Robust Multimodal Depth Estimation using Transformer based Generative Adversarial Networks

Md Fahim Faysal Khan
The Pennsylvania State University
University Park, PA, USA
mzk591@psu.edu

Anusha Devulapally
The Pennsylvania State University
University Park, PA, USA
akd5994@psu.edu

Siddharth Advani
Samsung Electronics America
Plano, TX, USA
s.advani@samsung.com

Vijaykrishnan Narayanan
The Pennsylvania State University
University Park, PA, USA
vijaykrishnan.narayanan@psu.edu

ABSTRACT

Accurately measuring the absolute depth of every pixel captured by an imaging sensor is of critical importance in real-time applications such as autonomous navigation, augmented reality and robotics. In order to predict dense depth, a general approach is to fuse sensor inputs from different modalities such as LiDAR, camera and other time-of-flight sensors. LiDAR and other time-of-flight sensors provide accurate depth data but are quite sparse, both spatially and temporally. To augment missing depth information, generally RGB guidance is leveraged due to its high resolution information. Due to the reliance on multiple sensor modalities, design for robustness and adaptation is essential. In this work, we propose a transformer-like self-attention based generative adversarial network to estimate dense depth using RGB and sparse depth data. We introduce a novel training recipe for making the model robust so that it works even when one of the input modalities is not available. The multi-head self-attention mechanism can dynamically attend to most salient parts of the RGB image or corresponding sparse depth data producing the most competitive results. Our proposed network also requires less memory for training and inference compared to other existing heavily residual connection based convolutional neural networks, making it more suitable for resource-constrained edge applications. The source code is available at: *https://github.com/kocchop/robust-multimodal-fusion-gan*

CCS CONCEPTS

• **Computing methodologies** → **Reconstruction**; **Adversarial learning**; **Perception**; **Robustness**; • **Hardware** → *Sensor applications and deployments.*

KEYWORDS

Multimodal Sensing; Sensor Fusion; Depth Completion; Generative Adversarial Nertworks (GAN); Sensor Failure; Robustness

MM '22, October 10–14, 2022, Lisboa, Portugal

ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3548418

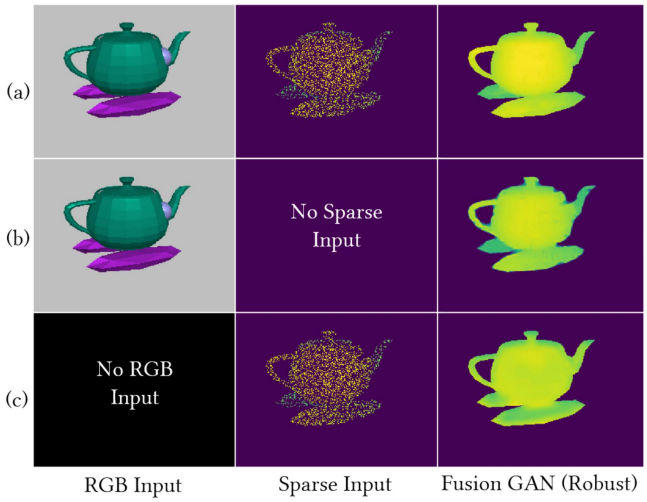

Figure 1: The figure reflects upon the model robustness due to sensor failures. In (a), both the RGB and sparse depth is present. However, in (b) and (c), one of the modalities is missing. Trained with our proposed novel recipe, a single model can produce reasonable outputs for all three scenarios.

ACM Reference Format:
Md Fahim Faysal Khan, Anusha Devulapally, Siddharth Advani, and Vijaykrishnan Narayanan. 2022. Robust Multimodal Depth Estimation using Transformer based Generative Adversarial Networks. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3503161.3548418

1 INTRODUCTION

Dense and accurate depth prediction is one of the most fundamental challenges for tasks such as autonomous vehicle guidance, 3D mapping and surveillance, scene understanding, assisting physically challenged people and augmented reality applications like modeling virtual environments such as digital twins [1, 2, 3, 4]. Forecasts predict that globally one in ten vehicles will be automated by 2030 [5]. Dense depth also helps in extracting better semantics, object detection and 3D object reconstruction. In order to obtain true depth information, generally, different time-of-flight (ToF) sensors such as LiDAR/RADAR (Light/Radio Detection and Ranging)

are widely used. ToF sensors can provide reliable depth data but are limited by heavy sparsity in their outputs. For example, LiDAR which is the most accurate and has the longest range provides only 32 or 64 depth scan lines at a frequency of 15-20 Hz [6]. Hence, predicting the dense depth map from such sparse data is a popular research topic in machine learning. Typically, in order to find the missing depth values, corresponding RGB images are utilized. RGB guidance can provide different contextual shape information that can be fused with the LiDAR data to obtain dense depth.

