UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

On the benefit of adversarial training for monocular depth estimation

Groenendijk, R.; Karaoglu, S.; Gevers, T.; Mensink, T.

DOI
10.1016/j.cviu.2019.102848

Publication date
2020

Document Version
Final published version

Published in
Computer Vision and Image Understanding

License
CC BY-NC-ND

Link to publication

Citation for published version (APA):
Groenendijk, R., Karaoglu, S., Gevers, T., & Mensink, T. (2020). On the benefit of adversarial training for monocular depth estimation. *Computer Vision and Image Understanding*, *190*, Article 102848. https://doi.org/10.1016/j.cviu.2019.102848

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:12 Nov 2026

Computer Vision and Image Understanding 190 (2020) 102848

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

On the benefit of adversarial training for monocular depth estimation ☆

Rick Groenendijk [a,*], Sezer Karaoglu [b], Theo Gevers [a,b], Thomas Mensink [a,c]

[a] University of Amsterdam, Science Park 904, 1098XH Amsterdam, The Netherlands
[b] 3DUniversum, Science Park 400, 1098 XH Amsterdam, The Netherlands
[c] Google Research, Claude Debussylaan 34, 1082 MD Amsterdam, The Netherlands

ARTICLE INFO

Communicated by: Nikos Paragios

MSC:
68T45

Keywords:
Monocular depth estimation
Adversarial training
GAN

ABSTRACT

In this paper we address the benefit of adding adversarial training to the task of monocular depth estimation. A model can be trained in a self-supervised setting on stereo pairs of images, where depth (disparities) are an intermediate result in a *right-to-left* image reconstruction pipeline. For the quality of the image reconstruction and disparity prediction, a combination of different losses is used, including L1 image reconstruction losses and left–right disparity smoothness. These are local pixel-wise losses, while depth prediction requires global consistency. Therefore, we extend the self-supervised network to become a Generative Adversarial Network (GAN), by including a discriminator which should tell apart reconstructed (fake) images from real images. We evaluate Vanilla GANs, LSGANs and Wasserstein GANs in combination with different pixel-wise reconstruction losses. Based on extensive experimental evaluation, we conclude that adversarial training is beneficial *if and only if* the reconstruction loss is not too constrained. Even though adversarial training seems promising because it promotes global consistency, non-adversarial training outperforms (or is on par with) any method trained with a GAN when a constrained reconstruction loss is used in combination with batch normalisation. Based on the insights of our experimental evaluation we obtain state-of-the art monocular depth estimation results by using batch normalisation and different output scales.

1. Introduction

We are interested in estimating depth from single images. This is fundamentally an ill-posed problem, since a single 2D view of a scene can be explained by many 3D scenes, among others due to scale ambiguity. Therefore the corresponding 3D scene should be estimated – *implicitly* – by looking at the global scene context. Using the global context, a model prior can be estimated to reliably retrieve depth from a single image.

To take into account global scene context for single image depth estimation, elaborate image recognition models have been developed (Eigen et al., 2014; Godard et al., 2017; Saxena et al., 2006). Currently used deep convolutional networks (ConvNets) have enough capacity to understand the global relations between pixels in the image as well as to encode the prior information. However, they are only trained with combinations of per-pixelwise losses.

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014; Isola et al., 2017) force generated images to be realistic and globally coherent. They do so by introducing an adaptable loss, in the form of a neural discriminator network that penalises generated output that looks different from real data. GANs have seen much research attention in recent years, with ever increasing quality of the generated data (e.g. Karras et al. (2019)). Moreover, they have shown success in many tasks, including image reconstruction (Isola et al., 2017), image segmentation (Ghafoorian et al., 2018), novel viewpoint estimation (Galama and Mensink, 2018) and binocular depth estimation (Pilzer et al., 2018). In this paper, we add an adversarial discriminator network to an existing monocular depth estimation model to include a loss based on global context.

Training deep ConvNets requires large datasets with corresponding depth data, preferably dense depth data where ground truth depth is available per pixel. These are not always easy to obtain, either due to a complicated acquisition setup or due to sparse depth data of LiDaR sweeps. To circumvent this, depth prediction has recently been formulated as a novel viewpoint generation task from stereo imagery, where depth is never directly estimated, but just the valuable intermediate in an image reconstruction pipeline. Godard et al. (2017) start from a stereo pair of images, and use the *left* image to estimate the disparities from the *right-to-the-left* image, which is combined with the *right* image to reconstruct the *left* image (see Fig. 1). Due to the constrained image reconstruction setting, disparities are learned from

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.cviu.2019.102848.

* Corresponding author.
E-mail address: r.w.groenendijk@uva.nl (R. Groenendijk).

https://doi.org/10.1016/j.cviu.2019.102848
Received 24 July 2019; Received in revised form 9 October 2019; Accepted 9 October 2019
Available online 18 October 2019
1077-3142/© 2019 Elsevier Inc. All rights reserved.

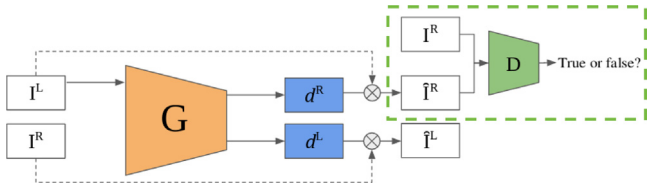

Fig. 1. Illustration of the depth prediction architecture. We use the baseline architecture of Godard et al. (2017), and extend this by a discriminator (in green) to enable adversarial training. A single input image into the network results in both *left-to-right* and *right-to-left* disparities, which are used to reconstruct both left and right images. The GAN extension, uses the same generator, however a discriminator network D is added to enforce adversarial loss on the generated data of the generator.

a single view. We use this model as our baseline and extend it with a discriminator network, which we train in an adversarial setup.

Previous literature (Godard et al., 2017; Yang et al., 2018; Luo et al., 2016; Mayer et al., 2016) shows impressive performance of depth estimation using mostly engineered photometric and geometric losses. However, most of these loss functions are defined as a sum over per-pixel loss functions. The global consistency of the scene context is not taken into account in the loss formulation. We address the question: *"To what extent can monocular depth estimation benefit from adversarial training?"* We do so by studying the influence of adversarial learning on several combinations of photometric and geometric loss functions.

2. Related work

Monocular depth estimation. Estimating depth from single images is fundamentally different from estimating depth from stereo pairs, because it is no longer possible to triangulate points. In a monocular setting, contextual information is required, *e.g.* texture variation, occlusions, and scene context. These cannot reliably be detected from local image patches. For example, a patch of blue pixels could either represent distant sky, or a nearby blue coloured object. The global picture thus has to be considered. Global context information can for example be modelled by using manually engineered features (Saxena et al., 2009), or by using CNNs (Eigen et al., 2014).

