Single-Image Stereo Depth Estimation using GANs

Sharan Ramjee Nikhil Parab Naveed Zaman
Stanford University
{sramjee, nik17, naveedz}@stanford.edu

Abstract

Supervised Deep Learning based monocular methods are among the state-of-the-art methods for estimating depth maps from from single images. However, they still lack in performance in comparison to stereo depth estimation methods that estimate depth maps from stereo image pairs. Unfortunately, due to the complexity of setting up stereo systems, monocular depth estimation methods are still the go-to approach when it comes to depth estimation, despite their inferior performance in comparison to stereo depth estimation methods. We propose a novel pipeline to make use of stereo depth estimation methods through the use of Generative Adversarial Networks (GANs). Our single-image stereo depth estimation pipeline makes use of a GAN (DepthGAN) to generate a "plausible" depth map given a single input image, followed by the generation of a stereo counter-part (depth2stereo) to the original input image given the "plausible" depth map, followed by the use of another GAN (StereoDepthGAN) to generate the final depth map given the stereo image pair. The DepthGAN and StereoDepthGAN, pretrained on the KITTI and Cityscapes datasets and fine-tuned on the NYU Depth V2 dataset, enable our pipeline to outperform current state-of-the-art monocular depth estimation methods, as examined in this paper through our extensive experiments. The source code is available on GitHub: `https://github.com/sharanramjee/single-image-stereo-depth-estimation`

1. Introduction

Depth information is crucial in understanding the 3D geometry of a scene. Traditional depth estimation methods, like stereo vision matching and structure from motion, require multiple viewpoint images as they rely on feature correspondences across these images for computing depth maps. Extracting depth from a single image i.e. monocular depth estimation is an ill-posed problem that makes it very challenging. Recent advances in Artificial Intelligence (AI) have led to the development of Deep Neural Network (DNN) based monocular depth estimation techniques that have achieved promising results. In particular, the use of Generative Adversarial Networks (GANs) [8] for monocular depth estimation has gained widespread popularity.

However, recent studies by Smolyanskiy *et al*. [18] and Tosi *et al*. [19] show that the performance of GAN-based monocular depth estimation methods are still inferior to those of GAN-based stereo depth estimation methods with the caveat that stereo depth estimation methods require a stereo pair of input images for depth estimation. The complexity of a stereo camera setup for stereo depth estimation has deterred the use of stereo depth estimation methods in real-world applications and has thus led to a widespread adoption of monocular depth estimation methods [3], despite their relatively inferior performance. In order to address this issue, we propose a novel pipeline for monocular depth estimation in order to generate stereo image pairs for improved depth estimation through the use of GAN-based stereo depth estimation methods.

2. Related Work

Several papers have been published aimed at solving depth estimation problems using deep learning. Notably, Laina *et al*. [11] propose a convolutional architecture that encompasses residual learning to model the ambiguous mapping between monocular images and depth maps. Zhou *et al*. [22] use an unsupervised learning framework for the task of monocular depth and camera motion estimation from unstructured video sequences. An end-to-end learning approach is used with view synthesis as the supervisory signal. Godard *et al*. [6] propose a CNN to learn to perform single image depth estimation despite the absence of ground truth depth data. Exploiting epipolar geometry constraints, disparity images are generated by training the network with an image reconstruction loss. Liu *et al*. [13] present a DNN for piece-wise planar depth map reconstruction from a single RGB image. The proposed end-to-end DNN learns to directly infer a set of plane parameters and corresponding plane segmentation masks from a single RGB image. Watson *et al*. [20] generate plausible disparity maps from single images which are used in a carefully designed pipeline to generate stereo training pairs.

3. Problem Statement

We use existing implementations of the DepthGAN [9], depth2stereo algorithm [14], and StereoDepthGAN [15] and employ transfer learning (pretrained on the KITTI [5] and Cityscapes [4] datasets) to further fine-tune the model parameters using the train set of the NYU Depth V2 dataset. The training curves for fine-tuning the discriminator, generator, and image reconstruction losses are reported for both the DepthGAN and the StereoDepthGAN in Sec. 5.3.

