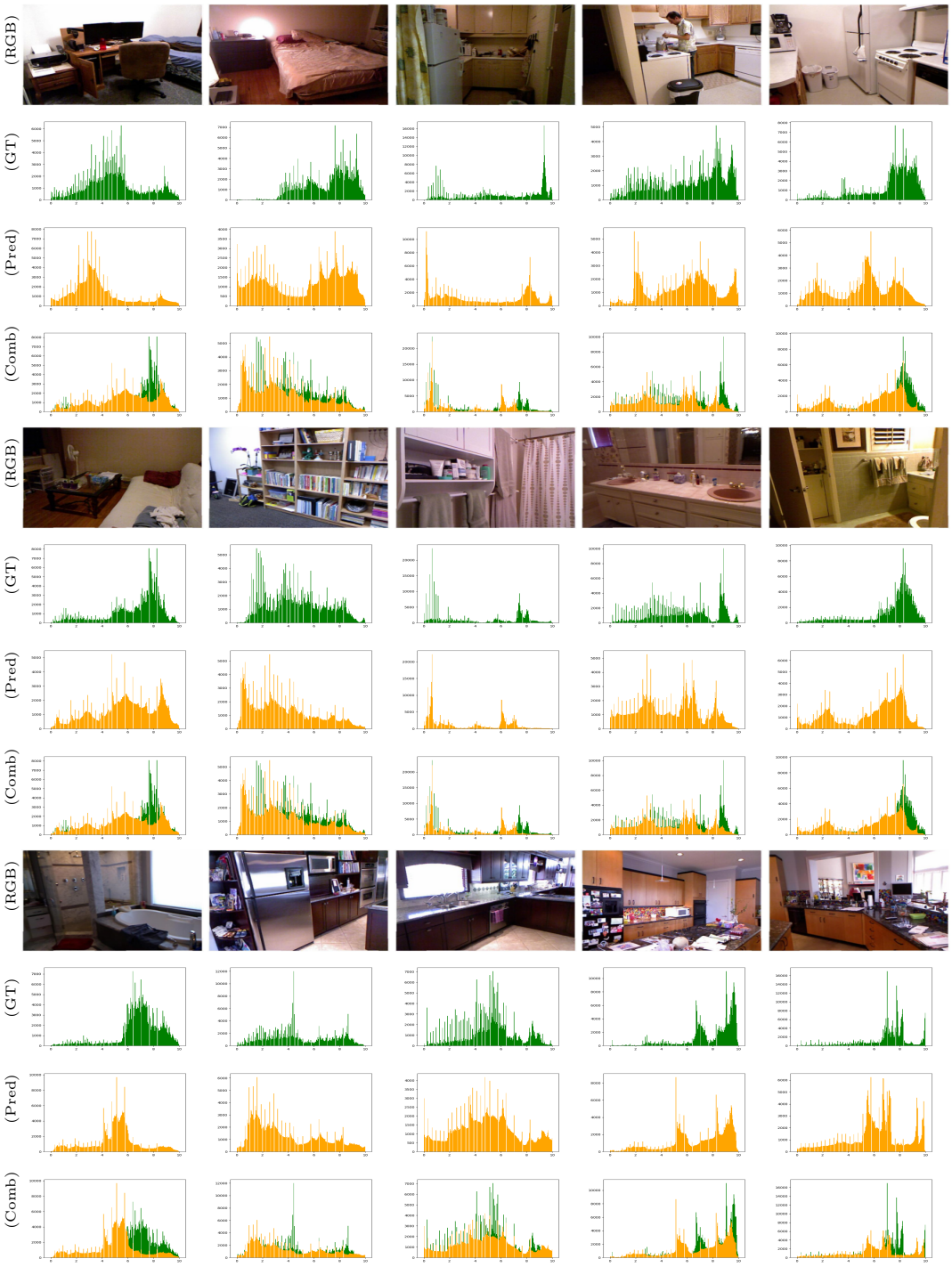


Figure 5.25: Depth value histogram from random images on NYU Depth v2 (cont.)

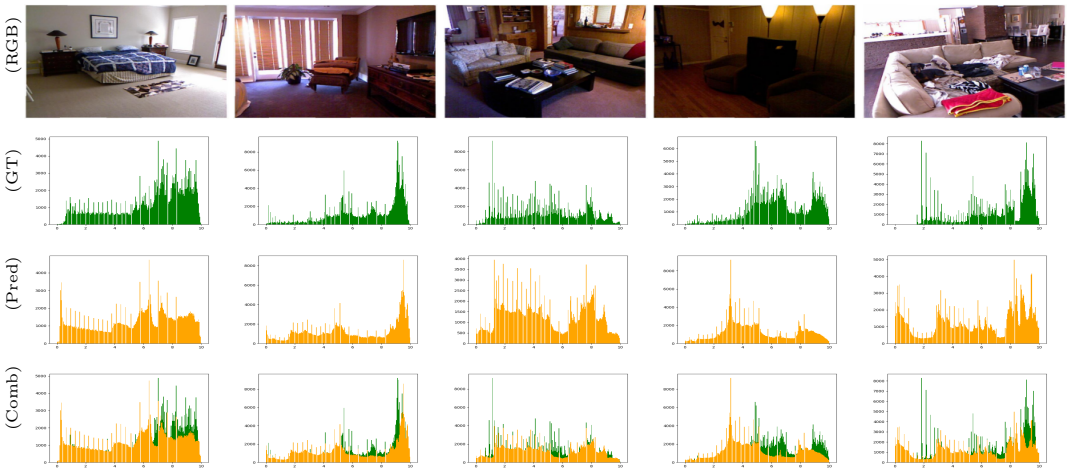


Figure 5.26: Depth value histogram from random images on NYU Depth v2(cont.)

On the indoor NYU Depth V2 dataset shown in Fig. 5.23, 5.24, 5.25, and 5.26, while the majority of the predicted depth value distribution histograms closely matched those of the ground truth, there were instances where some deviations were obscured. These deviations could be attributed to various factors such as challenging scenes, occlusions, or limitations in the depth estimation process. This signifies the accuracy and reliability of our method in estimating depth values and preserving the overall distribution characteristics.

Similar situation were observed in Fig. 5.27, 5.28, 5.29, and 5.30 on the outdoor KITTI dataset, where some of the predicted depth value distribution histograms obscured deviations from the ground truth histogram. These deviations may arise due to complexities in outdoor scenes. Despite the challenges posed by outdoor scenes with varying complexities, our method consistently estimated depth values that aligned well with the ground truth distribution.

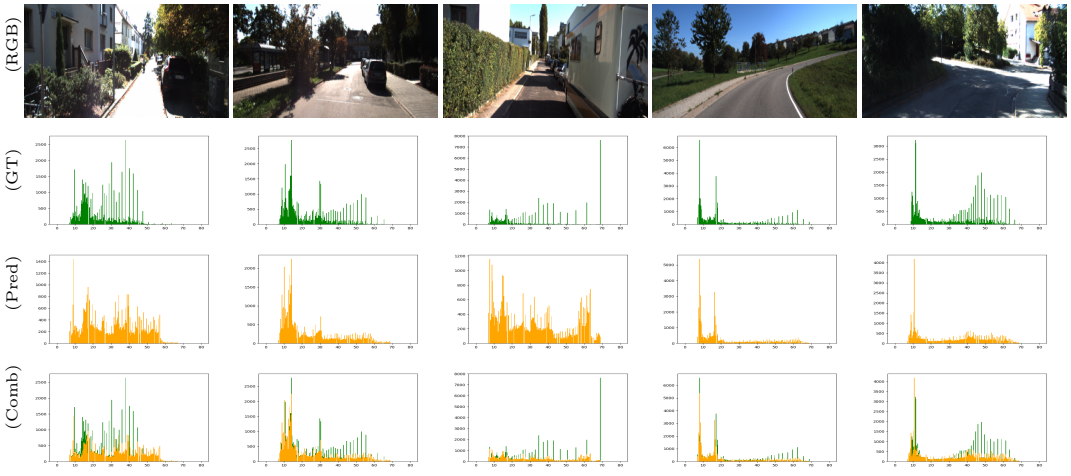


Figure 5.27: Depth value histogram from random images on KITTI data.

Figure 5.28: Depth value histogram from random images on KITTI data (cont.)

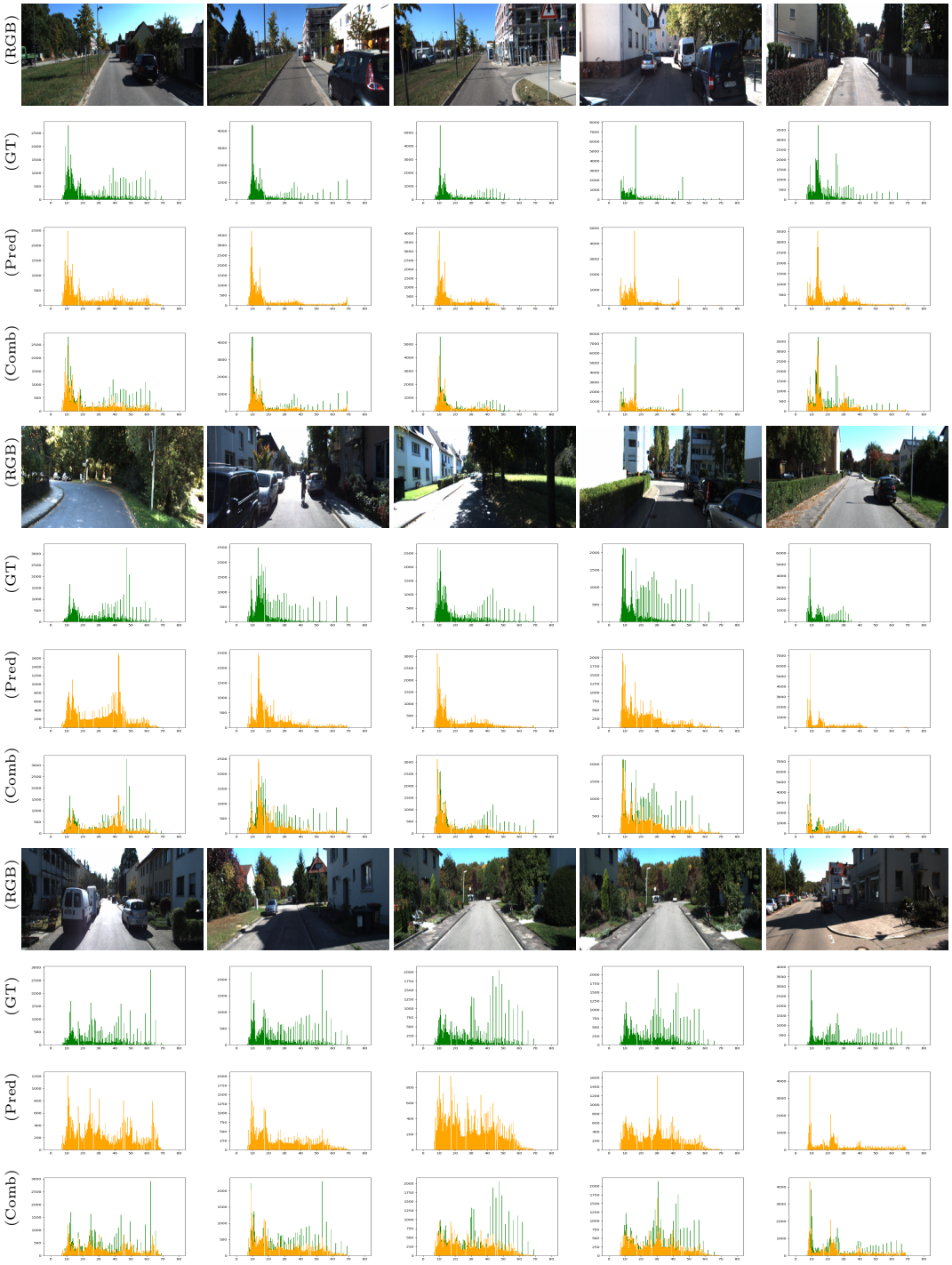


Figure 5.29: Depth value histogram from random images on KITTI data(cont.)

Figure 5.30: Depth value histogram from random images on KITTI data (cont.)

Despite these obscured deviations, it is important to note that the overall trend and similarity between the predicted depths and the ground truth histograms were still apparent. The majority of the histograms exhibited similar shapes, peaks, and distribution patterns, indicating that our method captures the general depth value distribution effectively.

While some deviations may be obscured in certain instances, it is crucial to interpret the evaluation results in the context of the overall performance of our method. The consistent alignment and similarity between the predicted depths and the ground truth histograms, in the majority of cases, still validate the accuracy and reliability of our approach.

5.5.6 Cross-Data Dependency

We examine the cross-dataset adaptation capabilities to provide a comprehensive understanding of how well our model can generalize to unseen data from different sources or domains.

We present the experimental results of evaluating the cross data validation between the indoor NYU Depth V2 and outdoor KITTI datasets using the Structural Similarity Index (SSIM) metrics. The objective was to assess the performance of our model when applied to data from a different domain, specifically examining the impact of using an outdoor-trained model on indoor data and vice versa.

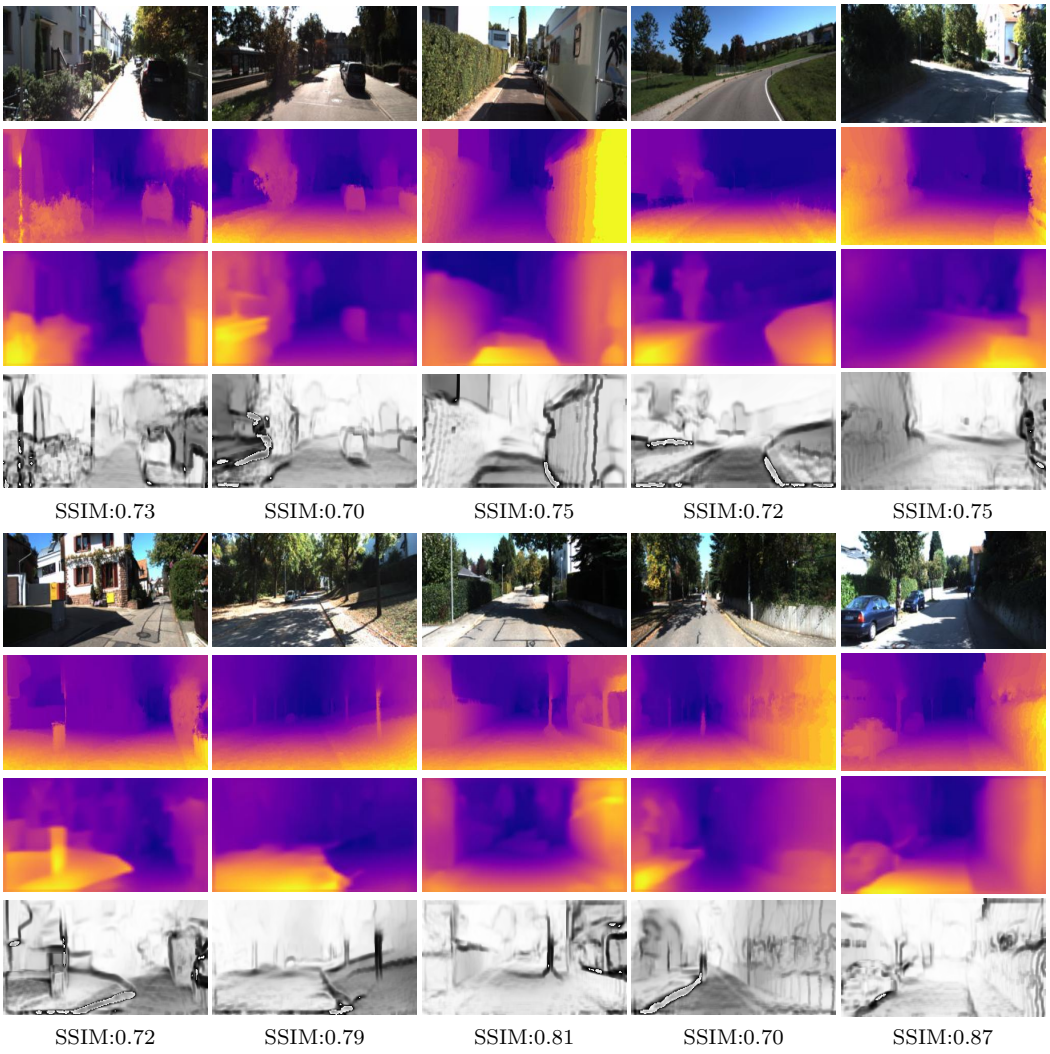


Figure 5.31: Qualitative results of our adversarial model approach (trained on NYU depth v2) on images of the KITTI data.