Depth ground truth is expensive and time-consuming to obtain, and the readings might be inaccurate, *e.g.* due to infrared interference or due to surface specularities. An alternative is to use self-supervised depth estimation (Garg et al., 2016; Godard et al., 2017, 2018), where training data consists of pairs of *left* and *right* images. A disparity prediction model can be trained, to warp the *left* image into the *right* image, using photometric reconstruction losses. Depth can be recovered from the disparities, by using the camera intrinsics, making depth ground truth data unnecessary at train time. While stereo pairs are necessary during training, during test time depth can be predicted from a single image. In this work we use the work of Godard et al. (2017) as a baseline. Their follow-up work is also concerned with depth estimation, yet based on temporal sequences of (monocular) frames (Godard et al., 2018). The scope of our paper is on stereoscopic learning of depth.

GANs for image generation. Depth estimation from single images can be formulated as an image warping or image generation task. Often image generation is done by means of encoder–decoder networks that output newly generated images. Encoder–decoder networks trained using L1 or L2 produce blurry results (Pathak et al., 2016; Zhao et al., 2017), since output pixels are penalised conditioned on their respective input pixels, but never on the joint configuration of the output pixels. GANs (Goodfellow et al., 2014; Mirza and Osindero, 2014; Isola et al., 2017) counteract this by introducing a structured high-level adaptable loss. GANs are used in tasks where generating globally coherent images is important. Since monocular depth estimation is largely dependent on how well global contextual information is extracted from the input view, there is reason to believe GANs can be of benefit.

Pairing the high-level adversarial loss with a low-level reconstruction loss such as L1 may boost performance even more (Isola et al., 2017). This may be due to the fact that the adversarial loss punishes high-level detail, but only slowly updates low-level detail. Combining adversarial losses and pixel-based local losses has been shown to work well for a number of tasks, including novel viewpoint estimation (Huang et al., 2017; Wu et al., 2016; Galama and Mensink, 2018), predicting future frames in a video (Yin et al., 2018; Mathieu et al., 2015), and image inpainting (Pathak et al., 2016).

GANs for depth estimation. Adversarial losses have already been explored for depth estimation. Chen et al. (2018) shows that adversarial training can be beneficial when directly regressing on depth from single images using ground-truth depth data. The authors use a CNN and a CNN-CRF architecture using either L1 or L2 norm as similarity metrics for the predicted depths. Since they only ever use single images during training, they do not exploit scene geometry for more involved geometric losses. Kumar et al. (2018) predict depth maps from monocular video sequences and successfully use an adversarial network to promote photo-realism between frames. Their generator is composed of two separate networks: A depth network and a pose network. Together these networks enable the authors to generate frames over time. Compared to the current work the problem is less constrained, because the static scene assumption is violated. That is, objects in the scene may themselves move between frames. Pilzer et al. (2018) suggest to use a cycled architecture for estimating depth, in which two generators and two discriminators jointly learn to estimate depth. Their half-cycle architecture is close to our approach, since it uses a single discriminator. However, the generator requires the input of both left and right images to predict a disparity map and even then does not explicitly enforce consistency between the two images. Concurrently with our research is the work of Aleotti et al. (2018), where the method of Godard et al. (2017) is extended with a vanilla GAN. They address the weighting of loss components and find that a subtle adversarial loss can possibly yield improved performance, albeit marginally. In our experiments we find the opposite, none of the used GAN variants improve performance, and we show that small variations of performance could also be explained by initialisation or attributed to the use of batch normalisation. Unlike the methods above the purpose of the current work is to evaluate adversarial approaches when constrained reconstruction losses are used.

We evaluate different GAN objectives for depth estimation, in part based on the results of Lucic et al. (2018), who conclude that no variant is (necessarily) better than others, given a sufficiently large computational budget and extensive hyper-parameter search. We compare the following GAN variants:

1. Vanilla GAN, with a PatchGAN (Isola et al., 2017) discriminator;
2. Gradient-Penalty Wasserstein GANs (WGAN-GP) (Arjovsky et al., 2017; Gulrajani et al., 2017), which minimises the Wasserstein distance, to overcome the saturating loss of the original GAN formulation;
3. Least Square GANs (LSGAN) (Mao et al., 2017), where the sigmoid *real or fake* prediction is replaced by an L2-loss.

While GANs provide a powerful method to output realistic data with an adaptable loss, they are notoriously difficult to train stably (Salimans et al., 2016). Many strategies to improve training stability of GANs have been proposed, including using feature matching (Ghafoorian et al., 2018), using historical averaging (Shrivastava et al., 2017), or adding batch normalisation (Ioffe and Szegedy, 2015). We include the latter in our GAN variants; initial results have shown that batch normalisation is beneficial for vanilla GANs and LSGANs to counteract internal covariate shift.

3. Method

The baseline of our single depth estimation model is the reconstruction-based architecture for depth estimation from Godard et al. (2017), which we describe first. Then, we extend this baseline with an adversarial discriminator network.

Depth from image reconstruction. The problem of estimating depth from single images could be formulated as an image reconstruction task, like in Garg et al. (2016), where a generator network takes in a single left view image and outputs the left-to-right disparity. The right image is reconstructed from the left input image and the predicted disparity. As a consequence this network could be trained from rectified left and right image pairs, without requiring ground-truth depth-maps. At test time depth estimation is based on the disparity predicted from the (single) left image, see Fig. 1.

Godard et al. (2017) improve on the work of Garg et al. (2016) by using both right-to-left and left-to-right disparities and by using a bilinear sampler (Jaderberg et al., 2015) to generate images, which makes the objective fully differentiable.

We first describe in detail the method of Godard et al. (2017), our baseline: The generator G uses the left image of the pair to reconstruct both the left and right images. Consider a left image $\mathbf{I}^L$ and a right image $\mathbf{I}^R$. Image $\mathbf{I}^L$ will be used as the sole input to the generator G. The generator G, however, outputs two disparities d^L and d^R. Using left-to-right disparity d^R the right image $\hat{\mathbf{I}}^R$ can be reconstructed, using warping method f_w:

$$\hat{\mathbf{I}}^R = f_w(d^R, \mathbf{I}^L) \tag{1}$$

And similarly, the reconstructed left image $\hat{\mathbf{I}}^L$, is obtained by:

$$\hat{\mathbf{I}}^L = f_w(d^L, \mathbf{I}^R) \tag{2}$$

A good generator G should predict d^L, d^R such that the reconstructed images ($\hat{\mathbf{I}}^L$ and $\hat{\mathbf{I}}^R$) are close to the original image pair ($\mathbf{I}^L$ and $\mathbf{I}^L$). To measure this several image reconstruction losses are used.

Image reconstruction losses. The quality of the reconstruction is based on multiple loss components, each with different properties for the total optimisation process. We combine:

1. L1 loss to minimise the absolute per-pixel distance:

$$\mathcal{L}^l_{L1} = \frac{1}{\mathrm{N}} \sum_{i,j} \|\mathrm{I}^l_{ij} - \hat{\mathrm{I}}^l_{ij}\|. \tag{3}$$

   Note that L1 has been reported to outperform L2 (Zhao et al., 2017).

