Adversarial Training for Unsupervised Monocular Depth Estimation

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science
in
Computer Science and Engineering
by Research

by

Ishit Mehta
201403008
ishit.mehta@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
July 2019

Copyright © Ishit Mehta, 2019
All Rights Reserved

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "**Adversarial Training for Unsupervised Monocular Depth Estimation**" by **Ishit Mehta**, has been carried out under my supervision and is not submitted elsewhere for a degree.

_____ _____

Date Adviser: Prof. P J Narayanan

Acknowledgments

This thesis is a culmination of research ideas that have originated from numerous discussions involving Prof. P J Narayanan and Parikshit Sakurikar. Their constant support and advice were indispensable in this journey.

I would like to express my sincere gratitude towards Kalpit, Rajvi and Saurabh for paying heed to my ramblings and for teaching me lessons that I will carry forward.

I am also grateful to have been a part of the IIIT community which has provided me with more than I could have asked for.

Abstract

The problem of estimating scene-depth from a single image has seen great progress recently. It is one of the foundational problems in computer vision and hence been studied from various angles. Since the advent of deep learning, most of the approaches are data driven. These methods train high-capacity models with large amount of data in an end-to-end fashion. They rely on ground-truth depth, which is hard to capture and process.

Recently self-supervised methods have been proposed which rely on view supervision as an alternative. These methods minimize photometric reconstruction error in order to learn depth. In this work, we propose a geometry-aware generative adversarial network to generate multiple novel views from a single image. Novel views are generated by learning depth as an intermediate step. The synthesized views are discerned from real images using discriminative learning. We show the gains of using the adversarial framework over previous methods. Furthermore, we present a structured adversarial training routine to train the network, going from easy examples to difficult ones. The combination of adversarial framework, multi-view learning, and structured training produces state-of-the-art performance on unsupervised depth estimation for monocular images.

We also compare our method with human depth perception by conducting a series of experiments. We investigate the existence of monocular depth cues like relative size, occlusion and height in the visual field in artificial vision systems. With quantitative and qualitative experiments, we highlight the shortcomings of artificial depth perception and propose future avenues for research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The human visual system has been under extensive scrutiny since the beginning of human consciousness. Questions pertaining to the inception of concepts like visual encoding and semantic understanding, are still being pondered upon. Notwithstanding the difficulty of some of these questions, the vitality of vision in human evolution is uncontested. Perhaps the most curious aspect about human vision is our ability to perceive the layout of objects in environments around us. In understanding spatial layout of our surroundings, perceiving the distance (or depth) of an object from the position of the viewer becomes fundamental. Without this perception, even locomotion and navigation become difficult. To build intelligent systems mimicking humans, it is imperative to endow them with 3D perception. To this end, there have been efforts towards understanding biological vision systems and using them as inspiration. This thesis is one of such several attempts at building artificial depth perception, which is founded on a few fundamental elements of human vision.

Although human depth perception is highly intricate and difficult to replicate, it is grounded on a set of fundamental depth cues which can be broadly categorized into monocular or binocular. Stereopsis, convergence and shadow stereopsis [60] are some of the binocular cues, which when combined with monocular cues like motion parallax, object size, lighting and occlusion help in robust depth perception. Stereopsis or binocular disparity is the most dominant feedback amongst these cues while also being the most widely studied aspect of human vision [5, 23, 35]. The differences in the two 2D images captured by the two horizontally displaced eyes, indicate the depth of objects in a scene. Motion parallax is the other dominant cue and it uses the relative motion of objects with respect to the observer. One can judge the distance of an object from its apparent motion in the direction opposite to the motion of the viewer.

Some of these cues are simple enough to be expressed as computer algorithms, while others are more complicated. Estimating depth from single images has been one of the long standing problems in computer vision and there have been a number of solutions. Early methods try to take inspiration from biological depth cues and leverage geometry to extract 3D scene information. However, with the success of Convolutional Neural Networks (CNNs), there has been a shift from modelling geometry to modelling neural networks to incorporate priors and semantics. This chapter delineates this shift and highlights the major issues in previous methods.

1.1 Artificial Depth Perception

Majority of the literature in computer vision which address 2D to 3D perception is based on replicating the two dominant cues *viz.,* stereopsis and motion parallax. However, estimating depth from a stereo camera evolves into a much simpler problem of stereo matching or disparity mapping. In contrast, depth from monocular videos is a severely under-constrained problem, with estimating depth from monocular *images* being even harder since we do not have access to the most dominant monocular cue, *viz.*, motion parallax.

Despite the challenges, from early methods like shape-from-shading and shape-from-defocus to recent semantics based techniques, monocular depth estimation has come a long way. It has enabled a variety of computer vision applications like post-capture scene refocusing [88], parallax simulation for virtual reality, fog simulation and higher-level computer vision tasks like object recognition. Robotic systems which need to be deployed in restrictive locations cannot be equipped with multiple cameras. For instance, invasive surgeries like endoscopy are performed by inserting tiny cameras into the body to analyze critical areas that need be paid attention to. Depth maps also provide richer scene understanding while being used to analyze object interactions in the scene. Moreover, images of objects with corresponding depth maps can be used to recover the shape and the material of the object as well as estimate the relevant BRDF (Bidirectional Reflective Distribution Function). Recently, the applications have become widespread, especially with smartphone cameras which provide post-capture aesthetic controls.

The following sections in this chapter discuss the previous attempts on single-image depth prediction, along with their successes and failures. We highlight some of the challenges in recent methods and later propose a few solutions.

1.1.1 Geometry Based Strategies

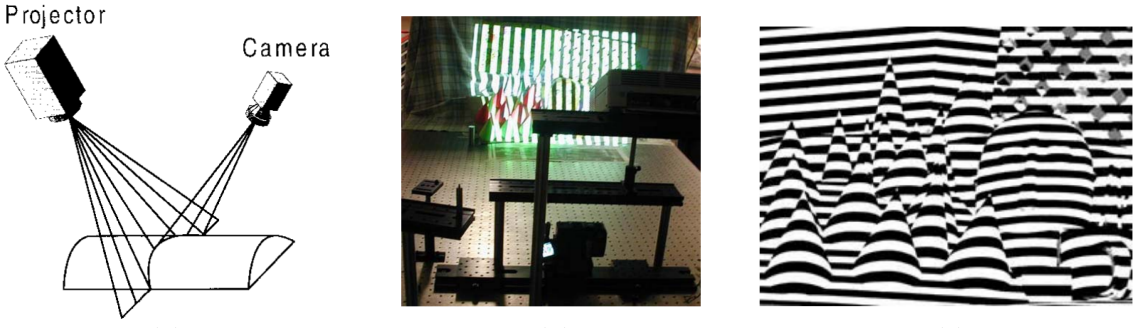


Figure 1.1: Active depth estimation using geometric light patterns. (a) shows the projector and camera configuration used by Albrecht *et al.* [2]. (b) shows a scene illuminated by gray codes to capture ground-truth depth in Middlebury [65] dataset. (c) shows a grayscale image of the same scene.

Geometry based depth estimation algorithms can be categorized into active methods and passive methods [62]. The active methods use an external energy source (e.g., light) to strategically change the

Input Depth Map using Defocus

Figure 1.2: Depth from defocus cue used by Shi *et al.* [68]. They use small-scale defocus blur present in camera lens to get crude estimation of depth, which is further refined using semantic cues. Adapted from [68].

scene, which is captured passively to estimate distances. The active methods, in most cases do not work for dynamic scenes but are in general more accurate than passive methods.

1.1.1.1 Active Methods

Using incandescent light sources to illuminate the scene, was one of the earliest active depth estimation strategies [49]. By observing the scene under different lighting conditions, depth can be inferred from the light reflected off the objects. This method, known as photometric stereo, was proposed by Woodham [80]. Horn *et al.* [28] propose shape from shading to solve the harder case of using a single image with a given lighting condition. Such methods however are not suitable for objects with specularity since they depend heavily on the colour of the objects. Projecting incandescent light in known geometric patterns and capturing the distortions in these patterns due to the layout of the scene can result in accurate depth measurements. These structured-light techniques [2] typically use a single camera to capture a scene illuminated by known light patterns projected using a single projector as shown in Figure 1.1. Scharstein *et al.* (Middlebury dataset) [65] use stereo cameras and gray code light patterns to capture near perfect depth and pixel-accurate stereo disparities. Figure 1.1 shows the Cones scene from Middlebury illuminated by the light patterns.

Illumination based methods however cannot be used for outdoor scenes. Methods which use time-of-flight cameras resolve this by sending pulses of electromagnetic signals into the environment and measure distances to the objects by estimating the round trip time of the pulses reflected off them. Most popularly Laser or IR lights are used as sources of the pulses [66]. For close range objects, Ultrasound based time-of-flight techniques have also been proposed [15].

These methods are highly accurate but suffer from problems like inter-object reflectance and other interferences from existing light sources in the environment. Additionally, they work only for static scenes. Passive methods are directed towards images or image sequences captured in the wild.

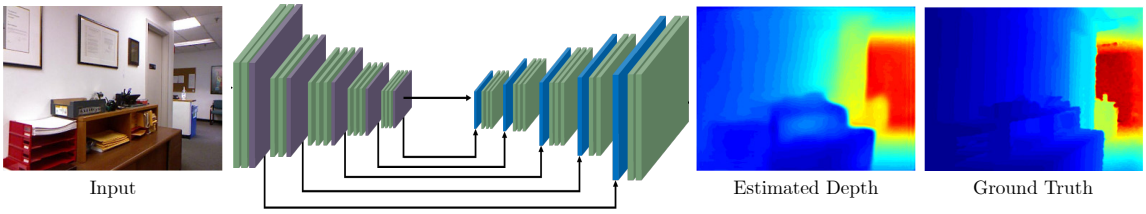


Figure 1.3: Representation of CNN based supervised monocular depth estimation. The input RGB image is passed through an FCN-like architecture to generate corresponding depth map. The estimated depth map is compared pixel-by-pixel with ground-truth depth captured usually using LIDAR or Kinect.

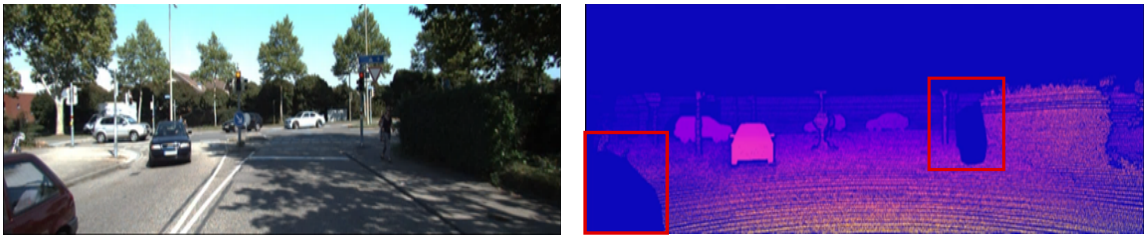
1.1.1.2 Passive Methods

The passive depth estimation methods use traditional cameras with no external influence in the scene. A large number of passive methods have been proposed due to both the difficulty of the problem as well as its potential applications. In relevance to this thesis, we discuss only monocular depth estimation methods. Depth using single cameras is particularly hard for its under-constrained nature. One of the early methods by Francois *et al.* [17] add reasonable constraints to the structure of the objects in the scene. Their interactive system take user inputs for feature extraction and edge detection. Such a method however assumes the scene to be of a known configuration. Non-interactive methods by Nagai *et al.* [54] and Delage *et al.* [14] avoid user input, but make stronger scene assumptions with objects like humans and indoor rooms. These solutions, however, work only for static scenes.

For dynamic scenes, it is necessary to account for individual object motions to extract 3D information from a sequence of images. Ozden *et al.* [57] use optical flow to group features belonging to the same object and propose a dynamic large-scale SfM (Structure from Motion) framework. All the aforementioned passive methods for monocular depth predict relative depth up to a scale. This is one of the fundamental problems still discussed in state-of-the-art monocular depth estimation methods [90].

Defocus blur has been shown to be an important depth cue both in human visual systems [51] and computer vision. The benefit of using defocus cue over other sources of information is that it provides depth at the correct scale. This benefit comes from the prior knowledge of the camera intrinsics like focal length and focus distance. Using spatial derivative kernels, the amount of deblurring at a pixel can be estimated [26]. Shape from focus methods depend on moderate to high frequencies in an image and hence do not work for texture-less surfaces.

Failing to incorporate high-level semantics has been one of the major challenges in geometry based solutions. Conversely, recent deep learning based methods are capable of predicting depth at high accuracy. Wang *et al.* [77] propose a method which estimates high quality depth (close to LIDAR quality) with denser predictions than sensor based depth.



RGB Ground Truth Depth

Figure 1.4: Sparsity in depth captured using LIDAR. Post-processing for completion can lead to artefacts as highlighted in red.

1.1.2 Deep Learning Methods

In light of its importance in vision and graphics, monocular depth estimation is one of the most popular problems in computer vision. Early shape from X methods exploit only low-level depth cues and fail to incorporate scene semantics. Recently, this has changed due to the availability of large-scale datasets [20, 64], highly parallelized compute devices and high capacity deep learning models. Methods have been proposed which use highly-parameterized functions to translate RGB images to dense depth maps—the parameters of which are learnt using non-linear optimization. Figure 1.3 gives an overview of one such method which performs single view depth estimation in an end-to-end fashion. It uses a fully convolutional network which is trained using datasets with a large number of RGB-Depth pairs. Conventionally, for every training step the generated depth maps are compared with the ground truth and the errors are used to train the parameters of the network. Section 1.1.3 highlights the major challenges with such methods.

1.1.3 Charactertistics of Existing Methods

Deep learning based methods learn high-dimensional non-linear functions which require access to massive datasets. The success of CNN based methods in monocular depth estimation is due to the availability of such datasets [20, 11]. Building them is expensive and requires extensive post processing after capturing raw data. For depth estimation in particular, aligned RGB-Depth pairs are required—which are captured using high-cost apparatus like LIDAR (Light Detection and Ranging) sensors. Good quality LIDAR sensors are expensive (about $25,000) and are not easy to use. Due to the nature of technology being used in these sensors, the depth maps generated are sparse. It is difficult to train deep networks which generate dense depth maps using such data as there is no full supervision. One may need to rely on depth completion techniques to obtain dense ground truth depth maps which adds more uncertainty into the problem. Figure 1.4 shows one such method which uses CAD models to complete certain regions (like cars) in the depth map, can be unreliable.

Recently, there have been attempts to circumvent this problem by substituting RGBD data used for training, by imaging data captured using multi-aperture cameras like light fields or stereo images [21, 71]. These methods use models which learn depth for an input image as intermediate representations

and use view synthesis and image-based rendering to generate novel views of the same scene. Using differentiable modules for view synthesis and depth estimation, the photometric errors in the generated views are backpropagated to improve the quality of depth maps. However, these methods by relying on pixel-wise photometric loss inadvertently assume that the scenes under consideration are Lambertian. They also assume colour constancy across different views of the same scene. Structural Similarity (SSIM) can be used in addition to a pixel-based loss which although produces better results, but is unreliable as a perceptual similarity measure as shown by Zhang *et al.* [87].

One of the other challenges with depth estimation methods is that most of the deep learning based models are trained in an end-to-end fashion. As a result, these models lack interpretability and it is hard to recognize the priors learnt by them. For systems to be deployed in the real world, it is necessary that these models have not overfit on the training data. Even a simple domain switch like from KITTI [20] to Cityscapes [11] does not work for these models as we show in chapter 3. For action-critical systems it is crucial to know if the perception modules are adaptive and if the learnt priors are interpretable. In the following chapters, we try to address some of these problems.

