To complete or to estimate, that is the question: A Multi-Task Approach to Depth Completion and Monocular Depth Estimation

Amir Atapour-Abarghouei[1] Toby P. Breckon[1,2]
[1]Department of Computer Science – [2]Department of Engineering
Durham University, UK
{amir.atapour-abarghouei,toby.breckon}@durham.ac.uk

Abstract

Robust three-dimensional scene understanding is now an ever-growing area of research highly relevant in many real-world applications such as autonomous driving and robotic navigation. In this paper, we propose a multi-task learning-based model capable of performing two tasks:- sparse depth completion (i.e. generating complete dense scene depth given a sparse depth image as the input) and monocular depth estimation (i.e. predicting scene depth from a single RGB image) via two sub-networks jointly trained end to end using data randomly sampled from a publicly available corpus of synthetic and real-world images. The first sub-network generates a sparse depth image by learning lower level features from the scene and the second predicts a full dense depth image of the entire scene, leading to a better geometric and contextual understanding of the scene and, as a result, superior performance of the approach. The entire model can be used to infer complete scene depth from a single RGB image or the second network can be used alone to perform depth completion given a sparse depth input. Using adversarial training, a robust objective function, a deep architecture relying on skip connections and a blend of synthetic and real-world training data, our approach is capable of producing superior high quality scene depth. Extensive experimental evaluation demonstrates the efficacy of our approach compared to contemporary state-of-the-art techniques across both problem domains.

1. Introduction

With the growing demand for accurate 3D scene understanding as an integral part of various computer vision applications, efficient and accurate depth estimation has received significant attention within the research community in the past few years. Conventional depth estimation techniques such as stereo correspondence [49], structure from motion [11], depth from light diffusion [53, 58] and alike have led to significant strides in real-world scene understanding applications. However, pervasive issues and complications ever-present in depth-reliant vision systems (*e.g.* missing depth, temporal and in-scene consistency, intensive computational and calibration requirements and alike), have led to the emergence of entire areas of research focusing on refinement procedures post estimation or capture [3, 8, 10, 16, 39, 55] to render scene depth more useful for downstream applications.

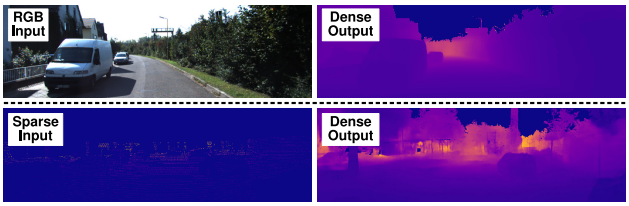

Figure 1: Exemplar results - a single network architecture facilitates seamless performance of both monocular depth estimation (from RGB, upper) and sparse depth completion (from LiDAR, lower).

In recent years, monocular depth estimation (*i.e.* estimating scene depth from a single RGB image) has received widespread attention within both academia and industry as a more effective, economical and innovative alternative to more conventional depth estimation strategies [5, 6, 14, 18, 20, 31, 66].

In this work, we propose a multi-task depth prediction approach capable of performing sparse depth completion and monocular depth estimation in a joint network trained end to end using a mixture of publicly available synthetic [17] and naturally-sensed real-world [19] training data from urban driving scenarios. In order words, this work specifically handles two practical depth estimation/completion scenarios:- (a) dense depth image estimation from an RGB input (monocular depth estimation) and (b) sparse to dense depth completion from a sparse LiDAR (laser scanner) input (depth completion). Consisting of two sub-networks jointly trained, our approach can seamlessly perform either task without any need for re-training. The first sub-network is solely trained to regress to sparse depth information, sim-

ilar to that obtained via a 64-channel LiDAR sensor [19]. This network carries out its objective based on the information available in the RGB view of the scene, thus mostly focusing on low-level feature extraction to estimate depth values for various objects and components within a constrained region of the scene. This sparse depth output is subsequently utilised by the second sub-network to generate a full dense depth output of the entire scene, requiring high-level inferences and a deeper semantic and geometric understanding of the scene.

During inference in the deployment stage, the entire model can be used as a single unit to perform monocular depth estimation from an RGB colour input, or alternatively, the second sub-network can be utilised alone to yield a dense depth image given a sparse depth input acquired by a LiDAR (laser scanner). Using advances in adversarial training [21], a deep architecture relying on skip connections to preserve high-level spatial features [42, 46] and a combination of synthetic and real-world training data to ensure both the density of the entire depth output and dispensing with any potential domain adaptation requirements [1, 5, 65], our approach can generate accurate scene depth. In short, our primary contributions are as follows:

- A joint multi-task framework for depth prediction encouraging improved geometric and contextual learning to boost performance.
- Monocular depth estimation via adversarial training, a deep architecture with skip connections and a robust compound objective function directly supervised using this framework to outperform prior contemporary work [5, 7, 14, 20, 31, 36, 62, 66].
- Sparse to dense depth completion via the same multi-task model, capable of generating a dense depth output given a sparse depth input captured via a LiDAR sensor with results superior to prior contemporary work [10, 16, 40, 50, 54].
- Unique leverage of both synthetic [17] and real-world datasets [54] to ensure high-density complete depth outputs, despite such levels of density not existing in any real-world training images.
- Capable of generalising to previously unseen images from different environments since the training data is sampled from varying data domains.

2. Related Work

Here, relevant prior work is considered over two distinct areas, namely monocular depth estimation (Section 2.1) and sparse depth completion (Section 2.2).

2.1. Monocular Depth Estimation

The emergence of learning-based approaches capable of estimating depth from a single RGB image has caused revolutionary changes in the landscape of 3D scene understanding, leading to significant strides made within the field of monocular depth estimation in recent years. While traditionally, probabilistic graphical models [34, 47, 48] and non-parametric approaches [27, 35, 38] offered promising solutions, their use of hand-crafted features and intensive computational requirements created issues regarding their efficiency and performance capabilities.

With the advent of convolutional neural networks and a growing number of publicly available depth datasets [19, 51, 52], supervised approaches made significant improvements in the state of the art despite prevalent issues in the quality of the ground truth depth for supervision. For instance, [13, 14] generate depth from a two-scale network trained on RGB and depth and [32, 33] offer remarkable performances by providing higher quality outputs.