2. Structural similarity (SSIM) reconstruction loss to measure the perceived quality (Wang et al., 2004):

$$\mathcal{L}^l_S = \frac{1}{\mathrm{N}} \sum_{i,j} \frac{(1 - \mathrm{SSIM}(\mathrm{I}^l_{ij}, \hat{\mathrm{I}}^l_{ij}))}{2} \tag{4}$$

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{5}$$

   where illumination μ and signal contrast σ are computed around centre pixels x and y, $c_1 = 0.01^2$ and $c_2 = 0.03^2$ overcome divisions by (almost) zeros, these values are set in line with Godard et al. (2017) and Yang et al. (2018).

3. Left–Right Consistency Loss (LR), which enforces the consistency between the predicted *left-to-right* and *right-to-left* disparity maps:

$$\mathcal{L}^l_{LR} = \frac{1}{\mathrm{N}} \sum_{i,j} |d^l_{ij} - d^r_{i(j+d^l_{ij})}| \tag{6}$$

4. Disparity Smoothness Loss, which forces smooth disparities, *i.e.* small disparity gradients, unless there is an edge, therefore using an edge-aware L1 smoothness loss:

$$\mathcal{L}^l_{disp} = \frac{1}{\mathrm{N}} \sum_{i,j} |\partial_x d^l_{ij}| \ \exp(-\|\partial_x \mathrm{I}^l_{ij}\|) + |\partial_y d^l_{ij}| \ \exp(-\|\partial_y \mathrm{I}^l_{ij}\|), \tag{7}$$

   since the generator outputs disparities at different scales, this loss is normalised by $\frac{1}{2^s}$ at scale s to normalise the output ranges.

In Yang et al. (2018) an occlusion loss component was suggested in addition to the other loss terms. Initial experimentation shows no clear benefit of using it for our set-up. Results with the occlusion loss can be found in the supplementary material.

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

$$\mathcal{L}_{rec} = \sum_{s=0}^{3} \mathcal{L}_s \tag{8}$$

$$\mathcal{L}_s = \gamma_{L1} \ \mathcal{L}_{L1} + \gamma_S \ \mathcal{L}_S + \gamma_{lr} \ \mathcal{L}_{lr} + \gamma_{disp} \ \frac{1}{2^s} \mathcal{L}_{disp}, \tag{9}$$

where γ weighs the influence of each loss component. For each loss component we use the reconstruction of the *left and right* image, which are defined symmetrically.

Adversarial training for single image depth estimation. We extend the baseline model by a single discriminator network. The discriminator network is tasked with discerning between fake and real images on the right side. The schematic is shown in Fig. 1, by the green discriminator.

For adversarial training, we combine the reconstruction loss $\mathcal{L}_{rec}$ with the loss functions belonging to specific GAN variants. Note that unlike the original formulation of GANs actual data, in the form of the left image $\mathbf{I}^L$, is fed to the generator, not noise z (Mirza and Osindero, 2014). The generator G produces two disparities d^L and d^R. However the discriminator D is only presented with the right image $\hat{\mathbf{I}}^R = f_w(d^R, \mathbf{I}^L) = f_w(G(\mathbf{I}^L), \mathbf{I}^L)$ constructed from d^R. Thus the discriminator D examines only (reconstructed) right images $\mathbf{I}^R$ and $\hat{\mathbf{I}}^R$, to tell apart. This leads to the following losses:

1. Vanilla GAN:

$$\mathcal{L}^G_{\mathrm{V}} = -\mathbb{E}\left[\log D(\hat{\mathbf{I}}^R)\right], \tag{10}$$

$$\mathcal{L}^D_{\mathrm{V}} = \mathbb{E}\left[\log D(\mathbf{I}^R) + \log(1 - D(\hat{\mathbf{I}}^R))\right]. \tag{11}$$

2. LS-GAN:

$$\mathcal{L}^G_{\mathrm{LS}} = \tfrac{1}{2}\mathbb{E}\left[(D(\hat{\mathbf{I}}^R) - 1)^2\right] \tag{12}$$

$$\mathcal{L}^D_{\mathrm{LS}} = \tfrac{1}{2}\mathbb{E}\left[(D(\mathbf{I}^R) - 1)^2 + D(\hat{\mathbf{I}}^R)^2\right]. \tag{13}$$

3. WGAN:

$$\mathcal{L}^G_{\mathrm{W}} = \mathbb{E}\left[D(\hat{\mathbf{I}}^R)\right] \tag{14}$$

$$\mathcal{L}^C_{\mathrm{W}} = \mathbb{E}\left[D(\mathbf{I}^R) - D(\hat{\mathbf{I}}^R)\right] + \lambda\Omega_{GP}, \tag{15}$$

   where Ω_{GP} denotes the gradient penalty with $\lambda = 10$ from WGAN-GP (Gulrajani et al., 2017); and where the generator follows the NS-GAN loss (Goodfellow et al., 2014).

The final loss used for training the generator is:

$$\mathcal{L} = \mathcal{L}_{rec} + \phi_G \ \mathcal{L}^G_{\{\mathrm{V,\ LS,\ W}\}}, \tag{16}$$

which combines the reconstruction loss $\mathcal{L}_{rec}$ with the generator part of the GAN loss $\mathcal{L}^G$, where $\phi_G = 0.1$ weighs its influence.

Evaluation at test time. At test time only the generator is used to predict the *right-to-left* disparity d^L at the finest scale, which has the same resolution is the input image. The predicted disparity is transformed into a depth map by using the known camera intrinsic and extrinsic parameters.

Table 1
Evaluation of robustness against initialisation of the networks. Both the baseline model – our implementation of Godard et al. (2017)'s model – and the LSGAN model are trained 10 times. The results indicate that the models are robust against initialisation, albeit some minor variations in the performance remains.

		A	S	R	Rlog	δ^1	δ^2	δ^3
		Lower is better				Higher is better		
Baseline	min	0.141	1.163	5.639	0.236	0.806	0.926	0.967
	max	0.143	1.227	5.732	0.240	0.811	0.929	0.969
	avg	0.142	1.195	5.681	0.238	0.809	0.927	0.968
	std	0.001	0.017	0.027	0.001	0.002	0.001	0.001
LSGAN	min	0.130	1.010	5.359	0.222	0.819	0.936	0.972
	max	0.135	1.053	5.417	0.227	0.823	0.938	0.974
	avg	0.133	1.038	5.388	0.225	0.821	0.937	0.973
	std	0.001	0.014	0.019	0.001	0.001	0.001	0.001

Table 2
Baseline method performance comparison between different generator architecture backbones, VGG and ResNet variants. While the ResNet-50 architecture yields best performance, we use VGG for fair comparison.