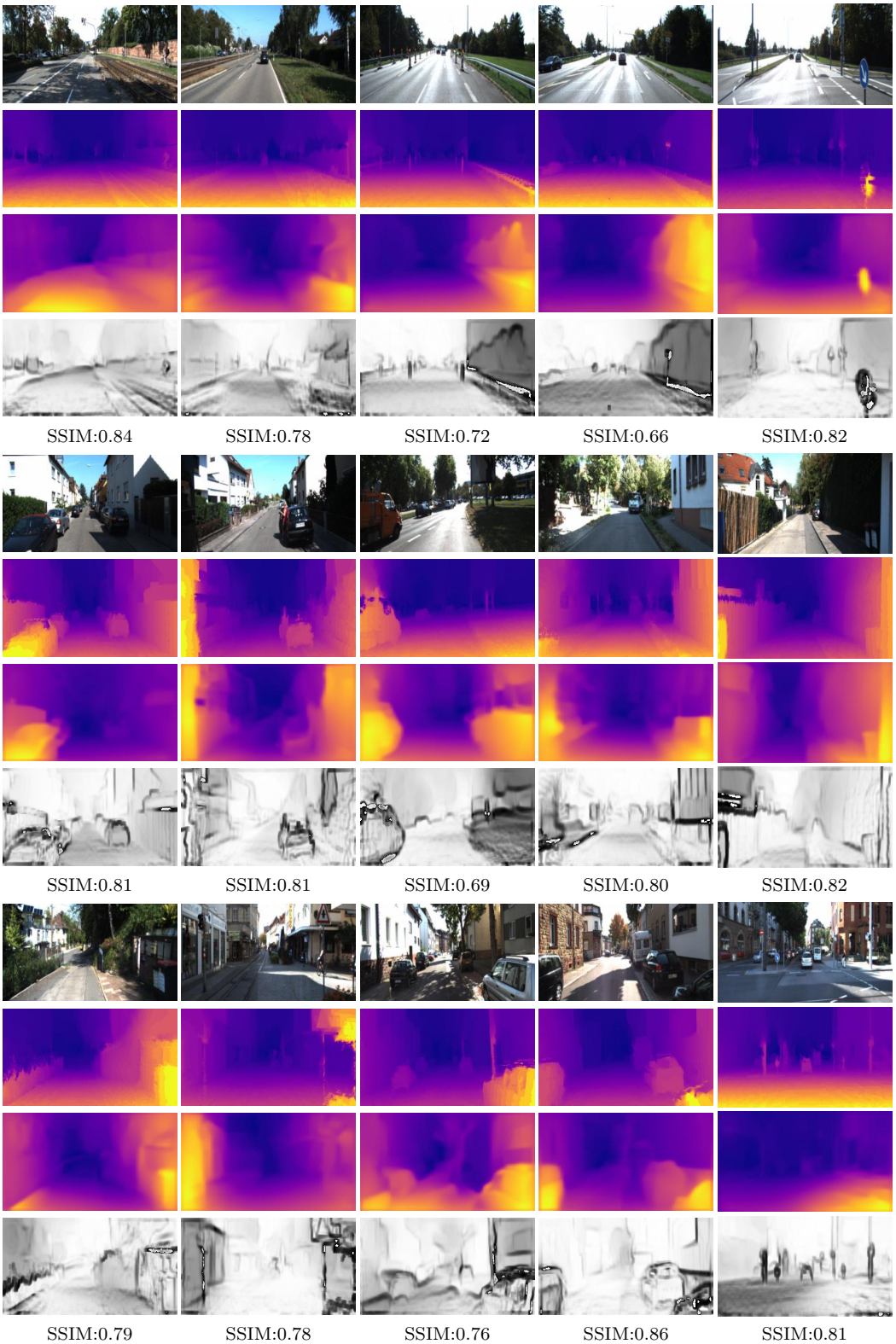


Figure 5.32: Qualitative results of our adversarial model approach (trained on NYU depth v2) on images of the KITTI data (cont.)

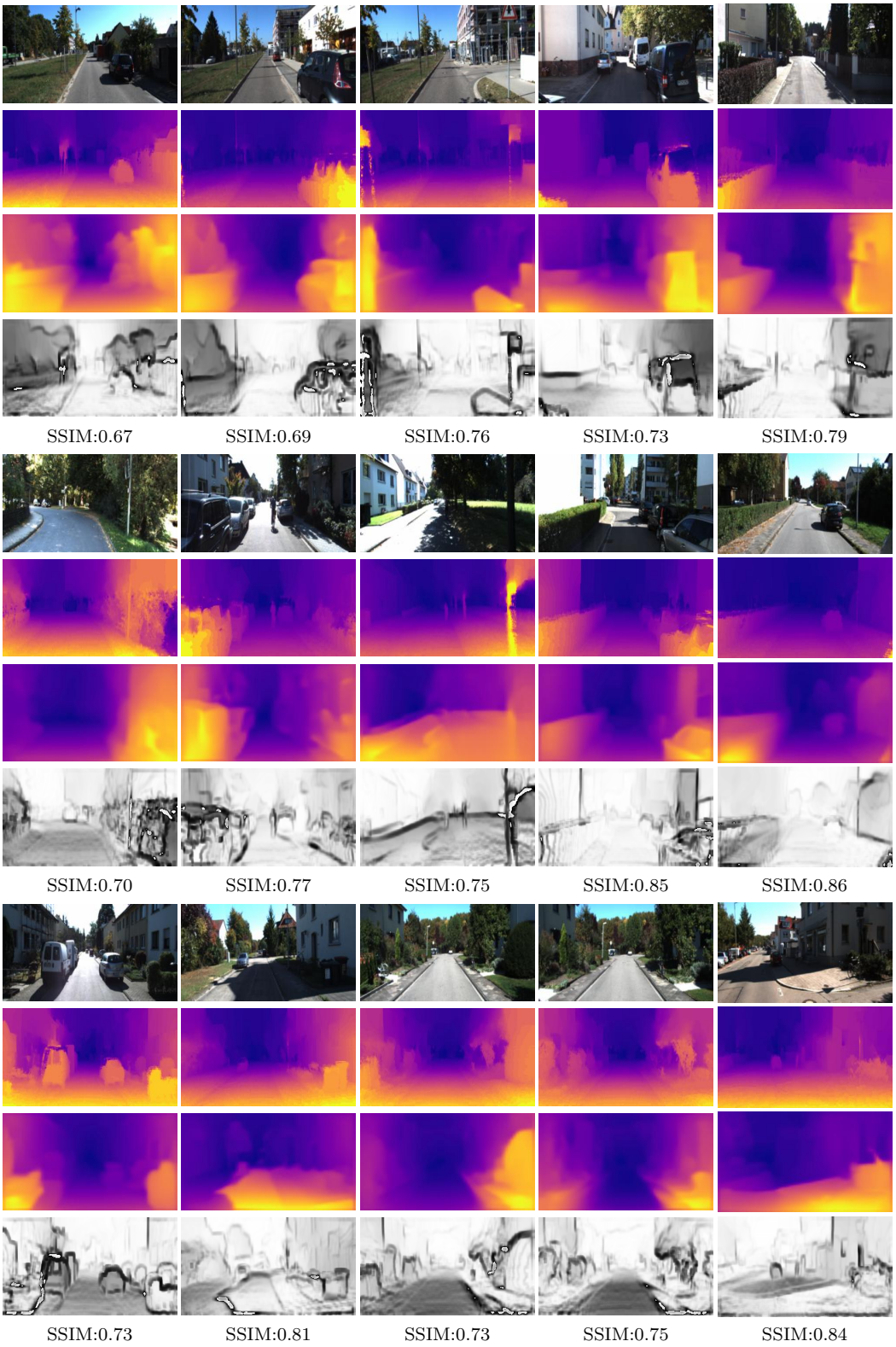


Figure 5.33: Qualitative results of our adversarial model approach (trained on NYU depth v2) on images of the KITTI data (cont.)

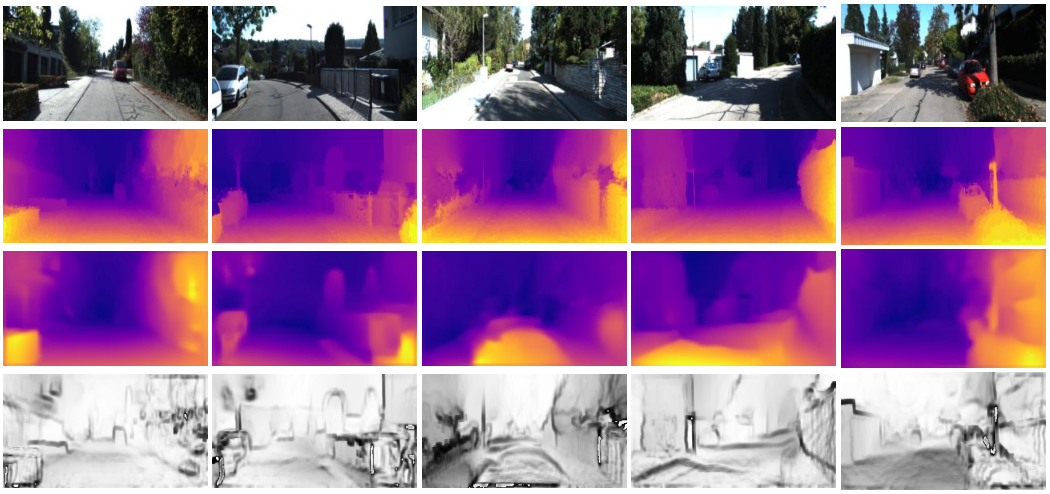


Figure 5.34: Qualitative results of our adversarial model approach (trained on NYU depth v2) on images of the KITTI data (cont.)

Upon thorough analysis, we observed a mixed performance when applying the model trained on one dataset to the other dataset. In some cases, our model's performance decreased when examining indoor data using the outdoor-trained model, and similarly, a decrease in performance was observed when evaluating outdoor data using the indoor-trained model. These findings indicate the challenges and limitations associated with cross-domain data validation.

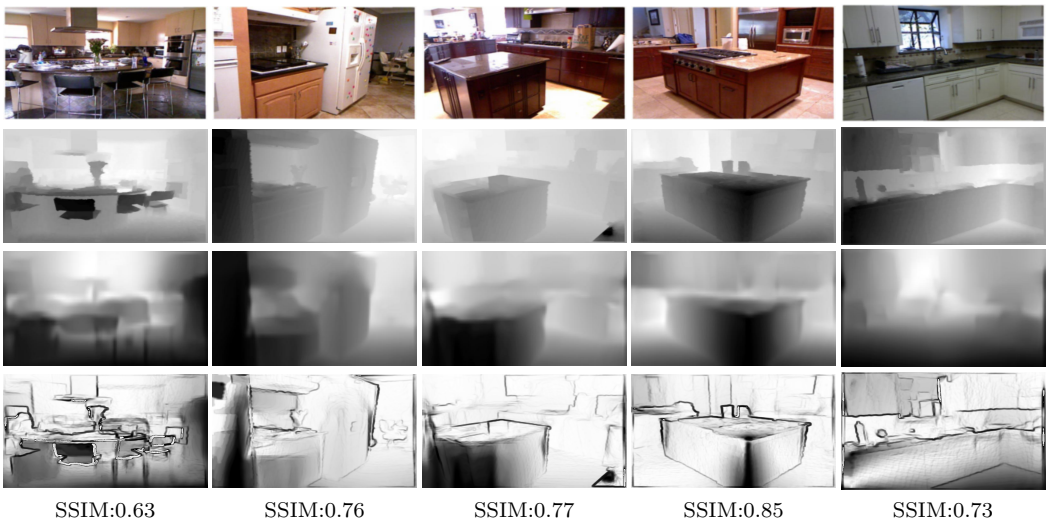


Figure 5.35: Qualitative results of our adversarial model approach (trained on KITTI) on images of the NYU depth v2.

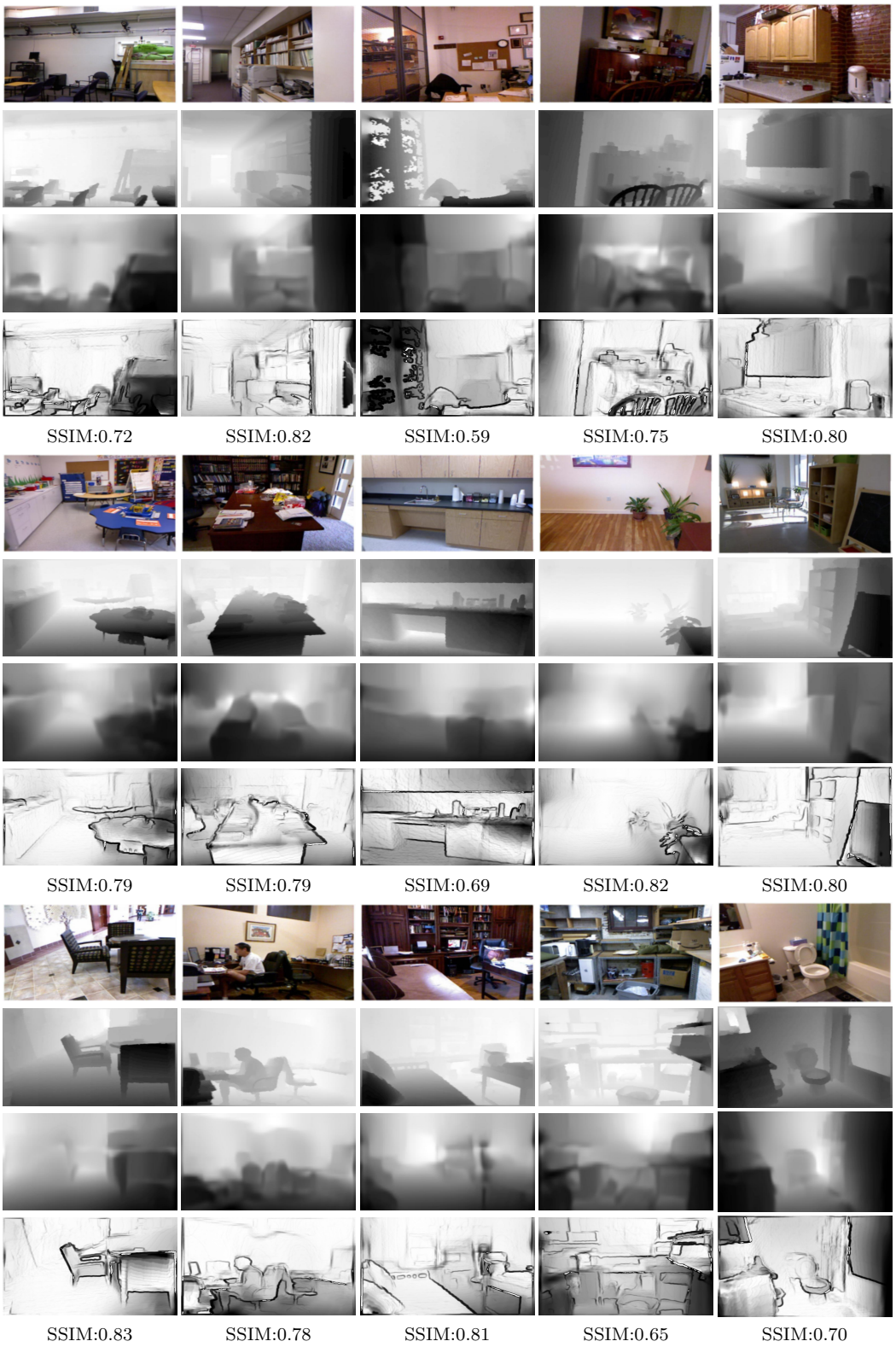


Figure 5.36: Qualitative results of our adversarial model approach (trained on KITTI) on images of the NYU depth v2 (cont.)