1.1.4 Our Contributions

We address the issue of photometric matching in existing methods by proposing a novel architecture, the parameters of which are learnt using adversarial training [22]. Our proposed method is robust against cross-view color inconsistencies which could occur due to non-Lambertian or texture-less surfaces. In addition to the adversarial loss, a perceptual loss is used which aids in generating sharper depth maps. Furthermore, we propose a curriculum learning framework which goes from generating novel views with smaller pose changes in the initial stages of the training, to generating views with larger pose changes in the later stages. This *structured training* approach is adaptable to other generative tasks which involve surrogate learning of proxy tasks. We also show the shortcomings of monocular depth estimation methods by investigating priors learnt by them. We use some of the strongest monocular depth cues in human visual system and show that they are partially reflected in artificial systems. Concretely, the highlights of this work are:

- Generative Adversarial Network (GAN) for monocular depth estimation. We show the first employment of a GAN that estimates dense depth maps for monocular images using stereo-view synthesis.

- Structured Adversarial Training (StrAT). A progressive training framework grounded on the idea of curriculum learning [8] is shown, which goes from generating easy samples to hard samples incrementally.

- Juxtaposition of biological depth estimation and recently proposed monocular depth estimation methods. We show with carefully designed experiments that some of the depth cues used by humans are self-learnt by the existing monocular depth estimation methods.

1.2 Thesis Organization

This thesis is organized into five chapters. Chapter 1 introduces monocular depth estimation, contextualizes the problems and our solutions to them. Chapter 2 summarizes the advances in 2D to 3D perception using alternate sources of supervision. It also provides a background on Generative Adversarial Networks to build a foundation for some of the ideas proposed in chapter 3. Chapter 3 describes our contributions to the problem of single image depth estimation. We provide a new framework for the problem and propose a robust training routine to train the model. Chapter 4 compares human depth perception and artificial depth perception. It provides insights into the priors learnt by state-of-the-art systems using qualitative and quantitative experiments. Chapter 5 concludes the thesis with some future directions for research in dense estimation problems like optical flow and depth estimation.

Chapter 2

Background and Related Work

2D-to-3D translation has been extensively investigated in forms of structure from motion, time-of-flight cameras, stereo vision, shape from shading, depth from defocus, etc. Considering the substantial volume of literature in this area, we restrict the focus of further discussion to monocular depth estimation.

2.1 Monocular Depth Estimation

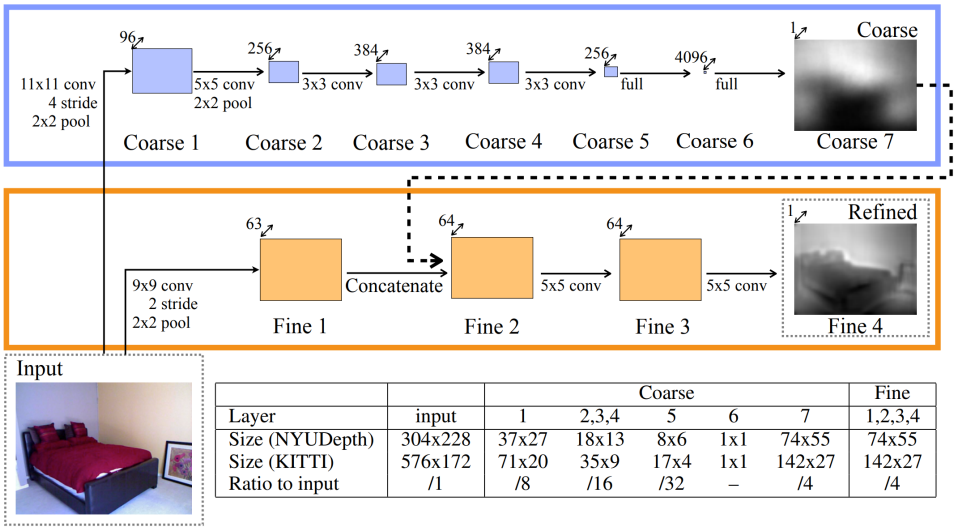
2.1.1 Supervised Depth Estimation

Most of these techniques are supervised in the nature that they require ground-truth RGBD data. Saxena *et al.* [63] estimate depth from single images using linear regression and Markov Random Fields (MRFs). They later extend this work to Make3D [64], where they propose a new RGBD dataset in addition to a more refined supervised learning method. In a more recent work, Hoiem *et al.* [27] suggest a method which decomposes a scene under consideration into planar surfaces. The scene is segmented into components like sky, ground, vertical, etc. to compose a 3D model.

Ladicky *et al.* [40] use *relative size*, one of the stronger monocular cues for depth perception. They suggest that the perceived size of an object scales inversely with depth. They use semantic labels and superpixels to segment an image with object labels and introduce classification-based depth estimation using hand-crafted features.

Karsch *et al.* [33] propose one of the first methods that use optical flow for depth. They use a static RGBD sensor (Kinect) to capture ground-truth data, which is later used to generated stereo RGBD videos. To estimate depth for a single frame, visually similar patches are extracted from the training data, and subsequently a kNN and SIFT-flow based depth transfer mechanism is used. For consecutive frames, optical flow is used to propagate depth priors and further refine the depth maps. One of the drawbacks of this work is that the entire training set needs to be available at runtime.

Eigen *et al.* [16] show the first prominent work in the area of CNN-based methods. Their approach learns pixel-aligned depth from monocular images and unlike previous methods [64] it does not use hand-crafted features. They use a coarse prediction network which generates a depth map



		Coarse					Fine
Layer	input	1	2,3,4	5	6	7	1,2,3,4
Size (NYUDepth)	304x228	37x27	18x13	8x6	1x1	74x55	74x55
Size (KITTI)	576x172	71x20	35x9	17x4	1x1	142x27	142x27
Ratio to input	/1	/8	/16	/32	–	/4	/4

Figure 2.1: Model architecture by Eigen *et al.* [16]. One of the first supervised *single image* depth estimation methods which are learning based. Two deep network stacks are employed: one for coarse global prediction and other which refines it.

subsequently refined by a refinement network. Concurrently, Liu *et al.* [46] propose a CNN-based learning scheme which learns unary and pairwise potentials for a CRF model. With the introduction of fully convolutional networks by Long *et al.* [48], Mayer *et al.* [52] propose an FCN for single image depth estimation and optical flow, which has subsequently resulted in several improvements in the area [21, 45, 50, 76, 83, 90].

Fu *et al.* [18] argue that previous CNN based methods pose the task of depth estimation as a regression problem. This leads to slow learning and locally optimal solutions. They pose it as an ordinal regression problem by discretizing depth. Recently, unsupervised or self-supervised methods have been proposed which use view synthesis as a proxy task. Semi-supervised methods like the one proposed by Kuznietsov *et al.* [39] use depth supervision as well as image alignment loss used in unsupervised methods to generate high quality depth maps.

2.1.2 Surrogate Supervision

2.1.2.1 Stereo View Supervision

Capturing ground-truth depth maps is expensive and time consuming. An alternate strategy to train CNNs for depth estimation is to use multi-view supervision as a proxy for depth supervision. Garg *et al.* [19] use stereo view supervision, in which the left-view of a stereo pair is warped into the right-view by learning dense disparities as an intermediary task. They make Taylor approximations to engineer differentiable loss functions. Figure 2.2 shows a system overview of their method. It is the first method in the area which performs unsupervised/self-supervised monocular depth estimation with end-to-end

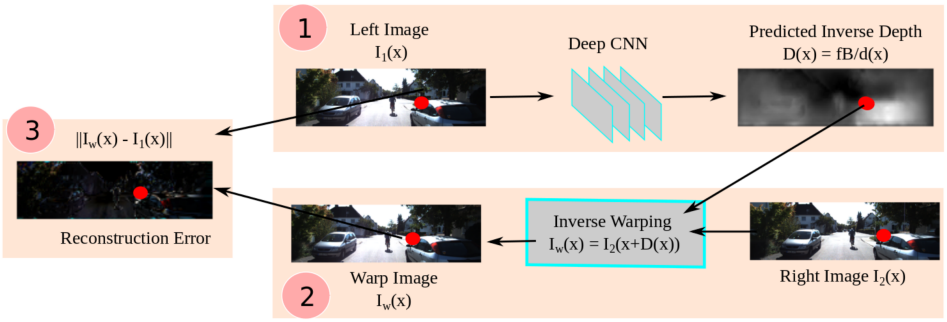


Figure 2.2: System overview of the method proposed by Garg *et al.* [19]. They use a stereopsis based auto-encoder setup in which the encoder translates a left stereo view I_1 to its corresponding depth map. The decoder forces the output from the encoder as a disparity map by synthesizing a backward warp image I_w from the right-view I_2 using the output generated by the encoder. The synthesized image I_w is then matched with the encoder input I_1 using reconstruction errors.

training. Note that although stereo-pairs are required at the time of training, at test time the depth CNN requires only monocular images.

Since Garg *et al.* [19] use Taylor approximation to linearize their loss, it is harder to optimize their overall objective. Godard *et al.* [21] instead use bilinear sampling and add a consistency check between the disparities learned for the left-view and the right-view. The consistency check in a stereo setting has been previously used as a post-processing step [85], but has not been directly incorporated in an end-to-end fashion. Figure 2.3 illustrates this approach with a comparison with previous naive stereo synthesis based depth estimation methods.

Deep3D [81] also addresses the problem of generating stereo pairs from a single image, specifically generating right images from left images. Existing stereo movies are used for training. Instead of generating disparity maps in a deterministic manner, they generate a distribution over disparity values. The synthesized view has pixel values as a combination of pixel values from the input view, estimated using the distribution over disparity values. However, since their image formation model is based on distributions over all possible disparity values, increasing the resolution of images greatly increases the memory consumption of the algorithm. This makes it difficult to use their approach on high resolution images.

Augmenting view supervision with depth supervision further improves the accuracy of depth estimation as shown by Kuznietsov *et al.* [39].

2.1.2.2 Monocular Video Supervision

For stereo synthesis, since the baseline is assumed to be fixed throughout the training set, most frameworks implicitly use camera-pose supervision. Video synthesis has also been used as an alternative. The additional challenge is that the camera transformation between two consecutive frames is not fixed. Zhou *et al.* [90] use a pose network in addition to a depth network and use video reconstruction

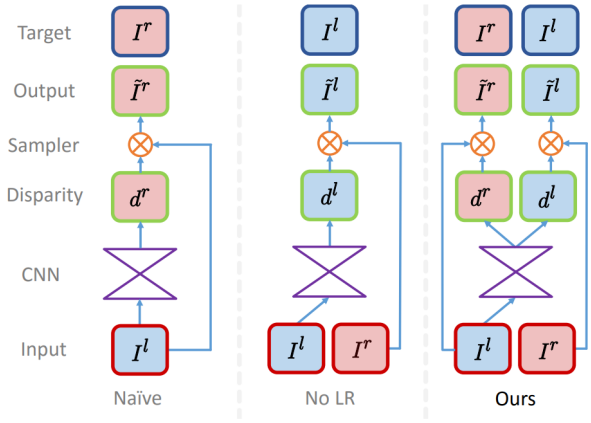


Figure 2.3: LR consistency check introduced by Godard *et al.* [21]. The naive strategy generates disparity map aligned with the output view instead of the input view. The No LR strategy corrects this but is still riddled with artifacts. The LR check approach produces disparity maps for both the input view and the output view and enforces mutual consistency.

errors to train them together. As shown in figure 2.4, a reference frame is warped into subsequent and preceding frames using estimated depth and camera transformation. The errors in the RGB domain are backpropagated to simultaneously train the two networks. Like the stereo synthesis based methods discussed in the previous section, at test time only a single image is expected as the input.

Simultaneously, Sfm-Net proposed by Vijayanarasimhan *et al.* [75] learns depth and pose change with various degrees of depth supervision. It predicts depth, segmentation masks, and camera and object motions. Using this information, it estimates optical-flow between two consecutive video frames which is used to differentiably warp frames and backpropagate pixel errors. In spite of using only view reconstruction errors, these approaches showcase better results than a depth supervised method [16].

In an attempt to add more geometric constraints, Yang *et al.* [83] add a consistency loss function for surface normals inferred from the depth and input RGB image. Alternatively, Mahjourian *et al.* [50] use ICP-based [9] loss on the 3D point clouds generated using a depth network. Furthermore, Zhan *et al.* [86] integrate both stereo and video synthesis in their method (like [83]) along with a deep-feature loss.

2.1.2.3 Other Alternatives

There also have been attempts at using other supervisory signals such as 4D light fields [71]. However, cameras which capture light field data are not readily accessible and are far from being mainstream. Another possible supervisory signal can be defocusing [71] (Figure 2.5). But stereo supervision [21] still remains to be the most accurate and memory-efficient technique. With the introduction of stereo cameras in several modern smartphones, capturing stereo images is almost as easy as capturing monocular images. Consequently, we make use of only stereo-supervision for our method.

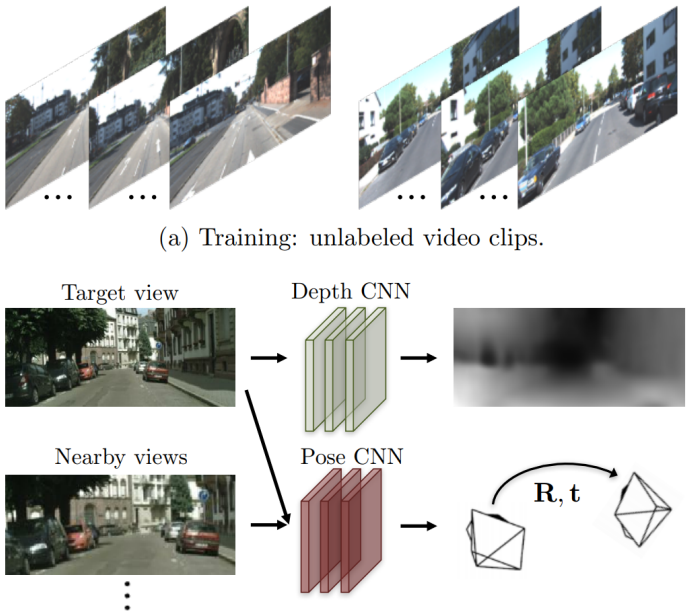


Figure 2.4: System overview of method by Zhou *et al.* [90]. The method uses unlabeled video clips as training data. Two networks—one for depth and the other for egomotion are simultaneously trained by using video reconstruction errors.