For qualitative evaluation, the error maps (i.e. pixel-wise differences) between the generated depth maps and the ground truth depth maps on the test set are reported in Sec. 6.1. For quantitative evaluation and performance comparison with other state-of-the-art monocular depth estimation methods, we use a set of common metrics [9]: Absolute Relative Distance (ARD), Squared Relative Distance (SRD), Root Mean Squared Error (RMSE), and log Root Mean Squared Error (log RMSE), as reported in Sec. 6.2.

While we expect the other monocular depth estimation methods to outperform the DepthGAN in terms of the plausible/intermediate depth maps generated, we expect the entire single-image stereo depth estimation pipeline (output from the StereoDepthGAN) to outperform these other methods in terms of the final depth maps generated.

Finally, in order to evaluate the performance gain achieved as a result of in-domain training, we also report both the qualitative and quantitative results of the pipeline before and after fine-tuning on the train set of the NYU Depth V2 dataset.

4. Technical Approach

The pipeline for generating the depth maps consists of three components: the DepthGAN [9], the depth2stereo [14] algorithm, and the StereoDepthGAN [15]. The DepthGAN takes in an RGB image as input to generate a plausible depth map, which is then fed into the depth2stereo algorithm along with the initial RGB input image in order to generate a right stereo counterpart. Finally, the stereo image pair is fed into the StereoDepthGAN to generate the final depth map as illustrated in Fig. 1.

4.1. DepthGAN

Groenendijk *et al*. [9] use the reconstruction-based architecture for depth estimation developed by Godard *et al*. [6] for constructing the DepthGAN and extend it with an adversarial discriminator through the formulation of the problem of depth estimation as an image reconstruction task, where a generator network takes a single left view image as input and gives the left-to-right disparity as output. More formally, the generator G used the left image $\boldsymbol{I}^L$ to reconstruct both the left $\hat{\boldsymbol{I}^L}$ and the right $\hat{\boldsymbol{I}^R}$ image. The warping function f_w [21] is then used to reconstruct the left and right stereo pairs using the two disparities outputted by the generator G: left-to-right disparity d^R and right-to-left disparity d^L. Groenendijk *et al*. [9] emphasize that a good generator G should predict d^L and d^R such that the reconstructed images $\hat{\boldsymbol{I}^L}$ and $\hat{\boldsymbol{I}^R}$ are close to the original image pair $\boldsymbol{I}^L$ and $\boldsymbol{I}^R$ as measured by the following image reconstruction losses:

L1 loss to minimize the absolute per-pixel distance:

$$\mathcal{L}_{L1}^l = \frac{1}{N}\sum_{i,j} ||\boldsymbol{I}_{ij}^l - \hat{\boldsymbol{I}}_{ij}^l|| \tag{1}$$

Structural similarity (SSIM) reconstruction loss to measure the perceived quality:

$$\mathcal{L}_{SSIM}^l = \frac{1}{N}\sum_{i,j} \frac{(1 - SSIM(\boldsymbol{I}_{ij}^l, \hat{\boldsymbol{I}}_{ij}^l))}{2} \tag{2}$$

where $SSIM(\cdot,\cdot)$ is the Structural Similarity Index as defined by Godard *et al*. [6].

Left-Right Consistency Loss (LR) to enforce the consistency between the predicted left-to-right and right-to-left disparity maps:

$$\mathcal{L}_{LR}^l = \frac{1}{N}\sum_{i,j} |d_{ij}^l - d_{i(j+d_{ij}^l)}^r| \tag{3}$$

Disparity Smoothness Loss to enforce smooth disparities i.e. small disparity gradients:

$$\begin{aligned}\mathcal{L}_{disp}^l = \frac{1}{N}\sum_{i,j} (&|\nabla_x d_{ij}^l| \exp(-||\nabla_x \boldsymbol{I}_{ij}^l||) + \\ &|\nabla_y d_{ij}^l| \exp(-||\nabla_y \boldsymbol{I}_{ij}^l||))\end{aligned} \tag{4}$$

The DepthGAN generator G outputs scaled disparities at intermediate layers of the decoder when it it upsamples from the bottleneck layer and for each subsequent scale, the height and width of the output image is halved. The reconstruction loss is computed at each scale and the final reconstruction loss is a combination of the losses at the different scales s:

$$\begin{aligned}\mathcal{L}_s = \gamma_{L1}\mathcal{L}_{L1} + \gamma_{SSIM}\mathcal{L}_{SSIM} + \\ \gamma_{LR}\mathcal{L}_{LR} + \gamma_{disp}\mathcal{L}_{disp}\end{aligned} \tag{5}$$

$$\mathcal{L}_{rec} = \sum_{s=0}^{3} \mathcal{L}_s \tag{6}$$

where the γ's weight the influence of each loss component.