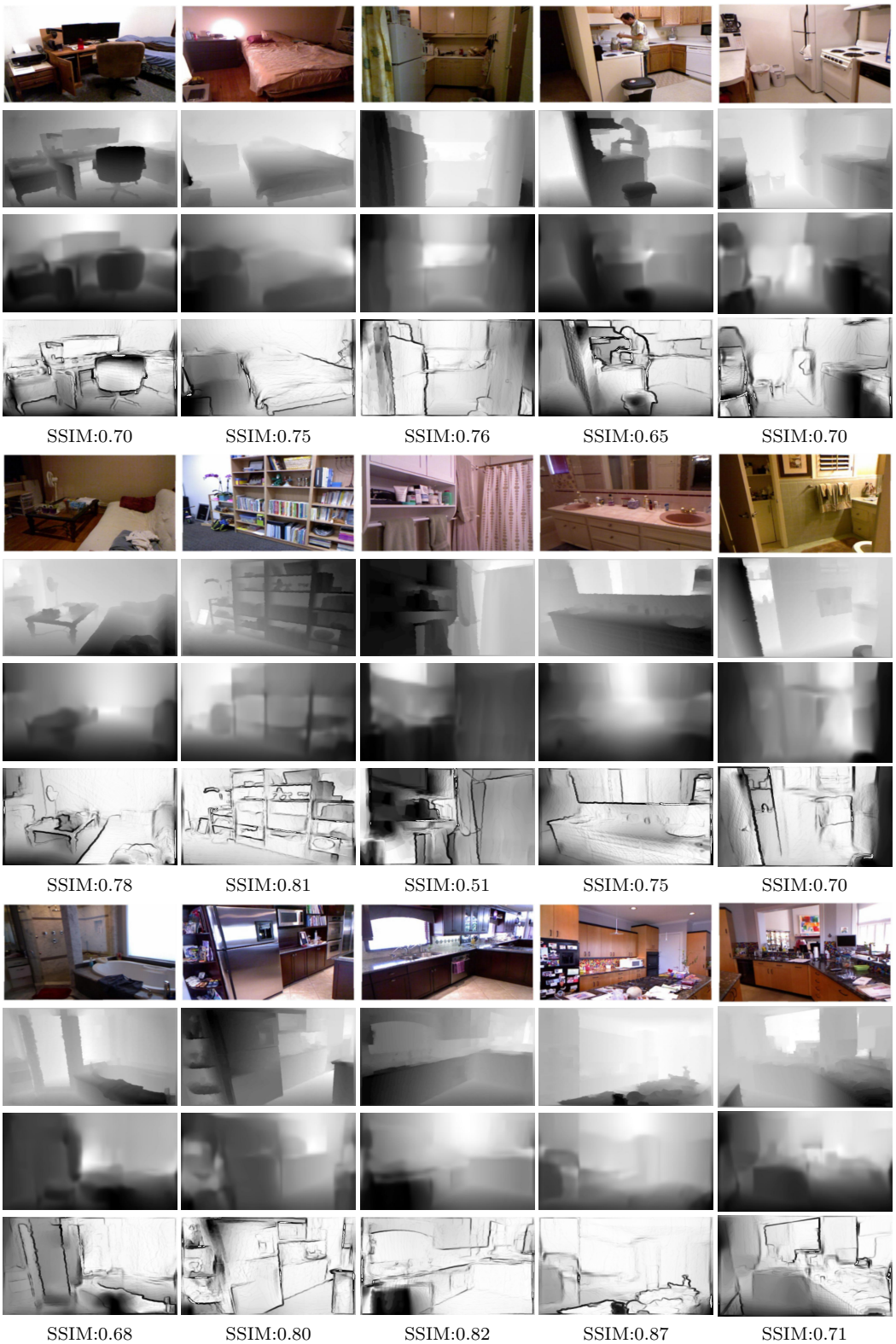


Figure 5.37: Qualitative results of our adversarial model approach (trained on KITTI) on images of the NYU depth v2 (cont.)

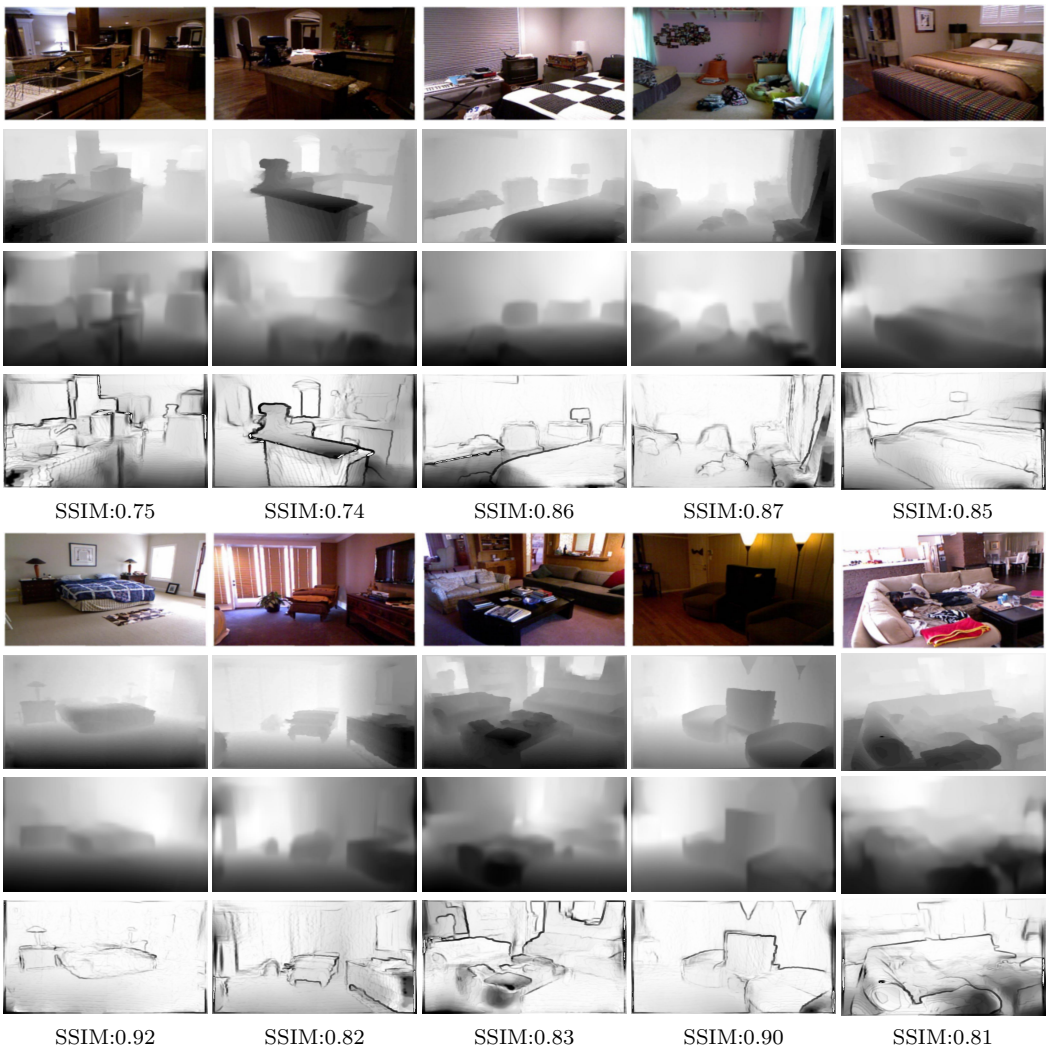


Figure 5.38: Qualitative results of our adversarial model approach (trained on KITTI) on images of the NYU depth v2 (cont.)

Specifically, when evaluating the SSIM metrics on the indoor NYU Depth V2 dataset using the outdoor-trained model, we observed a decrease in the SSIM scores compared to when using the model specifically trained on indoor data. This decrease suggests a degradation in the model's ability to capture the structural similarities between the predicted depths and the ground truth depths in the indoor scenes.

Similarly, when applying the indoor-trained model to the outdoor KITTI dataset, a decrease in performance was observed, indicating a mismatch between the model's learned features and the characteristics of the outdoor scenes. Since the model was primarily trained on indoor data, it lacks exposure to outdoor-specific variations and complexities, leading to difficulties in accurately estimating depths in outdoor environments.

Despite the decrease in performance during cross-dataset validation, our model still exhibits reliable performance overall. Further research and improvements in cross-domain generalization could enhance the model's performance and enable it to handle diverse datasets with higher consistency and accuracy.

5.5.7 Internet Images

We assess our model performance on random outdoor and indoor images downloaded from internet to provide preliminary insights into the model's performance. The testing images encompass a wide range of scenes, including both simple and complex scenarios.

We present the experimental results of evaluating the qualitative performance of our depth estimation model on a diverse set of indoor and outdoor images downloaded from the internet. Using internet images allows for assessing the model's performance on real-world, diverse data that may not be present in curated datasets. However, it also poses challenges related to data quality, ground truth availability, and potential biases.

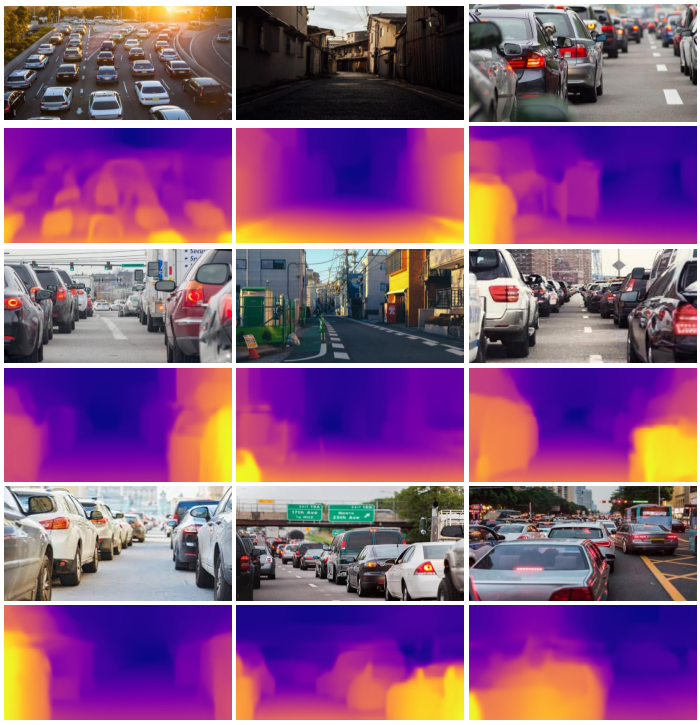


Figure 5.39: Qualitative results of our adversarial model on outdoor images from internet.

Upon visual examination, we observed distinct differences between the predicted depths using the NYU trained model and the KITTI trained model on the indoor and outdoor images. Both models demonstrated certain strengths and limitations depending on the environment they were trained on.

To evaluate the outdoor internet images, we employed the outdoor-trained model that was trained on the KITTI dataset. The outdoor images in Fig. 5.39 were sourced from various internet platforms and covered a wide range of outdoor scenes, including urban environments, and street views. Similar to the evaluation of indoor images, we relied on qualitative assessments to evaluate the model's performance on outdoor scenes. We examined the model's ability to handle complex scenes, dynamic objects, and challenging lighting conditions. Additionally, we assessed the accuracy and reliability of the estimated depth maps in representing the structure of the outdoor scenes.

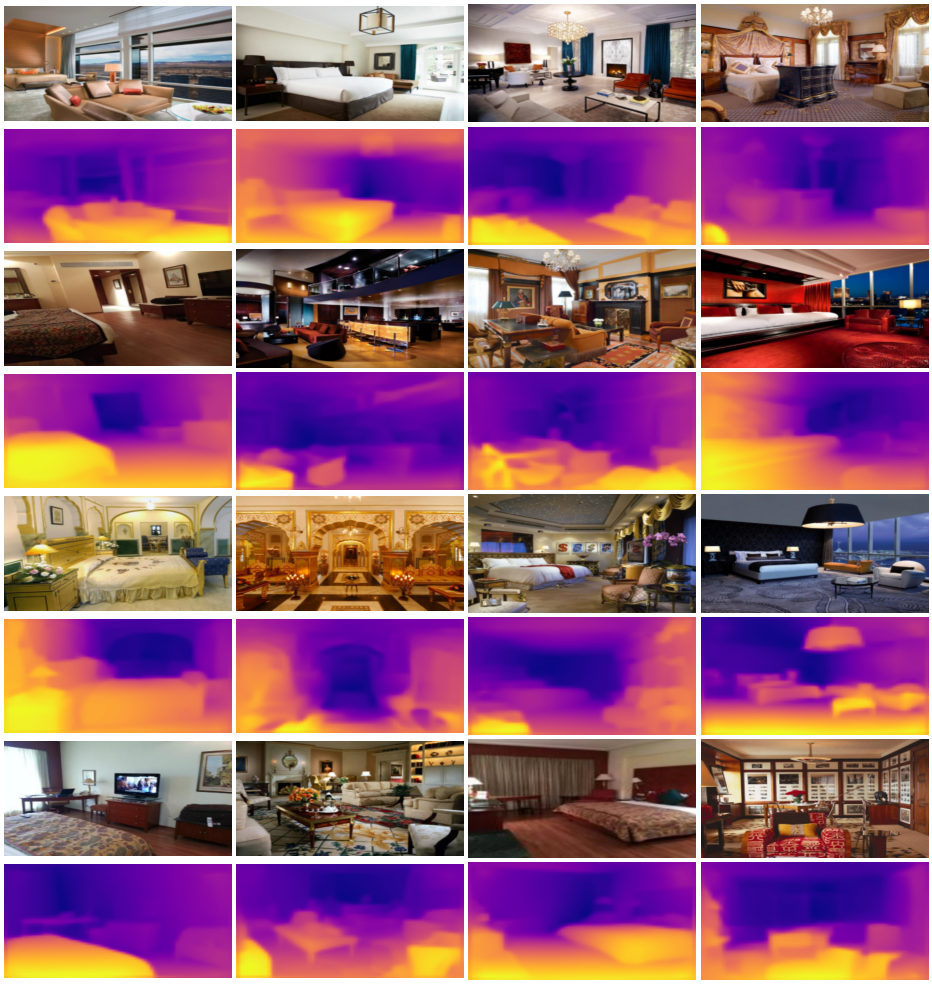


Figure 5.40: Qualitative results of our adversarial model on indoor images from internet.

In the case of indoor internet images, we utilized the indoor-trained model trained explicitly on the NYU Depth V2 dataset. Images in Fig. 5.40 represent diverse indoor scenes, including living rooms, and bedrooms. The objective is to assess the model's ability to estimate accurate depths in various indoor contexts. Due to the unavailability of ground truth depth information, we performed qualitative evaluations by visually inspecting the generated depth maps. We focused on evaluating the preservation of fine details, the representation of scene structure, and the overall visual quality of the depth maps.

Due to the absence of ground truth depth information in the downloaded internet images, the evaluation primarily relies on qualitative assessments rather than quantitative metrics. The focus is on the visual quality and perceptual accuracy of the generated depth maps in relation to the scene content and complexity.

5.5.8 Contrast Level

We present the experimental results of evaluating the qualitative performance of our depth estimation model on the KITTI dataset, specifically examining the impact of contrast variation on the predicted depth maps. We compared the predicted depths under normal, bright, and dark contrast conditions with the ground truth depth, using the Structural Similarity Index (SSIM) metrics as the evaluation criteria.

To assess the model's ability to handle contrast variation, we conducted a comprehensive analysis of the predicted depth maps across different contrast levels. We evaluate the robustness of our adversarial based model (TP-GAN) against light intensity by adjusting the contrast level. We compare our TP-GAN SSIM score with [4], [10], and our encoder-decoder model as shown in Fig. 5.41, 5.42, 5.43, and 5.44. We compare the similarity stucture of the adjusted constrast images (normal, brighter and darker) against the ground truth depth and then compute the error.