	#Params	A	S	R	Rlog	δ^1	δ^2	δ^3
		lower is better				higher is better		
VGG	31.6M	0.142	1.200	5.694	0.239	0.809	0.927	0.967
RN-18	20.2M	0.146	1.260	5.771	0.243	0.801	0.924	0.967
RN-50	43.9M	**0.123**	**0.936**	**5.145**	**0.216**	**0.843**	**0.943**	**0.975**
RN-101	62.9M	0.124	0.971	5.280	0.219	0.840	0.942	0.974

4. Experiments

In this section we experimentally evaluate the proposed GAN models on two public datasets KITTI and CityScapes.

4.1. Setup

Dataset. For the main set of experiments we use the KITTI (Geiger et al., 2013) dataset, which contains image pairs of a car driving in various environments, from highways to city centres to rural roads. We follow the Eigen data split (Eigen et al., 2014) to compare fairly against other methods, which uses 22.6K training images, 888 validation images and 697 test images, which are resized to 256 × 512. During training no-depth ground truth is used, only the available stereo imagery. For evaluation the provided velodyne laser data is used.

To test if the results generalise to another dataset, we use the CityScapes dataset (Cordts et al., 2016). This dataset consists of almost 25 thousand images, with 22.9K training images, 500 validation images and 1525 test images. Visual inspection of the CityScapes dataset reveals two things that have consequences for data pre-processing. First, some images contain artefacts at the top and bottom of the images. Second, both left and right cameras capture part of the car on which they are mounted. To compensate the top 50 and bottom 224 rows of pixels are cropped. Cropping is also performed at the sides of the images to retain width and height ratios.

Implementation details. The generator-only model from Godard et al. (2017) is used as a baseline.[1] For all experiments we use an adapted VGG30 generator network architecture, with ∼31.6M parameters, for fair comparison with other methods (Godard et al., 2017; Pilzer et al., 2018).

All models are trained for 50 epochs, in mini-batches of 8, with the Adam optimiser (Kingma and Ba, 2014) The initial learning rate $\lambda_g = \lambda_d$ is set to 10^{-4}, and updated using a plateau scheduler (Radford et al., 2015). Data augmentation is done in online fashion, including gamma, brightness, and colour shifts and horizontal flipping. In case of the latter left and right images are swapped in order to preserve their relative position. Based on recommendations from previous works and some initial experiments, we set $\gamma_{L1} = 0.15$, $\gamma_S = 0.85$, $\gamma_{lr} = 1.0$, $\gamma_{disp} = 0.1$, $\phi_G = 0.1$.

The discriminators for Vanilla GAN and LSGAN are convolutional network with five layers and for WGAN-GP a 3-layer fully-connected network was used.

Evaluation. At test times disparities are warped into depth maps, and the predicted depth is bounded between 0 and 80 m, which is close to the maximum in the ground truths. Similar to other methods we vertically centre-crop images, see *e.g.* Garg et al. (2016). Ground truth depth data of the Eigen split is sparse. For quantitative evaluation we use a set of common metrics (Eigen et al., 2014; Godard et al., 2017; Pilzer et al., 2018; Yang et al., 2018): Absolute Relative Distance (**ARD**), Squared Relative Distance (**SRD**), Root Mean Squared Error (**RMSE**), log Root Mean Squared Error (**log RMSE**), and accuracy within threshold t (δ_t, with $t \in [1.25, 1.25^2, 1.25^3]$).

Note that the disparity of pixels on the side of the image is ill-defined, the so-called disparity ramps. To compensate we use the test image $\mathbf{I}^L$ and its flipped version $\mathbf{I}^{L,f}$ to obtain disparity maps d^L and $d^{L,f}$, the latter is flipped again to align with d^L. Both have a disparity ramp, yet on opposite sides of the disparity map. Therefore we use d^L, flipped ($d^{L,f}$) for the 5% outer right (left) columns, and average predictions everywhere else.

Initialisation and backbone. In this set of initial experiments we study the robustness of our models to initialisation and the influence of the backbone architecture.

In this first experiment we study the robustness of our models with respect to the initialisation of the weights and the randomness of the training. Therefore, we have trained two of our models 10 times with the same hyper-parameters, namely our baseline model, with a VGG backbone, which combines the 4 reconstruction loss components without batch norm and the LSGAN model with batch norm. The results are shown in Table 1, where we show for each performance measure the minimum, the maximum and the average value, and include the standard deviation. We observe some small differences in performance, *e.g.* ARD in range $0.141 - 0.143$ and $\delta < 1.25$ in the range $0.806 - 0.811$. We conclude that minor differences in performance between models, might in fact be the result of initialisation rather than model design and training choices.

In the second experiment in this section, we study the influence of the backbone network used. We compare the VGG30 network, used in all other experiments, to variants of a ResNet backbone, using the full reconstruction loss definition (*i.e.* L1, LR, disparity & SSIM), without any adversarial components. The results are shown in Table 2. We conclude that the ResNet50 architecture yields best performance on the test set and while the shallow ResNet18 is outperformed by the VGG architecture, it is only by relatively little. This is in the line of expectations, since residual learning has been shown to be effective for deep convolutional architectures (He et al., 2016). For the other experiments, however, we use the VGG30 backbone for fair comparison to related work.

4.2. Loss components & GANs

To address our main research question we have performed an extensive set of experiments using different combinations of loss components paired with three different GAN variants. We compare the performance of the model when using different configurations of loss components: L1 Loss, adding Left-Right consistency, adding disparity smoothness, and adding the SSIM loss. For each of the loss component configurations, we also compare the performance of the model without using a GAN, against using a Vanilla GAN, LSGAN or WGAN-GP. The results are shown in Table 3, we refer to experiment indices in parentheses. From the results we observe the following:

[1] Available at https://github.com/rickgroen/depthgan.

Table 3
Performance of models using different loss configurations and GAN variants. The best results for each loss configuration are indicated by blue highlighting, the overall best results have been boldfaced. Model configuration 4.a is our implementation of Godard et al. (2017). We conclude that with the most constraint image reconstruction loss adversarial training does not improve depth estimation, see text for discussion.