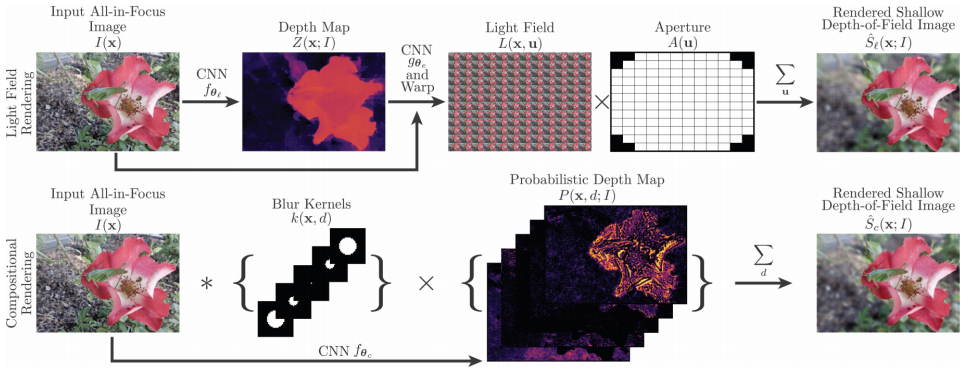


Figure 2.5: Monocular depth estimation using aperture supervision by Srinivasan *et al.* [71] with two aperture rendering pipelines. A CNN f_{θ_l} is trained to predict a depth map from an input all-in-focus image. Another CNN g_{θ_e} is used to render each view in the light field using the depth map and the input image. A differentiable rendering function is used to generate a shallow depth of field image by integrating the views rendered by g_{θ_e}.

2.2 Generative Adversarial Networks

This section presents the idea of Generative Adversarial Networks and their significance in the landscape of generative modelling. Adversarial training has been proven as an important tool for several generative vision tasks like deblurring, semantic segmentation and super-resolution.

Generative Adversarial Network (GAN) is a powerful tool for probabilistic generative modelling. It has been consistently shown to produce remarkable results for several generative tasks. The general idea of GANs is to pit two adversaries—a *generator* and a *discriminator* against each other in a min-max game. The players (or adversaries) are generally deep neural networks operated by a set of controlled parameters. Each training step comprises of two stages. In the fist stage, the generator G is tasked with producing a sample which ostensibly belongs to the training set distribution. The generator samples z from a latent space distribution to generate a sample $G(z)$. In the second stage, a sample x from the true training set distribution and the fake sample $G(z)$ are provided to the discriminator D. The discriminator strives for $D(x)$ to be 1 and $D(G(z))$ to be 0. With the feedback from the discriminator the parameters of the generator are trained such that $D(G(z))$ is pushed towards 1. At the end of training, the discriminator fails to discern between $G(z)$ and x, as a result of which the generator produces samples which apparently belong to the true distribution.

There are several loss functions used to train a GAN framework. The loss for the discriminator proposed in the original paper [22] is:

$$J^{(D)}(\theta^{(D)}, \theta^{(G)}) = -\frac{1}{2}\mathbb{E}_{x\sim p_{data}}\log D(x) - \frac{1}{2}\mathbb{E}_z \log(1 - D(G(z))),$$

where $\theta^{(D)}$ and $\theta^{(G)}$ are the parameters of the discriminator and the generator respectively. This is standard cross-entropy loss which is minimized during training. The discriminator is a binary classifier with a sigmoid output.

To generate samples belonging to a certain class, conditional GANs [53] (cGANs) are used. The generator and the discriminator take additional inputs on which the output is supposed to be conditioned on. The loss function used for cGANs looks very similar to the one proposed in [22]:

$$J^{(D)}(\theta^{(D)}, \theta^{(G)}) = -\frac{1}{2}\mathbb{E}_{x\sim p_{data}}\log D(x|c) - \frac{1}{2}\mathbb{E}_z \log(1 - D(G(z|c))),$$

where c is the conditional variable. A comprehensive overview of all the applications of GANs is beyond the scope of this work. The following discussion is of the problems which share some similarities to the task of monocular dense estimation.

2.2.1 Image-to-image Translation

Several vision tasks related to dense estimation like optical flow, semantic segmentation, depth estimation *etc.*, can be framed as image-to-image translation. Isola *et al.* [91] proposed a simple framework which can be used for most of these tasks. They argue that rather than finding an appropriate loss function for your problem, a high level goal of "Does this image look real?" is enough to learn a suitable

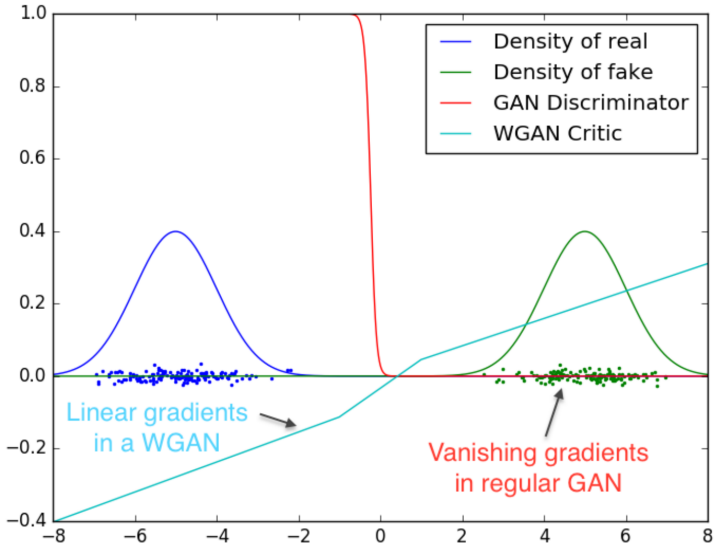


Figure 2.6: The vanishing gradient problem in GANs. When trying to discern between the two gaussian distributions, the discriminator in [22] saturates with zero gradients. On the contrary, the critic (or discriminator) in WGAN [6] has non-diminishing gradients on all parts of the space. Figure reproduced from [6].

loss for the given task. This naturally fits into the idea of using generative adversarial networks. They formulate image-to-image translation in the framework of a conditional GAN, where the input acts as the conditional variable. They use the traditional cGAN loss with an additional L1 term. The experiments in [91, 58] show that it is beneficial to mix the GAN objective with L2/L1 term. Their overall GAN objective is:

$$J^{(D)}(\theta^{(D)}, \theta^{(G)}) = \arg \min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda \mathcal{L}_{L1}(G).$$

2.2.2 Markovian Discriminator

It has been observed that L1/L2 loss functions when used exclusively, produce blurry outputs for generative tasks [42]. Although these loss functions do not generate outputs with high frequencies, they are usually sufficient for problems which require only low frequency outputs. However, since dense depth maps have high frequency regions at occlusion boundaries, using standalone L1/L2 loss is not sufficient. To generate a dense output with high frequency elements, a *PatchGAN* [91] discriminator can be used. This discriminator penalizes inaccuracies at the scale of patches and classifies each $N \times N$ patch in an image as real or fake. As shown in [44], such a discriminator models the input image as a Markovian random field with independence assumptions between pixels that do not belong to the same patch.

2.2.3 Wasserstein Loss

Training GANs with the loss function proposed in [22] is tedious. In generative tasks, there are two common measures which are popularly used: 1) Kullback-Leibler (KL) Divergence, and 2) Jensen-Shannon (JS) Divergence. Goodfellow *et al.* [22] show that minimizing the original GAN objective is equivalent to minimizing the JS-divergence between the generated distribution and the training set distribution. However, the gradient of JS-divergence diminishes when the distance between the two distributions is large. This is precisely the case in the initial stages of the training and so learning the parameters of the generator becomes hard. As a result, in practice, it is easier to train the discriminator than the generator. As an alternative Arjovsky *et al.* [6] propose to use a new loss function which uses Wasserstein distance which has non-diminishing gradients. Figure 2.6 illustrates the advantage of using WGAN critic over GAN discriminator. The overall object in WGAN is:

$$\mathbb{E}_{x \sim p_{data}} f(x) - \mathbb{E}_z f(G(z)),$$

where f is a 1-Lipschitz function.

2.3 Summary

Broadly, the recent solutions for the problem of monocular depth estimation either use ground truth depth which is noisy for large distances and sparse, or use view alignment errors. The problems with both kinds of solutions have been discussed in section 2.1. We try to address some of these issues by introducing a method which does not use ground truth depth and uses a novel loss for capturing perceptual similarity. We use GANs (section 2.2) which have been used in other generative problems in computer vision and have been shown to generate predictions with high perceptual quality [32]. We also propose a training curriculum which progressively generates difficult predictions. These ideas are discussed at length in chapter 3.

Chapter 3

Structured Adversarial Training

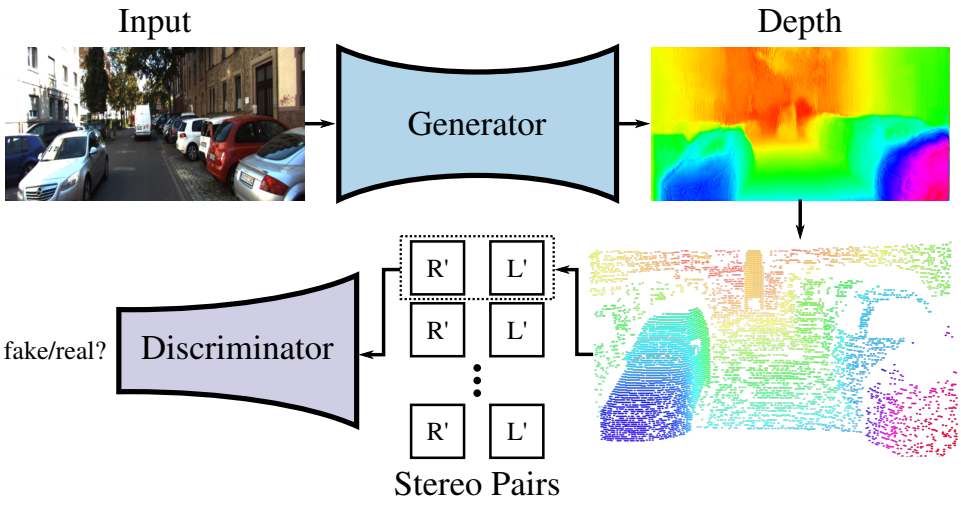


Figure 3.1: System overview. The generator produces stereo pairs with different baselines for every iteration using intermediate depth and the discriminator learns to disambiguate these generated images from real images. We use *structured adversarial training* in which the baseline for the stereo pairs is gradually increased over epochs.

The long-standing problem of estimating scene depth from a single image has seen promising solutions with the advent of CNNs. Like most CNN based solutions, depth estimating networks require large amount of data. Recently, unsupervised methods have been proposed which use view synthesis as an alternate supervisory signal. Novel views are synthesized by estimating scene depth in an intermediate step. These methods minimize photometric reconstruction errors in order to learn the intermediary depth estimator. In this chapter, we show the shortcomings of this approach and propose a geometry-aware generative adversarial network which uses depth to generate novel views from a single image. The synthesized views are discerned from real images using discriminative learning. We show the gains of using the adversarial framework over previous methods. Furthermore, we present a structured adversarial training scheme to train the network. In this training routine the estimator is fed easy samples in the initial stages of training, and as the training progresses, so does the difficulty of the samples. The com-

bination of adversarial framework, multi-view learning, and structured training produces state-of-the-art performance on unsupervised depth estimation for monocular images.

Human perception of the world in 3D has been studied by psychologists, physicists, mathematicians and by computer scientists. Our capacity to navigate about our surroundings relies heavily on our ability to locate ourselves with respect to the objects around us. We do this by aggregating several monocular and stereo depth cues like occlusion, motion parallax, binocular disparity, relative size and several others as discussed by Cutting and Vishton [12]. Apart from these cues, human depth perception is also based on semantic understanding of the scene. There have been several attempts at deducing 3D shape from 2D images in geometric computer vision such as shape from X, stereo matching and structure-from-motion. Early techniques try to infer depth from low-level depth cues but fail to incorporate scene semantics. Eigen *et al.* [16] showed that depth based on semantics can be learned by training CNNs on large datasets like KITTI [20] and NYU Depth [69]. Subsequently, several methods [10, 34, 47] have showcased the remarkable ability of CNNs in dense depth estimation from monocular RGB images.

These methods, however, are trained on RGB-Depth pairs. Capturing datasets with depth-aligned RGB images is expensive and requires extensive post-processing. Garg *et al.* [19] and Godard *et al.* [21] circumvent these problems by estimating dense depth (or disparity) as an intermediary task for stereo view-synthesis. They use photometric reconstruction errors to train a fully-convolutional network which estimates the adjacent stereo view corresponding to an input RGB image. On a similar line of work, Zhou *et al.* [90] use video synthesis with depth estimation as the auxiliary task. However, these methods make two implicit assumptions: 1) brightness and colour constancy across views, and 2) Lambertian scene composition. We address these underlying assumptions in previous unsupervised depth estimation methods by proposing a generative adversarial network (GAN [22]) for generating stereo pairs from single RGB images. The adversarial learning framework (shown in Figure 3.1) guides the generator part of the architecture to produce realistic stereo pairs, while the discriminator learns to disambiguate between the generated and real images. Since the discriminator tries to learn the real distribution of images, the reconstruction errors incorporate the changes in structural content between real and synthesized images, in addition to intensity changes. Separating synthesized and real images is a complex task. Using the depth-map, multiple samples at varying baselines can be generated to facilitate structured training of the GAN.

The success of adversarial learning in generative tasks is unchallenged. A diverse set of problems in computer vision like 3D reconstruction [24], novel view synthesis [61, 84], super-resolution [43] and image-deblurring [38] have been better solved by GANs [22]. GANs eliminate the need to find efficient loss functions for generative networks as the discriminator adapts to each task with the aim of discerning between generated and real samples. We thereby propose a generative adversarial network for the task of monocular depth estimation. To the best of our knowledge, our method is the first work which employs GANs for unsupervised depth estimation.

Most of the view supervision based methods use loss functions for low-level features like pixel intensities. These methods inadvertently make assumptions about the scene being composed of only

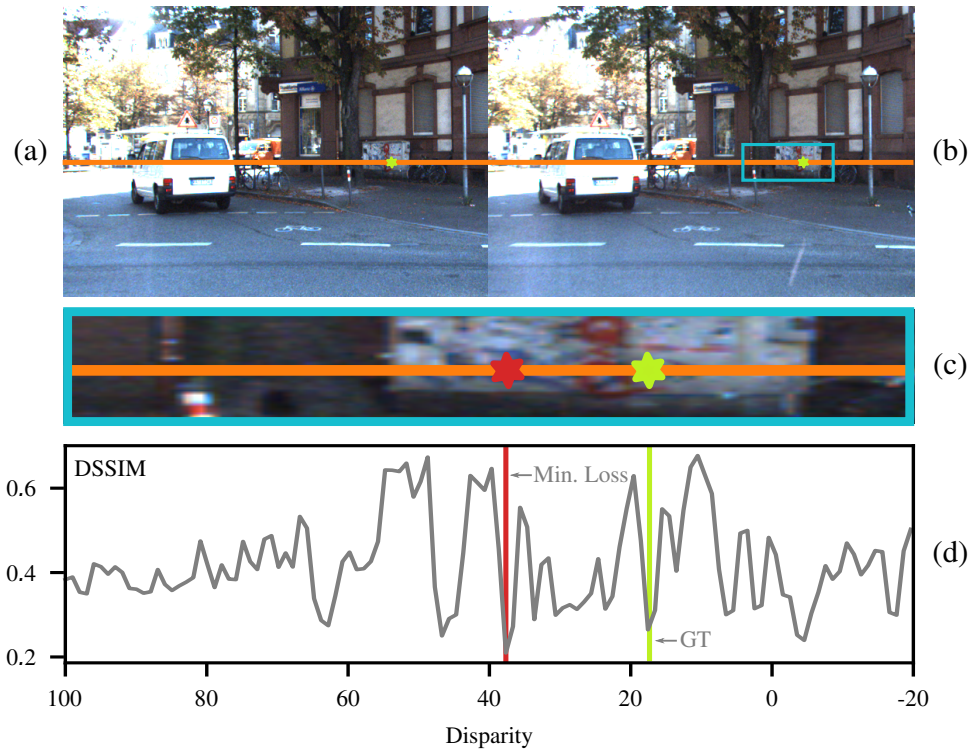


Figure 3.2: Unreliable stereo-matching using structural similarity (SSIM). (a) and (b) are a stereo pair from KITTI [20]. The line in orange denotes the epipolar line. The green stars show ground-truth correspondence. The red star shows the estimated match using SSIM. The search region for stereo matching is highlighted in blue. DSSIM $= \frac{1-\text{SSIM}}{2}$

Lambertian and densely-textured surfaces. For highly-reflective surfaces like glass and surfaces without unique textures, these loss functions are ineffective as shown in [21]. In [86], a deep feature loss is used along with an L1 loss measure on pixel intensities, but it does not integrate left-right disparity consistency checks. For the deep feature loss, however, their method requires a network to be trained on an alternate dataset [69] with ground truth depth. On the contrary, our method uses adversarial learning [22] as an alternative to the deep feature loss, which keeps our method to be unsupervised with comparable results.