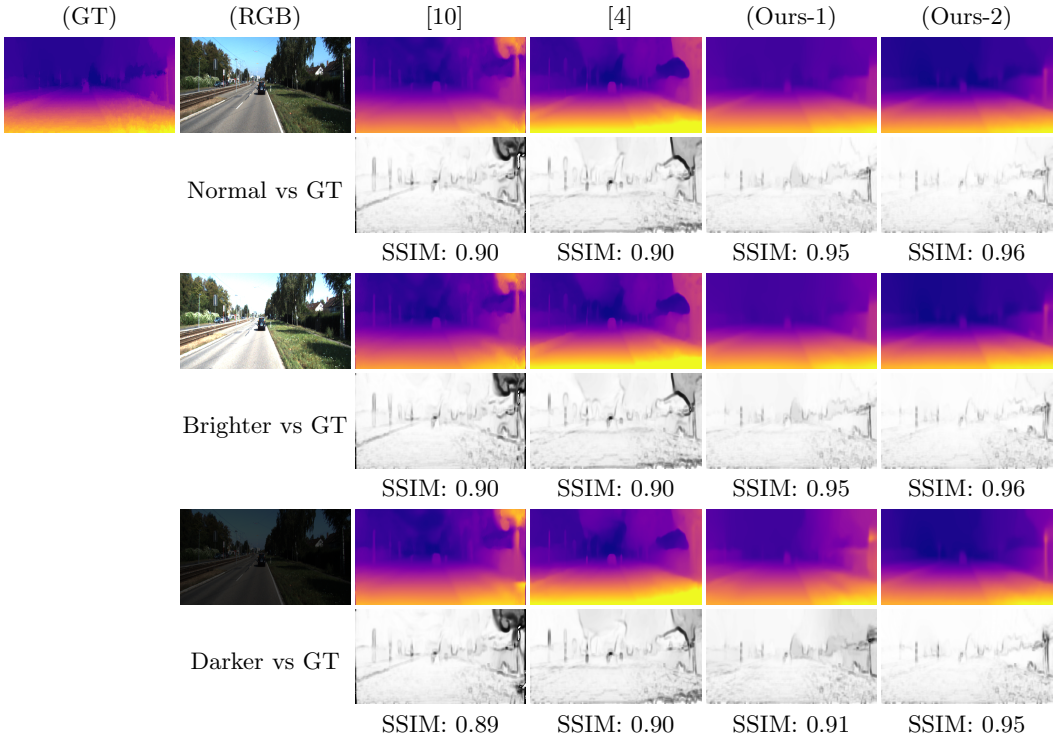


Figure 5.41: SSIM error on different contrast images; normal, brighter, and darker against ground truth

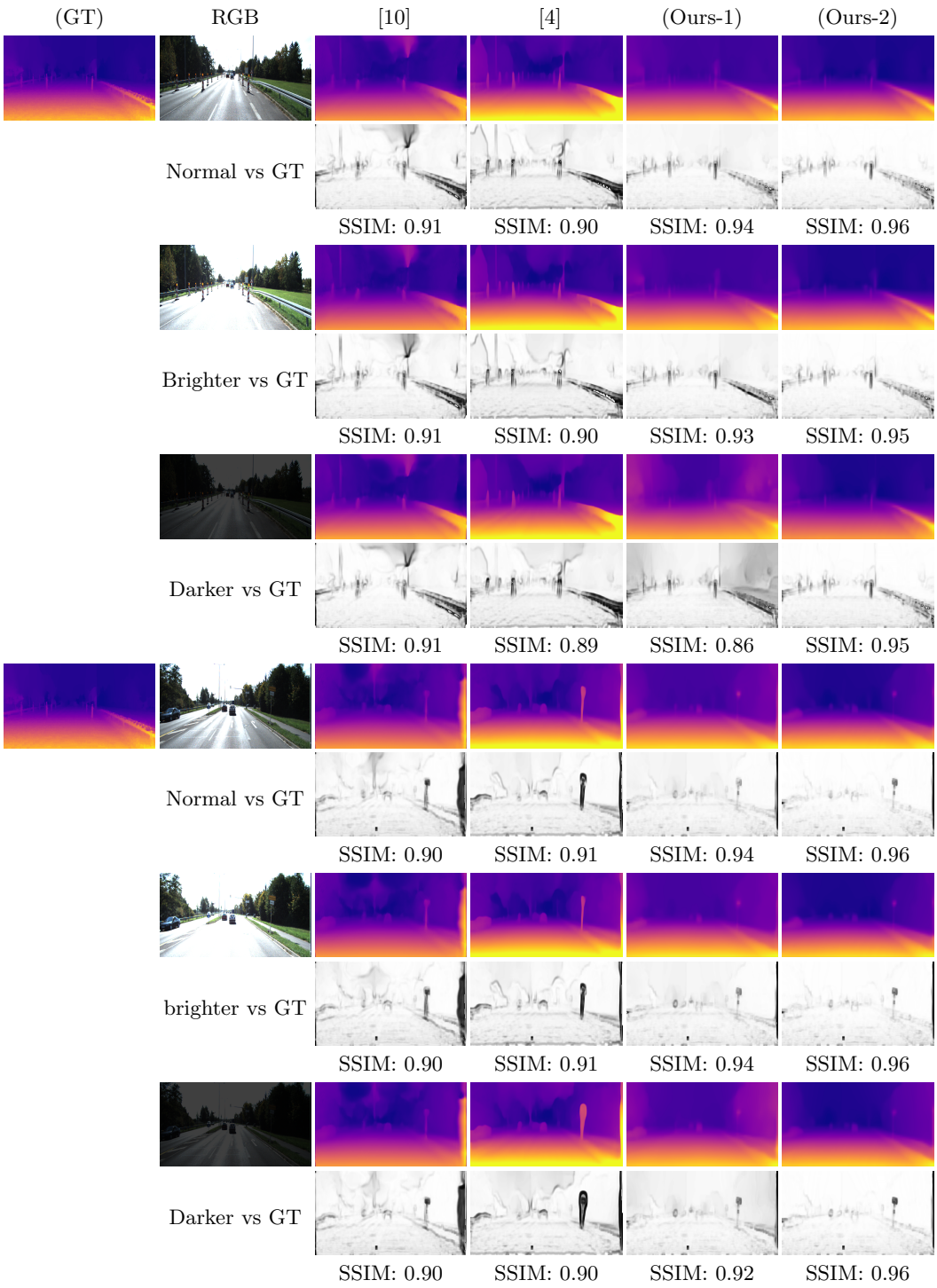


Figure 5.42: SSIM error on different contrast images; normal, brighter, and darker against ground truth (cont.)

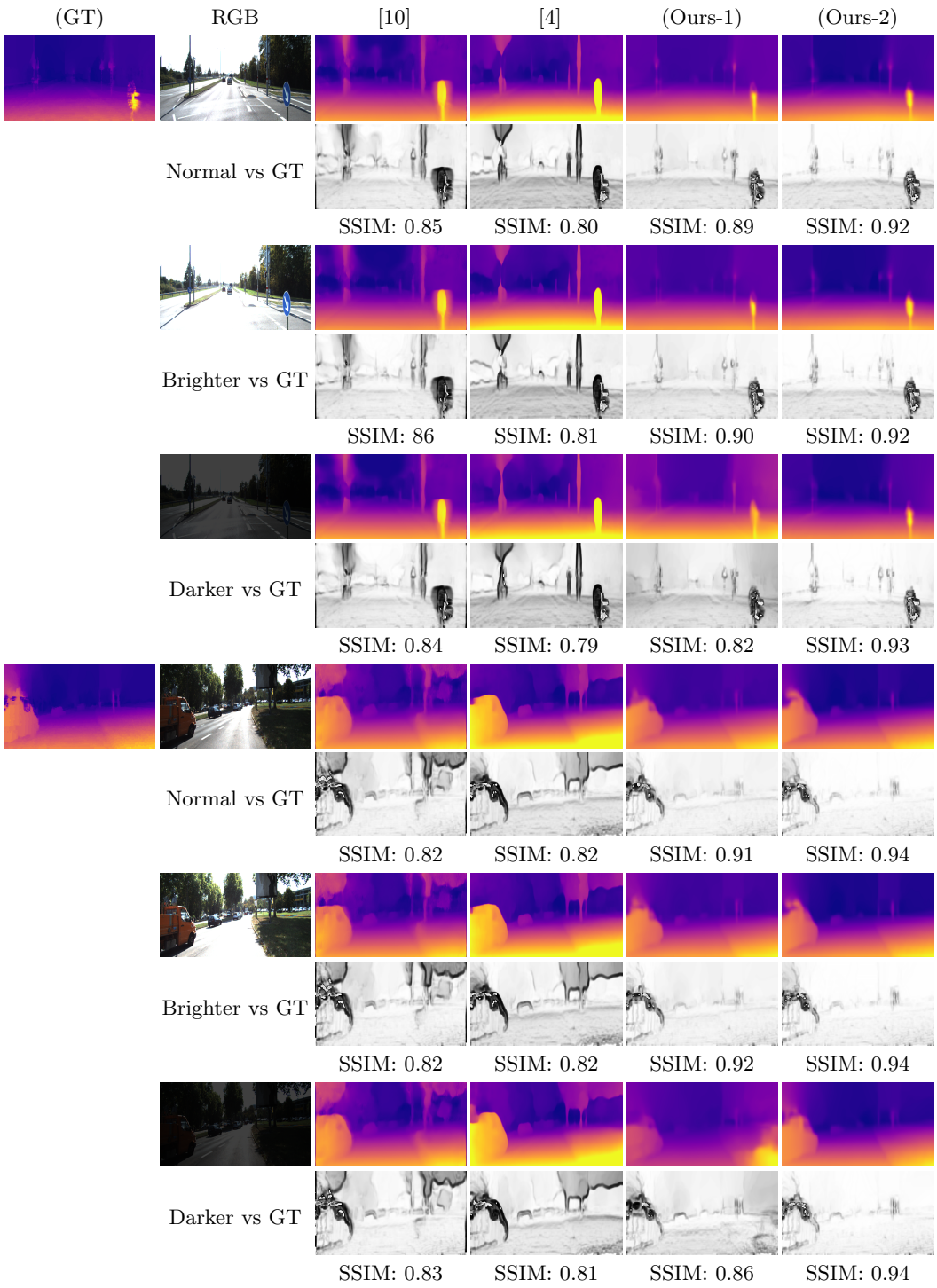


Figure 5.43: SSIM error on different contrast images; normal, brighter, and darker against ground truth (cont.)

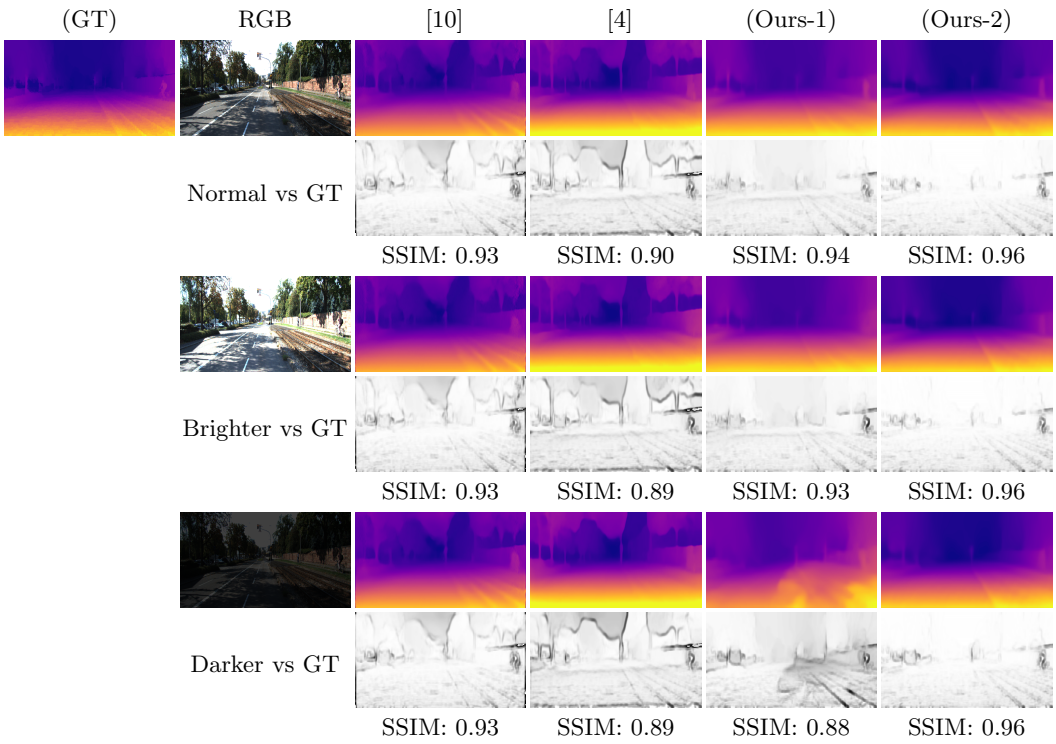


Figure 5.44: SSIM error on different contrast images; normal, brighter, and darker against ground truth

When comparing the predicted depths under normal contrast conditions, we observed a relatively high SSIM score, indicating a close resemblance between the predicted depths and the ground truth depths. The model demonstrated its capability to accurately estimate depths in scenes with balanced lighting and moderate contrast, effectively capturing structure of the scene.

Under brighter contrast conditions, where the scene's lighting was increased, the model's performance exhibited a slight decrease. The predicted depth maps showed some discrepancies when compared to the ground truth depths, suggesting a challenge in accurately estimating depths in scenes with high contrast. However, despite the decrease in SSIM scores, the model still captured the overall depth structure reasonably well, although with reduced accuracy and fidelity.

In contrast, under darker contrast conditions, where the scene's lighting was decreased, the model's performance also experienced a slight decrease in accuracy. The predicted depth maps exhibited more pronounced deviations from the ground truth depths, indicating a challenge in estimating depths accurately in low-contrast scenes. However, similar to the brighter contrast condition, the model still provided reliable depth information, albeit with a reduced level of accuracy.

These findings highlight the model's sensitivity to contrast variations and its ability to adapt to different lighting conditions. While the model performed best under normal contrast conditions, it showed a little decrease of accuracion for estimating depths in scenes with extreme contrast levels.

Additionally, We conduct experiments to evaluate the qualitative performance of our depth estimation model on the KITTI dataset, specifically focusing on the impact of contrast variation. We compared the predicted depth maps under normal contrast conditions with those under brighter and darker contrast conditions, using the Structural Similarity Index (SSIM) metrics as the evaluation criteria.

The objective of this analysis was to assess how well our model handled contrast variations and how it affected the quality of the predicted depth maps. By examining the SSIM scores, we could determine the degree of similarity between the predicted depths and the ground truth depths. We provide qualitative comparison in Fig. 5.45, 5.46, and 5.47.

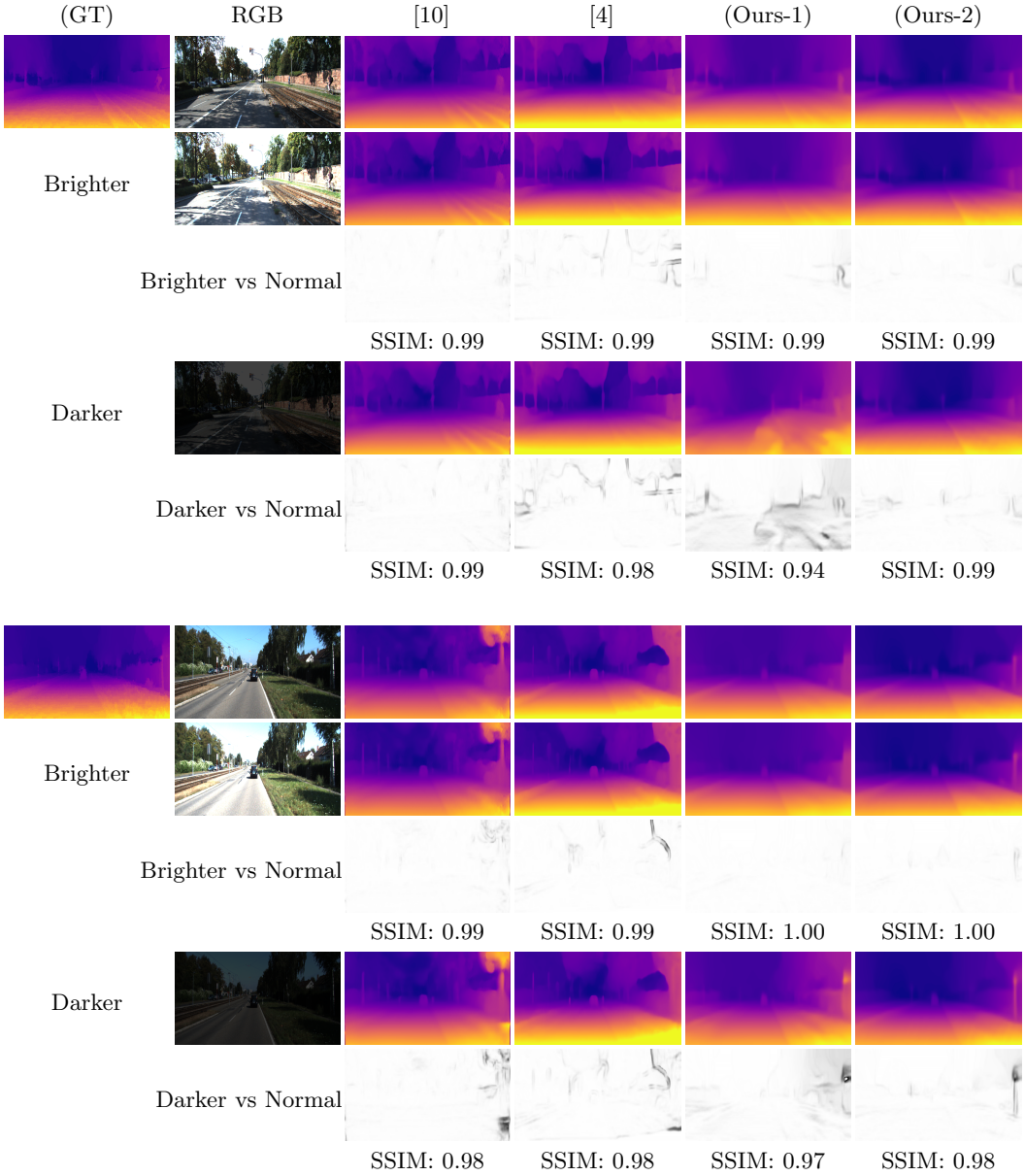


Figure 5.45: SSIM error on different contrast images; brighter, and darker against normal contrast.