One of the most effective ways to find out contextual information is by using attention models [7, 8]. Attention based models have gained a lot of popularity lately due to their capability of finding out the most salient features, and drawing necessary context across the entire input space. This leads to much better performance. Dosovitskiy et al. [9] introduced multi-headed self attention based transformer model which obtains the state-of-the-art results on language tasks outperforming the existing recurrent network approaches. Later, they extended the same concept to an image classification task where an entire image is divided into a set of 16x16 sized patches. These patches were transformed into tokens much similar to word embeddings and achieved better performance compared to the state-of-the-art CNN based designs. One of the most fundamental aspect contributing to such good performance is their ability to extract the most relevant information required for the given task. In depth completion tasks, one preliminary step is to extract the depth of field information from RGB images for which transformers seem to be an ideal candidate.

While multimodal fusion ensures better reconstruction quality, it also makes the model conditioned upon the input modalities. When one of the sensor modalities undergoes some technical, physical or environmental hindrance, the whole model performance becomes vulnerable [10]. Hence, even though fusion is essential for better quality, some provision for robustness must also be present to tackle with such failures. One probable solution is to make the model robust during training by reproducing the absence of different sensor modalities.

Generative adversarial networks or GANs [11, 12, 13, 14] have gained a lot of popularity lately due to their better modeling capability of the input data distribution and is the state-of-the-art for image generation tasks [15]. Since the end product of depth completion task is usually a dense depth image, GANs are being adopted by different studies for obvious reasons [16]. In this study, we also deploy a GAN towards the task of dense depth reconstruction. To summarize, the key contributions of this paper are,

(1) We deploy a transformer based multimodal fusion network to recover dense depth from RGB and sparse depth information. Our proposed network achieves state-of-the-art performance compared to other concurrent multimodal fusion studies.
(2) We propose a novel training scheme to improve the model's tolerance to sensor availability. We empirically show that our trained network is more robust compared to other uni or multimodal fusion networks.
(3) Our proposed network has much less run time memory requirement in contrast with other heavily residual connection based models, thus making our model suitable for deployment in memory constrained systems.

2 RELATED WORKS

2.1 Unimodal Depth Reconstruction

Eigen et al. [17, 18] proposed a multi-scale deep network that employs two network stacks, in which one takes the RGB image and gives a coarse global depth prediction, and the latter uses the previous stack output and refines the depth prediction locally. Liu et al. [19] used continuous conditional random field (CRF) along with deep CNN to predict the depth from RGB images. Laina et al. [20] used a Residual Network for depth prediction on RGB images. Liu et al. [21] used wavelet-contour let dictionaries for accurate reconstructions. Hawe et al. [22] used a conjugate sub-gradient method to reconstruct the dense disparity image. Eldesokey et al. [23] used normalized convolution layers to calculate the confidence and propagate it to the consecutive layers for sparse input. Uhrig et al. [24] used a similar approach of calculating the validity mask and propagating it to get dense sparsity maps. HMS-Net [25] used sparsity-invariant operations with the multi-scale encoder-decoder network for handling sparse inputs and sparse feature maps. The performance of the model further increased with the addition of RGB guidance. Jartiz et al. [26], Lu et al. [27] proposed a encoder-decoder for depth completion from only sparse depth. Ku et al. [28] used the basic image-processing algorithms on sparse LiDAR data to get dense depth. Most recently, Khan et al. [29] built a GAN architecture that generates dense depth using only sparse depth from LiDAR. However, without using multiple modalities it is very difficult to get the most accurate dense depth maps.

2.2 Multimodal Fusion

Various efforts have shown the effectiveness of the combination of RGB and sparse depth input to infer dense depth information. To merge these two modalities, most of the techniques in the literature used a two-branch network. Hua et al. [30], Jaritz et al. [31] created an encoder-decoder architecture in which two modalities are encoded separately to extract features before being fused into a single decoder. To extract features, Tang et al. [32] employed a two-level encoder-decoder network for RGB and LiDAR, fusing the RGB decoder output with the sparse Depth encoder at each level. Fusion-Net [1] merges RGB and LiDAR data by extracting global and local information via two branches. DeepLiDAR [33] is composed of surface normal and color pathways. Each pathway fuses both modalities at different phases. To integrate RGB and sparse depth, Rig Net [34] uses a repetitive guidance technique. This method includes an image guidance network with repetitive hourglass networks that feed the output of the previous network into the current network, as well as a depth generation network with a single hourglass network that refines the predicted depth using repetitive guidance modules and an efficient guidance algorithm to produce refined depth step by step. Ma et al. [2] employs a deep regression model that predicts a full resolution depth picture using RGBD (4-channel) as input. PE-Net [35] employs two branch networks, one color-dominant and the other depth-dominant, to merge RGB and sparse depth maps into a dense depth map. The color-dominant branch uses RGB, and sparse depth maps to forecast a coarse depth map, which is then sent into the depth-dominant branch, which produces a dense depth map. They use CSPN++ [36]