The computed errors are highly local and fail at capturing high-level semantics *e.g.* rigid objects like poles and trees must remain vertical across views. These errors *viz.* SSIM and $\ell 1$, do not account for the entire context of a scene and hence can result in unreliable depth estimates. (Figure 3.2). We propose a generative adversarial network in which the discriminator considers a larger context while computing the loss between a generated and a target image. Our method is unsupervised as it requires only stereo images for training and even performs better than most depth-supervised techniques.

3.1 Depth using Adversarial Training

Binocular disparity is one of the strongest cues for depth in human perception [12]. Point-to-point correspondence between the left-view and the right-view of a stereo pair can be used to estimate dense depth maps as:

$$D^i = \frac{Bf}{d^i}, \tag{3.1}$$

where D^i is the depth for the point i with disparity d^i, B is the baseline for the stereo camera and f is the focal length. With a known dense disparity map, a stereo pair can be generated from the image corresponding to the disparity map. Inspired by this, the naïve way to solve depth estimation would be to generate the adjacent stereo-view from a single image and use stereo-matching algorithms to estimate the depth of the scene, in a two-step approach. We use a one-step approach instead, in which a dense disparity map is generated from a single image, which is used to create the adjacent stereo view. The errors in the generated disparity map are computed using reconstruction errors between the generated and the ground-truth stereo view. This is similar to novel view synthesis using appearance flow [89]. In case of rectified stereo pairs, the appearance flow becomes one dimensional due to epipolar constraints. The magnitude of the flow for every pixel represents its corresponding disparity and dense depth can be inferred according to Equation 3.1.

Our goal is to generate a stereo pair (L'_i, R'_i) from a single image L_i, assuming it to be the left-view without loss of generality. To this end, we train a generator network as a feed-forward CNN G_{θ_G}, where θ_G are the parameters of the network. The generator takes L_i as the input and generates a pair of disparities (D_i^L, D_i^R). Subsequently, a bilinear sampler (S) [29] generates the stereo pair as $L'_i = S(D_i^L, R_i)$ and $R'_i = S(D_i^R, L_i)$. Note that (L_i, R_i) pairs are present only during training. During inference, a forward pass through G with L_i as the input is sufficient to produce dense depth maps from D_i^L using Equation 3.1. For training pairs $\{(L_i, R_i) \mid i = 1 \dots N\}$, we solve:

$$\hat{\theta}_G = \arg\min_{\theta_G} \frac{1}{N} \sum_i E^R(R'_i, R_i) + E^L(L'_i, L_i), \tag{3.2}$$

where E^L and E^R are linear combinations of a set of loss functions defined in later parts of this section. For the remainder of this chapter, we restrict the discourse to generation of only right-views R'_i.

Previous methods for depth estimation based on view-synthesis [21, 90] rely on primitive reconstruction loss functions like mean-squared error and SSIM [78]. These pixel-wise error measurements fail to capture the perceptual quality of a generated image as shown by Zhang *et al.* [87]. As shown in Figure 3.2, a pixel-based loss like structural dissimilarity, when used for stereo matching, produces unfavourable loss surfaces with multiple local minima and a global minimum far from the ground truth disparity.

One way to capture the perceptual loss between two images, is to use feature maps from a pre-trained network like VGG19 [70], as proposed by Johnson *et al.* [31]. However, a network like VGG19 trained

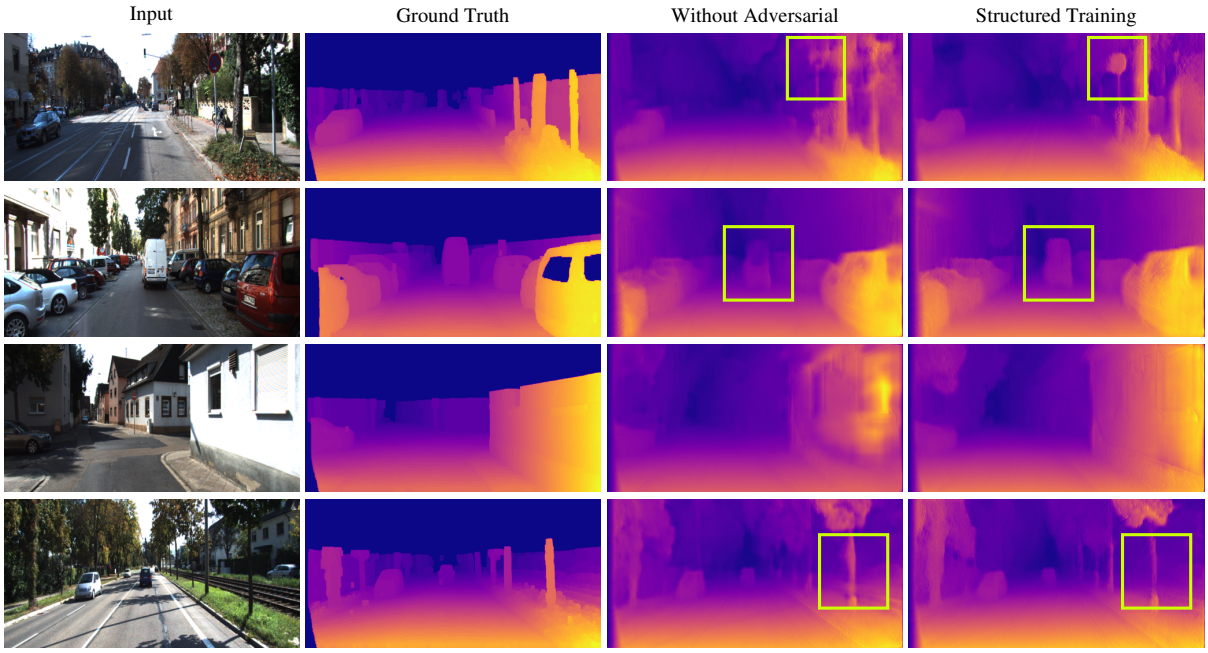


Figure 3.3: Qualitative comparison between our model with and without adversarial learning. The model using adversarial loss performs better for objects with low-texture density and for thin structures like poles and trees. The major differences are inside the boxed regions. For better visualization, we dilate the ground-truth depth maps which are Velodyne laser projections and hence sparse.

on a classification task does not capture high-frequency details required for stereo matching. We instead train a discriminative network which adapts to our particular task by learning to perceptually distinguish a generated image from a real image.

3.1.1 Adversarial Learning

We propose an adversarial framework for view synthesis, which eliminates the need to find an appropriate loss function that captures the perceptual quality of the generated view. We use a discriminator network which learns to predict whether the synthesized view belongs to the manifold of real views without relying on pixel-by-pixel matching. In the proposed framework, the generator G_{θ_G} learns to synthesize the adjacent right-view for an input image using the generated disparity D_i^R. The discriminator C_{θ_C} tries to distinguish the generated right-view from the real right-views in the training set, as it competes with the generator. This is done by optimizing the following objective in an alternating manner:

$$\begin{aligned}\mathcal{L}_{adv}^{R} = \min_{\theta_G}\max_{\theta_C}\ & \mathbb{E}_{L\sim p_G(L)}[\log(1 - C_{\theta_C}(R_i')] \\ & + \mathbb{E}_{R\sim p_{train}(R)}[\log\ C_{\theta_C}(R)]\end{aligned} \tag{3.3}$$

Equation 3.3 refers to the vanilla GAN [22] objective for generating right-view R'_i from left-view L_i. It is to be noted that for view synthesis of outdoor scenes like in KITTI [20] the transformation of objects across views is mostly rigid. Our goal is to learn a generator which is aware of such structural content in the scene and other semantics, *e.g.*, thin structures like poles remain vertical across all views.

In a naïve training procedure, the generator produces right-views with a fixed baseline B. In the early stages of training, since the estimated disparities are quite erroneous, using them at full scale (with baseline B) will generate right-views which are easily distinguishable from the original training set. As a result, the discriminator dominates the training procedure and hence equilibrium is not achieved for the adversarial objective (Equation 3.3). We resolve this by adding more consistency constraints by using multiple views corresponding to the same depth-map.

3.1.2 Multi-view Synthesis

The generator network produces a dense depth-map which is a 2.5D representation of the scene. As a result, given a camera-transformation matrix, a novel view can be generated corresponding to the transformation. With more views, we can add additional consistency constraints to the depth.

For accurate depth estimation, the generated views should be *real-looking* and consistent with each other. Inspired by this idea, we can train our framework to generate multiple views of the scene corresponding to an input image, using the dense depth-map from the generator. The views can be generated using the dense-depth map obtained from the generator. The task of the discriminator remains to discern these views from real views present in the training data. This results in a min-max zero-sum game in which the generator and the discriminator are the two agents.

In case of a complex scene, for generating views from 2.5D with severe changes in camera pose, an additional step of hole-filling becomes necessary. The task of the discriminator in such cases becomes fairly easy and equilibrium for the min-max game is not achieved, as we find in our experiments. To generate views with small pose changes, we restrict the pose transformation as a translation along the direction of the epipolar lines. This is equivalent to generating right-views with fractional baselines:

$$R'_{i,b} = S\big(D_i^R \times \frac{b}{B}, L_i\big), \tag{3.4}$$

where $R'_{i,b}$ corresponds to baseline b. For the sake of simplicity, we take R'_i as the right-view corresponding to baseline B. Note that the training data consists of stereo pairs with baseline B only. For every training iteration, a right-view is generated with a chosen baseline b. We show that the choice of b is crucial for training and propose a structured training scheme which varies b in a sequential and organized manner.

3.1.3 Structured Adversarial Training

Inspired by the work on curriculum learning by Bengio *et al.* [8], we argue that by training our framework with right-views in a meaningful order, helps the networks learn in a more structured manner.

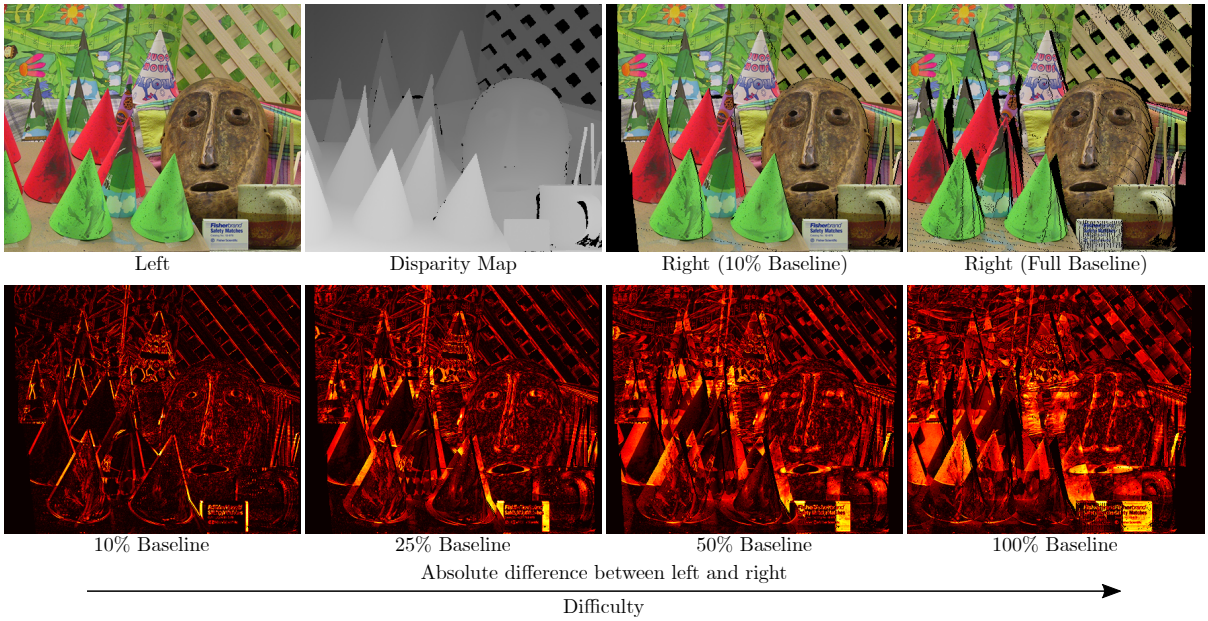


Figure 3.4: Representation of increase in difficulty of right view synthesis with increase in stereo baseline. The information loss between left-right pair increases with increasing baseline and so does the difficulty of generating stereo images from a single image.

We gradually increase the baseline such that the impact of the erroneous disparities in the initial learning stages is mitigated. In other words, b is linearly varied from 0 to B over epochs. The incremental nature of this scheme allows the training to first discover low-level mismatches like incorrect textures, and then shift attention to high-level content changes. In the early stages of training, the task is substantially simpler as there is minimal loss of information between stereo views with small baseline. As we increase the baseline b incrementally, the task becomes harder as the effect of erroneous disparities becomes pronounced (shown in figure 3.4). This procedure is conceptually similar to the recent work by Karras *et al.* [32], in which they progressively train GANs to generate low-resolution images and then high-resolution images as the training progresses.

We find that linearly varying b performs better than just randomly sampling b from $[0, B]$, which itself is better than using a fixed baseline B. Training with random baseline can be considered as data augmentation to regularize the discriminator. However, our experiments indicate that not only does the learning benefit from stereo-views with multiple baselines, but also from structured training. We find in our experiments that linearly varying b results in much more stable training compared to other training schemes.

3.1.4 Training the Generator

In this section, we describe the loss functions used to train the generator. We use a combination of adversarial loss and photometric loss to achieve the desired outcome.

Adversarial Loss The adversarial objective influenced by the parameters θ_G of the generator is minimized to train the generator:

$$\mathcal{L}^R_{G.adv} = -\log C_{\theta_c}(R'_{i,b}) \tag{3.5}$$

We optimize $-\log C_{\theta_c}(R'_{i,b})$ instead of $\log[1 - C_{\theta_c}(R'_{i,b})]$ for better gradient behaviour. The adversarial loss encourages the generator to generate right-views which reside on the manifold of ground-truth right-views and as a result, fool the discriminator. The discriminator is not conditioned by L_i and hence only learns to perceptually disambiguate R'_i from the distribution of real images and not whether R'_i corresponds to L_i. As a result, using the adversarial loss independently will lead to $R'_i = L_i$ as L_i belongs to the distribution of real images. To avoid this problem of mode collapse, we couple the adversarial objective with other metrics like $\ell 1$ and SSIM loss used in several prior works [21, 75, 83, 90].