On the other hand, more recent techniques began circumventing the need for ground truth depth by reconstructing corresponding views within a stereo correspondence framework to calculate disparity. The work in [59] generates the right view given the left while producing an intermediary disparity image. Similarly, [20] employs bilinear sampling [26] and a left/right consistency constraint for improved results. In [66], depth and pose prediction networks, supervised via view synthesis, are trained to estimate depth and camera motion. The work in [31] produces dense depth by enforcing a model supervised by sparse ground truth depth within a stereo framework via an image alignment loss. Although the training data for the majority of such approaches is abundant and easily obtainable, they still suffer from undesirable artefacts, such as blurring and incoherent content, due to the nature of their secondary supervision.

More recently, supervised approaches have begun using synthetic training images and are capable of producing better quality depth outputs, despite potential issues with domain shift [5, 64, 65]. In this work, we utilise a blend of real world [19] and synthetic data [17] in a supervised training approach to accurately estimate depth from a monocular RGB image without the need for any domain adaptation.

2.2. Sparse Depth Completion

Depth completion can refer to a range of related problems with different input modalities [3]. The existing literature contains a variety of techniques capable of completing relatively dense depth images that contain missing values, such as those utilising exemplar-based depth inpainting [4], low-rank matrix completion [60], object-aware interpolation [2], tensor voting [30], Fourier-based depth filling [8], background surface extrapolation [41], learning-based approaches using deep networks [1, 7, 63], and alike [9, 37].

However, depth completion can also refer to the problem of generating dense depth information from a scene when only a sparse representation of the scene depth is available. This is of particular interest in robotics applications such

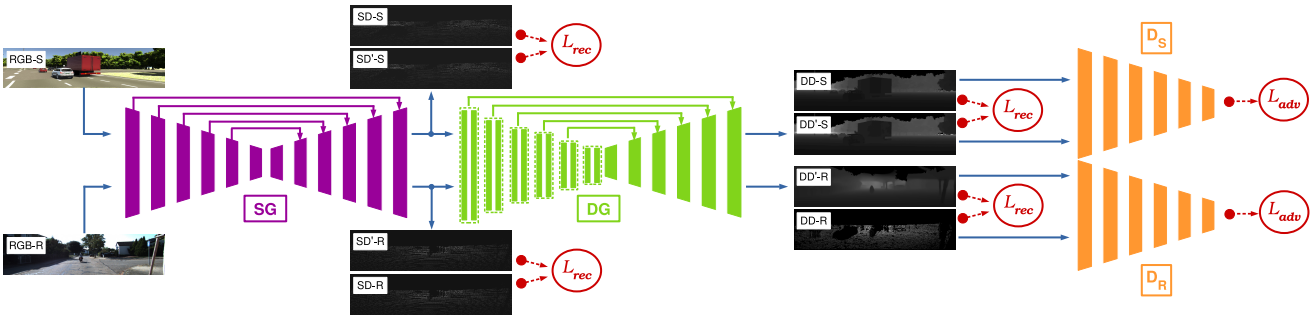

Figure 2: Overall training procedure of the approach. **S:** synthetic data from the virtual environment [17]; **R:** data captured from the real world [54]; **SD:** sparse depth; **DD:** dense depth; **SG:** sparse generator network; **DG:** dense generator network.

as autonomous vehicles where depth sensing technologies such LiDAR are commonly utilised. When depth measurements from such sensors are projected into the camera image space, the available scene depth information accounts for approximately 4% of the image pixels [54].

To improve the applicability of such sparse depth measurements, a growing number of novel approaches attempt to estimate dense depth based on the available sparse information. In [54], sparse convolutions are proposed with input normalisations in mind to take data sparsity into account while training a convolutional neural network. An end-to-end regression model is introduced in [40] to address the problem of sparse depth completion. [16] proposes a constrained convolution operation from which confidence values are propagated through the network. A compressed sensing approach in [10] utilises a binary mask to filter out unmeasured values in a depth completion framework. The approach in [39] addresses depth completion by employing a self-supervised training procedure based on sequential RGB and sparse depth images. In [50], a network is proposed that fuses contextual cues learned from RGB and sparse depth inputs to produce dense depth outputs.

Even though sparse depth completion is not the primary objective of this work, our approach is capable of generating dense depth from a sparse input along with its primary function (monocular depth estimation) and can outperform a variety of prior related work [10, 16, 40, 50, 54].

3. Proposed Approach

The approach proposed here is capable of performing two tasks within a single joint model, monocular depth estimation and sparse depth completion. This has been made possible using two publicly available datasets:- a depth completion dataset based on real-world images [54], in which relatively dense ground truth depth is created by accumulating measurements made by several laser scans with further consistency enforced between laser scans and stereo reconstruction [24]; and a synthetic dataset of images captured from a graphically-rendered virtual environment designed for urban driving scenarios [17].

The primary reason for using synthetic images [17] during training is that despite the increased depth density of the real-world imagery [54], depth information for the majority of the scene is still missing, leading to undesirable artefacts in regions where depth values are not available. A naïve solution would be to only use synthetic data to resolve the issue, but due to differences in the data domains, a model only trained on synthetic data cannot be expected to perform well on real-world images without domain adaptation [5, 63]. Consequently, we opt for randomly sampling training images from both datasets to force the overall model to capture the underlying distribution of both data domains, and therefore, learn the full dense structure of a synthetic scene while simultaneously modelling the contextual complexity of the naturally-sensed real-world images.

While the entirety of our approach can be considered a single generative model (G) that predicts full depth, as seen in Figure 2, it is composed of two stages. Each stage relies on a separate sub-network, both trained end to end. Based on the input RGB image, the first network, called the sparse generator (Figure 2 - SG), generates a sparse depth image (with non-valid pixels simply set to zero), which the second network, dense generator (Figure 2 - DG), subsequently uses to produce the final dense depth output.