The DepthGAN discriminator D is used to discern between the real $\boldsymbol{I}^R$ and fake $\hat{\boldsymbol{I}^R}$ right images. For adversarial training, Groenendijk *et al*. [9] combine the reconstruction loss $\mathcal{L}_{rec}$ with vanilla GAN loss:

$$\mathcal{L}_V^G = -\mathbb{E}[\log D(\hat{\boldsymbol{I}^R})] \tag{7}$$

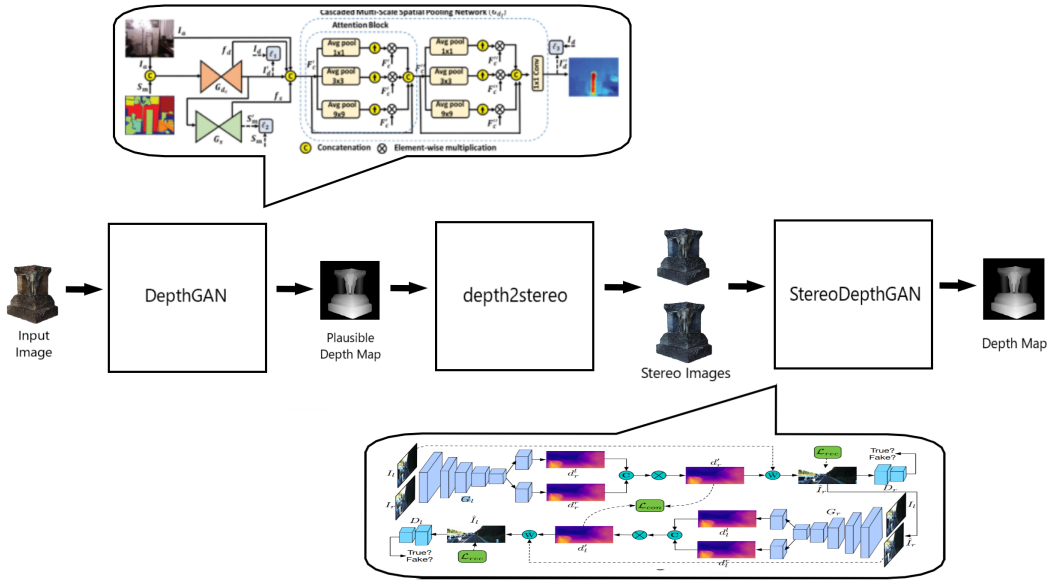

Figure 1. Single-image stereo depth estimation pipeline

$$\mathcal{L}_V^D = -\mathbb{E}[\log D(\boldsymbol{I}^R) + \log(1 - D(\hat{\boldsymbol{I}^R}))] \tag{8}$$

Finally, the generator is trained with the following loss:

$$\mathcal{L} = \mathcal{L}_{rec} + \phi_G \mathcal{L}_V^G \tag{9}$$

where $\phi_G = 0.1$ is the GAN loss weight.

4.2. depth2stereo

The depth2stereo algorithm [14] is a row-wise order-invariant algorithm that uses the premise that objects popping out of the image as observed in the depth map are farther apart from each other in comparison to objects sinking into the image. The algorithm operates in two steps: the tear identification step followed by the tear interpolation step.

In the tear identification step, the scanline is searched for instances of pixel ranges in which tearing (shifting when generating the right stereo counter-part) may occur. In general, tears will occur along regions where depth map intensities increase over the entire pixel range and this step is implemented to identify ranges of pixels along the scanline where the intensities strictly decrease from right to left. The shifted result is then obtained using a standard shift where all torn pixels are assigned an intensity of 0.

