Efficient Bayesian Deep Inversion for Depth Prediction

Catherine F. Higham[1,*]
Steven Johnson[2]
Neal Radwell[2]
Miles J. Padgett[2]
Roderick Murray-Smith[1]
[1] School of Computing Science
University of Glasgow, Glasgow, G12 8QQ, UK
[2] School of Physics and Astronomy
University of Glasgow, Glasgow, G12 8QQ, UK
[*]Catherine.Higham@glasgow.ac.uk

Abstract

We develop a deep learning method to enhance sensor detection for depth prediction. Our novel system combines sensor hardware and Bayesian inference to solve the underlying inverse problem, recovering depth from measurements. The hardware comprises single sensor non-scanning time-of-flight laser detection with synchronised video to produce a 3D depth map. The Bayesian framework provides depth prediction with uncertainty quantification. A conditional generator-discriminator adversarial network is adapted to learn a compact representation of the scene that recovers 3D depth at 30 Hz using a large training set. We transfer the network to a real hardware system and compare with ground truth depth information. Our novel synthesis of hardware and machine learning technologies addresses the important challenge of providing accurate absolute depth prediction at video rate with efficient and cost-effective non-scanning laser detection technology. This flexible and compact system has many exciting applications for autonomous vehicles, drones and wearable technology.

1 Introduction

Highly accurate scene reconstruction, in terms of reflectivity and depth, can be achieved using a time-of-flight laser detection and ranging system (LiDAR) [22, 16, 3]. However, recovery of the transverse spatial information requires laser scanning or detector arrays which adds expense, size and inflexibility to the system. Also, with such high-dimensional data arising from LiDAR, the overall acquisition and reconstruction cost is high. LiDAR measures the full temporal signal from a powerful pulse laser source and has a range of up to 100 metres outdoors, but limited to around 10 metres when laser eye-safe powers are a requirement. By contrast, single-photon sensitive LiDAR, operating in Geiger mode, provides a histogram of the arrival times of individual photon events, extending the range and improving the depth resolution. This technology also has its drawbacks as it requires high repetition rate lasers, limiting the unambiguous depth range, and is easily blinded by bright objects which are close. In this work we overcome these drawbacks by combining a single sensor non-scanning laser detector with 3D video. In order to do so, we build on state-of-the-art machine learning techniques.

1.1 Deep Learning for 3D Depth Reconstruction

Several studies have looked at reconstructing depth from RGB-only, including [8, 7, 20, 6, 18, 25, 5]. While these results can be perceptually pleasing, and suitable for certain tasks, they are based on relative spatial information and not absolute depth. Hence they are not appropriate for applications such as determining distance between cyclist and car in autonomous driving situations, where depth precision is required for responsible reasoning and reaction. Moreover, the computational expense associated with very deep networks (250 layers or more) makes attaining a video rate of 30 Hz infeasible. Recently, a study [10] investigated ways of fusing RGB with 'cheaper/faster to obtain', sparse, low-resolution data from bulky LiDAR equipment [12]. Computational analogues of depth estimation from context, parallax and motion cues have also been developed. So-called RGB-D techniques [2] can estimate depth from a single image by using a pre-trained neural network to learn context. However, these networks can be large and computationally expensive. Although they can produce convincing depth maps, these may have poor absolute depth accuracy, due to the limits of the cues they are using. Hence they are also vulnerable to optical illusions, in the same way the human eye can be tricked by dependence on particular cues such as shading and shadows.

The challenge of depth recovery is being tackled in many ways with data fusion techniques being used to combine single pixel [19, 26, 24, 11], dual pixel [11] and SPAD arrays [23, 15]. However, we are approaching this challenge in a different way with an emphasis on a compact solution with new applications for wearable tech, drones and mobility vehicles. We are also interested in providing uncertainty quantification, which is possible when stochasticity through noise is introduced into the system with a generative model, rather than a deterministic deep learning approach.

1.2 Deep Bayesian Inversion

Many applications need to reliably recover high dimensional parameters from noisy indirect observations. Such inverse problems are often ill-posed and unidentifiable: small errors in the data may lead to large errors in the model parameters; and several possible model parameter values may be consistent with the observa-

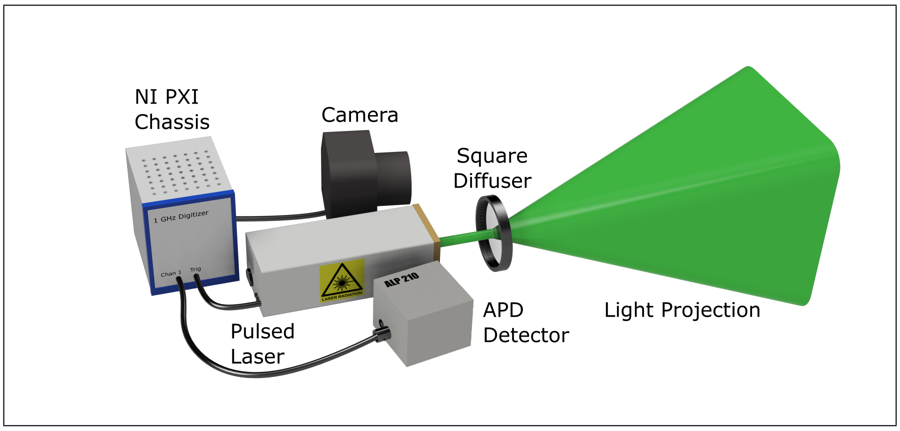

Figure 1: **Green Light Laser Experimental Set-up**. Illustration of the physical set-up of the camera, pulsed laser, and detector. The laser illuminates the scene and the detector measures the time signal of the back reflected signal, where the digitiser is synchronised with the output laser pulse via a trigger pulse.

tions. Conditional generative adversarial networks (GANs) as described by Adler & Öktem [1] provide a framework that combines a generating model with a prior information model to assign probabilities to a model parameter given data (posterior) for solving large scale inverse problems with deep learning methods. In summary, the posterior is explored by sampling from a generator trained using a discriminator critic that is defined by a conditional Wasserstein GAN (cWGAN). Exploring the posterior allows recovery of the model parameters in a reliable manner and provides uncertainty quantification.