3.1. Stage 1 - Generating Sparse Depth

Our sparse generator (SG) network produces its output by solving an image-to-image translation problem, in which an RGB image is translated to a sparse depth output. More formally, our first generative model (SG) encapsulates a mapping function that takes x (RGB image) as its input and outputs y_s (sparse depth) $SG : x \rightarrow y_s$. This can be done by minimising the Euclidean distance between the pixel values of the output ($SG(x)$) and the sparse ground truth (y_s). Such a reconstruction objective encourages the model to learn the structural composition of the scene by extracting lower-level features and estimating depth in a constrained window in the scene. This loss is therefore as follows:

$$\mathcal{L}_{rec_{SG}} = ||SG(x) - y_s||_1 \quad (1)$$

where x is the input RGB image, $SG(x)$ is the output and y_s

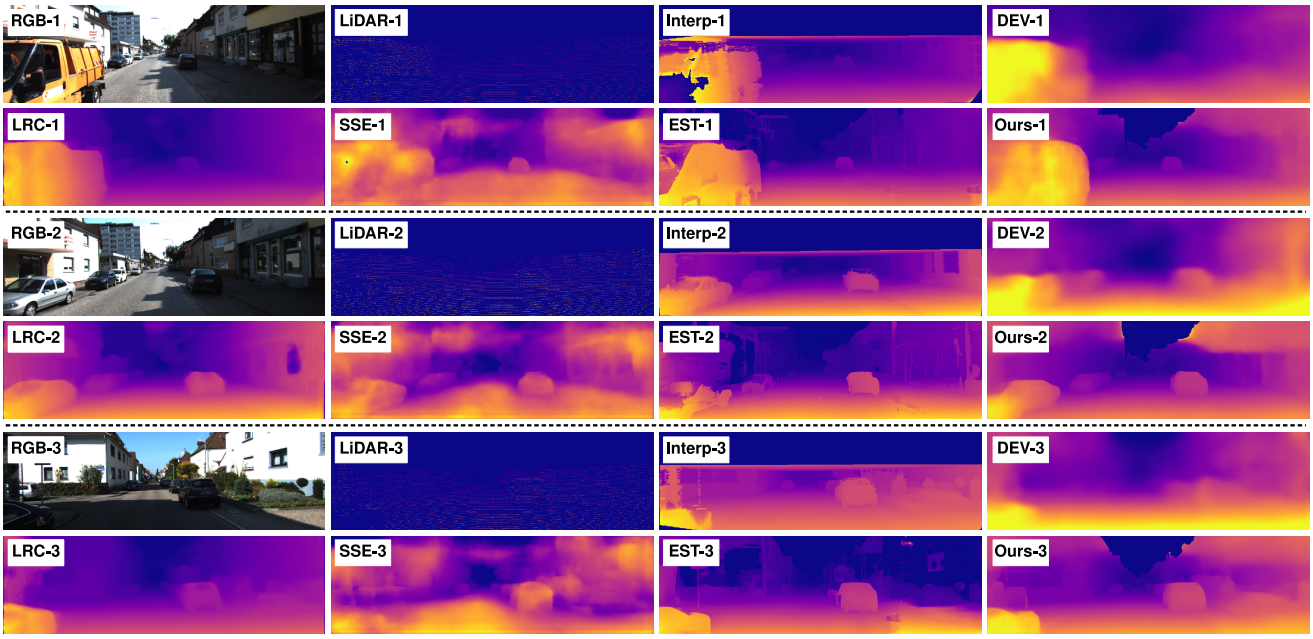

Figure 3: Comparing the results of our monocular depth estimation approach against [5, 20, 31, 66]. Adjusted disparity images are included for better visibility. **RGB:** input colour image; **DEV:** depth and ego-motion from video [66]; **LRC:** left-right consistency [20]; **SSE:** semi-supervised estimation [31]; **EST:** estimation via style transfer [5].

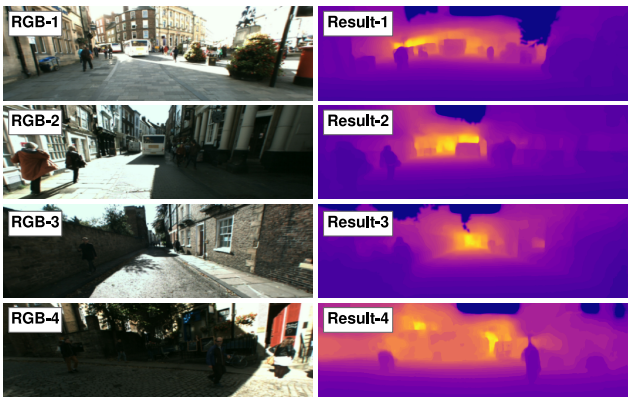

Figure 4: Demonstrating the generalisation capabilities of the approach using data captured locally from Durham, UK.

the ground truth sparse depth. However, since our training data is randomly sampled from synthetic and real-world images and no sparse ground truth depth is available in the synthetic dataset, it needs to be artificially created. While this could be achieved by training a separate network to predict which pixel values would exist in the sparse depth image (based on the details of the semantic scene objects such as their colour or reflectance qualities), a simpler and just as effective method would be to generate sparse synthetic depth based on a randomly selected sparse depth image from the real-world dataset.

Consequently, before the loss function in Eqn. 1 is calculated for a *synthetic* image, the ground truth sparse depth for said image is generated as follows:

$$y_{s_S}(p) = \begin{cases} 0, & \text{for } y_{s_R}(p) = 0 \\ y_{d_S}(p), & \text{for } y_{s_R}(p) \neq 0 \end{cases} \quad (2)$$

where y_{s_S}, y_{d_S} and y_{s_R} are sparse synthetic depth, dense synthetic depth and sparse real depth images respectively and p denotes the image pixel index. The output of our sparse generator network is subsequently passed to the second sub-network to produce the final result.