	Loss Components				BN	GAN	ARD	SRD	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
	L1	LR	Disp	SSIM			lower is better				higher is better		
1.a					✓	Vanilla	0.810	12.442	18.245	1.999	0.002	0.008	0.020
1.b						LSGAN	0.893	13.826	18.816	2.468	0.000	0.000	0.000
1.c						WGAN	0.813	12.310	18.119	1.932	0.001	0.003	0.011
2.a	✓				✓	-	0.200	3.149	6.795	0.289	0.760	0.904	0.956
2.b	✓				✓	Vanilla	0.205	3.781	7.045	0.288	0.771	0.911	0.958
2.c	✓				✓	LSGAN	0.190	2.826	6.612	0.281	0.766	0.909	0.959
2.d	✓					WGAN	0.177	2.398	6.504	0.275	0.770	0.905	0.957
3.a	✓	✓			✓	-	0.162	1.755	5.954	0.253	0.789	0.922	0.966
3.b	✓	✓			✓	Vanilla	0.168	2.090	6.104	0.261	0.784	0.919	0.964
3.c	✓	✓			✓	LSGAN	0.160	1.761	5.966	0.253	0.792	0.923	0.966
3.d	✓	✓				WGAN	0.170	1.521	6.121	0.258	0.769	0.909	0.960
4.a	✓	✓	✓	✓		-	0.142	1.200	5.694	0.239	0.809	0.927	0.967
4.b	✓	✓	✓	✓	✓	-	**0.132**	**1.049**	**5.376**	**0.224**	**0.822**	**0.937**	**0.974**
4.c	✓	✓	✓	✓	✓	Vanilla	0.135	1.052	5.428	0.229	0.818	0.935	0.972
4.d	✓	✓	✓	✓	✓	LSGAN	0.135	1.051	5.417	0.227	0.819	0.936	0.972
4.e	✓	✓	✓	✓		WGAN	0.152	1.357	6.003	0.249	0.788	0.917	0.963
5	**Training set mean**						0.361	4.826	8.102	0.377	0.638	0.804	0.894

1. Using only an adversarial loss, without an image reconstruction loss yields imprecise results (1), even worse than using the training set mean (5). The poor performance of the adversarial loss could be explained by the fact that many different disparity maps may reconstruct a correct image. The global coherency loss of the GANs seems to have difficulty converging to locally geometrically viable disparities, we conclude that *both* global and local consistency should be taken into account.
2. All models (with or without GAN) improve when more constraining image reconstruction losses are combined (2, 3, 4); However, where GANs do improve over using just L1 as image reconstruction losses (2.b vs. 2.d/2.e), they do not improve when multiple reconstruction losses are used (4.b vs. 4.c/d/e).
3. Where WGAN is the best (adversarial) model for the L1 loss, it is the worst (adversarial) model when more constrained losses are used. This could be partly explained by the difficulty of training GAN models, which are highly sensitive to parameters settings and network architectures.
4. When considering models with a constrained loss (4), adversarial learning boosts performance beyond the baseline (4.a vs. 4.c/4.d). However, upon closer inspection, the difference in performance is likely due to the use of batch normalisation (*cf.* 4.b vs. 4.c/4.d). Batch normalisation is often used for GANs because it facilitates stable training (Salimans et al., 2016) and it is the default in many open-source GAN implementations.
5. For the baseline model, we obtain 0.142 ARD (4.a) with our implementation, which is slightly better compared to 0.148 reported by Godard et al. (2017) (see also Table 8). Training with batch normalisation yields an increase of performance to 0.132 (4.b).

Batch & instance normalisation. In this experiment we evaluate the influence of normalisation strategies on the performance. We compare the L1 and the full reconstruction loss, trained without GAN or with Vanilla/LSGAN. The results are shown in Table 4. From the results, we observe that for any configuration of loss components, performance improves when batch normalisation is used. While instance normalisation does not yield better performance for a full loss model, it is beneficial for models trained with a L1 loss. We conclude that batch normalisation is important for training for any model. Qualitative investigations in Fig. 5 show that batch normalisation takes away some granularity in the predicted disparities.

Table 4
The effects of different kinds of normalisation on the performance of depth prediction (in ARD) using Vanilla or LSGAN adversarial training. We evaluate no-normalisation (-), instance normalisation (IN), and batch normalisation (BN). We conclude that BN is important for obtaining good results.

GAN	L1			Full		
	–	IN	BN	–	IN	BN
–	0.215	0.208	0.200	0.142	0.144	**0.132**
Vanilla	0.216	**0.183**	0.205	0.143	0.145	0.135
LSGAN	0.216	0.184	0.190	0.143	0.142	0.135

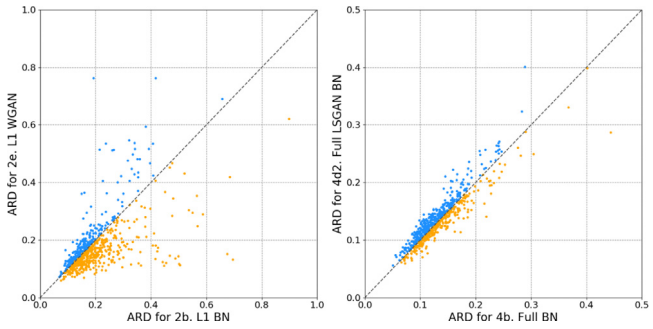

Fig. 2. Error scatter plots of the Absolute Relative Distance (ARD, lower is better) for each image, comparing models with and without GAN loss: (i) the left plot shows a L1 loss model with BN (entry in 3: 2.b) against a L1 loss WGAN (2.e); (ii) right plot shows a full loss model with BN (4.b) against a full component LSGAN with BN (4.d); The diagonal indicates equal performance for both model. E.g. in (i) the blue dots represent those images for which a L1 loss with BN outperforms the L1 loss complemented with WGAN. Note the different scaling between plots. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 3. Illustrative examples to show the quality of reconstruction and disparity maps for different loss configurations. (a) Reconstructed left images $\hat{\mathbf{I}}^L$; (b) Difference between the ground truth and the reconstructed images ($\hat{\mathbf{I}}^L - \mathbf{I}^L$), where red indicates regions that are much lighter in the reconstructed image than in the original image, and blue regions indicate spots that are darker. Red and blue areas indicate wrongly reconstructed areas, which yield incorrect disparity values; and (c) Generated disparity maps d^L. The top three images are among the top performing images for the full loss model, while the bottom three include failure cases nodes, which achieve poor performance. Also not that images are reconstructed at 256 × 512 resolution and then upsampled, such that they are less sharp than the input image at full resolution. Best viewed in colour. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.3. Reconstructed image quality & GANs

In the previous section we quantitatively show how performance of models is affected making use of adversarial losses. GANs optimise for photo-realism in reconstructed images, which could lead to well-reconstructed images while the predicted disparities poorly model accurate depth. This is because many disparity maps may reconstruct an image well even though they do not capture depth correctly. We evaluate the performance of the non-adversarial model versus GANs at the level of individual images to see if there are cases for which GANs are better suited. The results are presented in Fig. 2. When using only the L1 reconstruction loss (*left plane*), the variance in the performance difference between models trained with and without GANs is large (*i.e.* many outliers away from the diagonal). However, when using