1.3 Aims

Our overall aim is to develop cWGAN technology to create a physical, compact system (hardware and software) that combines, for the first time, low cost, low power and flexibility, and is capable of accurately reconstructing absolute depth (rather than perceptually pleasing relative depth) in a previously unseen scene at video rate.

2 Novel Light Laser Application

A prototype hardware system was built for model verification. A pulsed laser illuminates the scene, the reflected signal is recorded giving the time-of-flight laser response. The physical set-up is illustrated in Figure 1. The prototype system used a pulsed laser (Teem Photonics SNG-03E-100) with a 7 kHz repetition rate and 1 ns pulse width. The 532 nm laser light was incident on a square pattern diffuser (Thorlabs ED1-S20) to give a top-hat shaped square pattern of light to illuminate the scene. The light back-scattered by the scene was detected with a high-speed Si avalanche photodetector (MenloSystems APD 210). The time-of-flight signal from the APD was recorded with a 1 GHz digitizer card (NI PXI 5154) within a National Instruments PXI chassis. The digitizer was triggered via pick-off light from the laser pulse on a photodiode. A measurement time of 1 second was used to record the back-scattered signal, which was collated into a timing histogram. For a ground truth measurement of the 3D scene a Kinect for Windows V2 was used. The RGB camera from the Kinect was used as the camera for the 2D image. The single time bin histogram laser response was integrated with G channel video frames. Without further processing, the depth prediction was made by inputting this data into the pre-trained generator network. We present real-time scene reconstruction results showing the successful transfer of our neural network to data arising from real life situations, see Section 4.2.

3 Method

We develop a novel computational method that integrates input from our proposed system, denoted Green Light Laser (GLL), with a cWGAN to predict depth. In Section 3.1 we briefly discuss the theory. In Section 3.2 we describe the training data and simulations, and in Section 3.3 the cWGAN architecture and training. Key to the success of our approach is achieving generalisation by training on a range of different scenes. To do this we simulate the laser response (GSL) using an optics inspired forward model as described in Section 3.2.2. This important step leverages the training data and facilitates solution of the inverse problem. Transfer and calibration of the trained model to our system is discussed in Section 3.4. In Section 2 we present our novel hardware system GLL set-up which is applied in a real setting with results in Section 4.2.

3.1 Deep Bayesian Inversion Theory

Our computational method is designed to solve the underlying inverse problem and to reconstruct a scene in real time. For many reasons, including downstream decisions, it is important to build into such a system the ability to estimate the error underpinning our depth predictions. This can be done by posing the inverse problem in a Bayesian framework and using a generative model to sample from the posterior distribution [1]. This technique learns to generate new data with the same statistics as the training set.

Given data, x, GANs learn to generate new data, $\hat{x}$. The basic idea introduces a variable, z, commonly called a latent variable because it is unseen, that comes from a Gaussian distribution that is easy to sample from and the objective is to learn the conditional probability distribution, $\pi(x|z)$, so that given some z, also referred to as noise, $\hat{x}$ can be sampled form this learned distribution. A generator network is tasked with producing realistic $\hat{x}$ and a discriminator network is tasked with deciding whether $\hat{x}$ is real or fake. The discriminator 'sees' real x along with $\hat{x}$ and a loss value is passed back to the generator so that it can improve its output [13]. This loss function, defined by the discriminator, is potentially more flexible and task specific than a standard regression loss function.

Further development in GANs has introduced the Wasserstein GAN with stability of learning, to overcome issues such as mode collapse and to provide meaningful learning curves useful for debugging and hyper-parameter searches [4] with improved training using gradient penalty [14]. A notable extension is the conditional GAN [21]. The benefits of combining conditional and Wasserstein GANs in order to control image generation, conditioned on both discrete and continuous attributes are described in [9].

In Bayesian inversion, the ground truth, x, and measured data, y, are assumed to be generated by random variables X and Y respectively. The aim is to recover the posterior $\pi(x|y)$ which describes all possible solutions $X = x$ along with their probabilities given data $Y = y$. The deep posterior sampling approach samples from a generator trained using a conditional discriminator. We assume that $\pi(x|Y = y)$ can be approximated by a parameterised family $\{\mathcal{G}_\theta(y)\}_{\theta\in\Theta}$ of probability measures on X. The best approximation is defined as $\mathcal{G}_{\theta^*}(y)$ where $\theta^* \in \Theta$ solves

$$\theta^* \in \operatorname*{arg\,min}_{\theta\in\Theta} \mathcal{W}\left(\mathcal{G}_\theta\left(y\right), \pi\left(x|y\right)\right), \tag{1}$$

and where $\mathcal{W}$ quantifies the distance between probability measures on X. The distance should be finite and differentiable almost everywhere for computational feasibility using stochastic gradient descent. For this reason, we use the Wasserstein 1-distance $\mathcal{W}$ in (1). However this formulation requires access to the posterior. Also, the distribution of the data is often unknown and evaluating Wasserstein 1-distance from its definition is not computationally feasible. Results in [1] show that all these drawbacks can be circumvented by rewriting equation (1) as an expectation over the joint law $(x, y) \sim \mu$. This makes use of specific properties of the Wasserstein 1-distance (Kantorovich-Rubenstein duality) and defines (θ^*, ϕ^*) via

$$\operatorname*{arg\,min}_{\theta\in\Theta} \left\{ \sup_{\phi\in\Phi} \mathbb{E}_{\substack{(x,y)\sim\mu \\ z\sim\eta}} \left[D_\phi\left(x, y\right) - D_\phi\left(G_\theta\left(z, y\right), y\right)\right] \right\}. \tag{2}$$