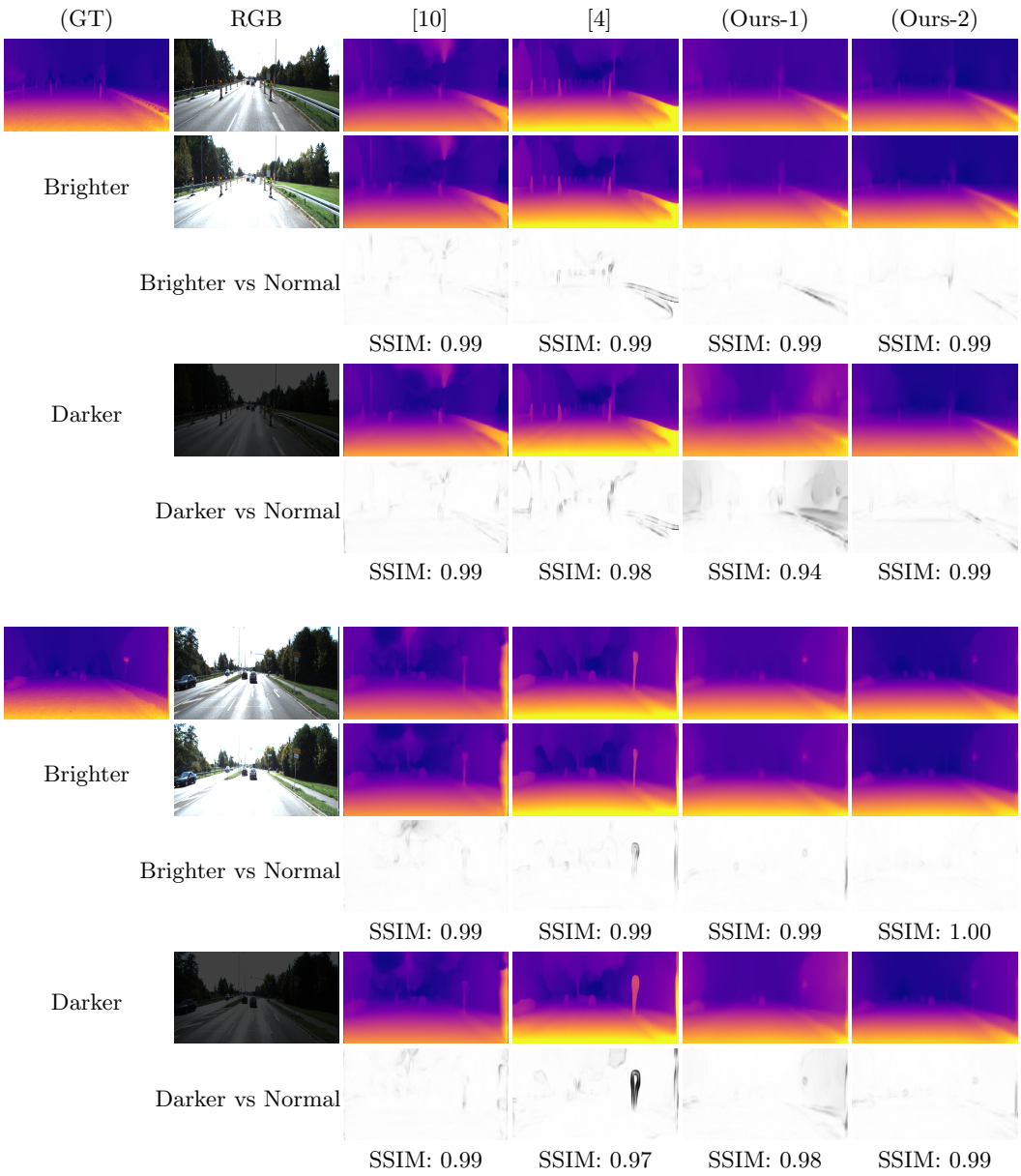


Figure 5.46: SSIM error on different contrast images; brighter, and darker against normal contrast (cont.)

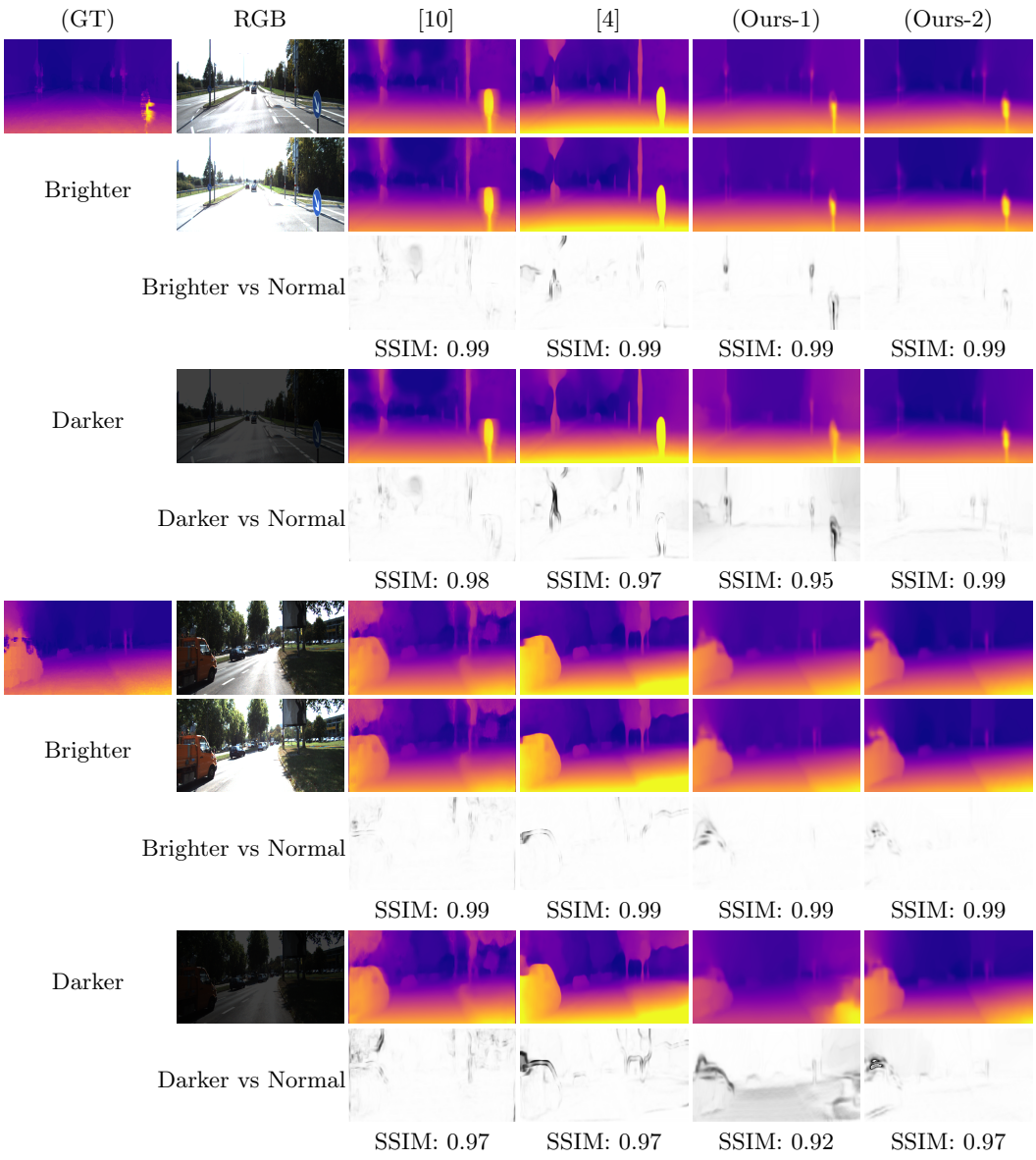


Figure 5.47: SSIM error on different contrast images; brighter, and darker against normal contrast (cont.)

We compare the performance of our model on images with normal contrast against images with brighter contrast. The objective is to evaluate how the model handles scenes with increased contrast levels. We selected images with normal contrast and adjusted a separate set for brighter contrast. The model's performance was assessed by computing the SSIM scores for the depth maps generated from these two sets of images. We analyze the results to determine if the model can accurately estimate depths in scenes with varying contrast levels.

Similarly, we evaluate the model's performance on images with normal contrast against darker ones. This comparison examines the model's ability to handle scenes with decreased contrast levels. Like the previous comparison, we selected a set of images with normal contrast and adjusted another set to have darker contrast. The SSIM scores are calculated for the depth maps generated from these sets of images. Analyzing the results, we assess the model's performance estimating depths in scenes

with reduced contrast.

The evaluation of contrasting conditions provides insights into how our model responds to variations in image contrast. By comparing the SSIM scores, we can determine whether changes in contrast levels affect the model's performance. We also examine the visual quality and reconstruction error of the depth maps in brighter and darker contrast conditions to better understand the model's robustness and reliability.

Overall, comparing depth estimation results between normal contrast and contrasting conditions (brighter and darker) using the SSIM metric allows us to evaluate the model's performance under different contrast scenarios. The results provide valuable insights into the model's ability to handle contrast variations and generate accurate depth maps.

5.6 Supplementary Results

In addition to all the previous results, we provide supplementary qualitative results to offer visual insights into the model's predictions, highlighting its strengths, weaknesses, and potential areas for improvement.

5.6.1 Depth from Different Environments

we present the experimental results of evaluating the qualitative performance of our depth estimation models on totally different environments compared to the training data. We conducted evaluations using three distinct types of images: natural images, underwater images, and coral reefs underwater images downloaded from internet. Additionally, we compared the performance of models trained on the NYU Depth v2 dataset and the KITTI dataset to assess their generalization capabilities across different environments.

Natural Images

In Fig. 5.48 and 5.49, we evaluated the models' performance on natural images. We conducted the evaluation using two trained models: one trained on the indoor NYU Depth V2 dataset and another trained on the outdoor KITTI dataset.

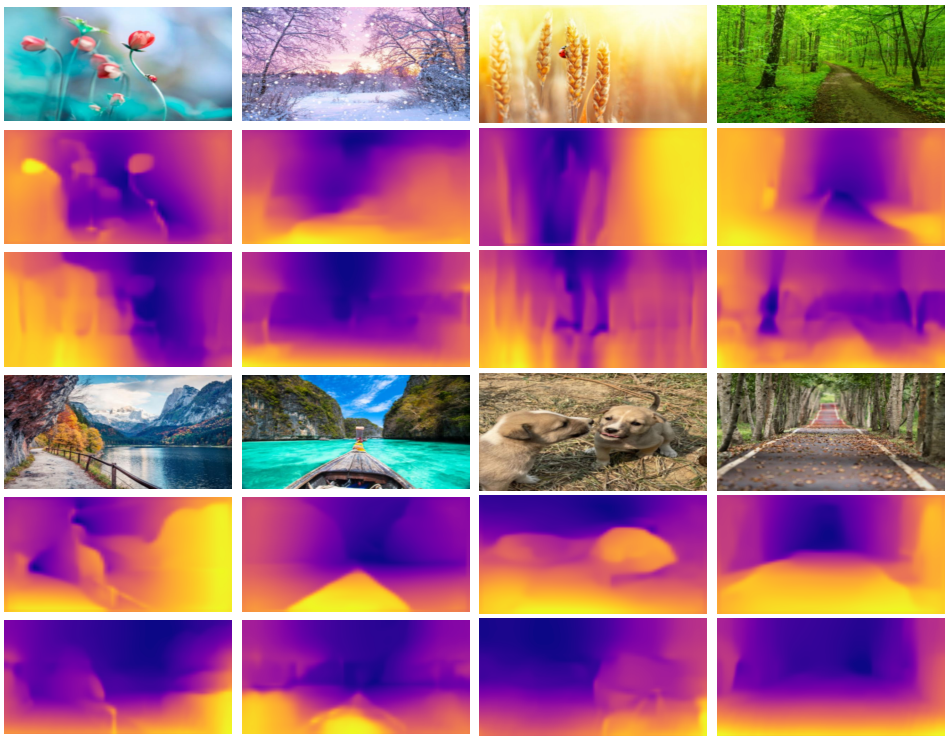


Figure 5.48: Qualitative results of our TP-GAN on nature images from internet, top (NYU) and bottom (KITTI).

In certain cases, using the NYU-trained model yielded better results, while in other cases, the KITTI-trained model outperformed the NYU-trained model. This indicates that the selection of the appropriate model depends on the specific characteristics and challenges of the natural image environment. The results demonstrated that both models are capable of generalizing to natural environments, despite being trained on different datasets with diverse characteristics.

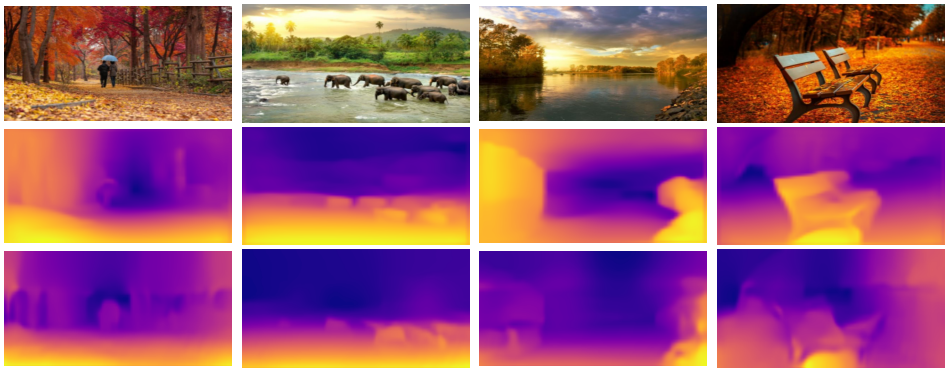


Figure 5.49: Qualitative results of our TP-GAN on nature images from internet, top (NYU) and bottom (KITTI) (cont.)

Underwater Images

We assessed the models' performance on underwater images. Underwater scenes present unique challenges due to light absorption, scattering, and color distortion.

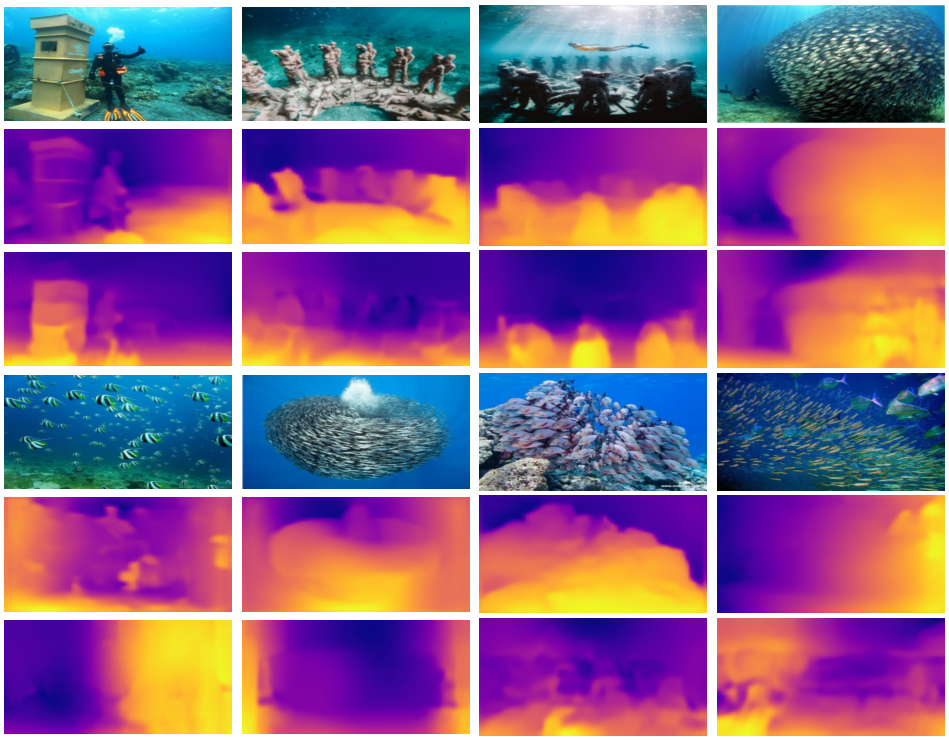


Figure 5.50: Qualitative results of our TP-GAN on underwater images from internet, top (NYU) and bottom (KITTI)

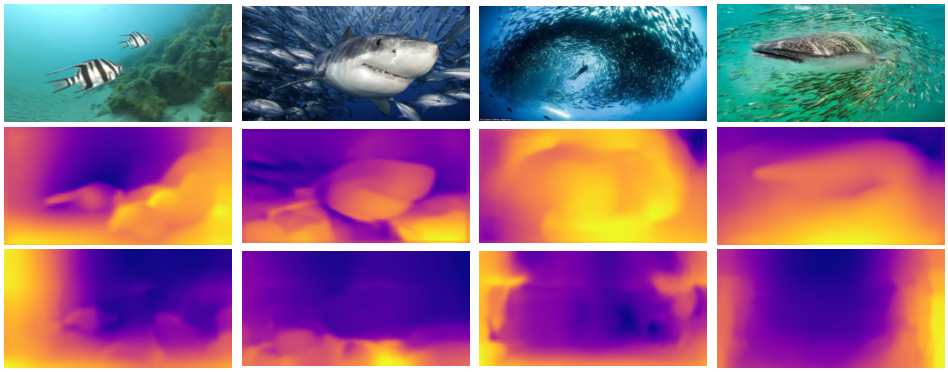


Figure 5.51: Qualitative results of our TP-GAN on underwater images from internet, top (NYU) and bottom (KITTI) (cont.)