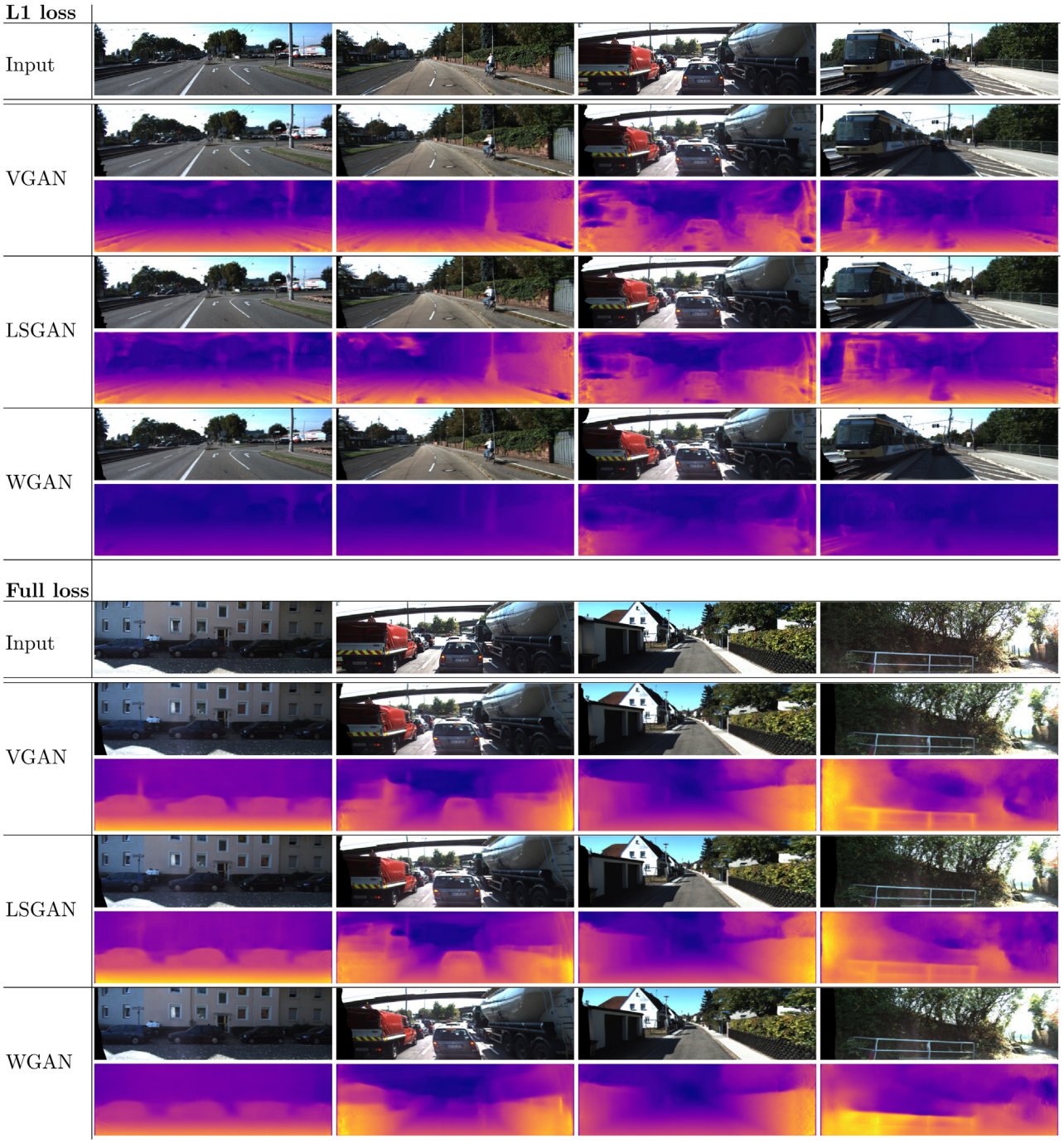

Fig. 4. Comparison of the three different GAN variants: Vanilla GAN, LSGAN and WGAN, compared for both models trained with only L1 loss (top) and models trained with the full loss (bottom). For each of the image blocks, the top row shows input images $\mathbf{I}^L$ to the generator. Then for each GAN variant we show the reconstructed left images $\hat{\mathbf{I}}^L$ and the right-to-left disparities d^L. Not especially that WGAN severely smooths disparity predictions when paired with only a L1 loss. This is effect is much less pronounced when it is paired with our full loss. Compared with the Vanilla GAN and LSGAN, the WGAN recognises objects in the foreground to a lesser extent (see the second column on the left). Best viewed in colour. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the full reconstruction loss, the scatter is aligned around the diagonal, indicating that both models perform similarly on most images. We have verified the few outliers visually, but do not find a noticeable patterns which indicates better photo-realism for GANs in these cases.

In Fig. 3 the reconstructed images and their corresponding disparities are shown for L1, WGAN L1, Full and Full WGAN models. The model trained using only a L1 loss can reconstruct images that have a low pixel-wise loss, but impose no structure on the disparities. This means that there are holes and strong transitions in the predicted disparities. For the WGAN that was trained alongside a L1 loss, it seems the WGANs prefers to predict low disparities. This way images that are very close to the input image are generated. As such, even though the disparities are poor, the reconstructed images are photo-realistic. Adding more geometric loss components that constrain the disparities alleviates these problems.

We also visually compare the performance of different GAN variants in Fig. 4 for models that were trained alongside a L1 loss and a full loss. Again, for models trained with a L1 loss, it can be seen that the WGAN predicts low disparities compared to the other GAN variants. This is beneficial for depth estimation, since it smooths the disparities