This is followed by the tear interpolation step, where the shifted image is corrected for areas where a tear has left an unintended blank pixel. Two pointers: a left pointer pointing to the pixels in the original image and a right pointer pointing to the shifted image traverse across the scanline from left to right. A tear is identified when the pixel values do not match and if this is the case, the empty pixel is filled with the pixel value from the left pointer i.e. the corresponding pixel from the original image. In this way, the depth2stereo algorithm is able to generate a shifted right stereo counterpart to an inputted left image and its corresponding depth map.

4.3. StereoDepthGAN

Pilzer *et al*. [15] target at estimating a disparity map given a pair of stereo images through the StereoDepthGAN, which performs an unsupervised adversarial depth estimation using cycled generative networks. Similar to the StereoGAN, the generator G of the StereoDepthGAN uses the warping function f_w [21] is used to generate or reconstruct the left $\hat{\boldsymbol{I}^L}$ and right $\hat{\boldsymbol{I}^R}$ stereo image pair using the estimated left-to-right d^R and right-to-left d^L disparity maps given a left $\boldsymbol{I}^L$ and right $\boldsymbol{I}^R$ stereo image pair as inputs. Here, the generator network G consists of two generative sub-networks G_l and G_r that exploit the same convolutional encoder-decoder architecture detailed by Pilzer *et al*. [15]. G_l is used to produce two distinct left-to-right disparity maps d_l^R and d_r^R given the left $\boldsymbol{I}^L$ and right $\boldsymbol{I}^R$ stereo input images, respectively.

$$d_l^R = G_l(\boldsymbol{I}^L) \quad \text{and} \quad d_r^R = G_l(\boldsymbol{I}^R) \tag{10}$$

The two disparity maps d_l^R and d_r^R are concatenated and passed through a 1×1 convolution layer to obtain an enhanced disparity map d^R, using which, the synthesized right image $\hat{\boldsymbol{I}^R}$ is generated using the warping function f_w. Moving forward, in order to establish the closed loop structure of cycled generative networks, the generative sub-network G_r is used to produce two distinct right-to-right disparity maps d_l^L and d_r^L given the original left $\boldsymbol{I}^L$ and synthesized

right $\hat{\boldsymbol{I}}^R$ image, respectively.

$$d_l^L = G_r(\boldsymbol{I}^L) \quad \text{and} \quad d_r^L = G_r(\hat{\boldsymbol{I}}^R) \tag{11}$$

Similarly, d_l^L and d_r^L are used to obtain an enhanced disparity map d^L, using which $\hat{\boldsymbol{I}^L}$ is generated. Unlike the DepthGAN, the StereoDepthGAN merely the L1 loss for the generator image reconstruction loss:

$$\mathcal{L}_{rec} = ||\boldsymbol{I}^R - f_w(d^R, \boldsymbol{I}^L)|| + ||\boldsymbol{I}^L - f_w(d^L, \hat{\boldsymbol{I}^R})|| \tag{12}$$

Pilzer *et al*. [15] also apply an L1 consistence loss to constrain the generated depth maps d^L and d^R on each other:

$$\mathcal{L}_{con} = ||d^L - f_w(d^L, d^R)|| \tag{13}$$

The StereoDepthGAN discriminator D, similar to the generator D, consists of two sub-networks D_l and D_r. D_l is used to discern between the original $\boldsymbol{I}^L$ and the synthesized $\hat{\boldsymbol{I}^L}$ left image while D_r is used to discern between the original $\boldsymbol{I}^R$ and the synthesized $\hat{\boldsymbol{I}^R}$ right images as given by the adversarial objective:

$$\begin{aligned} \mathcal{L}_{gan} = \mathbb{E}_{\boldsymbol{I}^L \sim p(\boldsymbol{I}^L)}[\log D_l(\boldsymbol{I}^L)] + \\ \mathbb{E}_{\boldsymbol{I}^R \sim p(\boldsymbol{I}^R)}[\log(1 - D_l(f_w(d^L, \hat{\boldsymbol{I}^R})))] + \\ \mathbb{E}_{\boldsymbol{I}^R \sim p(\boldsymbol{I}^R)}[\log D_r(\boldsymbol{I}^R)] + \\ \mathbb{E}_{\boldsymbol{I}^L \sim p(\boldsymbol{I}^L)}[\log(1 - D_r(f_w(d^R, \boldsymbol{I}^L)))] \end{aligned} \tag{14}$$

Finally, the full optimization objective is given by:

$$\mathcal{L} = \gamma_{rec}\mathcal{L}_{rec} + \gamma_{gan}\mathcal{L}_{gan} + \gamma_{con}\mathcal{L}_{con} \tag{15}$$

where the γ's weight the influence of each loss component.