	Error (lower the better)				Precision (higher the better)		
	Abs. Rel	Sq. Rel	RMSE	log RMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Yang *et al.* [83] *	0.172	1.559	6.794	0.252	0.744	0.910	0.969
Godard *et al.* [21]	0.124	1.388	6.125	0.217	0.841	0.936	0.975
Ours – No $\mathcal{L}_{adv}$	0.118	1.124	5.916	0.211	0.835	0.936	0.975
Ours – Fixed Baseline	0.115	1.154	5.766	0.205	0.846	0.944	0.979
Ours – Random Baseline	0.112	1.089	5.636	0.201	0.850	0.946	0.980
Ours – StrAT	**0.108**	**1.019**	**5.551**	**0.195**	**0.856**	**0.950**	**0.981**

Table 3.1: Results on KITTI [20] 2015 split. Comparison of our model with previous unsupervised methods. Our model with structured adversarial training (StrAT) scheme performs the best across all metrics. (* – trained on monocular videos)

Photometric Reconstruction Loss We use photometric losses to incentivize the generator to create images which are pixel-wise similar to right-views corresponding to input left-views. For generated right views $\{R'_i \mid i = 1 \dots N\}$:

$$\mathcal{L}^R_{\text{SSIM}} = \frac{1}{w \times h} \sum_x \sum_y \frac{1 - \text{SSIM}\{R'_i(x,y), R_i(x,y)\}}{2}, \tag{3.6}$$

where w is the width and h is the height of R_i. $R_i(x,y)$ is a patch at (x,y) with an empirically determined patch size of 3×3. Additionally, for local matching we use $\ell 1$ loss:

$$\mathcal{L}^R_{\ell 1} = \left\| \left(R'_i - R_i\right) \right\|_1 \tag{3.7}$$

Left-Right Consistency To ensure coherence between left-disparity D_i^L and right-disparity D_i^R, a consistency check is introduced [21]. The left-disparity is taken as the left-view of a stereo pair and is warped into its corresponding right-view (here, right-disparity $D_i^{R'}$) as $D_i^{R'} = S(D_i^R, D_i^L)$. An $\ell 1$ penalty is used for the projected and generated right-disparities:

$$\mathcal{L}_{CR}^R(D_i^R, D_i^L) = \left\| \left(D_i^{R'} - D_i^R \right) \right\|_1 \tag{3.8}$$

Relating equations 3.2,3.5,3.6,3.7 and 3.8, the final objective for the generator to generate right-views from left-views is:

$$E^R = \lambda_1 \mathcal{L}_{G.adv}^R + \lambda_2 \mathcal{L}_{\text{SSIM}}^R + \lambda_3 \mathcal{L}_{\ell 1}^R + \lambda_4 \mathcal{L}_{CR}^R(D_i^R, D_i^L), \tag{3.9}$$

where λ_{1-4} are empirically chosen, the details for which are mentioned in Section 3.2.1. All the discussed loss functions are mirrored to get a similar objective for E^L.

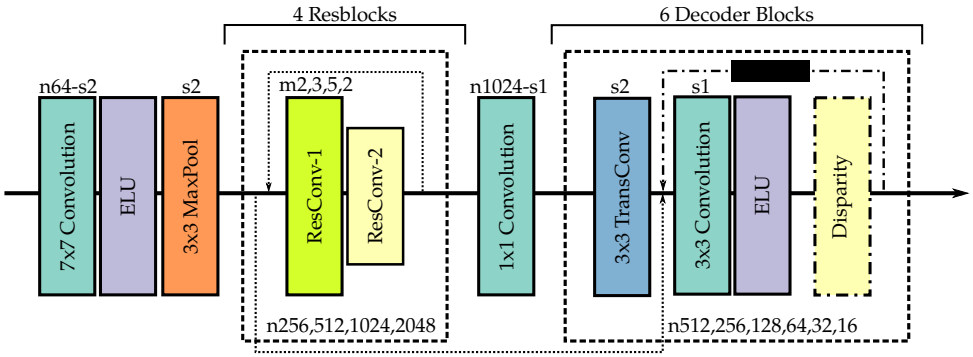
3.1.5 Network Architecture

The generator is a fully convolutional network with an encoder and a decoder. It is similar to the DispNet architecture proposed by Mayer *et al.* [52]. We use a ResNet-50 [25] based encoder for its large receptive field inspired by [41], and it is composed of four residual blocks. The encoder is followed by a convolutional layer with 1×1 kernel and then by a decoder. The decoder is composed of multiple transposed convolutions to upsample the encoded feature map. We use skip-connections to ensure that there is minimal information loss due to downsampling and that the gradients flow smoothly. Unlike the previous non-linearity layers which use ELU [73], our final layer is followed by ReLU [55] to ensure that the resultant disparities are positive. The output of the generator is a $512 \times 256 \times 2$ matrix with concatenated left and right disparity maps.

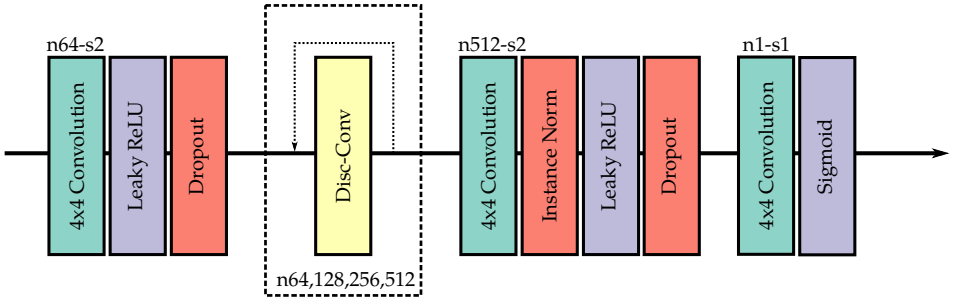
The discriminator is similar to PatchGAN [91] with a series of convolutional layers. Except the last layer, all of them are followed by Instance Normalization [74], Leaky ReLU [82] with $\alpha = 0.2$ and Dropout [72] regularization with keep-probability as 0.8. The last layer is followed by sigmoid activation. The input to the discriminator is a random patch of 256×256 from the generated and ground-truth images. The PatchGAN generates a 7×7 grid in which every cell indicates whether the input image patch in its receptive field is real or fake.

3.2 Experiments

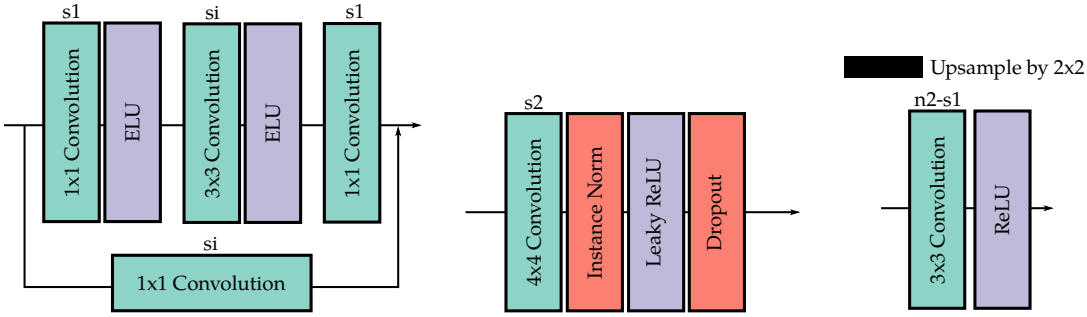
In this section we show thorough quantitative and qualitative evaluation by analyzing the performance of our method for monocular depth estimation for the training and test splits discussed in Section 3.2.2. We compare our work with prior methods on a standard benchmark and report promising results. To encourage reproducibility, we also list various implementation details. Ablation studies on the adversarial loss illustrate its validity. We also highlight the generalization of our model on unseen



(a) Generator



(b) Discriminator



(c) ResConv-i (d) Disc-Conv (e) Disparity

Figure 3.5: Network architecture. (a) and (b) are the generator and the discriminator respectively. (c), (d) and (e) are the submodules used by (a) and (b). nx denotes that there are x number of filters at the end of the module. sx denotes a stride length x, kept same for height and width for an input channel. mx represents the number of repetitions. The disparity module is used for the last 4 decoder blocks.

data. Finally, we show depth estimation performance individually on a few object categories using semantic segmentation.

3.2.1 Implementation Details

Our models are implemented using the Tensorflow [1] deep learning framework. Training is performed on two NVIDIA 1080 Ti GPUs. It is executed for 50 epochs with $b = 0.1 + \frac{e}{50} \times 0.9$, where b is the baseline for epoch e. The initial learning rate is 0.0001, which is linearly decreased to zero in the second half of the training. For gradient descent optimization, we use Adam [37] optimizer with the settings as $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. For loss balancing, we using $\lambda_1 = 0.01$, $\lambda_2 = 1.0$, $\lambda_3 = 0.1$ and $\lambda_4 = 1.0$ in Equation 3.9. Following [21] and [90], the photometric losses are calculated at four different scales and added together with equal weighting. To make the discriminator more robust against colour and intensity variation, we augment discriminator inputs by randomly scaling the colour channels individually with factors from 0.8 to 1.2.

3.2.2 Datasets

We evaluate our model primarily on the KITTI [20] dataset, having 42,382 rectified stereo pairs from 61 videos captured from a camera rig mounted on the roof of a car. All the images have a resolution of 1242×375, which we downsample to 512×256 for memory efficiency. We perform the evaluation at full-resolution by resizing the depth maps using bilinear interpolation. To compare our work with prior work on depth estimation, we use two commonly used train-test splits.

KITTI Split Out of the 42,382 stereo pairs, the training set comprises of 30,159 images. The test set contains 200 images with sparse disparities from 28 videos and the remaining 33 videos are used for training. We split the training set further in validation and train splits. The validation set contains 10 images randomly sampled from every video in the training set. The remaining 29,829 images make the final training set. For evaluation we use the metrics proposed by Eigen *et al.* [16] as shown in Table 3.1. Our method with structured adversarial learning estimates more accurate disparities compared to previous unsupervised methods. The disparity maps in this test split have been augmented with CAD models for vehicles. Consequently, the disparity values at the object boundaries are inaccurate. Therefore, we also evaluate our method on the Eigen split for the sake of completeness.

Eigen Split Table 3.3 shows comparisons with other methods on the train-test split proposed in [16]. The test set in this split comprises of 697 images with corresponding sparse depth maps. These images are present in 29 scenes and the training set encompasses the rest of the 32 scenes. We use the same train-val split strategy that we use for KITTI split, to form a training set with 23,168 images and a validation split with 320 images.

3.2.3 Comparisons with other Methods

We show quantitative comparison with previous monocular depth estimation methods in Table 3.1 for the KITTI split and Table 3.3 for the Eigen split. For the Eigen split, the evaluation is performed inside

Semantic Categories	Methods		
	[21]	Ours - Photo.	Ours - Adv
Flat	2.165	2.376	2.149
Human	3.190	2.763	2.770
Vehicle	3.690	3.968	3.507
Construction	9.713	9.936	9.364
Object	12.763	11.789	11.537

Table 3.2: RMS errors for semantically categorized objects. From top to bottom, the categories are arranged in increasing order of textural uncertainty. Our model with adversarial loss is less error prone for objects like sign-boards, traffic lights, poles, *etc.*, than our model trained with only photometric losses.

	Error (lower the better)				Precision (higher the better)		
	Abs. Rel	Sq. Rel	RMSE	log RMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen *et al.* [16] – Fine *	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Liu *et al.* [47] *	0.202	1.614	6.523	0.275	0.678	0.895	0.965
Kuznietsov *et al.* [39] *	0.122	0.763	4.815	0.194	0.845	0.957	0.987
Amir *et al.* [7] †	0.110	0.929	4.726	0.194	0.923	0.967	0.984
Zhou *et al.* [90]	0.183	1.595	6.709	0.270	0.734	0.902	0.959
Yang *et al.* [83]	0.182	1.481	6.501	0.267	0.725	0.906	0.963
Godard *et al.* [21]	0.148	1.344	5.927	0.247	0.803	0.922	0.964
Zhan *et al.* [86]	0.135	1.132	5.585	0.229	0.820	0.933	**0.971**
Ours – StrAT	**0.128**	**1.019**	**5.403**	**0.227**	**0.827**	**0.935**	**0.971**
Poggi *et al.* [59]	0.129	0.996	5.281	0.223	0.831	0.939	0.974
Aleotti *et al.* [3]	0.118	0.908	4.978	0.210	0.855	0.948	0.976
Zhou *et al.* [90]	0.201	1.391	5.181	0.264	0.696	0.900	0.966
Garg *et al.* [19]	0.169	1.080	5.104	0.273	0.740	0.904	0.962
Godard *et al.* [21]	0.140	0.976	4.471	0.232	0.818	0.931	0.969
Zhan *et al.* [86]	0.128	0.815	4.366	0.225	0.818	0.937	0.973
Ours – StrAT	**0.122**	**0.768**	**4.095**	**0.214**	**0.842**	**0.943**	**0.974**
Aleotti *et al.* [3]	0.112	0.673	3.804	0.198	0.868	0.953	0.979

Table 3.3: Results on KITTI [20] train-test split proposed by Eigen *et al.* [16]. All the methods except [16] are evaluated on the crop region used in [19]. The results are categorized according to the maximum depth cap (80m – upper half, 50m – lower half). * – trained with additional depth supervision. [7] † is trained on a synthetic dataset with depth supervision. Our method with structured adversarial training (StrAT) performs better than all other unsupervised techniques and is comparable with the supervised ones. The results in grey were published post 3DV 2018.

the crop regions proposed in [19]. Our method performs better than other unsupervised methods [21, 75, 83, 90] and is comparable with supervised methods [7, 39] both in terms of precision and errors. We use the metrics defined in [16] and calculate the errors in the depth space. Since our network predicts disparities and not depth, it could lead to precision issues.

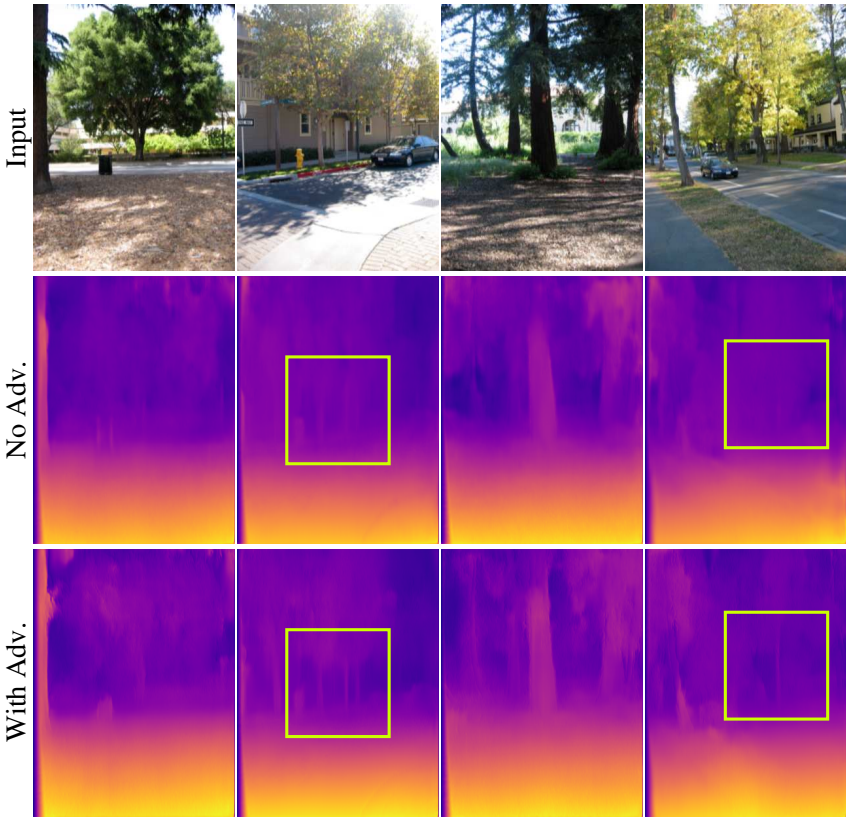


Figure 3.6: Generalization results. The model is trained only on KITTI [20] and the test images are from Make3D [64]. The model trained with adversarial loss performs better than the model trained on photometric loss terms. The differences are highlighted in the boxed regions. Adversarial training has helped in segmenting the objects better in the scene.