Here, $G_\theta : Z \times Y \to X$ (generator) is a deterministic mapping such that $z \sim \eta$ is a random variable that can be sampled in a computationally feasible manner and $D_\phi : X \times Y \to \mathbb{R}$ (discriminator) is a measurable mapping that is 1-Lipschitz in the X variable. With access to supervised training data, samples generated by $(x, y) \sim \mu$, the μ-expectation can be replaced by averaging over training data. The 1-Lipschitz condition on the discriminator is enforced by including a gradient penalty to the training objective.

3.2 Training Data and Simulated Laser Signal

3.2.1 Training Data

Our computational model takes as input a LiDAR signal and a light channel (green) image from a RGB camera and outputs the underlying depth map. The green channel is chosen, as most informative, for indoor scenes. The NYUdepth dataset comprises more than 100,000 sequential video frames and synchronised Kinect depth measurements from over 100 indoor scenes [8]. Further, the camera viewpoint of these sequential frames changes within scenes and hence creates a realistic training environment for general non-static scene reconstruction. In order to leverage this resource, we develop an optics inspired forward model to simulate the green light laser response from synchronised RGB video and Kinect depth. Our objective is to capture the most relevant depth information contained in the signal, namely the position and relative height of the peaks.

3.2.2 Green Light Laser Simulation

From the synchronised RGB frames and Kinect depth measurements, we extract the G channel, g_N, and the depth map, d_N, where N denotes the number of pixels in the image plane. We discretise d_N over the time bins, t, which are chosen to match the performance of the current technology (75 ps). The laser response is simulated by summing g_N over each time bin and correcting by $\frac{1}{(d_N)^2}$. This approach is adequate for training. Visualisation of the simulated signal and comparison with the raw and smoothed real signal is discussed in Section 4.2. The signals are standardised so that the peak signal takes value 1.

3.3 GAN Architecture and Training Options

Our key algorithmic developments are to adapt a cWGAN to fuse LiDAR and RGB inputs so that the learned feature representations of spatial and depth information can mutually aid depth recovery and scene reconstruction.

The generator and discriminator, G_θ and D_ϕ, are built and trained using the MATLAB Deep Learning Toolbox [27]. An overview of the generator architecture is shown in Figure 2. The architecture design and training options are informed by MATLAB code developed for a range of GANs in `https://github.com/zcemycl/Matlab-GAN`. Our approach is to combine translation at the pixel level [17] with a conditional improved Wasserstein GAN [4] by including a discriminator architecture and adding a gradient penalty to the discriminator loss. The patch discriminator penalizes structure at the scale of patches to improve modelling of high-frequencies. In addition an ℓ_1 penalty is added to the generator loss to enforce correctness at low frequencies.

A second important step that we introduce is to interpret the laser response, the timed arrival of photons, as a random variable from a Poisson distribution, z. Sampling depth values is computationally feasible and introduces randomness into the neural network model in a realistic way. It exploits knowledge of the physics and is robust to background noise and time bin step sizes and can be used without additional training across both simulated and real laser measurement scenarios where levels of background noise and time bin sizes may differ. The discretisation level is chosen to be 7.5mm resulting in 532 time bins.

3.3.1 Generator Architecture

The Generator U-net Architecture comprises *encoding layers* with convolutional blocks that each down-sample the input by a factor of two and *decoding layers* with convolutional blocks that each up-sample the encoder output by a factor of two. See Appendix A for more details about the composition of the convolutional blocks. Input into the network is a concatenation of the laser response, L, and image frames I. The U-Net architecture has skip connections between each layer i in the encoder and layer $n - i$ in the decoder, where n is the total number of layers. This addition allows the up-sampling decoder to see the corresponding down-sampling encoder and hence is a powerful design for applications, such as ours, mapping from one spatial domain (light) to another spatial domain (depth).

3.3.2 Discriminator Architecture

The Patch Discriminator Architecture comprises *encoding layers* with convolutional blocks. See Appendix B for more details about the composition of the convolutional blocks.As for the generator, convolutions are 4×4 spatial filters applied with stride 2 which down sample by a factor of 2.

3.3.3 Training

Training is conducted on a single TITAN Xp GPU. The discriminator is updated five times for every update of the generator. The model is stopped at 80 epochs when the validation set indicates over fitting.

Results are discussed in Section 4.1.

3.4 Transfer and Calibration

Having developed and tested a cWGAN for scenes up to 10 metres, and shown proof of principle, we refine and streamline the model for transfer to our system. We find that we can reduce the number of convolutional blocks (from 7 to 5) and also the number of filters at each layer. This modification was found to improve efficiency without compromising robustness and accuracy. Previously, we sampled from the laser signal to obtain input L. We now add a fully connected input layer with the objective of learning how to sample from the laser signal. We also add a convolutional layer after the last decoding block with the objective of learning how to weight the decoding output. Further, a simpler version of Wasserstein loss, clipping the weights rather than constraining them [4], is found adequate for our needs. This lightweight model was then retrained on 32 scenes, up to 5 metres, and tested on unseen real datasets.