5. Experiments

5.1. Dataset

NYU Depth V2 dataset [17] provides RGB images and corresponding depth maps for different indoor scenes captured at a resolution of 640×480 using Microsoft Kinect. The training dataset contains 120K samples. We train our method on a 50K subset and evaluate model on the official test split containing 654 samples. Missing depth values are filled using the inpainting method by Levin *et al*. [12]. The depth maps have an upper bound of 10 meters. All experiments and comparative analysis of our GAN based depth estimation pipeline with other baseline models are performed on the NYU Depth V2 dataset.

Figure 4. Sample example from the NYU depth V2 dataset

5.2. Baselines

DenseDepth [1] is a convolutional neural network for computing a high-resolution depth map given a single RGB image with the help of transfer learning. Following a standard encoder-decoder architecture, features extracted are leveraged using high performing pre-trained networks when initializing encoder along with augmentation and training strategies.

MonoDepth2 [7] uses a combination of appearance matching loss to address the problem of occluded pixels that occur when using monocular supervision, a simple automasking approach to ignore pixels where no relative camera motion is observed in monocular training, and a multi-scale appearance matching loss that performs all image sampling at the input resolution.

SGDepth [10] is a self-supervised depth estimation method to deal with moving objects. It uses cross-domain training of semantic segmentation and depth estimation with task-specific network heads, together with a semantic masking scheme to prevent moving objects from contaminating the photometric loss, and a detection method for frames with non-moving objects.

MiDaS [16] is based on novel loss functions that are invariant to the major sources of incompatibility between datasets. Implementation is based on optimized strategies for mixing datasets during training, using high-capacity encoders.

5.3. Experimental Setup and Fine-Tuning

Pre-trained models of the DepthGAN and StereoDepthGAN, whose implementations are available in their respective papers, were used for the single-image stereo depth estimation pipeline. For both implementations, hyperparameter tuning was done using an exhaustive grid search [2]. However, we found that the default set of hyperparameters that were empirically determined by Groenendijk *et al*. [9] for the DepthGAN and by Pilzer *et al*. [15] for the StereoDepthGAN work best and thus, these were the sets of hyperparameters that were used for fine-tuning each of the respective models.

We leveraged transfer learning using these pre-trained models, which were both pre-trained on the KITTI [5] and the Cityscapes [4] datasets in order to further fine-tune the models on the train set of the NYU Depth V2 dataset. The