3.2.4 Ablation Study

To establish the benefit of using adversarial learning for the problem of monocular depth estimation, we perform an ablation study. The results of the study are shown in Table 3.1. We evaluate the performance of our model with $\lambda_1 = 0$, *i.e.* without adversarial loss (shown as *Ours - No* $\mathcal{L}_{adv}$). We compare its performance with the model trained with adversarial loss using three baseline choices–fixed, random and structurally varying. Our model with structured adversarial training performs the best across all metrics. The model trained with fixed baseline training scheme is comparable to the model trained without adversarial loss. This indicates that training without using stereo-pairs with different baselines is ineffective which is in accordance with our hypothesis in Section 3.1.3.

We show qualitative comparisons between our model without adversarial learning and with structured adversarial training in Figure 3.3. The structured training scheme generates better depth for thin structures like poles and low texture density surfaces like a white wall. This can become noteworthy especially for challenging datasets.

3.2.5 Semantic Errors

To evaluate the depth estimation performance on specific object categories, we make use of semantic segmentation maps released for the KITTI test set [4]. We classify the pixels in every image into five object categories as shown in Table 3.2. The RMSE for flat surfaces like road and pavements is the lowest and for objects like sign-boards, traffic-lights, *etc.*, it is the highest. The structured training model achieves positive results for all the semantic categories. Note that the errors for the *constructions* category is higher than the errors for the *human* category. We believe that the errors are proportional to the uncertainty in the appearance of the objects in terms of their texture, shape and colour.

3.2.6 Generalization

We show generalization of our approach in Figure 3.6, in which our model, trained with and without adversarial learning are compared. Both the models are trained on KITTI [20] and the test samples are from Make3D [64]. Even though the appearance of the objects in Make3D is different from the ones in KITTI, adversarial training has helped in capturing more variance in object appearance.

3.2.7 Limitations

Just like other stereo-supervision methods [19, 21], our model produces erroneous depth for object boundaries. This is because stereo-matching implicitly assumes that the scene is devoid of occlusions. Also, generating adjacent stereo-views from a single image using only disparities is over-constrained. We show this problem with stereo-supervision methods and more failure cases in Figure 3.7. Unsurprisingly, our method also fails for low-intensity regions. This is solely because of information loss in these regions and is not a drawback of our method. As shown quantitatively in Table 3.2 and qualitatively in

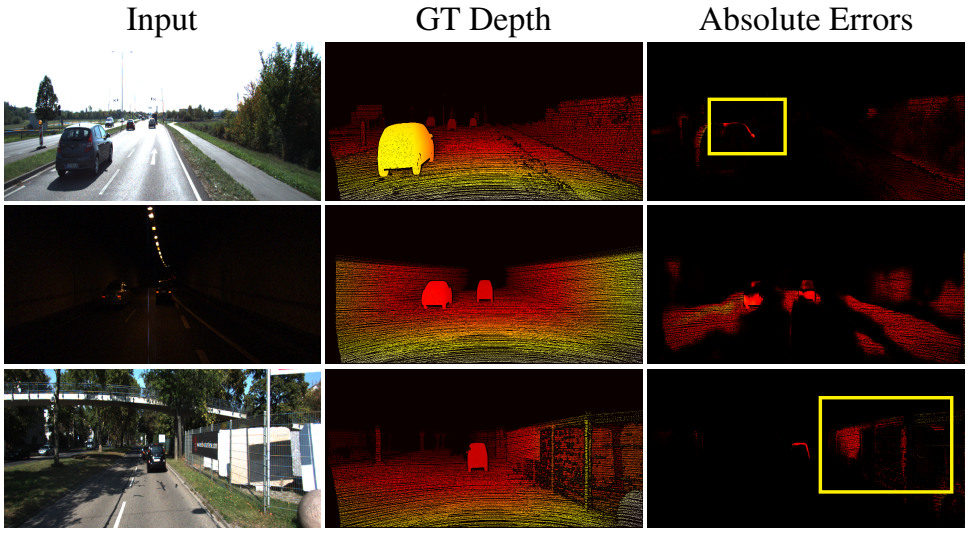


Figure 3.7: Failure cases. Our model estimates incorrect depth for object boundaries, low-intensity regions and objects with high textural uncertainty. Most of the issues can be resolved with more training data.

Figure 3.7, our model generates erroneous depth for the *constructions* category. This limitation can be attributed to epistemic uncertainty [36] and it can be resolved with more training data.

3.2.8 Other Results

Figure 3.9 shows a qualitative comparison between depth maps estimated using our method, with maps generated using a state-of-the-art method [21]. Our method performs well for thin objects (like poles and trees) and at object boundaries. We also show better view synthesis by our method compared to the method proposed by Godard *et al.* [21]. More results are shown in figure 3.8.

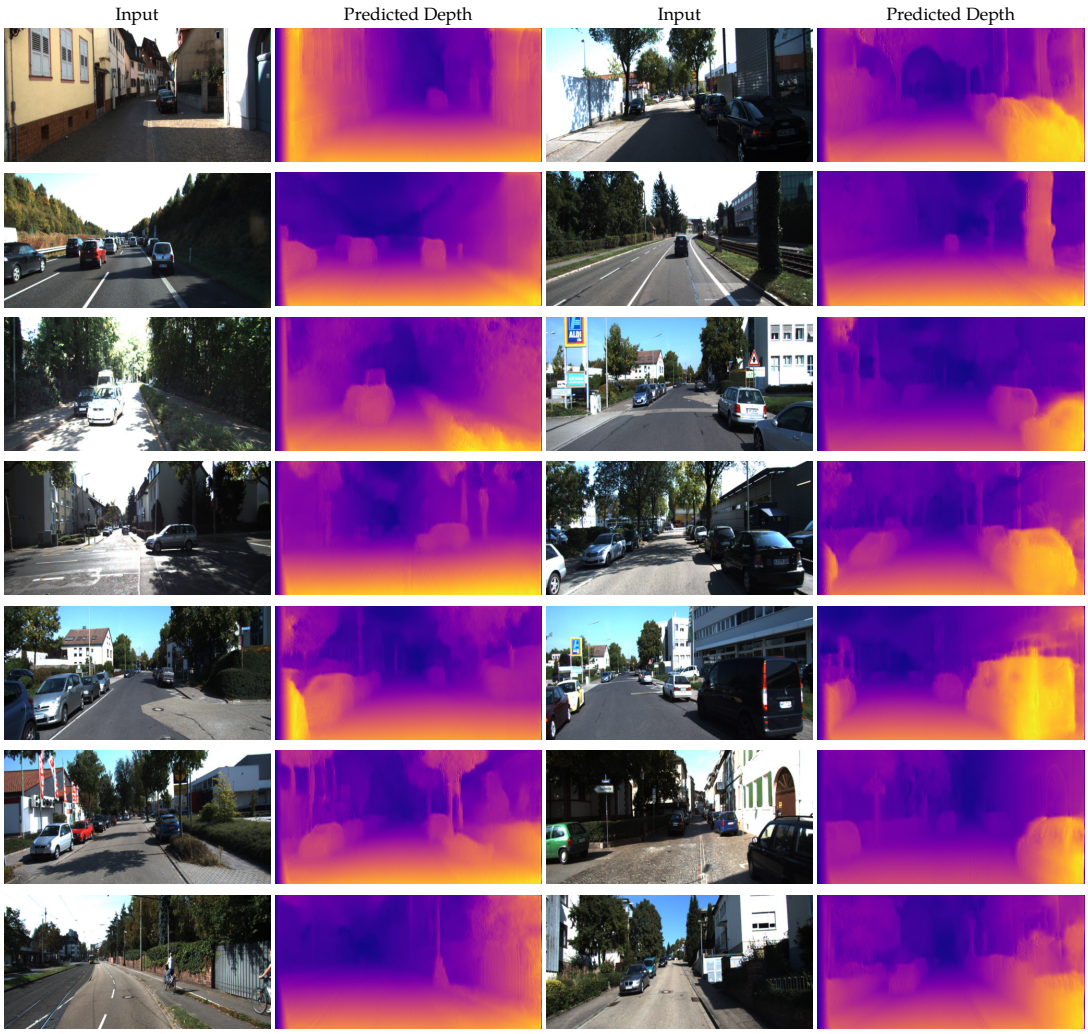


Figure 3.8: Estimated depth maps for images from KITTI test split using StrAT.

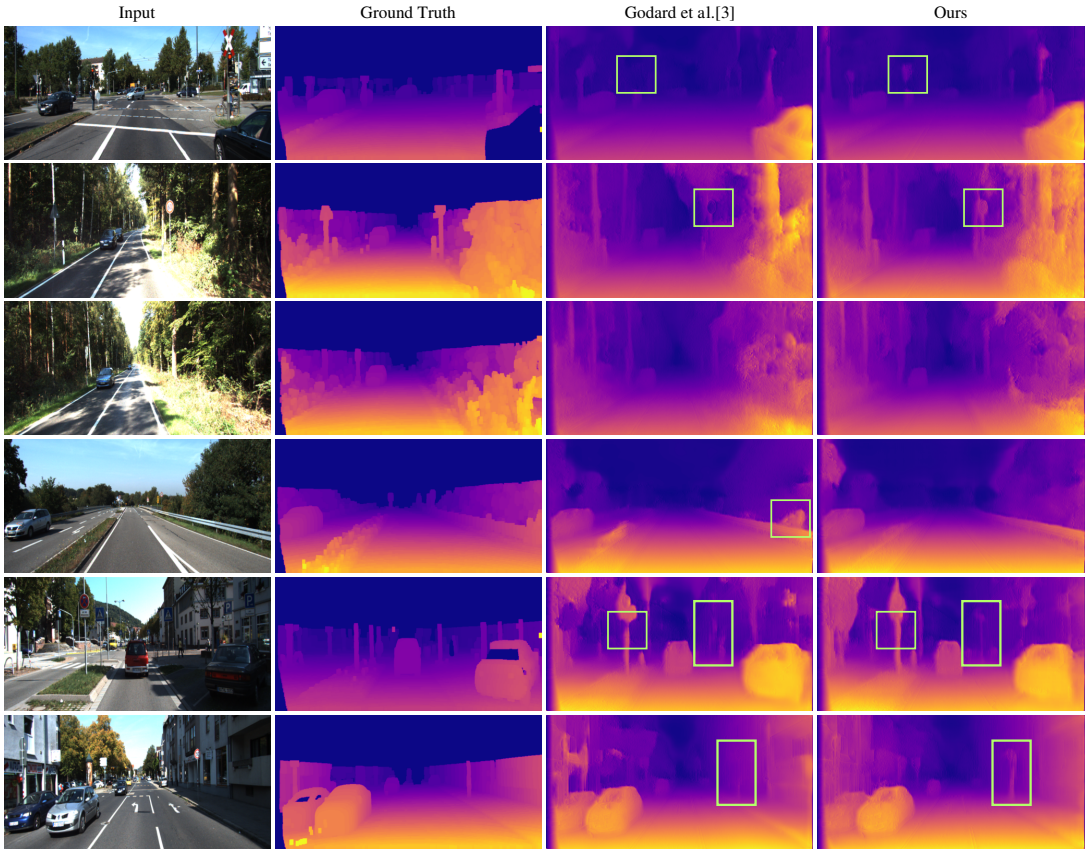


Figure 3.9: Qualitative comparison between our model trained with structured adversarial training and [21]. The input images are from KITTI [20].

Figure 3.10: Comparison of right stereo view at the widest baseline estimated using StrAT and [21]. There are four sets of images, with the top image in each set generated by StrAT and the bottom image by [21]. The views generated using fractional baselines are more realistic than the views generated by contemporary methods.

Chapter 4

Biological and Machine 3D Perception

In this chapter, we compare state-of-the-art monocular depth estimation methods with human monocular depth perception. By demonstrating qualitative and quantitative experiments, we draw parallels between biological 3D perception and a recent single-image depth estimation method explained in chapter 3.

4.1 Introduction

The progress of present deep learning models for single view depth estimation depends heavily on public datasets like KITTI [20] and Cityscapes [11]. Existing methods are almost always trained to increase their performance on the mentioned datasets using a few standardized metrics. These metrics, however, provide an incomplete picture. They inform us about how close the predictions of a model are to the ground-truth provided for a very specific dataset, but tell us nothing about why the model came to a given prediction. The state-of-the-art methods are very accurate but good predictions solve only half of the problem. The other half, interpretability of these predictions, is looked as a separate problem. Present depth estimation methods inadvertently incorporate biases present in the training datasets. Even minimal visual changes in the test samples leads to significant deterioration in predicted depth maps [7]. By training these models to not only maximize a given metric, but also increasing the explainability of the predictions will lead to models which are adaptive to change in sample domain.

Deep learning based computer vision models are being increasingly used in real-world applications. Some of the systems developed for these applications may have to take high-risk actions. A self-driving car for instance, depends on 3D perception algorithms to steer itself in a complicated environment involving humans. It is imperative that their 3D perception modules are tested thoroughly and take explainable actions. A lapse in their predictions could lead to serious consequences [79]. Designing safety-measures for such systems necessitates the knowledge of priors used by all of the decision-making modules.

In this chapter we look at the biases and priors in a CNN-based depth estimation method explained in chapter 3 (DCNN). We perform ablative experiments to isolate depth cues used in biological vision

systems. By contrasting the significance of these cues in DCNN against human depth perception, we highlight future directions for research.