3.2. Stage 2 - Generating Dense Depth

In this stage, our dense generator (DG) network uses the output of the sparse generator $SG(x)$ as its input and generates y_d (dense depth image) $DG : SG(x) \rightarrow y_d$. To ensure that the overall model produces structurally and contextually sound dense depth outputs, a second reconstruction loss component is introduced to ensure the similarity of the final result to the ground truth dense depth:

$$\mathcal{L}_{rec_{DG}} = ||DG(SG(x)) - y_d||_1 \quad (3)$$

where x and y_d are the input RGB and dense ground truth depth images respectively. The issue with this loss component arises from the use of synthetic and real-world training data together. While synthetic images are complete and without missing values (except for where necessary, *e.g.* sky and distant objects), real-world ground truth dense depth images from [54] still contain large missing regions. As a result, the reconstruction loss used in Eqn. 3 needs to be reformulated to account for missing values in the real-world

Method		Error Metrics (lower, better)				Accuracy Metrics (higher, better)		
		Abs. Rel.	Sq. Rel.	RMSE	RMSE log	$\sigma < 1.25$	$\sigma < 1.25^2$	$\sigma < 1.25^3$
Train Set Mean	[19]	0.403	0.530	8.709	0.403	0.593	0.776	0.878
Eigen *et al.*	[14]	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Liu *et al.*	[36]	0.202	1.614	6.523	0.275	0.678	0.895	0.965
Zhou *et al.*	[66]	0.208	1.768	6.856	0.283	0.678	0.885	0.957
Atapour *et al.*	[7]	0.193	1.438	5.887	0.234	0.836	0.930	0.958
Godard *et al.*	[20]	0.148	1.344	5.927	0.247	0.803	0.922	0.964
Zhan *et al.*	[62]	0.144	1.391	5.869	0.241	0.803	0.928	0.969
Atapour *et al.*	[5]	0.110	0.929	4.726	0.194	0.923	0.967	0.984
Our Approach		**0.080**	**0.836**	**4.437**	**0.157**	**0.929**	**0.970**	**0.985**

Table 1: Numerical comparison of our monocular depth estimation approach over data from [19] using the data split in [14].

ground truth dense depth data. For this purpose, a binary mask, M, is created to indicate which pixel values are missing from the ground truth dense depth:

$$M(p) = \begin{cases} 0, & \text{for } y_{d_R}(p) = 0 \\ 1, & \text{for } y_{d_R}(p) \neq 0 \end{cases} \quad (4)$$

Using this binary mask, Eqn. 3 is subsequently reformulated for *real-world* images as follows:

$$\mathcal{L}_{rec_{DG}} = ||M \odot DG(SG(x)) - y_{d_R}||_1 \quad (5)$$

where $\odot$ is the element-wise product operation. Since Eqn. 3 is used for synthetic images, the model will learn the full structure and context of the scene when the entirety of the scene is available, and at the same time, it will learn to ignore missing regions from real-world images using Eqn. 5.

With depth prediction being an ill-posed problem (several plausible depth outputs can correctly correspond to an RGB image), exclusively using a reconstruction loss would result in blurry outputs since our overall generative model, G, (consisting of both sparse and dense networks) tends to average all possible solutions rather than selecting one, leading to blurring effects. Adversarial training [21] can offer a solution to this problem [5, 12, 25, 61] since it pushes the model towards selecting single values from the distribution resulting in higher fidelity outputs. Consequently, our overall model (G) takes x as its input and outputs fake samples $G(x) = \tilde{y}_d$ while a discriminator (D) is adversarially trained to distinguish fake samples $\tilde{y}_d$ from ground truth samples y_d. The adversarial loss is thus as follows:

$$\mathcal{L}_{adv} = \min_G \max_D \mathop{\mathbb{E}}_{x,y_d \sim \mathbb{P}_d(x,y_d)} [logD(x, y_d)] + \mathop{\mathbb{E}}_{x \sim \mathbb{P}_d(x)} [log(1 - D(x, G(x)))] \quad (6)$$

where $\mathbb{P}_d$ is the data distribution defined by $\tilde{y}_d = G(x)$, with x being the generator input and y_d the ground truth. In our approach, we have opted for using two separate discriminators D_S and D_R for synthetic and real-world images respectively (using similar loss as per Eqn. 6 with the overall adversarial loss being $\mathcal{L}_{adv} = \mathcal{L}_{adv_S} + \mathcal{L}_{adv_R}$). In our experiments, using a single discriminator for both data types led to stability and convergence issues during training.

In addition to this, a smoothing term [20, 23] is used to force the model to produce more locally-smooth dense depth results. Depth gradients ($\partial G(x)$) are penalised using L_1 regularisation, and an edge-aware weighting term based on input image gradients (∂x) is used to produce smoother depth outputs since image gradients are stronger where depth discontinuities are most likely. The smoothing loss is thus calculated as follows:

$$\mathcal{L}_s = |\partial G(x)| e^{||\partial x||} \quad (7)$$

where x is the input RGB image and $G(x)$ the output of the overall model. The gradients are summed over vertical and horizontal axes. It is important to note that all the loss components introduced in this section are not only used to train the dense generator (DG) sub-network but the gradients of these loss functions are used to train the entire network end to end, including the sparse generator (SG) sub-network. The overall loss function is therefore as follows:

$$\mathcal{L} = \lambda_{rec_{SG}} \mathcal{L}_{rec_{SG}} + \lambda_{rec_{DG}} \mathcal{L}_{rec_{DG}} + \lambda_{adv} \mathcal{L}_{adv} + \lambda_s \mathcal{L}_s \quad (8)$$

where the weighting coefficients (λ) are empirically selected (Section 3.3). While the overall model can be used for monocular depth estimation, the second sub-network (DG) can be used alone as a sparse depth completion network since it takes a sparse depth image as its input and can produce an accurate dense depth image as its output.