The model trained on the KITTI dataset demonstrated reasonable qualitative performance, although with some limitations. It accurately estimated depths in most regions of the underwater images, providing valuable depth information for scene understanding.

However, there were instances where the model difficult to capture fine details and depth variations, potentially due to the differences between the training data and the specific characteristics of underwater scenes. Similarly, the model trained on the NYU Depth v2 dataset exhibited a comparable performance, suggesting its ability to adapt to underwater environments to some extent.

Our evaluation on underwater images using the NYU and KITTI trained models revealed that the choice of the trained model depends on the specific characteristics of the underwater environment.

Coral Reefs Images

Finally, we evaluated the models on coral reefs underwater images in Fig. 5.52 and 5.53. Coral reefs are known for their intricate structures and vibrant colors, making them a challenging environment for depth estimation. The model trained on the NYU Depth v2 dataset showed a reasonable performance, accurately capturing the overall structure of the coral reefs. However, there were instances where the model struggled

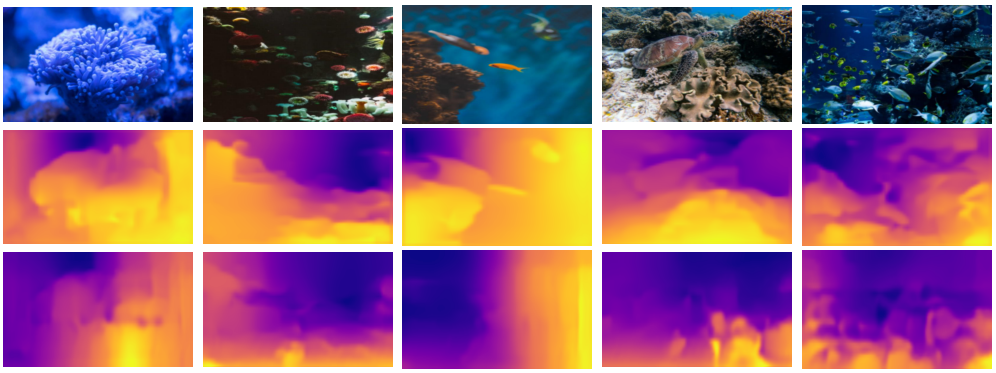


Figure 5.52: Qualitative results of our TP-GAN on coral reef images from internet, top (NYU) and bottom (KITTI)

to accurately estimate depths in regions with complex textures and color variations. The model trained on the KITTI dataset exhibited difficulty of generalizing across vastly different environments.

These results underscore the challenges and limitations of applying depth estimation models to environments that significantly differ from the training data. While both models demonstrated a degree of adaptability, they showed varying levels of performance depending on the environment. Future research should focus on developing models that are specifically trained on data from the target environment to achieve more accurate and reliable depth estimation results. Additionally, techniques such as domain adaptation and transfer learning can be explored to enhance the models' generalization capabilities across diverse environments.

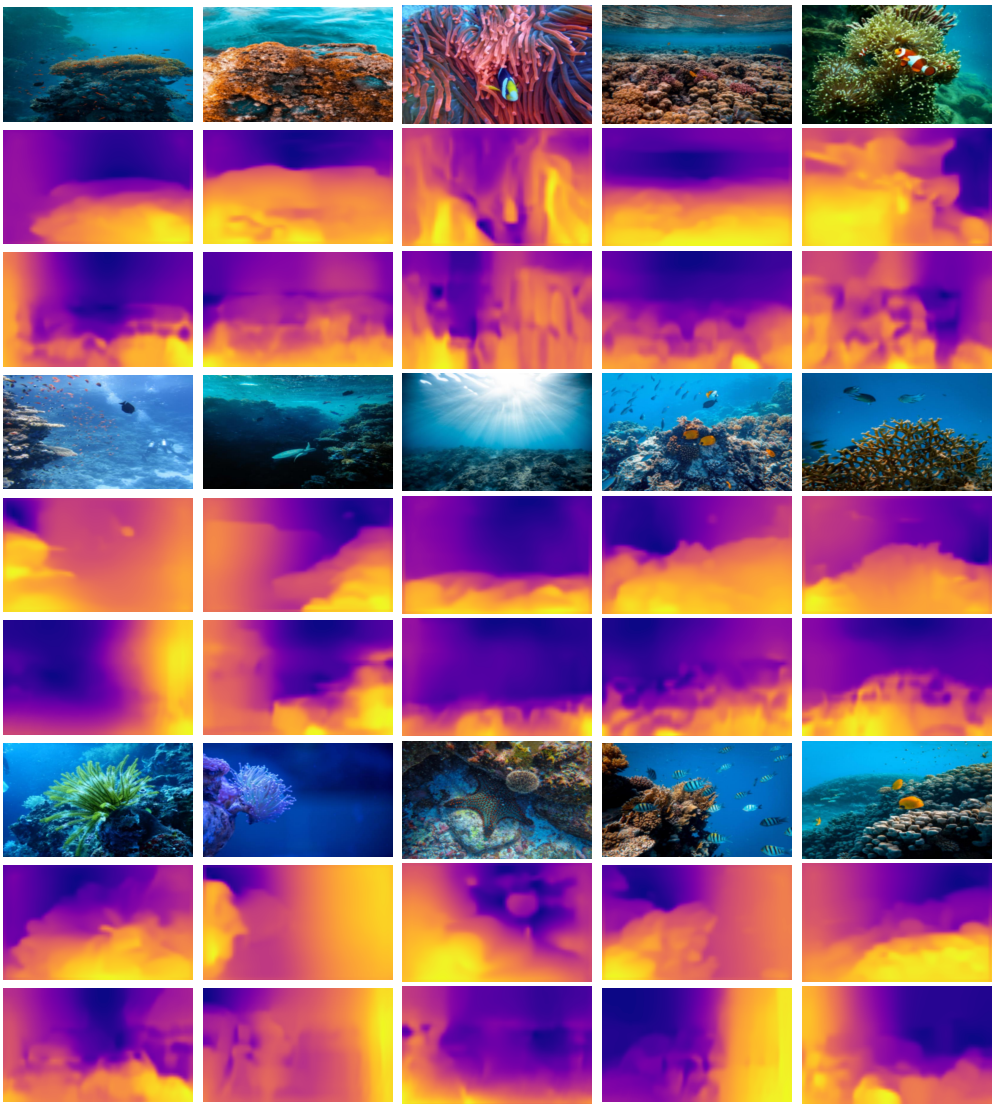


Figure 5.53: Qualitative results of our TP-GAN on coral reef images from internet, top (NYU) and bottom (KITTI) (cont.)

Note that some performance degradation was observed when the model was applied to datasets with significantly different characteristics from the training dataset.

5.6.2 3-D Point Clouds

We apply our model to generate 3-D point cloud visualizations from two different environments: indoor NYU and underwater coral reef sample images.

We present the experimental results of evaluating the qualitative performance of our depth estimation models in generating 3D point clouds based on the predicted depth maps. The 3D point clouds provide a geometric representation of the scene, allowing for a comprehensive understanding of the spatial structure and depth information.

Indoor NYU Images

The indoor NYU images, which are captured in controlled indoor environments, provided a favorable setting for our depth estimation models. The generated 3D point clouds exhibited smooth surfaces, accurate object shapes, and well-defined depth transitions. The point clouds accurately represented the relative distances between objects and conveyed a realistic sense of depth.

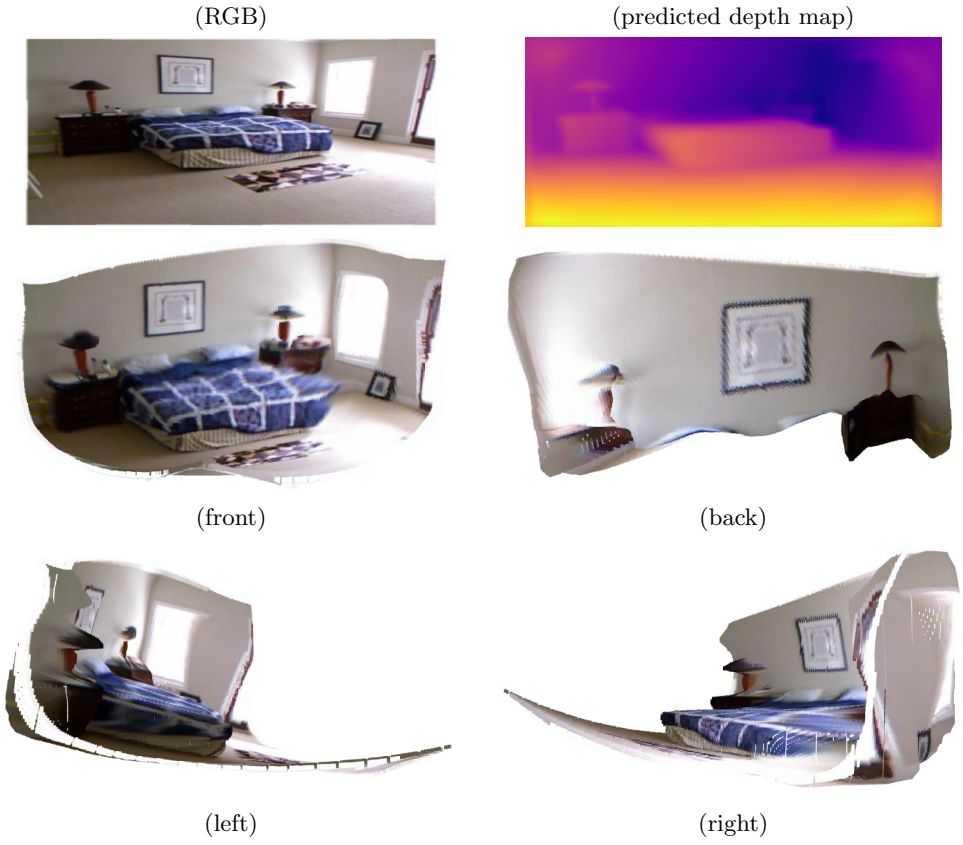


Figure 5.54: 3-D point cloud of sample NYU

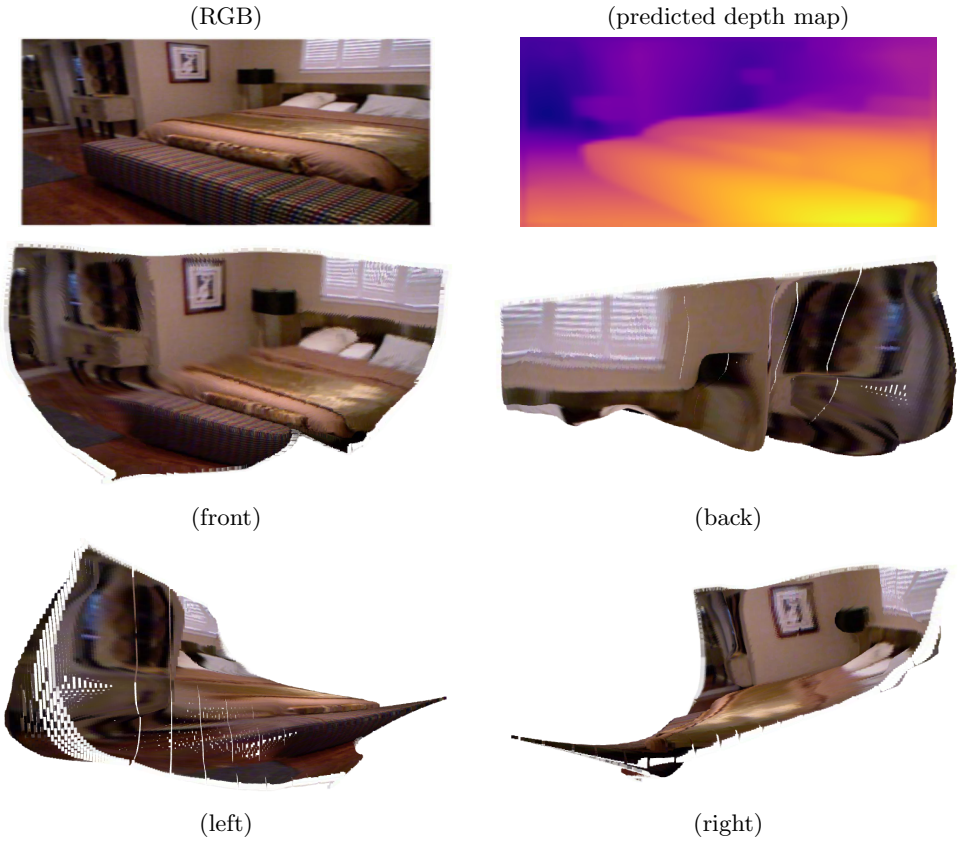


Figure 5.55: 3-D point cloud of sample NYU (cont.)

Coral Reefs Images

On the other hand, generating 3D point clouds from coral reef images presented more challenges due to the complex and dynamic nature of underwater scenes. Despite these challenges, our depth estimation models still produced reasonably accurate and visually plausible 3D point clouds from the coral reef images. However, the level of detail and accuracy in the generated point clouds was comparatively lower than those generated from the indoor NYU images.