4.2 Depth Contrast

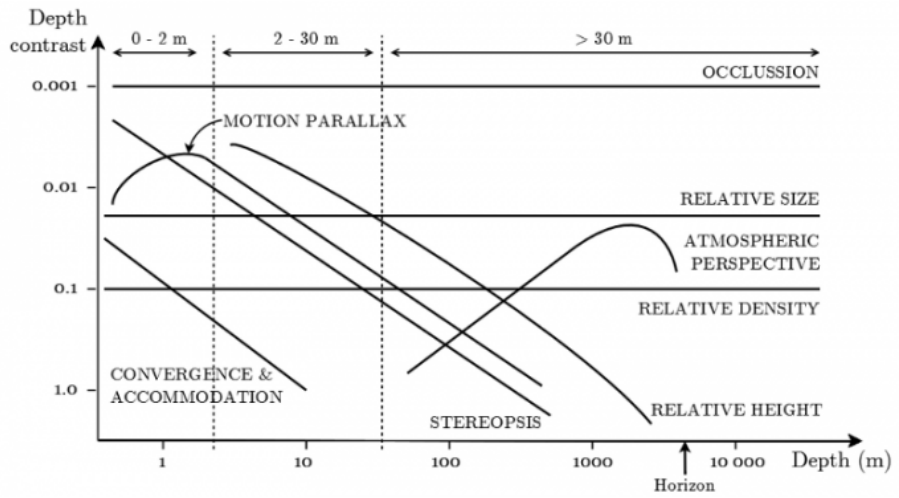


Figure 4.1: Significance of depth cues in estimating distance to an object from an observer. y-axis is the ratio of just-determinable distance and their mean distance ($2 \times \frac{d_1 - d_2}{d_1 + d_2}$) as a function of x-axis which is $\frac{d_1+d_2}{2}$. Adapted from Cutting and Vishton [12]

There are several depth cues used by humans to register the spatial layouts of our surroundings. Sources of information like linear perspective, texture, light and shading, occlusion, disocclusion and gravity are aggregated intelligently. These cues are highly context dependent and usually do not play a singular role. However, some of them are more dominant than others while creating a 3D impression. One way to evaluate the significance of a depth cue is to estimate its ability to discern between two distances from an observer. Figure 4.1 illustrates this with *depth contrast* ΔD [12] plotted against increasing distance from the observer. Here,

$$\Delta D = 2 \times \frac{d_1 - d_2}{d_1 + d_2},$$

where d_1 and d_2 are just-noticeable distances at a given depth from an observer. Considering the case of a non-stationary monocular observer, motion parallax and stereopsis render futile. Among the remaining sources, occlusion, relative size and relative height (or height in the visual field) have high depth contrast (low in magnitude) up to 30 meters. In this chapter we discuss these depth cues from the perspective of computer vision, where in we observe the importance and nuances of them in recent single-image depth estimation methods.

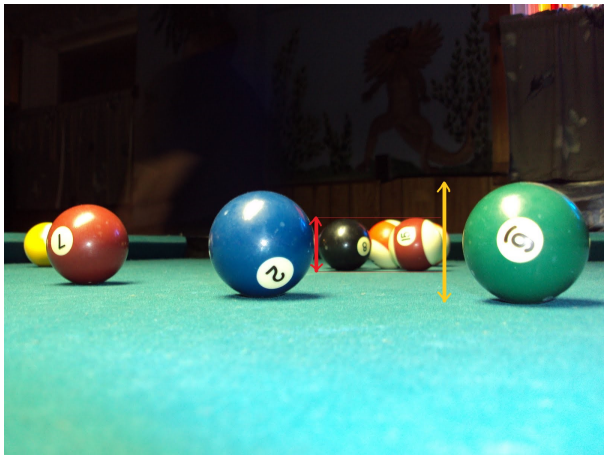


Figure 4.2: Relative size depth cue. The balls closer to the camera appear larger than the balls further away.

4.3 Relative Size

Relative Size is the perceived increase in distance to an object as it decreases in size—*i.e.* an object appears smaller as it moves away from the viewer. In words of Leonardo Da Vinci, "There is no object so large... that at a great distance from the eye it does not appear smaller than a smaller object near. Among objects of equal size, that which is most remote from the eye will look the smallest." [13] This is illustrated in Figure 4.2, where the balls closer to the viewer appear larger than the balls further away.

In this section, we investigate the importance of the prior relative-size in a contemporary depth estimation method, by conducting a series of synthetic experiments. We render images with varying sizes of a familiar object and use our depth estimation CNN (chapter 3) to infer distances to these objects. Section 4.3.1 goes into more detail about the procedure.

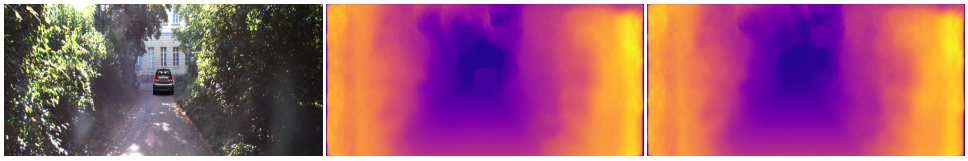
4.3.1 Method

To understand the role of the size of an object in estimating its distance from a camera, we create a dataset with images extracted from the KITTI test set [20] which are subsequently overlaid with objects/sprites (car and pedestrian) of varying sizes. Some sample images are shown in Figure 4.3. The dataset consists of 500 images with two synthetic object categories at 50 different scales. These images are fed into a DCNN (Depth CNN proposed in chapter 3) and the changes in the depth values are observed. The DCNN uses a Resnet50 [25] encoder and is trained on the KITTI train-test split.

We conduct an experiment in which for every image in our dataset, a sprite with a relative scale factor between 1-50 is overlaid as foreground. The resultant image is then used an input to DCNN which estimates a depth map for the same. The relative depth of the region in which the sprite is overlaid is then reported.

Background Foreground

Figure 4.3: Top: Samples from our synthetic dataset. Bottom: A car sprite is used as foreground with varying scale and position on a background image



Input (with sprite) Depth (with sprite) Depth (no sprite)

Figure 4.4: Left: Input image to DCNN. Middle: Estimated depth map with the car sprite. Right: Reference depth map without overlay. The depth map in the middle shows a plausible depth estimation for the car sprite.

4.3.2 Results and Discussion

Since the context of the scene is consequential for 3D perception, we restrict the location of the sprite on the images in the foveal region. We observe that DCNN falters at predicting depth for objects which do not conform to the overall scene structure.

Figure 4.5 shows a plot of depth estimated using DCNN with increasing size of a sprite. We conduct two experiements: one in which the object (in foregoround) is present in the training data and another in which the object is unseen by DCNN. Expectedly, the estimated distance to the object increases with decrease in its size. This is similar to human depth perception as we use the size of an object to guage its distance. However it becomes unusable for unfamiliar objects and we need to rely on other depth cues which do not consider the familiarity of the object, *e.g.*, stereopsis.

However, if there are multiple instances of the same object at varying distances, humans can perceive the ordinal relation between the depths of these instances. We investigate if such a prior exists in DCNN, by placing two instances of an unfamiliar object (*e.g.* a box crate), one of which has a fixed scale factor and the other varying in size. Figure 4.6 shows that DCNN does not have a relative-size depth prior,

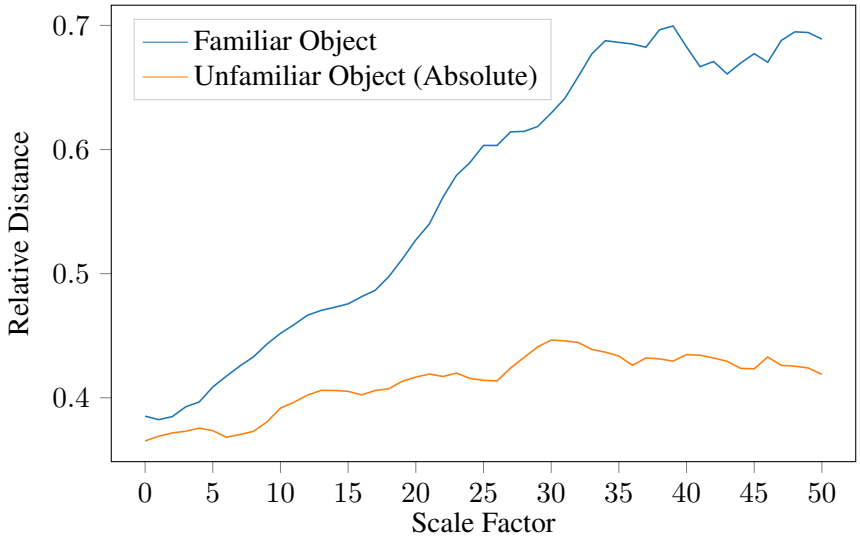


Figure 4.5: Variation in estimated (absolute) depth using DCNN with changes in the size of an object (car). The depth linearly increases with increase in relative size (scale factor) for familiar objects like cars, however for unfamiliar objects (which are not present in the training set), there is minimal variation in depth, which is akin to human depth perception.

unlike human depth perception. By adding an additional instance of an object, the depth estimated for the objects do not improve.

For a human observer, when an object is familiar, absolute information is available to turn relative size into absolute size. But, for unfamiliar objects, we do rely on the relative size of the objects if there are more than one instances available in the scene. We hypothesize that our depth estimation model and other contemporary methods fail to consider the object relationships (except for occlusion as we show in section 4.4) in a given scene. Adding additional priors in forms of weak supervision or strict supervision, can improve single image depth estimation. We propose this as a future avenue for research in 3D estimation.

4.4 Occlusion

Occlusion is one of the cues which does not provide any metric information of depth but provides highly reliable ordinality [12]. It occurs when an object partially hides behind another object, resulting in occluding edge from which one can infer discontinuity in depth. It is one of the most reliable sources of information at all distances (as shown in Figure 4.1). Although it does not provide absolute depth information, a multi-dimensional scaling procedure exists which generates near-metric depth information by only relying on ordinal information [67].

There are two major assumptions to be considered for occlusion as a useful source. One of them is called the Helmholtz's rule [12] in which the contour of the object which is the occluder, probably does not align with the contour of the object that is being occluded. It is highly unlikely that the object

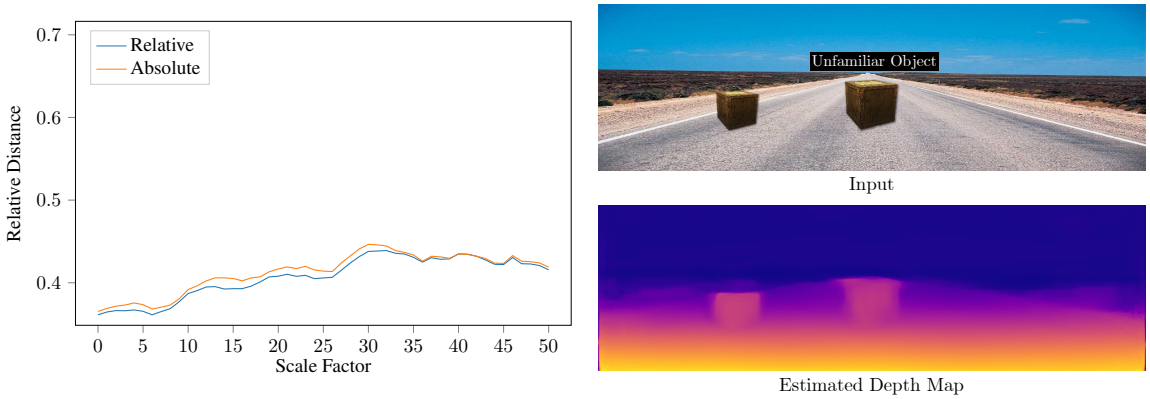


Figure 4.6: DCNN does not incorporate the relative size of an unfamiliar object for monocular depth estimation. There is no improvement in estimated depth by adding another instance of the same object with a fixed size.

boundaries align at the point of occlusion for a given view direction. The second assumption is that there is the luminance contrast at the occlusion boundary is considerable. If there are objects with the same color, one occluding the other, Gestalt's law of proximity and continuity may apply [12].

4.4.1 Experiments

In a vein similar to experiments in section 4.3.2, we investigate the importance of occlusion prior in DCNN. We overlay a background image devoid of any foreground objects like cars, pedestrians, etc., with two car sprites in a configuration where it seems like one car is occluding the other. The resulting image is used as an input to DCNN. We perform this experiment with varying sprites sizes and see perceptual discontinuity in depth at the occlusion boundary. Figure 4.7 (b) shows a sample depth map.

As mentioned in section 4.4, high luminance contrast is one of the fundamental assumptions to use occlusion as a depth prior. We observe that this idea is also reflected in depth estimation by DCNN. We perform the same experiments as discussed in the previous paragraph, with the exception that the contrast between the luminance of the car getting occluded and the car being occluded is high. Figure 4.7 shows that there is a steeper discontinuity in depth at the occlusion boundary with higher luminance contrast. The absolute difference (shown in figure 4.7 (d)) between the depth map estimated with the same sprites and the one estimated with sprites having higher contrast, at the occlusion boundary is high.

We also observe that occlusion plays a role only for familiar objects which are present in the training set. This may be explained by the overfitting nature of DCNN or that the model fails to interpret "objectness."

4.5 Height in the Visual Field

In Euclid's words—"In the case of flat surfaces lying below the level of the eye, the more remote parts appear higher." Height in the visual field is a result of perspective projection of objects on the

(a) Input (d) Absolute Difference

(b) Depth with low contrast (c) Depth with high contrast

Figure 4.7: Qualitative improvement in depth estimation with high luminance contrast at the occlusion boundary. (a) is the input image with two car sprites having high contrast. (c) is the depth map estimated by DCNN using (a). (b) is a depth estimated for an input image similar to (a) with the exception of using the same sprites (having the same colour). (d) is the absoluted difference between (b) and (c). It is apparent that the difference at the occlusion boundary is high.

retina. Like occlusion, it provides ordinal information about distance from the point of observation. It can also provide metric information if there are strong assumptions that can be made about the scene. Some of these assumptions are:

- The (ground) plane on which the camera is rested upon is opaque
- Gravity (no object is suspended or floating)
- The distance from the plane to the camera is known
- The ground plane is orthogonal to gravity

As shown in Figure 4.1, the relative height only starts being useful at a distance of about 2 m from the observer. At that distance it is just as resourceful as occlusion. However, after 2 m, its effectiveness diminishes curvilinearly. This is unlike occlusion and relative height as they remain effective at all distances. In the following section we show that this cue is one of the strongest priors used by DCNN and accurately replicates human perception.

4.5.1 Experiments

We follow the same methodology explained in section 4.3.1 to investigate height in the visual field as a depth prior in DCNN. We place a car sprite at varying row heights in a given background image and note the change in corresponding distance to the car estimated by DCNN. We use 20 images from the KITTI test split as foreground images. Figure 4.8 shows a sample test image and its corresponding depth map estimated using DCNN. One can observe that the areas in the depth map corresponding to the

(a) Input (b) Depth Map

Figure 4.8: (a) is from the KITTI [20] dataset overlayed with car sprites at different heights and sizes. (b) is the depth map estimated using DCNN. Objects higher in the visual field perceptually seem further away.

car sprites become darker as the contact point of the car on the road gets higher. Here, we also reduce the size of the object using the same scaling factor used to increase relative height of the object.

4.5.1.1 Ablation Study

We also show that by keeping the object size fixed, there is minimal change in the estimated distance by DCNN. To remove the relative size cue, we use a forgeound image which does not have any cars in the scene. We place car sprites of fixed size from row (*i.e.* edge of the image) up to the horizon. Figure 4.9 shows the change in depth with increase in relative height. It also shows that there is minimal change in the estimated distance between depths estimated with and without fixed object size.

4.5.2 Discussion

Ooi *et al.* [56] demonstrate the significance of height in the visual field (HVF) as a cue to depth by manipulating HVF using vertically displaced prisms. They show that HVF varies tangentially with distance:

$$tan(90°\text{- HVF}) = \frac{\text{observer's height}}{\text{distance from the observer}}. \tag{4.1}$$

Here, HVF is represented in terms of the visual angle subtended by the line of sight of the observer and the direction perpendicular to the ground plane on which the observer rests. 0° represents an object right in front of the observer and 90° is for an object on the horizon. We show that the plot in Figure 4.9 is very similar to a plot obtained from equation 4.1. The plot for equation 4.1 is shown in figure 4.10. We assume that the observer's height in equation 4.1 is 0.25 m and plot the values for angles in [0°, 60°]. We restrict the plot to 60° since the the model is trained on KITTI which does not have enough samples with objects on the horizon.