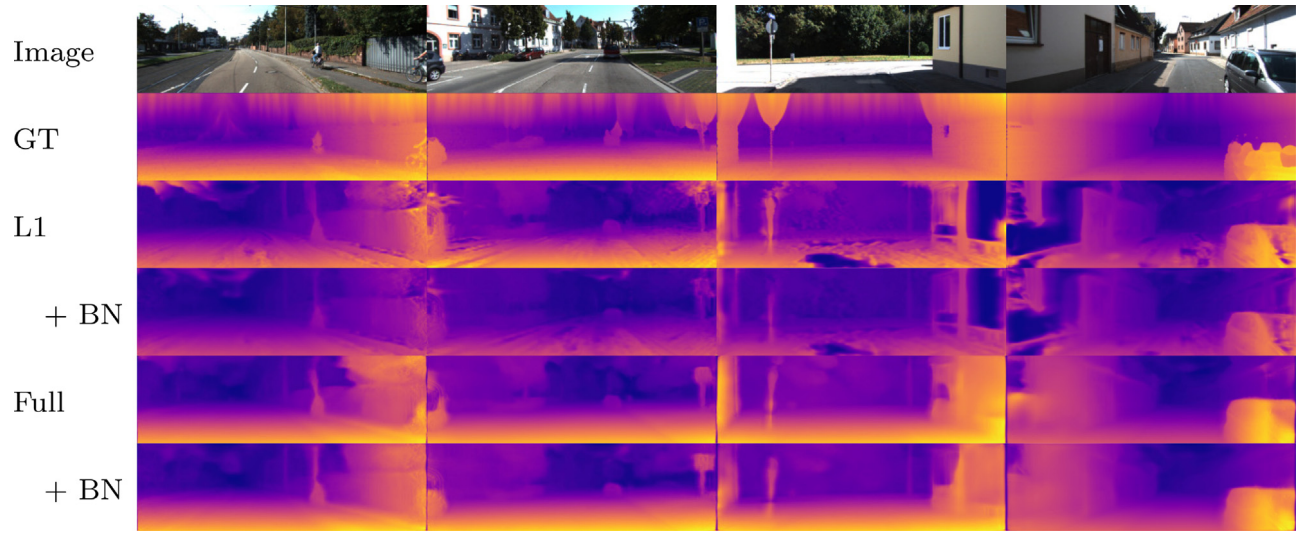

Fig. 5. Illustration of the influence of batch normalisation (BN) on depth prediction. Note the inaccurate values in the GT due to in-painting techniques at the top and bottom. BN results in smoother predictions, while keeping small and distinct objects. Best viewed in colour. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 5
Comparison of image reconstruction quality measured in L1, L2 norms and SSIM. The identity row indicates the losses that would be observed when comparing left and right images. This would happen when the generator learns to predict only zero-valued disparity maps, such that images are not warped using the disparities.

Method	L1	L2	SSIM
	Lower is better		Higher is better
Identity	0.1102	0.0387	0.4757
L1 loss — base	0.0499	0.0102	0.7110
L1 loss — Vanilla GAN	**0.0495**	**0.0101**	**0.7130**
L1 loss — LSGAN	0.0521	0.0112	0.7003
L1 loss — WGAN	0.0525	0.0111	0.6945
Full loss — base	**0.0514**	**0**.0110	**0.7183**
Full loss — Vanilla GAN	0.0521	0.0112	0.7115
Full loss — LSGAN	0.0518	0.0111	0.7125
Full loss — WGAN	0.0550	0.0126	0.6936

Table 6
Performance of depth estimation using different numbers of scales during training, measured in ARD. We conclude that using just 1 or 2 scales suffice for good performance.

Loss Components				BN	GAN	Scales		
L1	LR	Disp	SSIM			4	2	1
✓	✓					0.191	0.190	0.187
✓	✓			✓		0.162	0.145	**0.138**
✓	✓			✓	Vanilla	0.168	0.156	**0.138**
✓	✓				WGAN	0.170	0.154	0.181
✓	✓	✓	✓			0.142	0.137	0.137
✓	✓	✓	✓	✓		*0.132*	**0.128**	*0.131*
✓	✓	✓	✓	✓	Vanilla	0.135	0.136	*0.132*
✓	✓	✓	✓		WGAN	0.152	0.152	0.154

and prevents under-estimating depth. In a way, the WGAN trained alongside the L1 loss imitates the disparity smoothness loss component. This effect is much less pronounced for models trained with a full loss.

In a next experiment we compare the image quality (rather than the depth prediction quality) of the reconstructed images, measured by L1, L2, and SSIM scores on the test set. The results are shown in Table 5. The first row shows the identity mapping, indicating the scores comparing left and right ground truth images directly. In the rest of the table we can see that there are subtle differences between methods in image space. WGAN reconstruction is marginally worse than other methods. The results are interesting, because it seems image reconstruction score has relatively low correlation with depth estimation scores, the image quality of the L1 based models is higher than the full loss, yet the depth prediction is worse. This implies that there is a need for geometrically founded loss functions, like left–right consistency and disparity smoothness losses to obtain better depth predictions.

4.4. Output scales

In this set of experiments, we vary the number of output disparity predictions from 4 (as used in Godard et al. (2017)) to 1 (as used in Pilzer et al. (2018)). We do so for two settings of the reconstruction loss, using only L1 + LR loss, similar to Pilzer et al. (2018) and using the full reconstruction loss, similar to Godard et al. (2017). The results are shown in Table 6, using the ARD performance measure. From the results we observe that reducing disparity output scales contributes positively to depth estimation quality, this holds especially in the case of using just the L1 + LR loss. The intuition is that forcing the network to output coherent disparities early on can over-regularise disparities: With 4 output scales the smallest resolution of disparities is just 32×64, which acts as a (too) strict regulariser.

4.5. Generalising from KITTI to CityScapes

In this set of experiments we evaluate our models on the CityScapes dataset, after training on the KITTI dataset. The goal is to see if the insights and results generalise to this novel domain. Results are depicted in Table 7. The results indicate similar behaviour on the CityScapes dataset as on the KITTI dataset: when the reconstruction losses are sufficiently constrained, adversarial training does not improve the performance. We obtain the best generalising results by using a combination of all four image reconstruction losses, trained at a single scale, using batch normalisation.

Comparison to state-of-the-art. In the final set of experiments, we compare the performance of our current work to a few state-of-the-art methods on KITTI, see Table 8. For comparison, we report performance on the Eigen test set, using centre cropping (Garg et al., 2016). For reference we include a few seminal works on monocular depth estimation, when using the LiDAR data during training (Saxena et al., 2006; Eigen et al., 2014) and one of the newest methods (Yang et al., 2018). Then we compare our performance to the work of Godard et al. (2017, 2018), and Pilzer et al. (2018), since these serve as baselines

Table 7
Evaluation of KITTI trained models on the CityScapes dataset (Cordts et al., 2016).

	Loss Components				S	BN	GAN	ARD	SRD	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
	L1	LR	Disp	SSIM				lower is better				higher is better		
1.a	✓				4			0.409	11.281	20.515	0.569	0.310	0.605	0.781
1.b	✓				4		WGAN	0.329	7.557	19.646	0.512	0.370	0.644	0.803
2.a	✓	✓	✓	✓	1			**0.308**	**6.133**	**18.245**	**0.472**	**0.335**	**0.664**	**0.834**
2.b	✓	✓	✓	✓	1		WGAN	0.321	6.687	18.791	0.491	0.339	0.657	0.822
3.a	✓	✓	✓	✓	4			0.324	6.927	19.309	0.505	0.323	0.642	0.808
3.b	✓	✓	✓	✓	4		LSGAN	0.324	6.889	19.060	0.501	0.322	0.644	0.810
3.c	✓	✓	✓	✓	4	✓	LSGAN	0.310	6.310	18.576	0.479	0.337	0.660	0.826

Table 8
Comparison with state-of-the-art methods, both fully supervised (Yang et al., 2018) and using *left–right correspondence* as supervision (Pilzer et al., 2018; Godard et al., 2017). The improved work of Godard et al. (2018) is also shown, which reports better performance of their version 1 (i.e. Godard et al. (2017)) baseline. The improved version 1 has a ResNet50 backbone and up-samples low-resolution disparity maps before evaluating the loss. We only consider models that have not been pre-trained. Our model outperforms the work of Godard et al. (2017), due to using batch normalisation and better use of disparity prediction scales.