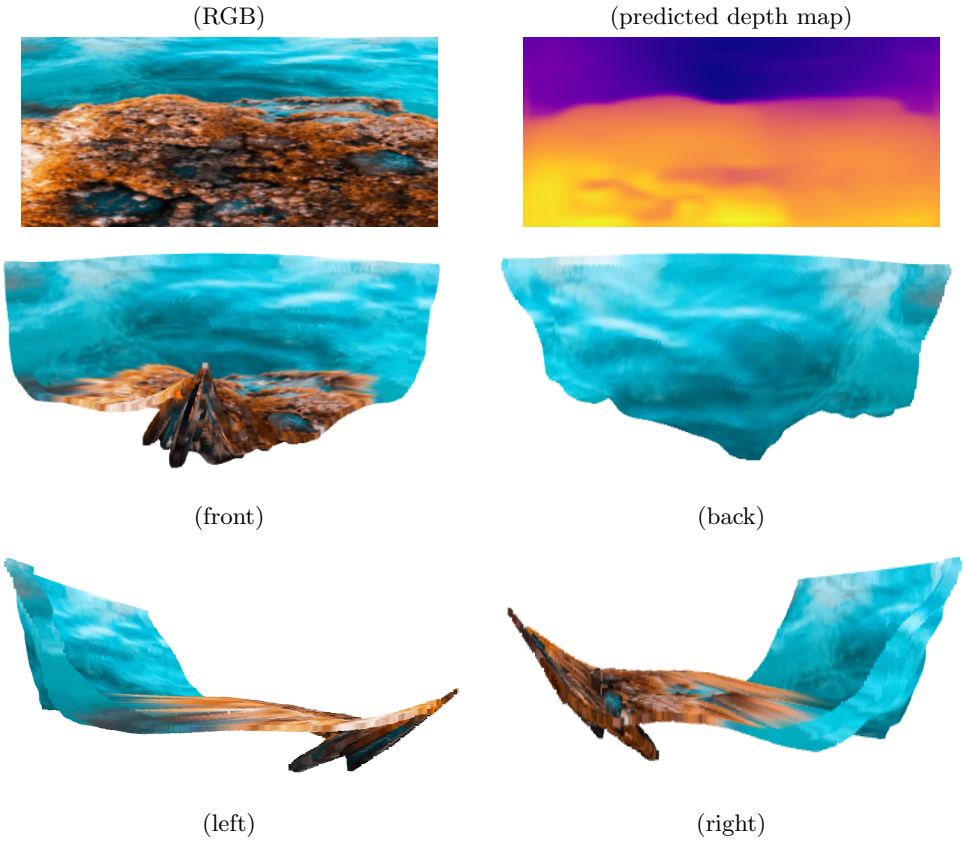


Figure 5.56: 3-D point cloud of sample coral reef images

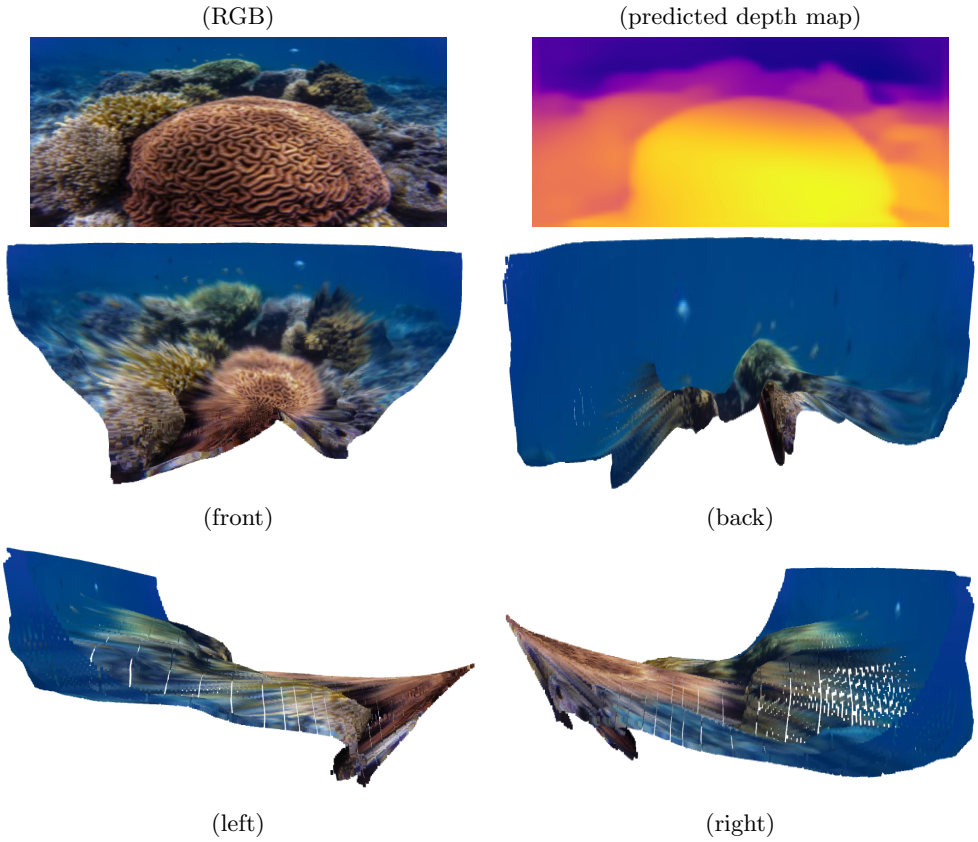


Figure 5.57: 3-D point cloud of sample coral reef images (cont.)

The results reveal that our depth estimation models successfully produce visually plausible and coherent 3D point clouds. The point clouds exhibit a consistent and connected structure, capturing the general shape and layout of the objects in the scene. Despite the absence of ground truth reference, the generated point clouds display a high level of detail, preserving fine geometric features such as object boundaries, surface irregularities, and depth variations.

Moreover, the generated 3D point clouds exhibit accurate representations of depth transitions, suggesting that our models effectively capture depth cues from the input images. The point clouds convey a sense of depth and spatial relationships between objects, contributing to a realistic and immersive representation of the scene's 3D structure.

While the lack of ground truth 3D point clouds limits our ability to quantitatively evaluate the accuracy of the generated point clouds, our qualitative assessment indicates that our depth estimation models are capable of producing visually plausible and reasonably accurate 3D representations of indoor scenes.

These results highlight the potential of our depth estimation models for applications that rely on 3D reconstruction and scene understanding. The visually plausible and coherent 3D point clouds generated from the predicted depths provide valuable insights into the scene's geometry and depth structure, paving the way for various applications in computer vision, robotics, and augmented reality.

5.6.3 Depth from videos

we present the experimental results of evaluating the qualitative performance of our depth estimation models in generating depth maps from videos. Specifically, we evaluated our models using two types of videos: high-resolution videos downloaded from the internet and low-resolution videos captured by a car dash camera, to evaluate using our outdoor KITTI trained model.

The high-resolution videos downloaded from the internet provided a diverse range of scenes and visual content. The depth estimation models were able to accurately estimate depth maps from these videos, capturing the scene's structure and depth information with good fidelity. The generated depth maps exhibited clear boundaries, accurate depth transitions, and preserved fine details, resulting in visually pleasing representations of the scenes.

(Internet video)

Figure 5.58: Depth generated from high-resolution video.

On the other hand, the low-resolution videos captured by the car dash camera presented additional challenges due to their lower quality and limited visual information. Despite these limitations, our depth estimation models were still able to generate reasonably accurate depth maps from these videos. The generated depth maps captured the overall depth structure of the scenes and provided a meaningful representation of the depth variations, despite the lower resolution and potential noise in the input videos. Qualitatively evaluating the generated depth maps from both types of videos revealed the capabilities and limitations of our models in capturing depth information. The high-resolution videos demonstrated the ability of our models to produce highly detailed and accurate depth maps. In contrast, the low-resolution videos showcased the models' adaptability to lower-quality inputs, generating depth maps that still conveyed the general depth structure of the scenes.

The visual inspection and analysis of the generated depth maps from high-resolution and low-resolution videos provided valuable insights into the performance and robustness of our depth estimation models in video-based depth estimation.

(Dash camera video)

Figure 5.59: Depth generated from low-resolution video.

These results demonstrate the potential of our depth estimation models in generating accurate and visually appealing depth maps from videos. The ability to estimate depth from videos opens up opportunities for applications in video analysis, object tracking, scene understanding, and augmented reality, among others.

Chapter 6

Discussion

In this study, we proposed a novel depth estimation method based on deep learning method using our proposed architectures. We evaluate depth estimation on indoor NYU Depth v2 up to maximum distance of 10 meters and 80 meters for the outdoor KITTI dataset. We delivered an in-depth analysis and interpretation of the research findings.

6.1 Model Performance

We compare our proposed architectures with some of the most notable single image depth estimation methods. We consider a range of approaches to be compared that span from the pioneering work [2] to the some current methods.To confirm an adequate and meaningful evaluation, we analyze the effectiveness of our model using the same metrics and similar dataset split validation technique as Eigen *et al.* [2].

6.1.1 Quantitative Results

We examine our model with several previous adversarial networks and non-adversarial methods on NYU Depth v2, as shown in Tab. 5.4 and Tab. 5.5. In the adversarial approach, ours performs slightly accuracy lower than [26] in the first threshold ($\delta < 1.25$) but perform better in other metrics with a significant margin. Compared to the non-adversarial based methods, our generative model outperforms our encoder-decoder model and the preceding works [2, 23, 24, 58, 59, 60, 61] and achieves comparative performance, even better than [6, 62]. Here, ours achieves better than [62] on the thresholds ($\delta < 1.25$ and $\delta < 1.25^2$), and error rate performances (root mean square error (RMS) and the average log error (LOG10)). Whereas [6] performs better only in the first threshold ($\delta < 1.25$) with a small margin, our generative consistently improves performance in the other two thresholds ($\delta < 1.25^2$ and $\delta < 1.25^3$) and performs the lowest RMS with a large margin.

We report the performance of comparison with several similar strategies on the KITTI dataset, both adversarial and non-adversarial. In terms of metrics of interest, as demonstrated in Tab. 5.6 and 5.7, our technique surpasses our encoder-decoder model and all the nine previous adversarial works [44, 45, 46, 47, 48, 63, 64, 65, 66] as well as non-adversarial methods [2, 3, 4, 5, 9, 10, 67, 68, 69, 70] by significant margins for all the three thresholds $\delta < 1.25$, $\delta < 1.25^2$, and $\delta < 1.25^3$, but performs lower than the work in [50] with a small margin. While transformer attention models have shown remarkable success in [50], their implementation presents a number of disadvantages

in comparison to encoder-decoder or generative models. Using transformer attention models for depth estimation is limited primarily by their complexity and increased more GPU memory requirements.

6.1.2 Qualitative Results

We provide qualitative visualization results for more analysis of our proposed method. We compare our predicted depth with the work of [2] in Fig. 5.8, [3] in Fig. 5.10, [4, 5, 6] in Fig. 5.11 and also the works of [7, 8] in Fig. 5.12 on the NYU Depth v2 dataset. To ensure a reasonable visualization comparison, we use similar sample images adopted from their paperworks. It demonstrates that our proposed approach is sufficient to generate more reproducible image depth estimation performance, in which some results are close to their ground truths.

Meanwhile, the performance of our depth estimation on the KITTI data along with the works [2, 3, 9] and [10] are shown in Fig. 5.13 and in Fig. 5.14 to that of [9] from differen sample images. Compared to their output depth, it can be seen that our method yields more visually satisfying predictions with more visible transitions that correlate with local depth information. We show that our strategy is more proficient at detecting the proper depth structure of the image for both datasets.

Further, we demonstrate the effectiveness of our proposed method in generating consistent better depth visualization by visualizing their SSIM error reconstruction images, and calculating the SSIM scores. Fig. 5.15, 5.16, 5.17, 5.18 and Fig. 5.19, 5.20, and 5.21 show the effective of our model to predicet depth on NYU and KITTI, respectively. In overall, our method prediction on achieves a good performance in which some results are relative to the ground truth, as represented by their SSIM error scores being close to 1.

In Fig. 5.22, 5.23, 5.24 and Fig. 5.25, 5.26, 5.27, we study the depth value distribution by analyzing the histogram from the ground truths and its predicted images for NYU and KITTI data, respectively. This analysis offers insights into the distribution characteristics and potential discrepancies between the predicted depths and the ground truth.

The histogram analysis revealed interesting patterns in the distribution of the depth values. We computed histograms for both the predicted depth values and the ground truth values, using 256 number of bins to capture the distribution across the entire range of depth values.

Upon comparing the histograms, we observed that the predicted depth values exhibited some obscured deviations from the ground truth. This skew indicated a tendency towards overestimation in certain depth ranges. Notably, we noticed a small number of outliers in the predicted depth values, representing extreme cases where the model struggled to estimate the depths accurately. Despite these limitations, it is important to note that the depth estimation system still demonstrated overall promising performance. The majority of the predicted depth values aligned reasonably well with the ground truth, indicating that the model captured the underlying depth structure to a satisfactory extent.

6.2 Conciseness

We demonstrated that the model we proposed in this research is rather concise, yet its performance is reliable. We focus on the ability of our model to achieve opti-

mal performance while maintaining simplicity and efficiency in terms of the model architecture and the number of parameters.

6.2.1 Architecture

To assess the conciseness of our model, we analyze its architecture and evaluating the investigate layers in our model. Our objective is to strike a balance between model complexity and performance, ensuring that our model is both effective and efficient. By emphasizing simplicity and efficiency of our model, we aim to provide valuable insights and advancements in depth estimation, enabling more practical and efficient solutions for various applications.

We reconstruct residual networks (ResNet-50) following two blocks of the up-sampling layer in the first stage of our encoder-decoder model. While in the second stage, we stacked five instead of four convolutional layers and reduced the filter size, which is proved effectively improved the final depth prediction. Since both networks on each stage consist of a very shallow layer and small filter size, our model is inexpensive with respect to the number of network parameters and produces a more accurate output image depth prediction.

Our adversarial model is comprised of only three simple sub-models, the first of which is a ResNet50V2-based generator sub-model. The second sub-model, a discriminator, consists of a stack of six convolution layers constructed as a patch GAN model. The final sub-model consists of a series of six convolution blocks which the first five blocks comprising convolution, batch normal- ization, Relu activation, and dropout. The last block contains a convolution layer following a linear activation.

We confirmed that regardless of its simple structure, the proposed architectures effectively improves the overall depth prediction performance of the model.

6.2.2 Parameteres

We focus on reducing excessive parameter redundancy without sacrificing depth estimation accuracy. We aim to achieve a more concise representation of the depth estimation task by optimising the model's parameter count.

The two architectures in this research utilizes their parameters efficiently. As shown in Tab. 5.8, 5.9, and 5.10 we could affirm that our proposed architectures strives to achieve good performance with a minimal number of parameters.

Our encoder-decoder method requires much fewer network parameters and less amount of input training data. For instance, compare with the work in [2] , our model required number of iterations about 234K vs 3.5M and with fewer input training data, 50K vs 120K for the NYU v2 depth data. While, for the KITTI data our model only need about 80K vs 3.5M iteration numbers and required 17K vs 40K samples training data.

Meanwhile, our adversarial model has around 59.2M training parameters, which 51.7M for the generator, 7M for the discriminator, and 520K for the refiner sub-model. During testing, only the refiner parameter is taken into account. It takes about 39.5 minutes and 50 minutes to finish one epoch for training KITTI and NYU data, respectively, measured in a single 8GB NVIDIA GeForce GTX 1080. All the results presented in this thesis work, the training process typically takes around 36 epochs for the NYU and 28 epochs for the KITTI dataset to converge with a batch size of 16.

In comparison to Eigen et al. work [2], our adversarial model utilizing $50K$ vs. $120K$ for training on the NYU v2 depth data and around $25K$ vs. $40K$ for the KITTI data. Whereas 25K vs. 39K training data on KITTI compared with the works [9, 10].

6.3 Robustness

We discuss the robustness of our method by analyzing its performance on different lighting conditions, and evaluating its generalization capabilities across two different datasets.

6.3.1 Different lighting conditions

In this study, we investigated the robustness of our depth estimation model under different contrast level. Variations in contrast level can result in differences in the intensity, color, and distribution of light, thereby affecting the overall appearance of the scene. We recognize that lighting variations, particularly in KITTI data, pose significant challenges to accurate depth estimation, and thus, it is crucial to assess the performance of our model under various lighting scenarios.

To evaluate the performance of our model, we generate visualization estimated depth and compute the SSIM score comprising images captured under varying contrast level conditions. We evaluate six random KITTI data which consists of scenes with normal contrast, lower contrast and higher contrast level as shown. We evaluate the robustness of our model performance on adjusted contast level (normal, brighter and darker) against ground truth in Fig. 5.36, 5.37, and 5.38. Whereas, SSIM reconstruction error of normal image contrast is computed against brighter and darker image in Fig. 5.39, 5.40, and 5.41. This evaluation helps assess the model's adaptability to different lighting conditions and provides insights into its real-world performance.

We conducted a comparative analysis to benchmark our model against existing depth estimation methods on the same dataset. The results demonstrated that our model outperformed other methods, reflected the SSIM metrics consistently indicated superior performance across the dataset. This comparison validates the robustness of our approach and highlights its superiority in accurately estimating depth under varying lighting conditions.

Although our model exhibited robustness, there are some limitations to consider. Certain extreme lighting conditions, such as scenes with extremely low-light or scenes with overexposed regions, still posed challenges to the accuracy of depth estimation.

The qualitative evaluation of our depth estimation model on KITTI data with contrast variation revealed variations in performance under different contrast levels. The model achieved accurate depth estimation under normal contrast conditions, but faced challenges under both brighter and darker contrast conditions. These results emphasize the importance of further development to enhance the model's adaptability to scenes with extreme contrast, leading to more accurate and reliable depth estimation in practical applications.

6.3.2 Cross dataset adaptation

We perform cross dataset validation by training on one dataset and testing on another to evaluate the generalization of the proposed architecture across diverse datasets. The results of the cross-dataset validation indicated promising performance of the proposed

method while trained using indoor NYU data and testing on outdoor KITTI as shown in Fig. 5.28, 5.29, and 5.30.

However, it is important to note that the proposed method exhibited certain limitations when applied to different datasets. In some cases, our model performance decreases while examining indoor data for the outdoor trained model. As shown in Fig. 5.31, 5.32, and 5.33 trained KITTI model has difficulty generating depth for some particular objects on the NYU indoor dataset. The observed performance degradation can be attributed to the differences in data distribution and characteristics between the training and testing datasets. NYU scene introduced novel variations in terms of scene complexity, lighting conditions, and object compositions, which were not adequately captured during the training phase on KITTI data.

These results emphasize the importance of domain-specific training for achieving optimal performance in depth estimation tasks. The inherent differences between indoor and outdoor scenes, such as lighting conditions, object appearances, and scene complexities, pose challenges for models trained on one domain when applied to another.