Results are discussed in Section 4.2.

4 Results

4.1 Depth Prediction Comparison

For 1,000 test video frames, depth prediction based on RGB-only was performed using the NYUdepth pre-trained model and toolbox [8]. The simulated laser response for each test frame was combined with the video information and the depth prediction obtained from our pre-trained generator network. The predictions are evaluated against the Kinect depth map in terms of the reconstruction signal-to-noise ratio (RSNR), which weights the error with the reference depth. RSNR is defined as RSNR = $10 \log_{10} \|x\|_2 / \|\hat{x} - x\|_2$ where x is the reference depth and $\hat{x}$ is the reconstructed depth, with components ranging over all pixels. RSNR scores are evaluated for RGB-only and Green Simulated Laser (GSL).

Results for two bedroom scenes are illustrated in Figures 3 and 4. The average RSNR scores for GSL are 17.8 and 15.0 which are higher than the averages of 15.4 and 11.8 for RGB-only, for bedroom scenes 1 and 2 respectively. GSL outperforms RGB-only almost everywhere in bedroom scene 1 (Figure 3) and by at least 3 points throughout the scene (Figure 4).

Figure 5 shows results for a living room scene with over 250 frames. Here, the average RSNR score for GSL is 18.9 compared with 14.6 for RGB-only. For illustration, the data and reconstruction are shown for frames 44 and 213. GSL outperforms RGB-only across the whole sequence as the room viewpoint changes. Inspection of the reconstructions indicates that GSL better captures the full room depth than RGB-only. The RGB-only reconstruction looks correct to the human eye but over-reliance on RGB information comes at the expense of relative accuracy over absolute accuracy.

4.1.1 Average RSNR performance

Based on 35 test video sequences, Table 1 shows the average RSNR scores grouped by scene type. GSL performs best on five out of six scene types and generally does better in scenes such as bedrooms and some study/living rooms with predominantly low spatial frequencies. The scenes vary in terms of content, viewing angles and light sources. We aimed for model transferability by including 100 different scenes in our training and these results indicate that we have, on average, achieved this goal. Improvements could be obtained by increasing scene content, for example to include more objects with high spatial frequency such as chair and table legs.

In terms of computing time, the GSL reconstruction is an order of magnitude faster than that of RGB-only, due to the use of a much smaller network.

Table 1: **Average RSNR performance.** Based on 35 test video sequences, GSL performs best on five out of six scene types and generally does better in scenes such as bedrooms and some study/living rooms with predominantly low spatial frequencies.

Scene Type	Number of Scenes	RGB-only	GSL
Classroom	(7)	14.7	**15.5**
Dining room	(7)	11.8	**12.7**
Home office	(4)	**13.5**	13.4
Living room	(7)	13.5	**14.1**
Study room	(3)	12.4	**12.9**
Bedroom	(10)	13.6	**15.5**

4.2 Real-time Reconstruction using Novel Hardware

4.2.1 Calibration of the Laser Signal

Using the novel hardware system described in Section 2, one hundred video frames and laser responses were collected for a scene, along with depth estimates from a Kinect camera. Also collected was the instrumental response, measured within the camera, and hence noise-free, shown in Figure 6. Our system is designed to take, as input, one noisy frame so the option of removing background noise by collecting several frames is not considered. Instead we filter the raw signal with the instrumental response and then smooth using a sliding window. The aim of this pre-processing step is to capture the relevant weight distribution of light over the depth time-bins to enable transfer of the neural network trained with a simulated signal to the real set-up. We compare the filtered and smoothed signal with the simulated signal and measure the accuracy of the predicted results in terms of the peak signal-to-noise ratio (PSNR) performance. The results are consistent between the frames and between the simulated (PSNR mean 25.2, std 0.080) or real laser signal (PSNR mean 25.5, std 0.131) indicating that our system transfers well to real data and is robust to varying input.

4.2.2 Depth Prediction for Real Datasets

The results in Section 4.1 used simulated laser responses. We now test the method with real data acquired with our own hardware. We use GLL to denote the Green Light Laser method. Pixel reconstruction results for four input images and corresponding smoothed LiDAR signals, and for three methods RGB-only, GLL and Kinect are shown in Figure 7 with the horizontal axis representing depth in metres. The depth maps are viewed from above (top view) to more clearly show the depth line. PSNR performance scores for methods GLL and RGB-only (in brackets), compared with Kinect, are 36.6 (32.0), 34.8 (32.2), 39.9 (24.4) and 26.9 (13.3) respectively. Figure 7 shows that GLL is more accurate than RGB-only at matching the Kinect depth line and indicates that GLL is able to predict absolute depth by fusing the image with the LiDAR signal.

4.2.3 Uncertainty Quantification

The results in Section 4.2.2 were obtained by sampling once from the generator model which, in this Bayesian framework, is equivalent to sampling from the posterior. This is the standard real-time operating mode. We can however, extend the approach to explore the posterior landscape by repeated sampling from the generator model. In this way we can check the reliability of the model parameters and quantify uncertainty. For illustration, Figure 8 shows histograms of obtained depth prediction values, computed by applying posterior sampling to the test data 100 times, for a randomly chosen pixel from each of four test scenes with different depth values in Figure 7.

4.2.4 New datasets and simulation code

The new GLL datasets produced for this work, and code for simulating the laser response are available at https://doi.org/10.5525/gla.researchdata.1542.

5 Discussion and Conclusion

We have developed a novel system for fusing a single non-scanning LiDAR depth signal with a single color channel that provides more accurate depth prediction than a state-of-the-art deep learning approach using RGB information alone. Furthermore, by using ten times fewer layers in the network, our approach runs at least an order of magnitude more quickly (on both CPU and GPU), allowing for video rate to be achieved.