(convolutional spatial propagation network) for depth map refinement. CSPN [37] is also another method for depth refinement and depth completion used with state of the art architectures. Zhang et al. [38] proposed a multi-task GAN for both semantic segmentation and depth completion. It includes the computed semantic images in addition to RGB and sparse depth to improve the dense depth output. Parallel to this, another line of work explores other modality based fusion to recover depth. For example, Parida et al. [39] uses binaural audio and RGB to predict the depth. They also use the echo signal to predict the material type which gets further fed to the attention network that weighs the depth maps generated from the echo and visual pipelines to produce the final depth map.

2.3 Attention Model in depth reconstruction

Prakash et al. [40] proposed a transformer-based model to fuse the RGB and LiDAR inputs as these samples complement each other while describing the scene. Both modalities are encoded separately, but intermediate features are fused at each level using a transformer. When the two modalities' results are added together, the global context of the 3D scene emerges. For monocular depth estimation, Rantfl et al. [41] employs transformers as encoders and convolution decoders, resulting in fine-grained predictions. It overcomes the limitations of down sampling with convolutions, such as granularity loss and feature resolution loss in deeper networks that are difficult to extract using a decoder.

The use of multiple sensor modalities with attention mechanism have been able to generate very close to true depth maps. However, most of the models are very much input dependent and hence cannot handle sudden sensor failures or unavailability. They lag behind in terms of robustness and reliability which is essential in practical scenario. In order to tackle this, recent approaches like [42] consider asymmetric distortion in different sensing modalities that they fuse. Complimentary to their approach our proposed training scheme is able make robust inference even when one of the sensor modalities is completely unavailable due to sensor malfunction or other challenges. For example, an RGB image in low light conditions or when the camera gets occluded. In such scenario, the system can disable the RGB modality and infer with only LiDAR. Similarly, the system can also leverage in case of malfunction of the LiDAR system to reduce expensive energy acquisition of this active sensing technique.

3 METHODOLOGY

In this section, we formulate the depth completion problem mathematically. We elaborate on the variation of our proposed task specific optimization rule with respect to typical GAN settings. Finally, we discuss the loss functions followed by the model architecture.

3.1 Problem Statement

In this study, we aim to solve the problem of dense depth prediction using multimodal sensor inputs. We leverage a generative adversarial network (GAN) [11] to solve it. A typical GAN consists of a **generator** block and a **discriminator** block where the generator delivers the desired output and the discriminator tries to detect if it is a counterfeit example. It is in essence a two player min-max game, where the training ends when the generator is able to make the discriminator believe that the generated samples are indeed real. In our case, we want the generator to produce dense depth samples from sparse depth data and a corresponding RGB image. Let P^{SP}, P^{DN} and P^{RGB} denote the input distributions of sparse depth, dense depth and the RGB image respectively. We consider the generator network G has a parameter set θ_G with a loss function L^G. If there are $n = 1, ..., N$ training samples for each of modalities, then we optimize to find $\hat{\theta}_G$:

$$\hat{\theta}_G = \arg\min_{\theta_G} \frac{1}{N} \sum_{n=1}^{N} L^G \left(G_{\theta_G} \left(P_n^{SP}, P_n^{RGB} \right), P_n^{DN} \right) \tag{1}$$

where $\hat{\theta}_G$ is the parameter set of the trained generator model. The loss function L^G consists of multiple weighted loss components designed to capture certain properties of the output dense depth map and is described in section 3.3. It is to be noted that unlike a typical GAN, our generator does not sample from a random distribution. We explain the reasons in the following section.