3.3. Implementation Details

Our sparse generator follows an encoder/decoder architecture with every layer containing modules of convolution, BatchNorm and leaky ReLU ($slope = 0.2$) with skip connections [46] between every pair of corresponding layers in the encoder and the decoder (Figure 2 - SG). The

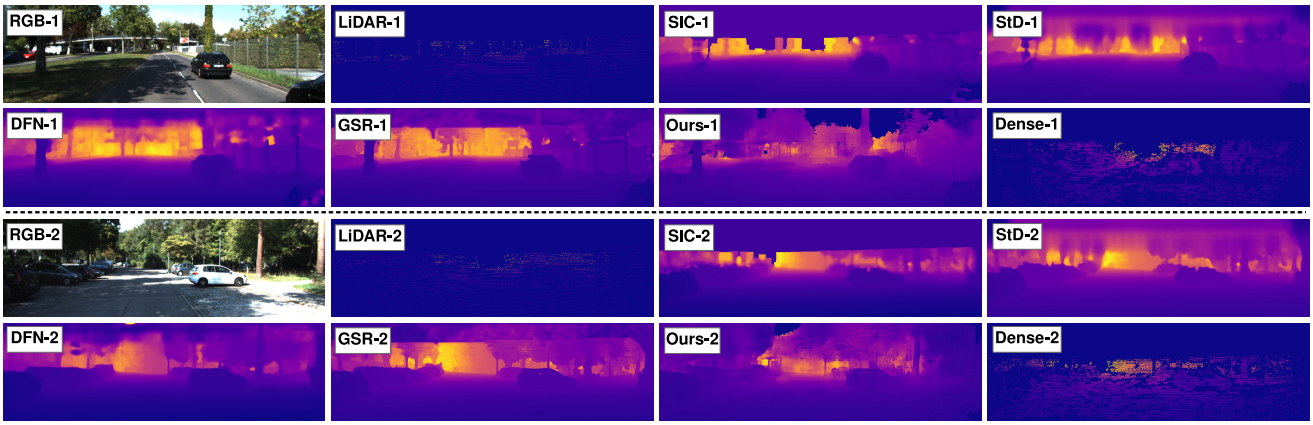

Figure 5: Comparing our depth completion results against [15, 40, 54, 50]. The depth images have been adjusted for better visualization. **RGB:** input colour image; **LiDAR:** sparse depth input; **SIC:** sparsity invariant CNN [54]; **StD:** sparse to dense completion [40]; **DFN:** deep fusion network [50]; **GSR:** guided sparse regression [15]; **Dense:** dense ground truth from [54].

dense generator follows a somewhat similar architecture, save that in its encoder, residual blocks [22] form each layer, with the output from each passed to corresponding decoding layers via skip connections. Both discriminators contain an architecture similar to that of [45] with each layer including the same modules as those in the generators. All implementation is done in *PyTorch* [43], with Adam [29] providing the best optimization ($\beta_1 = 0.5$, $\beta_2 = 0.999$, $\alpha = 0.0001$). The weighting coefficients in the loss function are empirically chosen, using a basic grid search, to be $\lambda_{rec_{SG}} = 150, \lambda_{rec_{DG}} = 100, \lambda_{adv} = 10, \lambda_s = 1$.

4. Experimental Results

To rigorously evaluate our approach, we conduct extensive ablation studies and both qualitative and quantitative comparisons with state-of-the-art methods applied to publicly available datasets [19, 54]. We additionally make use of randomly selected synthetic test images [17] and data captured locally to further evaluate the approach. It is worth noting that using a GeForce GTX 1080 Ti, the entire two passes for monocular depth estimation take an average of 33.4 milliseconds and a single pass through the dense generator for depth completion takes 18.1 milliseconds.

4.1. Monocular Depth Estimation

As the primary focus of our proposed approach, our monocular depth estimation model is evaluated against contemporary state-of-the-art approaches [5, 14, 20, 36, 62, 66]. Following the conventions of the existing literature, we use the data split suggested in [14] as the test set.

As seen in Table 1, our approach numerically outperforms all comparators across all metrics, mainly due to the superior scene representation learned by the model. It has been established within the literature that de-noising and completion tasks can lead to learning more robust features and a deeper representation of the scene [1, 44, 56, 57].

Method		Error Metrics (lower, better)	
		RMSE $[mm]$	MAE $[mm]$
Uhrig *et al.*	[54]	1729	503
Chodosh *et al.*	[10]	1431	460
Eldesokey *et al.*	[16]	1370	377
Shivakumar *et al.*	[50]	1303	446
Eldesokey *et al.*	[15]	909	**210**
Ma *et al.*	[39]	879	261
Van Gansbeke *et al.*	[55]	**802**	214
Our Approach		892	243

Table 2: Comparison of our depth completion approach against [10, 15, 16, 39, 50, 54, 55] using the validation set in [54]. Despite not being the primary focus, our completion approach remains competitive with the state of the art.

Since a portion of our model (DG) attempts to complete sparse depth information from the scene, a better understanding of the scene geometric content and semantic context is encapsulated within the model, aiding the approach to not only gain a secondary capability to preform sparse depth completion, but also to perform the primary function of monocular depth estimation more effectively. Qualitative results illustrated in Figure 3 also point to the same conclusions. Not only can our approach generate more accurate depth for the entire scene, it does so without undesirable artefacts such as blurring or bleeding effects. As seen in Figure 3, the object boundaries in the results are sharp and crisp, even for more distant scene components.

Additionally, to test the generalisation capabilities of our approach, we apply our model to previously unseen data captured locally in an urban environment in the city of Durham, UK. As seen in Figure 4, despite significant differences between the environmental conditions of the training data [17, 19] and those of the locally captured test data,