With this approach an inexpensive video camera provides 2D light levels for a scene. A single detector sensor records luminance levels when the whole scene is flood-illuminated, with very high timing accuracy (75 ps). A trained network is introduced into the computation stream and this new technique removes the need for a bulky complex scanning system. We also note that the use of laser technology enables a greater range of depths than typical 3D imaging cameras, making it more suitable for outdoor use. The lack of scanning electronics and the miniaturisation of cameras allows this technology to have a very compact sensor head package, potentially down to optical fibre sizes. This opens the way for a single central laser, detector and computational system to sense from tens of sensor heads distributed around the platform; applications include driver-less cars, drones, underwater vehicles, and wearable technology.

The use of Bayesian inversion offers two key advantages. First, prior knowledge can be incorporated to tackle the inherent ill-posedness and unidentifiability associated with the inverse problem. Second, this framework provides samples from the posterior that can be used to quantify and manage the inherent uncertainty in the model. These issues impact on robustness, performance, transparency and interpretability, which are important for safety-related applications.

References

[1] Jonas Adler and Ozan Öktem. Deep Bayesian Inversion. *arXiv preprint arXiv:1811.05910*, 2018. 3, 4, 5

[2] Ahmed J. Afifi and Olaf Hellwich. Object depth estimation from a single image using fully convolutional neural network. In *International Conference on Digital Image Computing: Techniques and Applications*, pages 1–7, 12 2016. 2

[3] Markus-Christian Amann, Thierry Bosch, Marc Lescure, Risto Myllylä, and Marc Rioux. Laser ranging: a critical review of usual techniques for distance measurement. *Opt. Eng.*, 40:10–19, 2001. 1

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR, 06–11 Aug 2017. 5, 6, 7

[5] S. Chen, M. Tang, and J. Kan. Predicting depth from single RGB images with pyramidal three-streamed networks. *Sensors (Basel)*, 3:667, 2019. 2

[6] D. Eigen and R. Fergus. Monocular depth estimation using neural regression forest. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 2800–2809, 2015. 2

[7] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE Conference on Computer vision and pattern recognition (CVPR)*, page 2650–2658, 2015. 2

[8] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems (NIPS)*, page 2366–2374, 2014. 2, 6, 8

[9] Cameron Fabbri. Conditional Wasserstein generative adversarial networks, 2017. 5

[10] W. Van Gansbeke, D. Neven, B. De Brabandere, and L. Van Gool. Sparse and noisy LiDAR completion with RGB guidance and uncertainty. In *16th International Conference on Machine Vision Applications (MVA)*, pages 1–6, 2019. 2

[11] R. Garg, N. Wadhwa, S. Ansari, and J. Barron. Learning single camera depth estimation using dual-pixels. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7627–7636, 2019. 2

[12] Andreas Geiger. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, pages 3354–3361, Washington, DC, USA, 2012. IEEE Computer Society. 2

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems (NIPS)*, page 2672–2680, 2014. 5

[14] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved Training of Wasserstein GANs. *arXiv:1704.00028*, 2017. 5

[15] A. Gupta, A. Ingle, and M. Gupta. Asynchronous single-photon 3d imaging. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7908–7917, 2019. 2

[16] Cheng Ho, Kevin L. Albright, Alan W. Bird, Jeffrey Bradley, Donald E. Casperson, Miles Hindman, William C. Priedhorsky, W. Robert Scarlett, R. Clayton Smith, James Theiler, and S. Kerry Wilson. Demonstration of literal three-dimensional imaging. *Appl. Opt.*, 38:1833–1840, 1999. 1

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with Conditional Adversarial Networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 6

[18] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *Proceedings of the Fourth International Conference on 3D Vision (3DV)*, page 239–248, 2016. 2

[19] David B. Lindell, Matthew O'Toole, and Gordon Wetzstein. Single-photon 3d imaging with deep sensor fusion. *ACM Trans. Graph.*, 37(4), July 2018. 2

[20] F.Y. Liu, C.H. Shen, G.S. Lin, and I. Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Trans. Pattern Anal.*, 38:2024–2039, 2015. 2

[21] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv:1411.1784*, 2014. 5

[22] Risto Myllylä, Janusz Marszalec, Juha Kostamovaara, Antti Mäntyniemi, and Gerd-Joachim Ulbrich. Imaging distance measurements using TOF lidar. *J. Opt.*, 29:188, 1998. 1

[23] M. Nishimura, D. B. Lindell, C. Metzler, and G. Wetzstein. Disambiguating Monocular Depth Estimation with a Single Transient. *European Conference on Computer Vision (ECCV)*, 2020. 2

[24] M. O'Toole, F. Heide, D. B. Lindell, K. Zang, S. Diamond, and G. Wetzstein. Reconstructing transient images from single-photon sensors. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2289–2297, 2017. 2

[25] A. Roy and S. Todorovic. Monocular depth estimation using neural regression forest. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 5506–5514, 2016. 2

[26] T. Siddiqui, R. Madhok, and M. O'Toole. An extensible multi-sensor fusion framework for 3d imaging. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[27] MATLAB Deep Learning Toolbox, 2019. 6

Acknowledgements

The authors would like to thank and acknowledge the financial support from the Engineering and Physical Sciences Research Council (EPSRC) (QuantIC Nos. EP/M01326X/1, EP/T00097X/1 and Looking and Listening in Complex Media No. EP/S026444/1); and also the H2020 European Research Council (ERC) (TWISTS, No. 340507). For the purpose of open access, the authors have applied a creative commons attribution (CC BY) licence to the accepted manuscript.