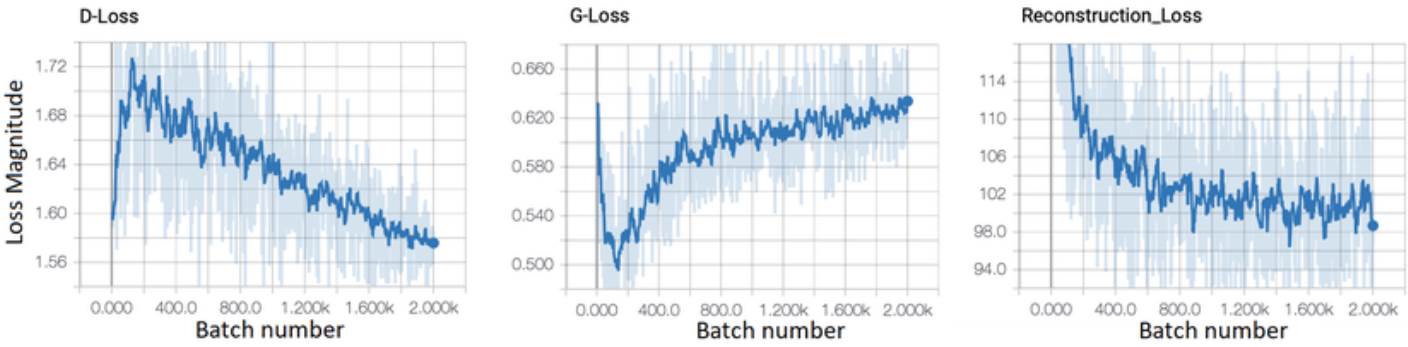

Figure 2. DepthGAN fine-tuning losses

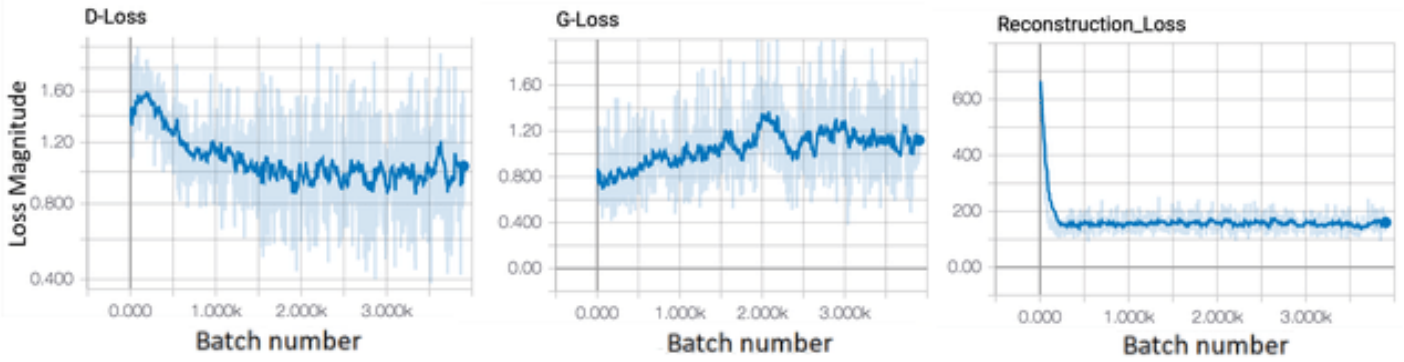

Figure 3. StereoDepthGAN fine-tuning losses

DepthGAN and StereoDepthGAN generators and discriminators were fine-tuned locally on a GeForce GTX 960M GPU with an initial learning rate of 0.1 using the Adam and SGD optimizers, respectively. The fine-tuning discriminator loss (D-Loss), generator loss (G-Loss), and image reconstruction loss of the DepthGAN and the StereoDepthGAN are given in Fig. 2 and Fig. 3, respectively.

6. Results

In order to justify using the depth2stereo algorithm and using the generated stereo counter-part to generate the final depth map, we report the qualitative and quantitative results for both the plausible depth map (output of DepthGAN) and the final depth map (output of StereoDepthGAN). Furthermore, we also report the results of the plausible and final depth maps before and after fine-tuning in order to observe the performance gain achieved as a result of in-domain fine-tuning on the NYU Depth V2 dataset.

6.1. Qualitative Evaluation

The error maps (i.e. pixel-wise differences) for some example images from the NYU Depth V2 test set between the ground truth and generated depth maps for our pipeline and other state-of-the-art monocular depth estimation baselines for qualitative evaluation are given in Fig.5.

Here, although the differences are subtle, we observe that the final depth maps generated by the pipeline after fine-tuning are closer to the ground truth depth maps and smoother near edges in comparison to the depth maps generated by the baselines. This can be attributed to the various image reconstruction losses that are enforced by the DepthGAN and StereoDepthGAN.

6.2. Quantitative Evaluation

We measure the performance of our pipeline and other state-of-the-art monocular depth estimation baselines as measured by a set of commonly used depth estimation metrics [9] on the NYU Depth V2 test set: Absolute Relative Distance (ARD), Squared Relative Distance (SRD), Root Mean Squared Error (RMSE), and log Root Mean Squared Error (log RMSE). Lower values are better for all of these metrics and the results are given in Table. 1.

Here, we observe that while some baselines outperform both the plausible and final depth map generated by the pipeline before fine-tuning, the final depth map generated by the pipeline after fine-tuning outperform all the state-of-the-art monocular depth estimation baselines across all the metrics considered on the NYU Depth V2 test set.