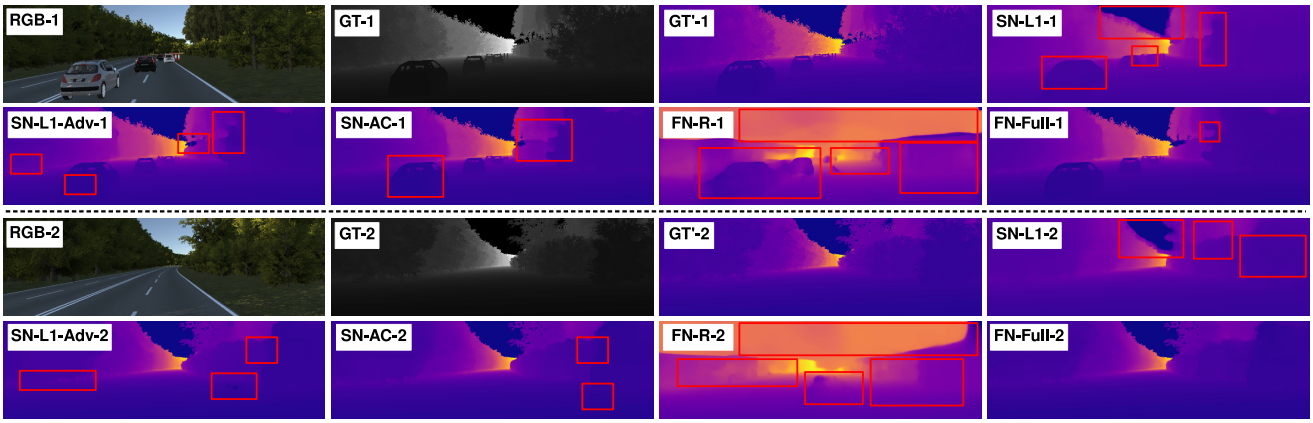

Figure 6: Comparing the performance of the approach using a randomly selected set of synthetic test images with differing components of the full monocular depth estimation model removed. **SN:** single network (sparse generator architecture); **FN:** full network (sparse and dense generators); $\mathbf{L_1}$**:** reconstruction loss component; **Adv:** adversarial loss component; **AC:** all loss components; **R:** real-world data only used for training. For clarity, errors in the images are signified by red boxes.

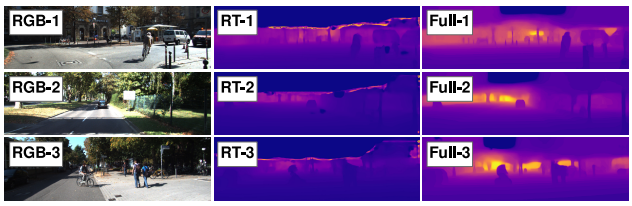

Figure 7: Demonstrating the importance of using synthetic and real-world data for training using test data from [54]. **RT:** real-world images used only; **Full:** full combined dataset of synthetic and real-world data used for training.

such as lighting, saturation levels, style, overall shape of the urban environment and alike, the results contain minimal anomalies, are sharp, crisp and very convincing with well-preserved object boundaries and thin structures.

4.2. Sparse Depth Completion

While sparse depth completion is not the primary focus of this work we attempt to extensively evaluate this part of our approach using the publicly available validation dataset in [54] to enable better reproducibility.

As seen in Figure 5, due to the use of dense synthetic depth for training and improved scene representation learned by the model, our depth completion approach is capable of predicting depth in the entire scene and visually outperforms all comparators [15, 40, 50, 54]. Since the upper regions of the available ground truth images in [54] are missing, all comparators are completely incapable of predicting reasonable depth values for said regions and synthesise erroneous degenerate content. While our approach is certainly not entirely immune to this issue (Figure 8), it produces visually improved outputs compared to the other techniques, as seen in Figure 5. Numerical results in Table 2 demonstrate that our completion approach quantitatively outperforms many contemporary state-of-the-art completion methods [10, 16, 50, 54] and remains competitive with others [15, 39, 55], despite the fact that it is primarily incorporated into our pipeline to improve the main functionality of the approach (monocular depth estimation) and lacks the complex training objectives of many of the comparators.

4.3. Ablation Study

To demonstrate the importance of every component of the proposed approach, we re-train our model as varying components of the loss function and the overall approach are removed. It is intuitively expected that using two networks (SG and DG) trained to carry out different stages of a task will lead to better performance than when a single network with half the depth of the architecture is used. We experimentally illustrate this by training a single network with the architecture of the SG to regress to the full dense depth and perform monocular depth estimation in a single pass through the network. Additionally, we remove the components of the loss function to evaluate the influence they have over the performance of the approach. As seen in Table 3, the numerical results of experiments over a randomly selected set of synthetic test images (chosen over real images due to their higher level of density) indicate that the model performs better when all elements of the loss function are used during training (SN/AC) and that our full architecture and training procedure (Full Approach) outperforms a single sub-network (SN) by a large margin.

Another important aspect of our approach is incorporating synthetic data into the training process. To evaluate the necessity of this, the full model is trained using real-world data [54] only (FN/R) and the results point to superiority of the joint synthetic/real training data (Table 3). Qualitative results in Figure 6 also indicate that our full approach with the complete architecture trained on the mixed dataset using the full loss function (Figure 6 - FN-Full) outperforms

Method	Error (lower, better)				Accuracy Metrics (higher, better)		
	Abs. Rel.	Sq. Rel.	RMSE	RMSE log	$\sigma < 1.25$	$\sigma < 1.25^2$	$\sigma < 1.25^3$
SN/L_1	0.147	1.319	5.810	0.249	0.821	0.933	0.959
SN/L_1/Adv	0.115	1.122	5.128	0.221	0.898	0.862	0.977
SN/AC	0.108	0.982	4.911	0.198	0.913	0.962	0.980
FN/R	0.286	1.652	6.328	0.298	0.701	0.822	0.958
Full Approach	**0.075**	**0.829**	**4.212**	**0.143**	**0.951**	**0.979**	**0.991**

Table 3: Numerical results with different components of the monocular depth estimation approach. **SN:** single network (sparse generator architecture); **FN:** full network (sparse and dense generators); **L_1:** reconstruction loss component; **Adv:** adversarial loss component; **AC:** all loss components; **R:** real-world data only used for training.

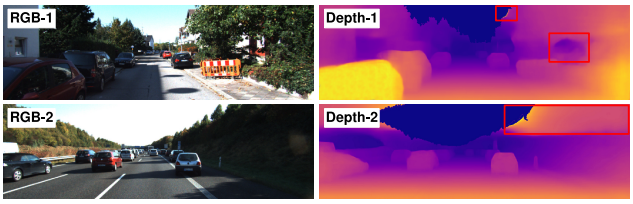

Figure 8: Examples of the limitations of the approach (red).

all ablated versions. Since for these experiments, the test images are chosen from the synthetic dataset [17], but a portion of our ablation studies (FN-R) focuses on the use of real data only, we also evaluated the approach on real-world images [19], and as seen in Figure 7, utilising a mixed dataset is more effective than the the use of real images only, even if the test images are selected from the latter.