4.6 Summary

In this chapter we showed three depth cues that are used by the human visual system and investigated if these cues or priors also exist in our monocular depth estimation model (DCNN). We used three most

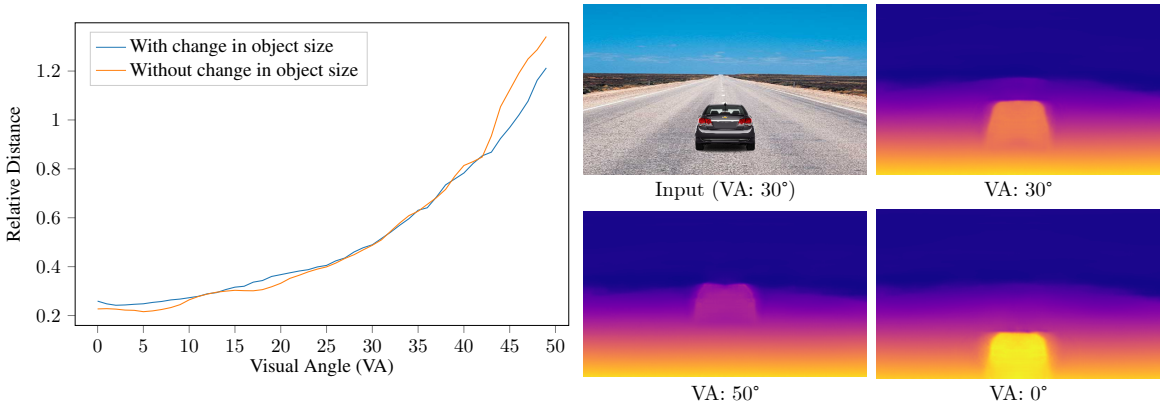


Figure 4.9: Results for ablative study on object size combined with height in the visual field as depth priors for DCNN. The height is represented as visual angle: the angle between the line of sight and the direction perpendicular to the ground plane.

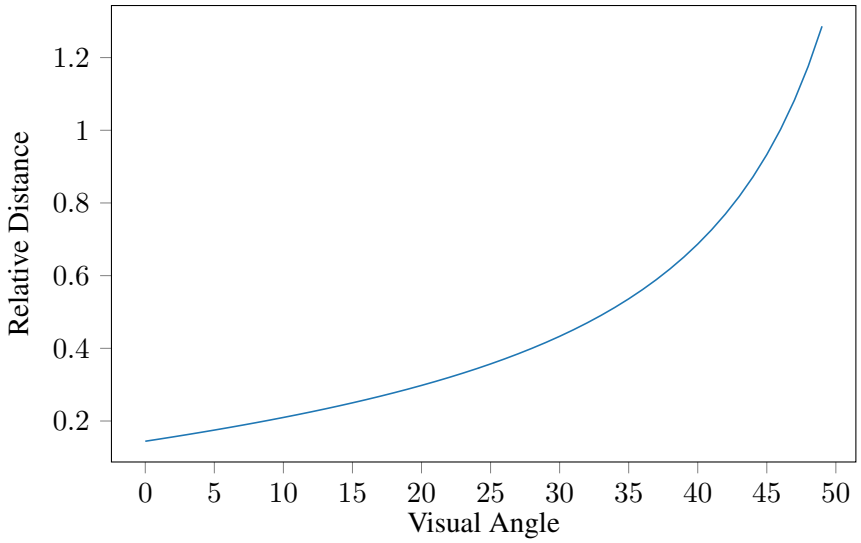


Figure 4.10: The height in the visual field varies tangentially with distance. A visual angle of 90° is for objects on the horizon and 0° is for objects right in front of the observer. This plot is highly similar to the plot in figure 4.8 which validates the strength of HVF prior in monocular depth estimation.

important monocular cues: relative size, occlusion and height in the visual field, to conduct a series of experiements to validate that these cues do exist as priors in DCNN. We also show that these priors exist with varying significance, with height in the visual field being closest to human perception. Explicitly modelling such cues in forms of weak supervisory signals can improve the current monocular depth estimation methods. The benefits of such modelling are two-fold: 1) It will force the models to be interpretable, and 2) Reduce overfitting to a particular dataset which might not capture all the depth cues. We propose these ideas as future directions for research.

Chapter 5

Conclusions

Several dense estimation tasks like scene-refocusing, optical flow and depth estimation can be approached using view-synthesis. Using view supervision eliminates the need for ground truth for these tasks. The recent works in these areas rely only on data which is available in the training set, *e.g.* light-fields, stereo images or monocular videos. In this work, we propose that more views can be generated and used for training even when the corresponding ground-truth is not present in the training set. The new views can be validated perceptually with the views in the training set. We propose a generative adversarial network, in which the discriminator captures this perceptual quality and does not rely on pixel-by-pixel matching. We also show that training with views generated without any meaningful order can be an impediment to the learning process. We argue using quantitative analysis that by adding structure into the process of view-synthesis can substantially improve depth estimation. In future, this idea can be extended to other dense estimation tasks like optical flow. It will also be interesting to discover new training schemes and understand the learning behaviour of both the generator and discriminator.

In chapter 4 we showed three depth cues that are used by the human visual system and investigated if these cues or priors also exist in our monocular depth estimation model. We showed that height in the visual field is the strongest of these priors in artificial depth perception, and that other priors also loosely exist. Current methods can be improved by explicitly modelling such cues in forms of weak or strict supervisory signals.

One of the ways to inject priors in view-synthesis based depth estimators is to design architectures which explicitly estimate the important depth cues–which can be subsequently combined using a differentiable aggregator. Analogous to using depth representations for view synthesis, these depth cues can be used as intermediate representations for depth estimation. Inspiration can be drawn from CNN based image re-lighting modules [30] which decompose a given image into intrinsic images. These methods use differentiable renderers to reconstruct an image from the intrinsic parameters. This will ensure that the models are interpretable and will reduce overfitting on a specific dataset. For action-critical systems, these results are pre-requisites. We propose to explore these ideas in the future.

Related Publications

- Ishit Mehta, Parikshit Sakurikar and P. J. Narayanan, "Structured Adversarial Training for Unsupervised Monocular Depth Estimation", International Conference on 3D vision (3DV), Verona, Italy, 2018

Bibliography

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, 2016.

[2] P. Albrecht and B. Michaelis. Improvement of the spatial resolution of an optical 3-d measurement procedure. *IEEE Transactions on Instrumentation and Measurement*, 47(1):158–162, 1998.

[3] F. Aleotti, F. Tosi, M. Poggi, and S. Mattoccia. Generative adversarial networks for unsupervised monocular depth prediction. In *European Conference on Computer Vision*. Springer, 2018.

[4] H. A. Alhaija, S. K. Mustikovela, L. Mescheder, A. Geiger, and C. Rother. Augmented reality meets deep learning for car instance segmentation in urban scenes. In *British Machine Vision Conference*, 2017.

[5] A. Arditi. Binocular vision. *Handbook of perception and human performance.*, 1, 1986.

[6] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[7] A. Atapour-Abarghouei and T. Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation. In *CVPR*, 2018.

[8] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *ICML*, 2009.

[9] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *PAMI*, 1992.

[10] Y. Cao, Z. Wu, and C. Shen. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 2017.

[11] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[12] J. E. Cutting and P. M. Vishton. Perceiving layout and knowing distances: The integration, relative potency, and contextual use of different information about depth. In *Perception of space and motion*. Elsevier, 1995.

[13] L. da Vinci. *The Notebooks of Leonardo Da Vinci.* 1888.

[14] E. Delage, H. Lee, and A. Y. Ng. Automatic single-image 3d reconstructions of indoor manhattan world scenes. In *Robotics Research*, pages 305–321. Springer, 2007.

[15] T. Douglas, S. Solomonidis, W. Sandham, and W. Spence. Ultrasound imaging in lower limb prosthetics. *IEEE Transactions on neural systems and rehabilitation engineering*, 10(1):11–21, 2002.

[16] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NIPS*, 2014.

[17] A. R. François and G. G. Medioni. Interactive 3d model extraction from a single image. *Image and Vision Computing*, 19(6):317–328, 2001.

[18] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018.

[19] R. Garg, V. K. BG, G. Carneiro, and I. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*, 2016.

[20] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012.

[21] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017.

[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014.

[23] W. L. Gulick and R. B. Lawson. *Human stereopsis: A psychophysical analysis*. Oxford University Press, USA, 1976.

[24] J. Gwak, C. B. Choy, M. Chandraker, A. Garg, and S. Savarese. Weakly supervised 3d reconstruction with adversarial constraint, 2017.

[25] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[26] F. S. Helmli and S. Scherer. Adaptive shape from focus with an error estimation in light microscopy. In *ISPA 2001. Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis. In conjunction with 23rd International Conference on Information Technology Interfaces (IEEE Cat.*, pages 188–193. IEEE, 2001.

[27] D. Hoiem, A. A. Efros, and M. Hebert. Automatic photo pop-up. In *ACM transactions on graphics (TOG)*, volume 24. ACM, 2005.

[28] B. K. Horn. Obtaining shape from shading information. *The psychology of computer vision*, pages 115–155, 1975.

[29] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015.

[30] M. Janner, J. Wu, T. D. Kulkarni, I. Yildirim, and J. Tenenbaum. Self-supervised intrinsic image decomposition. In *Advances in Neural Information Processing Systems*, pages 5936–5946, 2017.

[31] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[32] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[33] K. Karsch, C. Liu, and S. B. Kang. Depth extraction from video using non-parametric sampling. In *ECCV*, 2012.

[34] K. Karsch, C. Liu, and S. B. Kang. Depth transfer: Depth extraction from videos using nonparametric sampling. *PAMI*, 2016.

[35] L. Kaufman. *Sight and mind: An introduction to visual perception.* Oxford U. Press, 1974.

[36] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NIPS*, 2017.

[37] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. arxiv. org, 2014.

[38] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. *arXiv preprint arXiv:1711.07064*, 2017.

[39] Y. Kuznietsov, J. Stückler, and B. Leibe. Semi-supervised deep learning for monocular depth map prediction. In *CVPR*, 2017.

[40] L. Ladicky, J. Shi, and M. Pollefeys. Pulling things out of perspective. In *CVPR*, 2014.

[41] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3D Vision (3DV), 2016 Fourth International Conference on*, 2016.

[42] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*, 2015.

[43] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. *CVPR*, 2016.

[44] C. Li and M. Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European Conference on Computer Vision*. Springer, 2016.

[45] R. Li, S. Wang, Z. Long, and D. Gu. Undeepvo: Monocular visual odometry through unsupervised deep learning. *arXiv preprint arXiv:1709.06841*, 2017.

[46] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. *ICCV*, 2015.

[47] F. Liu, C. Shen, G. Lin, and I. Reid. Learning depth from single monocular images using deep convolutional neural fields. *PAMI*, 2016.

[48] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[49] J. M Benjamin. The laser cane. *Bulletin of prosthetics research*, pages 443–50, 02 1974.

[50] R. Mahjourian, M. Wicke, and A. Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. *arXiv preprint arXiv:1802.05522*, 2018.

[51] G. Mather. Image blur as a pictorial depth cue. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 263(1367):169–172, 1996.

[52] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016.

[53] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[54] T. Nagai, T. Naruse, M. Ikehara, and A. Kurematsu. Hmm-based surface reconstruction from single images. In *Proceedings. International Conference on Image Processing*, volume 2, pages II–II. IEEE, 2002.

[55] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML)*, 2010.

[56] T. L. Ooi, B. Wu, and Z. J. He. Distance determined by the angular declination below the horizon. *Nature*, 414(6860), 2001.

[57] K. E. Ozden, K. Schindler, and L. Van Gool. Simultaneous segmentation and 3d reconstruction of monocular image sequences. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

[58] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[59] M. Poggi, F. Tosi, and S. Mattoccia. Learning monocular depth estimation with unsupervised trinocular assumptions. In *2018 International Conference on 3D Vision (3DV)*. IEEE, 2018.

[60] A. M. Puerta. The power of shadows: shadow stereopsis. *JOSA A*, 6(2), 1989.

[61] K. Regmi and A. Borji. Cross-view image synthesis using conditional gans. *arXiv preprint arXiv:1803.03396*, 2018.

[62] P. R. Sanz, B. R. Mezcua, and J. M. S. Pena. Depth estimation - an introduction. In A. Bhatti, editor, *Current Advancements in Stereo Vision*, chapter 5. IntechOpen, Rijeka, 2012. doi: 10.5772/45904. URL `https://doi.org/10.5772/45904`.

[63] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *Advances in neural information processing systems*, 2006.

[64] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. *PAMI*, 2009.

[65] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 1, pages I–I. IEEE, 2003.

[66] S. Schuon, C. Theobalt, J. Davis, and S. Thrun. High-quality scanning using time-of-flight depth superresolution. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–7. IEEE, 2008.

[67] R. N. Shepard. Multidimensional scaling, tree-fitting, and clustering. *Science*, 210(4468), 1980.

[68] J. Shi, X. Tao, L. Xu, and J. Jia. Break ames room illusion: Depth from general single images. *ACM Trans. Graph.*, 34(6):225:1–225:11, Oct. 2015. ISSN 0730-0301.

[69] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[70] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[71] P. P. Srinivasan, T. Wang, A. Sreelal, R. Ramamoorthi, and R. Ng. Learning to synthesize a 4d rgbd light field from a single image. In *ICCV*, 2017.

[72] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 2014.

[73] L. Trottier, P. Gigu, B. Chaib-draa, et al. Parametric exponential linear unit for deep convolutional neural networks. In *Machine Learning and Applications (ICMLA), 2017 16th IEEE International Conference on*, 2017.

[74] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *CoRR*, abs/1607.08022, 2016.

[75] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki. Sfm-net: Learning of structure and motion from video. *arXiv preprint arXiv:1704.07804*, 2017.

[76] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey. Learning depth from monocular videos using direct methods. *arXiv preprint arXiv:1712.00175*, 2017.

[77] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. *arXiv preprint arXiv:1812.07179*, 2018.

[78] 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*, 2004.

[79] Wikipedia. Death of Elaine Herzberg — Wikipedia, the free encyclopedia, 2019.

[80] R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical engineering*, 19(1):191139, 1980.

[81] J. Xie, R. Girshick, and A. Farhadi. Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks. In *ECCV*, 2016.

[82] B. Xu, N. Wang, T. Chen, and M. Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.

[83] Z. Yang, P. Wang, W. Xu, L. Zhao, and R. Nevatia. Unsupervised learning of geometry with edge-aware depth-normal consistency. *arXiv preprint arXiv:1711.03665*, 2017.

[84] X. Yin, H. Wei, X. Wang, Q. Chen, et al. Novel view synthesis for large-scale scene using adversarial loss. *arXiv preprint arXiv:1802.07064*, 2018.

[85] J. Zbontar and Y. LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 17(1-32), 2016.

[86] 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. *arXiv preprint arXiv:1803.03893*, 2018.

[87] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. *arXiv preprint arXiv:1801.03924*, 2018.

[88] W. Zhang and W.-K. Cham. Single-image refocusing and defocusing. *IEEE Transactions on Image Processing*, 21(2), 2012.

[89] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros. View synthesis by appearance flow. In *ECCV*, 2016.

[90] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017.

[91] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.