A Generator U-net Architecture

The Generator U-net Architecture comprises an encoder with seven blocks C128-C256-C512-C1024-C1024-C1024-C1024 and a U-net decoder with seven blocks C1024-C1024-C1024-C1024-C512-C256-C128, where Ck denotes a Convolution-BatchNorm-ReLU layer with k filters. Input is a concatenation of z and two sequential image frames y. All convolutions are 4×4 spatial filters applied with stride 2. Convolutions in the encoder down-sample by a factor of 2, whereas in the decoder they up-sample by a factor of 2. After the last layer in the decoder, a convolution is applied to map to the number of output channels followed by an activation function. As an exception to the above notation, Batch-Norm is not applied to the first C128 layer in the encoder. All ReLUs in the encoder are leaky, with slope 0.2, while ReLUs in the decoder are not leaky. The U-Net architecture has skip connections between each layer i in the encoder and layer $n - i$ in the decoder, where $n = 7$ is the total number of layers.

B Patch Discriminator Architecture

The Patch Discriminator Architecture comprises an encoder with five blocks C128-C256-C512-C1024-C1024. Input is a concatenation of x and y. As above, convolutions are 4×4 spatial filters applied with stride 2 which down sample by a factor of 2. After the last layer, a convolution is applied to map to a 1-dimensional output, followed by a sigmoid function. As an exception to the above notation, Batch-Norm is not applied to the first C128 layer. All ReLUs are leaky, with slope 0.2.

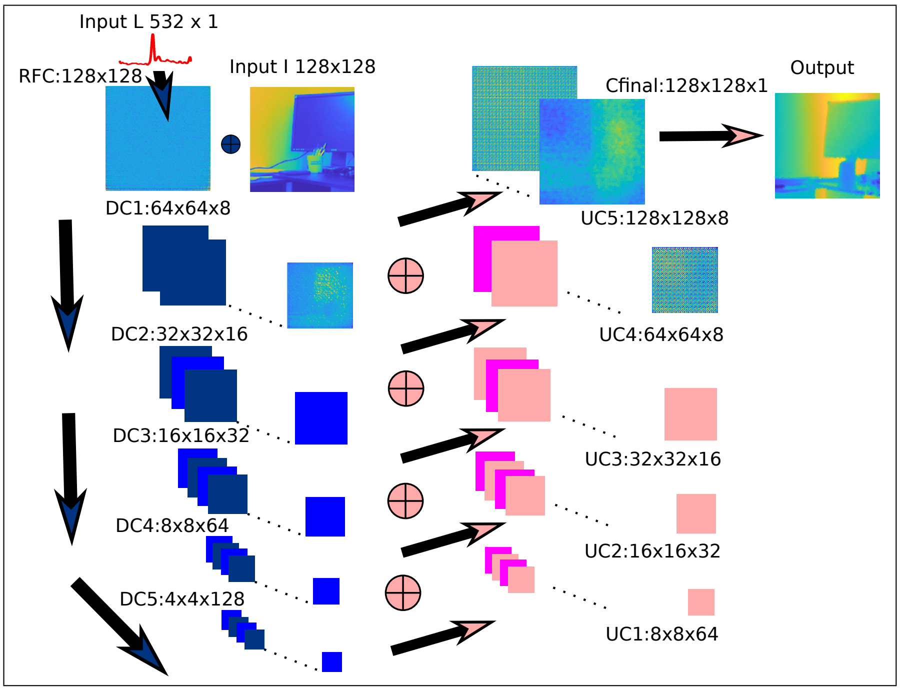

Figure 2: **Generator U-net Architecture**. The input layer concatenates the image I and the laser response, L, up-sampled and reshaped (RFC) to the size of I. The lightweight generator U-net architecture comprises five encoding convolutional blocks (denoted DC1, DC2, DC3, DC4 and DC5) that each down-sample the input by a factor of two whilst increasing the number of filters, producing 8, 16, 32, 64 and 128 feature maps respectively. The size of these feature maps, in terms of width, height and number, are indicated after the block name. After DC5 the output is up-sampled by a factor of two (UC1) and concatenated with the similarly sized output from DC4, indicated by $\oplus$. This step is repeated four times (UC2, UC3, UC4 and UC5) resulting in eight feature maps with width and height 128 pixels. A final convolutional layer (Cfinal) learns to weight these features and produces a depth map. The addition of skip connections between the decoder and the encoder is a powerful design for applications, such as ours, fusing data and mapping from one spatial domain (light) to another spatial domain (depth).