5. Discussions and Future Work

Even though the use of both synthetic and real-world training data results in better depth predictions, the relatively sparse and flawed real-world ground truth depth images push the underlying model distribution towards degenerate content where depth values are unknown. Examples of such issues can be seen in Figure 8, where despite most scene components having correctly been discerned by the model (even the sky), the upper regions still contain incoherent content. Such issues can potentially be addressed in any future work by adding a weighted loss component that can penalise the generator when content is wrongly synthesised in the approximate regions where sky and other distant objects with no depth values are likely found. Additionally, by calculating confidence values for the generated output or propagating confidences through every convolution operation within the network [15, 16, 28], this and many other issues such as anomalies and artefacts can be resolved.

Moreover, while our approach performs both tasks plausibly, it can benefit from improvements in its training procedure. Even though splitting the overall objective of the approach into two stages performed by two sub-networks has led to more robust features within the model and thus its improved results, significant enhancements can be made to the performance by taking advantage of the abundance of information available within the training data [17, 19]. Inspired by [39, 66] and using the sequential order of frames available in [17, 19], photometric transformations and temporal continuity can provide highly beneficial supervisory signals to enforce a deeper contextual learning of the scene.

6. Conclusion

Here, we propose a multi-task model that can perform two fundamental scene understanding tasks:- sparse depth completion and monocular depth estimation. This is accomplished using two sub-networks jointly trained on a mixture of publicly available synthetic [17] and natural real-world [54] training data from urban driving scenarios. The first network within the overall pipeline attempts to regress to a sparse depth image, not unlike those generated by projecting depth measurements captured via a LiDAR sensor into image space. This sparse depth output produced by the first sub-network is subsequently passed into the second sub-network which generates a full dense depth image of the entire scene.

The low-level feature extraction and high-level inferences carried out by these two networks lead to better representation learning within the model and consequently its superior performance. Additionally, the entire model can be used to perform monocular depth estimation or, alternatively, the second sub-network can be utilised alone to carry out sparse depth completion. Using adversarial training, a deep architecture with skip connections and a blend of synthetic and real-world training data to guarantee the accuracy and density of the depth output, our approach can produce high quality scene depth. Our extensive experimental evaluation demonstrates the efficacy of our approach compared to contemporary state-of-the-art methods across both domains of monocular depth estimation [5, 7, 14, 20, 31, 36, 62, 66] and sparse depth completion [10, 16, 40, 50, 54].

*We kindly invite the readers to refer to the **video**: https://vimeo.com/351624727 for more information and larger improved-quality result images and video sequences.*

References

[1] A. Atapour-Abarghouei, S. Akcay, G. Payen de La Garanderie, and T. Breckon. Generative adversarial framework for depth filling via Wasserstein metric, Cosine transform and domain transfer. *Pattern Recognition*, 91:232–244, 2019. 2, 6

[2] A. Atapour-Abarghouei and T. Breckon. DepthComp: Real-time depth image completion based on prior semantic scene segmentation. In *British Machine Vision Conference*, pages 1–13. BMVA, 2017. 2

[3] A. Atapour-Abarghouei and T. Breckon. A comparative review of plausible hole filling strategies in the context of scene depth image completion. *Computers and Graphics*, 72:39–58, 2018. 1, 2

[4] A. Atapour-Abarghouei and T. Breckon. Extended patch prioritization for depth filling within constrained exemplar-based RGB-D image completion. In *Int. Conf. Image Analysis and Recognition*, pages 306–314, 2018. 2

[5] A. Atapour-Abarghouei and T. Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 2800–2810, 2018. 1, 2, 3, 4, 5, 6, 8

[6] A. Atapour-Abarghouei and T. Breckon. Monocular segment-wise depth: Monocular depth estimation based on a semantic segmentation prior. In *Int. Conf. Image Processing*, 2019. 1

[7] A. Atapour-Abarghouei and T. Breckon. Veritatem dies aperit-temporally consistent depth prediction enabled by a multi-task geometric and semantic scene understanding approach. In *IEEE Conf. Computer Vision and Pattern Recognition*, 2019. 2, 5, 8

[8] A. Atapour-Abarghouei, G. Payen de La Garanderie, and T. P. Breckon. Back to Butterworth - a Fourier basis for 3D surface relief hole filling within RGB-D imagery. In *Int. Conf. Pattern Recognition*, pages 2813–2818. IEEE, 2016. 1, 2

[9] W. Chen, H. Yue, J. Wang, and X. Wu. An improved edge detection algorithm for depth map inpainting. *Optics and Lasers in Engineering*, 55:69–77, 2014. 2

[10] N. Chodosh, C. Wang, and S. Lucey. Deep convolutional compressed sensing for LiDAR depth completion. *arXiv preprint arXiv:1803.08949*, 2018. 1, 2, 3, 6, 7, 8

[11] L. Ding and G. Sharma. Fusing structure from motion and lidar for dense accurate depth map estimation. In *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, pages 1283–1287. IEEE, 2017. 1

[12] A. Dosovitskiy and T. Brox. Generating images with perceptual similarity metrics based on deep networks. In *Advances in Neural Information Processing Systems*, pages 658–666, 2016. 5

[13] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Int. Conf. Computer Vision*, pages 2650–2658, 2015. 2

[14] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*, pages 2366–2374, 2014. 1, 2, 5, 6, 8

[15] A. Eldesokey, M. Felsberg, and F. S. Khan. Confidence propagation through CNNs for guided sparse depth regression. *arXiv preprint arXiv:1811.01791*, 2018. 6, 7, 8

[16] A. Eldesokey, M. Felsberg, and F. S. Khan. Propagating confidences through CNNs for sparse data regression. *arXiv preprint arXiv:1805.11913*, 2018. 1, 2, 3, 6, 7, 8

[17] E. Francis, K. Theodora, H. Alexander, and L. Bastian. Exploring spatial context for 3D semantic segmentation of point clouds. In *IEEE Int. Conf. Computer Vision Workshop*, 2017. 1, 2, 3, 6, 8

[18] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao. Deep ordinal regression network for monocular depth estimation. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 2002–2011, 2018. 1

[19] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *Robotics Research*, pages 1231–1237, 2013. 1, 2, 5, 6, 8

[20] C. Godard, O. M. Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 6602 – 6611, 2017. 1, 2, 4, 5, 6, 8

[21] 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*, pages 2672–2680, 2014. 2, 5