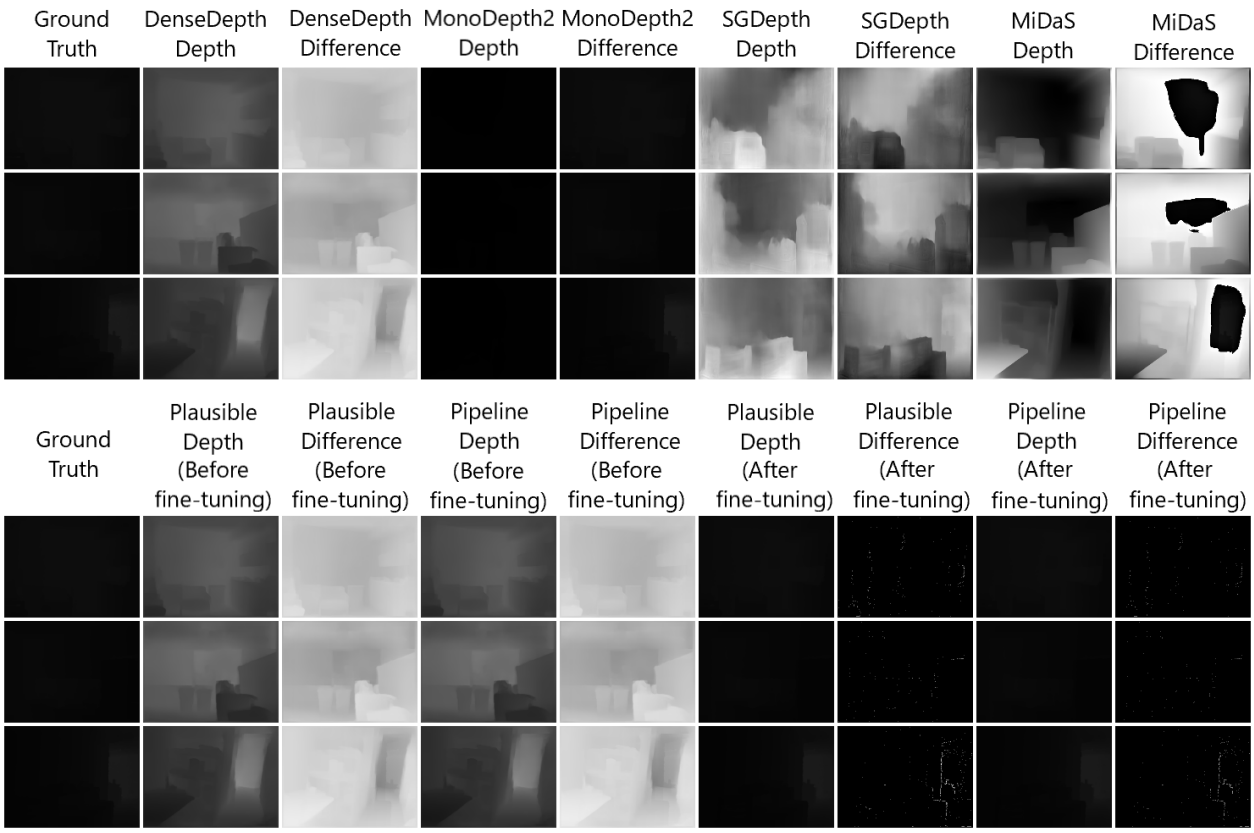

Figure 5. Qualitative evaluation of our pipeline and baselines

Method	ARD	SRD	RMSE	log RMSE
DenseDepth [1]	0.74	0.56	0.74	1.48
MonoDepth2 [7]	0.534	0.381	0.612	1.521
SGDepth [10]	0.550	0.345	0.584	1.027
MiDaS [16]	0.024	0.015	0.027	0.057
Ours - Plausible	0.792	0.646	0.802	2.560
Ours - Final	0.745	0.567	0.751	1.499
Ours - Plausible (After fine-tuning)	0.025	0.002	0.027	0.056
Ours - Final (After fine-tuning)	**0.019**	**0.013**	**0.022**	**0.046**

Table 1. Quantitative evaluation of our pipeline and baselines

7. Conclusion

Through the extensive experiments conducted, we observe that indeed, stereo depth estimation methods are superior to monocular depth estimation methods. The pipeline leverages the generative power of GANs to reinforce image reconstruction losses that lead to estimation of better depth maps in comparison to the baselines considered, both in the qualitative and quantitative sense. Another advantage of the single-image stereo depth estimation pipeline is that it does not require stereo image pairs for training. Finally, the pipeline is also modular, where the DepthGAN and StereoDepthGAN and be swapped out for other monocular and stereo depth estimation methods, respectively.