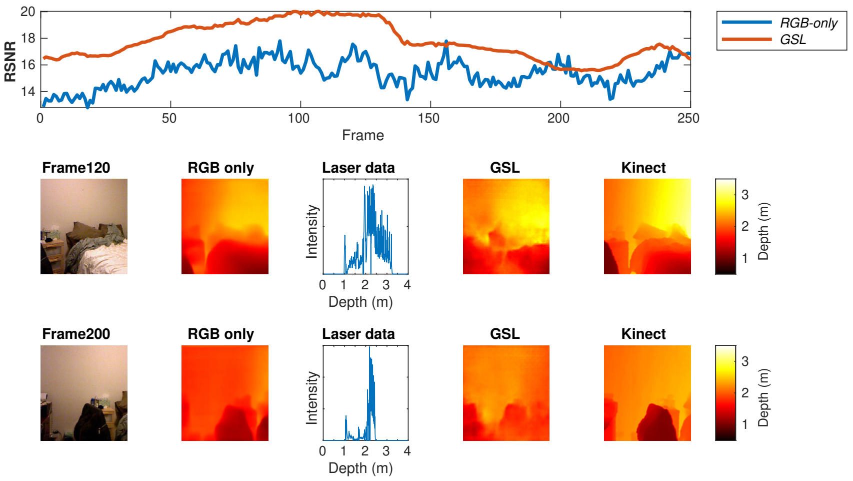

Figure 3: **Depth Prediction Comparison of RGB-only with G and Simulated Laser (GSL): Bedroom Scene 1.** RSNR scores (higher is better) *upper row*: RGB-only *blue line* and GSL *red line* for a previously unseen bedroom video sequence scene with 250 frames. The average RSNR score was 17.8 for GSL and 15.4 for RGB-only. GSL outperforms RGB throughout the sequence except where the depth range is reduced making the problem easier, e.g., Frame 200, and the GSL advantage of predicting actual depth is consequently lessened. Frames 120 and 200 are illustrated in the *middle row* and *bottom row*. *Second column* shows RGB-only reconstruction. *Third column* shows simulated laser response data. *Fourth column* shows the GSL reconstruction and *fifth column* the Kinect depth.

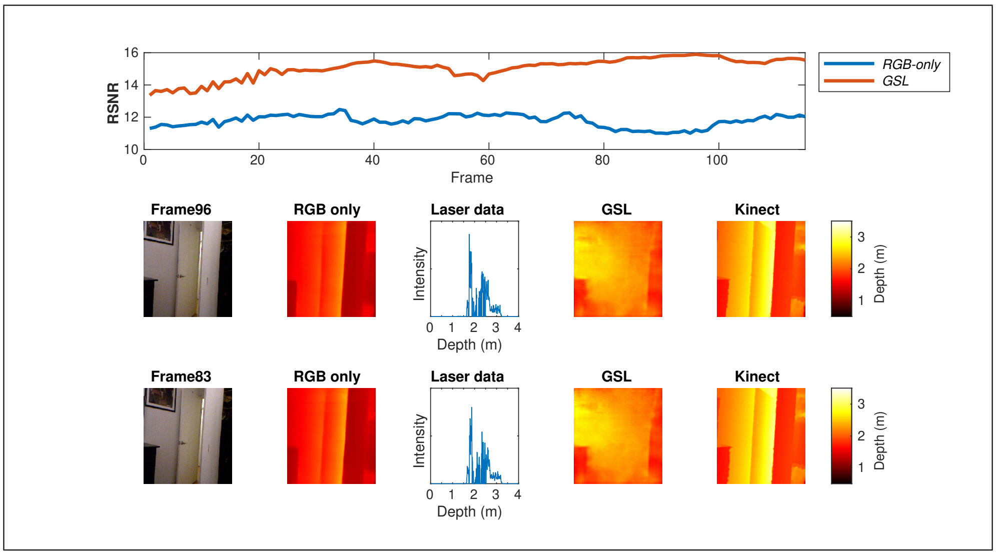

Figure 4: **Depth Prediction Comparison of RGB-only with G and Simulated Laser (GSL): Bedroom Scene 2.** RSNR scores (higher is better) *upper row*: RGB-only *blue line* and GSL *red line* for a previously unseen bedroom video sequence scene with just over 100 frames. The average RSNR score was 15.0 for GSL and 11.8 for RGB-only. GSL outperforms RGB throughout the sequence. Frames 96 and 83 are illustrated in the *middle row* and *bottom row*. *Second column* shows RGB-only reconstruction. *Third column* shows simulated laser response data. *Fourth column* shows the GSL reconstruction and *fifth column* the Kinect depth.

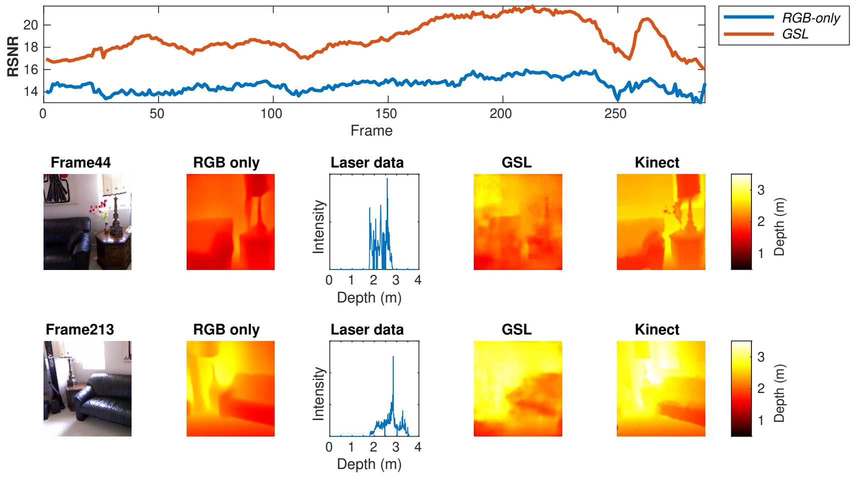

Figure 5: **Depth Prediction Comparison of RGB-only with G and Simulated Laser (GSL): Living Room Scene.** RSNR scores (higher is better) *upper row*: RGB-only *blue line* and GSL *red line* for a previously unseen living room video sequence scene with just under 300 frames. The average RSNR score was 15.2 for GSL and 11.1 for RGB-only. GSL outperforms RGB throughout the sequence. Frames 44 and 213 are illustrated in the *middle row* and *bottom row*. *Second column* shows RGB-only reconstruction. *Third column* shows simulated laser response data. *Fourth column* shows the GSL reconstruction and *fifth column* the Kinect depth. Depth range is also indicated by colour with blue low and red high.