[22] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 770–778, 2016. 6

[23] P. Heise, S. Klose, B. Jensen, and A. Knoll. Pm-huber: Patchmatch with huber regularization for stereo matching. In *Int. Conf. Computer Vision*, pages 2360–2367, 2013. 5

[24] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2007. 3

[25] P. Isola, J.-Y. Zhu, T. Zhou, and A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 5967–5976, 2017. 5

[26] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015. 2

[27] K. Karsch, C. Liu, and S. B. Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 36(11):2144–2158, 2014. 2

[28] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 1–10, 2018. 8

[29] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Int. Conf. Learning Representations*, pages 1–15, 2014. 6

[30] M. Kulkarni and A. Rajagopalan. Depth inpainting by tensor voting. *J. Optical Society of America A*, 30(6):1155–1165, 2013. 2

[31] Y. Kuznietsov, J. Stückler, and B. Leibe. Semi-supervised deep learning for monocular depth map prediction. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 6647–6655, 2017. 1, 2, 4, 8

[32] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *Int. Conf. 3D Vision*, pages 239–248, 2016. 2

[33] B. Li, C. Shen, Y. Dai, A. van den Hengel, and M. He. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 1119–1127, 2015. 2

[34] B. Liu, S. Gould, and D. Koller. Single image depth estimation from predicted semantic labels. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 1253–1260, 2010. 2

[35] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 33(5):978–994, 2010. 2

[36] F. Liu, C. Shen, G. Lin, and I. Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 38(10):2024–2039, 2016. 2, 5, 6, 8

[37] M. Liu, X. He, and M. Salzmann. Building scene models by completing and hallucinating depth and semantics. In *Euro. Conf. Computer Vision*, pages 258–274, 2016. 2

[38] M. Liu, M. Salzmann, and X. He. Discrete-continuous depth estimation from a single image. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 716–723, 2014. 2

[39] F. Ma, G. V. Cavalheiro, and S. Karaman. Self-supervised Sparse-to-Dense: Self-supervised depth completion from LiDAR and monocular camera. *arXiv preprint arXiv:1807.00275*, 2018. 1, 3, 6, 7, 8

[40] F. Ma and S. Karaman. Sparse-to-Dense: Depth prediction from sparse depth samples and a single image. In *IEEE Int. Conf. Robotics and Automation*, pages 1–8. IEEE, 2018. 2, 3, 6, 7, 8

[41] K. Matsuo and Y. Aoki. Depth image enhancement using local tangent plane approximations. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 3574–3583, 2015. 2

[42] E. Orhan and X. Pitkow. Skip connections eliminate singularities. In *Int. Conf. Learning Representations*, pages 1–11, 2018. 2

[43] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in PyTorch. In *Advances in Neural Information Processing Systems*, pages 1–4, 2017. 6

[44] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. Efros. Context encoders: Feature learning by inpainting. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 2536–2544, 2016. 6

[45] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 6

[46] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Int. Conf. Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015. 2, 5

[47] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *Advances in Neural Information Processing Systems*, pages 1161–1168, 2006. 2

[48] A. Saxena, M. Sun, and A. Y. Ng. Make3D: Learning 3D scene structure from a single still image. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 31(5):824–840, 2008. 2

[49] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Computer Vision*, 47:7–42, 2002. 1

[50] S. S. Shivakumar, T. Nguyen, I. D. Miller, S. W. Chen, and C. J. Taylor. DFuseNet: Deep fusion of RGB and sparse depth information for image guided dense depth completion. *https://arxiv.org/pdf/1902.00761.pdf*, 2019. 2, 3, 6, 7, 8

[51] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from RGB-D images. In *Euro. Conf. Computer Vision*, pages 746–760. Springer, 2012. 2

[52] S. Song, S. P. Lichtenberg, and J. Xiao. Sun RGB-D: A RGB-D scene understanding benchmark suite. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 567–576, 2015. 2

[53] M. Tao, P. Srinivasan, J. Malik, S. Rusinkiewicz, and R. Ramamoorthi. Depth from shading, defocus, and correspondence using light-field angular coherence. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 1940–1948, 2015. 1

[54] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger. Sparsity invariant CNNs. In *Int. Conf. 3D Vision*, pages 11–20. IEEE, 2017. 2, 3, 4, 6, 7, 8

[55] W. Van Gansbeke, D. Neven, B. De Brabandere, and L. Van Gool. Sparse and noisy LiDAR completion with RGB guidance and uncertainty. *arXiv preprint arXiv:1902.05356*, 2019. 1, 6, 7

[56] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Int. Conf. Machine learning*, pages 1096–1103. ACM, 2008. 6

[57] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Machine Learning Research*, 11(Dec):3371–3408, 2010. 6

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

[59] J. Xie, R. Girshick, and A. Farhadi. Deep3D: Fully automatic 2D-to-3D video conversion with deep convolutional neural networks. In *Euro. Conf. Computer Vision*, pages 842–857, 2016. 2

[60] H. Xue, S. Zhang, and D. Cai. Depth image inpainting: Improving low rank matrix completion with low gradient regularization. *IEEE Trans. Image Processing*, 26(9):4311–4320, 2017. 2

[61] R. Yeh*, C. Chen*, T. Y. Lim, S. Alexander, M. Hasegawa-Johnson, and M. Do. Semantic image inpainting with deep

generative models. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 6882–6890, 2017. 5

[62] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. Reid. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 340–349, 2018. 2, 5, 6, 8

[63] Y. Zhang and T. Funkhouser. Deep depth completion of a single RGB-D image. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 175–185, 2018. 2, 3

[64] S. Zhao, H. Fu, M. Gong, and D. Tao. Geometry-aware symmetric domain adaptation for monocular depth estimation. *arXiv preprint arXiv:1904.01870*, 2019. 2

[65] C. Zheng, T.-J. Cham, and J. Cai. T2Net: Synthetic-to-realistic translation for solving single-image depth estimation tasks. In *Euro. Conf. Computer Vision*, pages 767–783. Springer, 2018. 2

[66] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *IEEE Conf. Computer Vision and Pattern Recognition*, pages 6612–6619, 2017. 1, 2, 4, 5, 6, 8