However, the pipeline suffers from two main disadvantages. It makes use of Deep Learning based depth estimation methods in a sequential fashion, which makes depth estimation computationally expensive using the pipeline, and thus, it is not suitable for low-power mobile devices with limited computational power. Furthermore, since the pipeline uses a three-step sequential process that cannot be parallelized, it suffers from high inference times in comparison to other single-step Deep Learning based depth estimation methods, which makes it infeasible for use in scenarios where latency and fast inference is critical.

8. Future Work

Some of the future work that can be worked on in order to improve the performance of the pipeline are to investigate other components of the pipeline. Training GANs is computationally expensive and slow and replacing them with other Deep Learning based methods could lead to improved fine-tuning, especially in scenarios such as online learning, where adapting to new domains faster is useful. Furthermore, other replacements for the depth2stereo algorithm can be investigated. In the current state of the pipeline, the DepthGAN and the StereoDepthGAN have to be trained individually. If a differentiable implementation of the depth2stereo algorithm can be developed, then the entire pipeline can be trained as a whole, where the gradients can flow through the entire pipeline during training, thus simplifying the training or fine-tuning process.

References

[1] I. Alhashim and P. Wonka. High quality monocular depth estimation via transfer learning. *arXiv preprint arXiv:1812.11941*, 2018.

[2] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for hyper-parameter optimization. In *25th annual conference on neural information processing systems (NIPS 2011)*, volume 24. Neural Information Processing Systems Foundation, 2011.

[3] A. Bhoi. Monocular depth estimation: A survey. *arXiv preprint arXiv:1901.09402*, 2019.

[4] 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*, pages 3213–3223, 2016.

[5] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[6] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 270–279, 2017.

[7] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3828–3838, 2019.

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

[9] R. Groenendijk, S. Karaoglu, T. Gevers, and T. Mensink. On the benefit of adversarial training for monocular depth estimation. *Computer Vision and Image Understanding*, 190:102848, 2020.

[10] M. Klingner, J.-A. Termöhlen, J. Mikolajczyk, and T. Fingscheidt. Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance. In *European Conference on Computer Vision*, pages 582–600. Springer, 2020.

[11] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)*, pages 239–248. IEEE, 2016.

[12] A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. In *ACM SIGGRAPH 2004 Papers*, pages 689–694. 2004.

[13] C. Liu, J. Yang, D. Ceylan, E. Yumer, and Y. Furukawa. Planenet: Piece-wise planar reconstruction from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2579–2588, 2018.

[14] D. Marchese. depth2stereo. `https://github.com/marchese29/depth2stereo`, 2016.

[15] A. Pilzer, D. Xu, M. Puscas, E. Ricci, and N. Sebe. Unsupervised adversarial depth estimation using cycled generative networks. In *2018 International Conference on 3D Vision (3DV)*, pages 587–595. IEEE, 2018.

[16] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *arXiv preprint arXiv:1907.01341*, 2019.

[17] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012.

[18] N. Smolyanskiy, A. Kamenev, and S. Birchfield. On the importance of stereo for accurate depth estimation: An efficient semi-supervised deep neural network approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1007–1015, 2018.

[19] F. Tosi, F. Aleotti, M. Poggi, and S. Mattoccia. Learning monocular depth estimation infusing traditional stereo knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9799–9809, 2019.

[20] J. Watson, O. Mac Aodha, D. Turmukhambetov, G. J. Brostow, and M. Firman. Learning stereo from single images. In *European Conference on Computer Vision*, pages 722–740. Springer, 2020.

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

[22] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858, 2017.