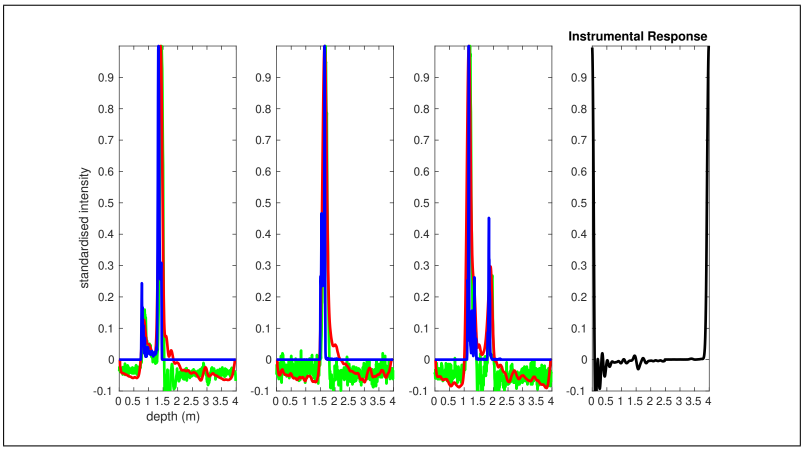

Figure 6: **Comparison of the real signal (*green*), the simulated signal (*blue*) and the filtered+smoothed real signal (*red*) for 3 different single frame scenes.** The signals were filtered using the instrumental response (*column four*) and smoothed using a width of 10 depth/time bins. The signals have been standardised so that the peak signal takes value 1.

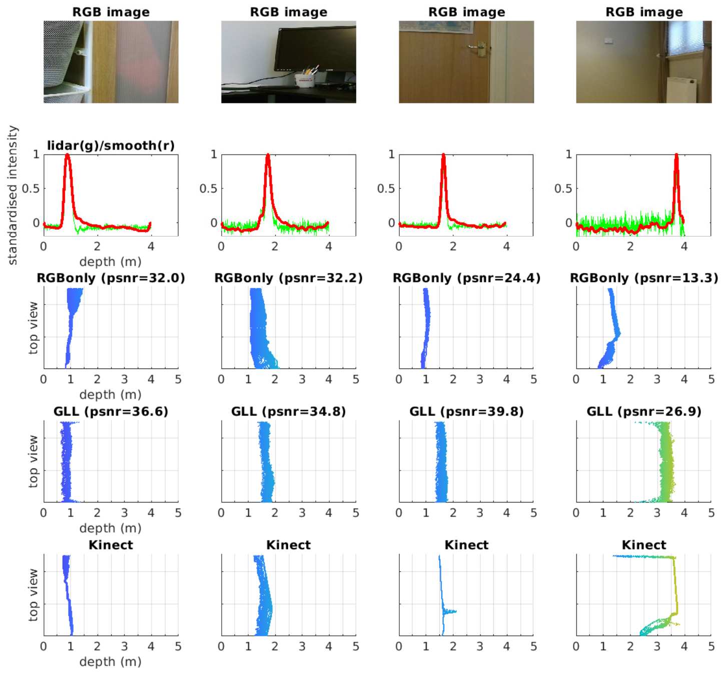

Figure 7: **Depth Prediction for real datasets acquired with our own hardware.** Pixel reconstruction results for four input images *first row* and corresponding smoothed LiDAR signals *second row*, and for three methods: RGB-only *third row*, GLL *fourth row* and Kinect *fifth row*; are shown in *top view*, with the horizontal axis representing depth in metres. Performance scores (PSNR) for methods RGB-only and GLL, compared to Kinect, are indicated above these reconstructions. GLL results are closer to the 'gold standard' Kinect than the RGB-only in each of the above images.

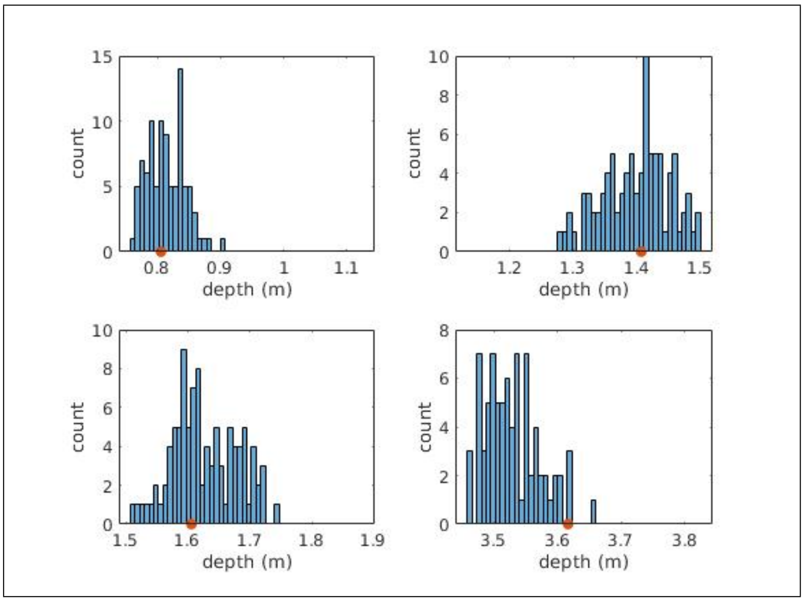

Figure 8: **Uncertainty Quantification: Posterior Sampling**. Histograms of obtained depth prediction values, computed by applying posterior sampling to the test data 100 times, for pixels from the four test scenes with different depth values in Figure 7. The true value is indicated on the horizontal axis by a red filled circle.