Chapter 7

Conclusion and Future work

7.1 Conclusion

Performance of our two model architectures have been demonstrated. First, we proposed a distinct multi-stages architecture in the form of smaller residual network for predicting depth from a single image along with the multi-loss and adjustable learning rates. Our model achieves a reliable yet better performance compare with several previous related works while required fewer iteration number and input training data.

Next, the use of an additional sub-model to integrate global scene structure and local scene information in a generative adversarial network (GAN)has been successfully demonstrated for single image depth estimation. We confirmed that regardless of its simple structure, the presence of the third player (TP) in adversarial learning effectively improves the overall depth prediction performance of the model. Extensive experimental results demonstrate that employing a third player along with the SSIM loss is beneficial in a single image depth estimation. The global performance of our proposed method revealed adequate depth prediction. we demonstrated that our proposed model required less training time to converge compared with the aforementioned related methods regardless the GPU device.

7.2 Future Work

Our future work is encouraging to develop a robust single image depth estimation, greater generalization capability across different datasets to be applied not only for indoor or outdoor data, but also will be applicable for such a complex environment e.g. underwater or coral reefs.

Bibliography

[1] D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *International Journal of Computer Vision*, vol. 47, pp. 7–42, 2001.

[2] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *Proc. of the 27th Int. Conf. on Neural Inf. Process. Syst.*, vol. 2, pp. 2366–2374, 2014.

[3] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields," *Proc. IEEE on Computer Vision and Pattern Recognition (CVPR)*, vol. 38, no. 10, pp. 2024–2039, June 2015.

[4] C. Godard, O. M. Aodha, M. Firman, and G. Brostow, "Digging into self-supervised monocular depth estimation," *Proc. The International Conference on Computer Vision (ICCV)*, pp. 3828–3838, October 2019.

[5] Z. Wang, S. Liu, and Y.-J. Liu, "Towards Better Generalization: Joint Depth-Pose Learning without PoseNet," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9151–9161, 2020.

[6] J.-W. Bian, H. Zhan, N. Wang, T.-J. Chin, C. Shen, and I. Reid, "Auto-rectify network for unsupervised indoor depth estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, pp. 9802–9813, 2021.

[7] W. Yuan, X. Gu, S. Zhu, and P. Tan, "New CRFs: Neural Window Fully-connected CRFs for Monocualr Depth Estimation," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3916–3925, 2022.

[8] A. Agarwal and C. Arora, "Attention Attention Everywhere: Monocular Depth Prediction with Skip Attention," *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 5861–5870, 2023.

[9] Y. Kutznietzov, J. Stuchler, and B. Leibe, "Semi-supervised deep learning for monocular depth map prediction," *In CVPR*, pp. 6647–6655, 2017.

[10] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," *Proc. IEEE on CVPR*, pp. 6602–6611, July 2017.

[11] D. Crandall, A. Owens, N. Snavely, and D. Huttenlocher, "Discrete-continuous optimization for large-scale structure from motion," *Proc. IEEE on CVPR*, vol. 35, pp. 3001–3008, June 2011.

[12] H. Ha, S. Im, J. Park, H. G. Jeon, and I. S. Kweon, "High-quality depth from uncalibrated small motion clip," *Proc. IEEE on CVPR*, pp. 5413–5421, June 2016.

[13] P. Fua and Y. G. Leclerc, "Object-centered surface reconstruction: combining multi-image stereo and shading," *Int. J. of Comput. Vis.*, vol. 16, pp. 35–56, September 1995.

[14] A. N. Rajagopalan, S. Chaudhuri, and U. Mudenagudi, "Depth estimation and image restoration using defocused stereo pairs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1521–1525, November 2004.

[15] D. Hoiem, A. A. Efros, and M. Hebert, "Automatic photo pop-up," *Proc. of ACM SIGGRAPH*, pp. 577–584, August 2005.

[16] A. Saxena, S. H. Chung, and A. Y. Ng, "Learning depth from single monocular images," *Proc. of Int. conf. on Neural Inf. Process. Syst.*, vol. 18, pp. 1161–1168, December 2005.

[17] A. Saxena, M. Sun, and A. Y. Ng, "Make3D: Learning 3D scene structure from a single still image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 824–840, May 2009.

[18] F. Liu, C. Shen, and G. Lin, "Deep convolutional neural fields for depth estimation from a single image," *Proc. IEEE on Computer Vision and Pattern Recognition (CVPR)*, vol. 07, pp. 5162–5170, June 2015.

[19] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," *Proc. of Int. Conf. on 3D Vision*, pp. 239–248, October 2016.

[20] Y. Cao, Z. Wu, and C. Shen, "Estimating depth from monocular images as classification using deep fully convolutional residual networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 11, pp. 3174–3182, 2018.

[21] X. Chen, X. Chen, and Z. Zheng-Jun, "Structure-aware residual pyramid network for monocular depth estimation," *Proc. of the 28th International Joint Conferences on Artificial Intelligence*, pp. 694–700, 2019.

[22] W. Yin, J. Zhang, O. Wang, S. Niklaus, L. Mai, S. Chen, and C. Shen, "Learning to recover 3D scene shape from a single image," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (CVPR)*, pp. 204–213, 2021.

[23] S. Gur and L. Wolf, "Single image depth estimation trained via depth from defocus cues," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (CVPR)*, pp. 7683–7692, 2019.

[24] H. Ye and D. Xu, "Inverted pyramid multi-task transformer for dense scene Understanding," *Proc. of the European Conference on Computer Vision.*, pp. 514–530, 2022.

[25] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, and Y. Bengio, "Generative Adversarial Nets," *Advances in Neural Information Processing Systems*, vol. 27, pp. 2672–2680, 2014.

[26] D.-h. Kwak and S.-h. Lee, "A Novel Method for Estimating Monocular Depth Using Cycle GAN and Segmentation," *Sensors*, vol. 20, no. 9, pp. 2567–2573, 2020.

[27] D. S. Tan, C.-y. Yao, C. Ruiz, Jr., and H. Kai-Lung, "Single-Image Depth Inference Using Generative Adversarial Networks," *Sensors*, vol. 19, no. 7, pp. 1708–1723, 2019.

[28] A. C. Kumar, S. M. Bhandarkar, and M. Prasad, "Monocular Depth Prediction using Generative Adversarial Networks," *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 413–4138, 2018.

[29] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," *ArXiv*, no. 1411.1784, 2014.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE on CVPR*, pp. 770–778, June 2016.

[31] P. Isola, J.-y. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image Translation with Condiitonal Adversarial Networks," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, 2017.

[32] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Depth from familiar objects: a hierarchical model for 3D scenes," *Proc. IEEE on CVPR*, vol. 2, pp. 2410–2417, June 2006.

[33] E. Coupeté, F. Moutarde, and S. Manitsaris, "Gesture recognition using a depth camera for human robot collaboration on assembly line," *Procedia Manufacturing*, vol. 3, pp. 518–525, 2015.

[34] M. Mancini, G. Costante, P. Valigi, and T. A. Ciarfuglia, "Fast robust monocular depth estimation for Obstacle Detection with fully convolutional networks," *Proc. IEEE Int. Conf. Intell. Robot. Syst. (IROS)*, pp. 4296–4303, July 2016.

[35] T. Kim, M. Motro, P. Lavieri, S. S. Oza, J. Ghosh, and C. Bhat, "Pedestrian Detection with Simplified Depth Prediction," *International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2712–2717, 2018.

[36] A. K.-F. Lui, Y.-H. Chan, and M.-F. Leung, "Modelling of Destinations for Data-driven Pedestrian Trajectory Prediction in Public Buildings," *IEEE International Conference on Big Data (Big Data)*, pp. 1709–1717, 2022.

[37] A. K.-F. Lui, Y.-H. Chan, and M.-F. Leung, "Modelling of Pedestrian Movements near an Amenity in Walkways of Public Buildings," *International Conference on Control, Automation and Robotics (ICCAR)*, pp. 394–400, 2022.

[38] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[39] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, pp. 346–359, 2008.

[40] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2564–2571, 2011.

[41] J. Zbontar and Y. LeCun, "Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches," *Journal of Machine Learning Research.*, vol. 17, no. 1, pp. 1–32, 2016.

[42] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," *Proc. of Eur. Conf. on Comput. Vis. (ECCV)*, vol. 7546, pp. 746–760, October 2012.

[43] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *International Journal of Robotics Research (IJRR)*, vol. 32, no. 11, pp. 1232–1237, 2013.

[44] F. Aleotti, F. Tosi, M. Poggi, and S. Mattoccia, "Generative Adversarial Networks for Unsupervised Monocular Depth Prediction," *Proc. of the European Conference on Computer Vision (ECCV)*, pp. 337–354, 2018.

[45] C. Zheng, T.-J. Cham, and J. Cai, "T2Net: Synthetic-to-Realistic Translation for Solving Single-Image Depth Estimation Tasks," *Proc. of Eur. Conf. on Comput. Vis. (ECCV)*, pp. 767–783, 2018.

[46] A. C. Kumar, S. M. Bhandarkar, and M. Prasad, "Monocular Depth Prediction Using Generative Adversarial Networks," *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 413–421, 2018.

[47] A. Pilzer, X. Dan, M. M. Puscas, E. Ricci, and N. Sebe, "Unsupervised Adversarial Depth Estimation Using Cycled Generative Networks," *International Conference on 3D Vision (3DV)*, pp. 587–595, 2018.

[48] C. Zhao, G. G. Yen, Q. Sun, C. Zhang, and Y. Tang, "Masked GANs for Unsupervised Depth and Pose Prediction with Scale Consistency," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 12, pp. 5392–5403, 2021.

[49] R. Ranftl, A. Bochkovsky, and V. Koltun, "Vision transformer for dense prediction," *In IEEE International Conference on Computer Vision (ICCV)*, pp. 12159–12168, 2021.

[50] G. a. Manimaran, "Focal-WNet: an architecture unifying convolution and attention for depth estimation," *In IEEE International Conference on Computer Vision (ICCV)*, pp. 1–7, 2022.

[51] A. Hendra and Y. Kanazawa, "Smaller Residual Network for Single Image Depth Estimation," *IEICE Trans. Inf. and Syst.*, vol. E104-D, no. 11, pp. 1991–2001, November 2021.

[52] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," *Proc. IEEE on CVPR*, pp. 9799–9809, December 2018.

[53] F. Chollet *et al.*, "Keras," *https://keras.io*, accessed August 28. 2022.

[54] A. Hendra and Y. Kanazawa, "TP-GAN: Simple Adversarial Network with Additional Player for Dense Depth Image Estimation," *IEEE Access*, vol. 11, pp. 44176–44191, 2023.

[55] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: from Error Visibility to Structural Similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[56] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: a system for large-scale machine learning," *Proc. of USENIX conference on Operating Systems Design and Implementation*, pp. 265–283, 2016.

[57] A. C. Wilson, R. Roelofs, S. Mitchell, N. Srebro, and B. Recht, "The Marginal Value of Adaptive Gradient Methods in Machine Learning," *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 4151–4161, 2018.

[58] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantics labels with a common multi-scale convolutional architecture," *Proc. of International Conference on Computer vision (ICCV)*, pp. 2650–2658, 2015.

[59] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. Yuille, "Towards Unified Depth and Semantic Prediction from a Single Image," *Proc. IEEE on CVPR*, pp. 2800–2809, June 2015.

[60] A. Roy and S. Todorovic, "TMonocular Depth Estimation Using Neural Regression Forest," *Proc. IEEE on CVPR*, pp. 5506–5514, June 2016.

[61] A. Chakrabarti, J. Shao, and G. Shakhnarovich, "Depth from a single image by harmonizing overcomplete local network predictions," *Proc. of the 30th International Conference on Neural Information Processing System (NIPS)*, pp. 2566–2674, December 2016.

[62] J. Li, C. Yuce, R. Klein, and A. Yao, "a two-streamed network for estimating fine-scaled depth maps from single RGB images," *Computer Vision and Image Understanding*, vol. 186, pp. 25–36, September 2019.

[63] Y. Almalioglu, M. R. U. Saputra, P. P. B. d. Gusmo, A. Markham, and N. Trigoni, "GANVO: Unsupervised Deep Monocular Visual Odometry and Depth Estimation with Generative Adversarial Networks," *Proc. of International Conference on Robotics and Automation (ICRA)*, pp. 5474–5480, 2019.

[64] S. Li, F. Xue, X. Wang, Z. Yan, and H. Zha, "Sequential Adversarial Learning for Self-Supervised Deep Visual Odometry," *Proc. of the IEEE International Conference on Computer Vision*, pp. 2851–2860, 2019.

[65] M. M. Puscas, D. Xu, A. Pilzer, and N. Sebe, "Structured Coupled Generative Adversarial Networks for Unsupervised Monocular Depth Estimation," *International Conference on 3D Vision (3DV)*, pp. 18–26, 2019.

[66] R. Groenendijk, S. Karaoglu, T. Gevers, and T. Mensink, "On the Benefit of Adversarial Training for Monocular Depth Estimation," *Computer Vision and Image Understanding*, vol. 190, pp. 102848–102858, 2020.

[67] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. Reid, "Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction," *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 340–349, 2018.

[68] Y. Zou, Z. Luo, and J.-B. Huang, "DF-Net: unsupervised joint learning of depth and flow using cross-task consistency," *In European Conference on Computer Vision (ECCV)*, pp. 38–55, 2018.

[69] A. Ranjan, V. Jampani, L. Balles, K. Kim, D. Sun, J. Wulff, and M. J. Black, "Competitive Collaboration: joint unsupervised learning of depth, camera motion, optical flow and motion segmentation," *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12240–12249, 2019.

[70] J.-W. Bian, H. Zhan, N. Wang, Z. Li, L. Zhang, C. Shen, M.-M. Cheng, and I. Reid, "Unsupervised scale-consistent depth learning from video," *INternational Journal of Computer Vision*, pp. 2548–2564, 2021.

Acknowledgements

This thesis paper is the outcome of my hard work, long days and nights, moments of desperation, anxiety, excitement, curiosity, creativity, and happiness mixed together. During this journey, several people directly or indirectly contributed to the final result. Some of them followed this project from the beginning until the end. To these, I would like to express my deepest appreciation.

First and foremost, I would like to express my greatest gratitude to my honorable supervisor Dr. Yasushi Kanazawa, whose guidance, patience, and encouragement have been invaluable during my study and research. Without his guidance, it would be impossible for me to conduct this research project.

I would also like to extend my deepest appreciation to Professor Jun Miura and Professor Shigeru Kuriyama for their valuable insight and precious time, helping me to review and improve my thesis.

To the faculty and staff of the Toyohashi University of Technology for their support, encouragement, and resources during my academic career. Their dedication and commitment to academic excellence have inspired me to pursue my academic goals.

To all the staff and everyone at the Japan International Cooperation Agency (JICA) for their support and warm welcome since I came to Japan. I am so proud to be a part of the JICA scholarship grantee.

My heartfelt thanks to my family, who have supported me endlessly. My lovely parents, sisters, and brother, for their constant encouragement. My uncle and aunts also gave me a lot of support, especially by visiting me to encourage me to keep up with my studies. To my beloved ones, thank you for all the great memories and for constantly sending me a wake-up call and reminding me about my research progress.

To all my friends in our laboratory members for their friendly welcome, for teaching me to speak awesome daily Japanese, playing sports, gundam and tamiya together, taking me to enjoy the wonderful culture of Japan, singing karaoke, having dinner and coffee together, and also helping me whenever I could not things understand.

My last words go to all Indonesian friends without exception. I got a lot of support from them. Thank you for all your support, encouragement, and inspiration.

List of Publications

Journal Articles

[1] Andi Hendra and Yasushi Kanazawa, Smaller Residual Network for Single Image Depth Estimation, IEICE Transactions, Vol. E104-D, No. 11, pp.1992-2001, Nov. 2021.

[2] Andi Hendra and Yasushi Kanazawa, TP-GAN: Simple Adversarial Network with Additional Player for Dense Depth Image Estimation, IEEE Access, Vol. 11, pp. 44176-44191, 2023.

International Conference Papers

[1] Andi Hendra and Yasushi Kanazawa, Depth Estimation from a Single image using multi Stream and Scale Deep Learning, ICAICTA2019, Yogyakarta, Indonesia, Sept. 20-22, 2019.

[2] Andi Hendra and Yasushi Kanazawa, Accuracy Improvement of Depth Estimation from A Single Image using 3rd Player in GAN, The 7th IIEEJ International Conference on Image Electronics and Visual Computing (IEVC 2021), Yumehall Shiretoko, Shiretoko (Shari), Hokkaido, Japan, Sept. 8-11, 2021 (Online).