Method		Trained on	ARD	SRD	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
			Lower is better				Higher is better		
Supervised using LiDAR depth									
Saxena et al. (2006)	LiDAR	K	0.280	–	8.734	–	0.601	0.820	0.926
Eigen et al. (2014)	LiDAR	K	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Yang et al. (2018)	LiDAR	K	**0.097**	**0.734**	**4.442**	**0.187**	**0.888**	**0.958**	**0.980**
Supervised using Left-Right Correspondence									
Pilzer et al. (2018)		K	0.152	1.388	6.016	0.247	0.789	0.918	0.965
Godard et al. (2017)	VGG	K	0.148	1.344	5.927	0.247	0.803	0.922	0.964
Godard et al. (2018)	R50, video based, v1	K	0.133	1.142	5.533	0.230	0.830	0.936	0.970
Godard et al. (2018)	R18, v2	K	**0.130**	**1.144**	**5.485**	**0.232**	**0.831**	**0.932**	**0.968**
This paper									
Baseline	VGG, S4	K	0.142	1.200	5.694	0.239	0.809	0.927	0.967
Optimised settings	VGG, BN + S2	K	0.128	1.026	5.313	0.222	0.830	0.939	0.973
ResNet — Baseline	R50, S4	K	0.123	0.936	5.145	0.216	0.843	0.943	**0.975**
ResNet — Optimised settings	R50, BN + S2	K	**0.122**	**0.928**	**5.119**	**0.215**	**0.847**	**0.945**	**0.975**

and inspiration for the current work. While adding adversarial losses to take into account scene context does not improve over a combination of reconstruction based lasses, using batch normalisation and just 2 output scales significantly boost performance.

5. Conclusions

This work has sought to investigate whether using adversarial losses benefits the estimation of depth maps in a monocular setting. For many tasks in which global consistency is important, adversarial training improves image reconstruction tasks. However, after extensive experimental evaluation, we conclude that adversarial training is beneficial in monocular depth estimation *if and only if* the reconstruction loss does not impose too many constraints on reconstructed images. When more involved geometrically or structurally inspired losses are introduced, adversarial training contributes hardly to the quality of the predicted depth maps and may even be harmful.

Based on our extensive experiments we also conclude that:

(i) Batch normalisation improves depth prediction quality significantly; and
(ii) evaluating reconstruction losses at many output scales over-regularises the disparity at the final scale, this effect is stronger when the loss function itself is more constrained by SSIM and left–right consistency components;

Using those two insights, we have been able to set a new state-of-the-art monocular depth prediction based on reconstruction losses, improving from 0.148 (Godard et al., 2017) to 0.128 (using batch norm and 2 output scales).

Future research could investigate the influence of specific architectures of the discriminator network and the use of conditional GANs for guiding monocular depth estimation.

Acknowledgement

This research was supported in part by the Dutch Organisation for Scientific Research, Netherlands via the VENI grant **What & Where** awarded to Dr. Mensink.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cviu.2019.102848.

References

Aleotti, F., Tosi, F., Poggi, M., Mattoccia, S., 2018. Generative adversarial networks for unsupervised monocular depth prediction. In: Proceedings of the European Conference on Computer Vision (ECCV).
Arjovsky, M., Chintala, S., Bottou, L., 2017. Wasserstein gan. arXiv preprint arXiv:1701.07875.
Chen, R., Mahmood, F., Yuille, A.L., Durr, N.J., 2018. Rethinking monocular depth estimation with adversarial training. Technical Report, ArXiV.
Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B., 2016. The cityscapes dataset for semantic urban scene understanding. In: CVPR.
Eigen, D., Puhrsch, C., Fergus, R., 2014. Depth map prediction from a single image using a multi-scale deep network. In: NIPS.
Galama, Y., Mensink, T., 2018. Itergans: Iterative gans for rotating visual objects. In: International Conference on Learning Representations - Workshop (ICLRw).
Garg, R., Vijay Kumar, B.G., Carneiro, G., Reid, I., 2016. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In: ECCV.
Geiger, A., Lenz, P., Stiller, C., Urtasun, R., 2013. Vision meets robotics: The kitti dataset. Int. J. Robot. Res. (IJRR).
Ghafoorian, M., Nugteren, C., Baka, N., Booij, O., Hofmann, M., 2018. El-gan: Embedding loss driven generative adversarial networks for lane detection. arXiv:1806.05525.
Godard, C., Mac Aodha, O., Brostow, G.J., 2017. Unsupervised monocular depth estimation with left-right consistency. In: CVPR.
Godard, C., Mac Aodha, O., Firman, M., Brostow, G., 2018. Digging into self-supervised monocular depth estimation. arXiv preprint arXiv:1806.01260.

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

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C., 2017. Improved training of wasserstein gans. In: NIPS.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778.

Huang, R., Zhang, S., Li, T., He, R., 2017. Beyond face rotation: Global and local perception GAN for photorealistic and identity preserving frontal view synthesis. arXiv:1704.04086.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A., 2017. Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1125–1134.

Jaderberg, M., Simonyan, K., Zisserman, A., et al., 2015. Spatial transformer networks. In: Advances in Neural Information Processing Systems. pp. 2017–2025.

Karras, T., Laine, S., Aila, T., 2019. A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4401–4410.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kumar, A., Bhandarkar, S., Prasad, M., 2018. Monocular depth prediction using generative adversarial networks. In: CVPR Workshop on Deep Learning for Visual SLAM.

Lucic, M., Kurach, K., Michalski, M., Gelly, S., Bousquet, O., 2018. Are GANs created equal? a large-scale study. In: NIPS.

Luo, W., Schwing, A.G., Urtasun, R., 2016. Efficient deep learning for stereo matching. In: CVPR.

Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S., 2017. Least squares generative adversarial networks. In: CVPR.

Mathieu, M., Couprie, C., LeCun, Y., 2015. Deep multi-scale video prediction beyond mean square error. arXiv preprint arXiv:1511.05440.

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

Mirza, M., Osindero, S., 2014. Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784.

Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A., 2016. Context encoders: Feature learning by inpainting. In: CVPR.

Pilzer, A., Xu, D., Puscas, M., Ricci, E., Sebe, N., 2018. Unsupervised adversarial depth estimation using cycled generative networks. In: Int. Conf. on 3D Vision.

Radford, A., Metz, L., Chintala, S., 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X., 2016. Improved techniques for training gans. In: NIPS.

Saxena, A., Chung, S.H., Ng, A.Y., 2006. Learning depth from single monocular images. In: Advances in Neural Information Processing Systems. pp. 1161–1168.

Saxena, A., Sun, M., Ng, A.Y., 2009. Make3d: Learning 3d scene structure from a single still image.

Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., Webb, R., 2017. Learning from simulated and unsupervised images through adversarial training. In: CVPR.

Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process..

Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J., 2016. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In: NIPS.

Yang, N., Wang, R., Stückler, J., Cremers, D., 2018. Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In: ECCV.

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

Zhao, H., Gallo, O., Frosio, I., Kautz, J., 2017. Loss functions for image restoration with neural networks. IEEE Trans. Comput. Imaging 3 (1), 47–57.