3.2 Conditional GAN settings

In an ideal setting, the generator samples from a random distribution in order to generate a real-like sample. However, Mirza et al. proposed that a GAN can be conditioned upon its particular inputs [43] and many studies reported state-of-the-art results adopting it [44, 45]. In this study, we also use a conditional GAN setting where we condition the network upon the input sparse depth and the corresponding RGB image. Hence, instead of sampling noise from a random distribution, our generator network takes the sparse depth and the RGB image as its input and we say the network to be conditioned upon such. If D represents the discriminator model, then our generator and discriminator function would solve the following min-max problem:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{P_{DN}} \sim P_{\text{train}}(\boldsymbol{P_{DN}})} [\log D(\boldsymbol{P_{DN}})] + \mathbb{E}_{\boldsymbol{P_{SP}} \sim P_G(\boldsymbol{P_{SP}}), \boldsymbol{P_{RGB}} \sim P_G(\boldsymbol{P_{RGB}})} [\log(1 - D(G(\boldsymbol{P_{RGB}}, \boldsymbol{P_{SP}})))] \tag{2}$$

With this approach, our generator is able to generate images very similar to original data distribution making it harder for the discriminator to distinguish between the real and fake samples. One other aspect of our GAN problem formulation that we want to particularly highlight is that unlike the generator, we do not condition the discriminator. Instead, we use a relativistic average discriminator [46], which is explained further in section 3.3.

3.3 Loss Functions

Since there are two networks which get simultaneously trained in a GAN, there are two loss functions, one for the generator and another for the discriminator. We follow prior works [29] to get suitable loss functions to train our GAN. The different loss components are discussed below briefly.

3.3.1 Discriminator Loss. For the discriminator, as mentioned before, we use a Relativistic average Discriminator (RaD) [46] which instead of comparing a binary label in a deterministic way, calculates the relative likelihood between the generated and the fake data. In other words, it finds out the probability of a generated sample being more realistic than the fake data or the probability of the

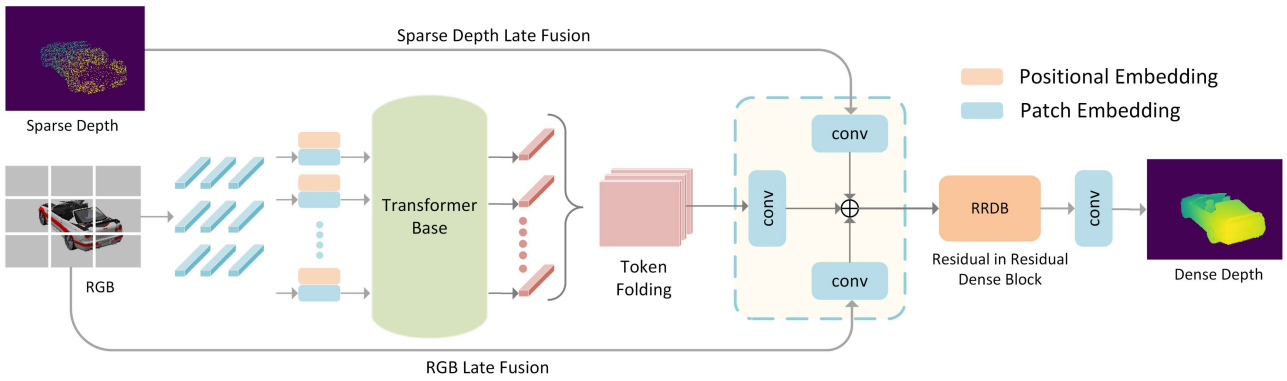

Figure 2: Transformer-based GAN architecture for Depth Completion. The RGB and sparse depth has separate late fusion pipelines. The transformer-based encoder outputs get folded first and then passed to the multimodal fusion block. The residual in residual dense block (RRDB) generates the final depth map.

generated sample being less realistic than the real data. Finally, the discriminator loss is obtained by taking an average of the two. Let us assume, $\hat{P}_{DN} = G_{\hat{\theta}_G}(P_{SP}, P_{RGB})$ be the distribution of generated dense depth maps. Then the discriminator loss, l^D_{RaD} can be written as,

$$l^D_{RaD} = -\mathbb{E}_{P^{DN}}\left[\log\left(D^{RaD}\left(P^{DN}, \hat{P}^{DN}\right)\right)\right] - \mathbb{E}_{\hat{P}^{DN}}\left[\log\left(1 - D^{RaD}\left(\hat{P}^{DN}, P^{DN}\right)\right)\right] \tag{3}$$

which automatically leads to its symmetrical adversarial loss for the generator:

$$l^G_{RaD} = -\mathbb{E}_{P^{DN}}\left[\log\left(1 - D^{RaD}\left(P^{DN}, \hat{P}^{DN}\right)\right)\right] - \mathbb{E}_{\hat{P}^{DN}}\left[\log\left(D^{RaD}\left(\hat{P}^{DN}, P^{DN}\right)\right)\right] \tag{4}$$

Finally, the discriminator loss, L^D can be written as the average of the two.

$$L^D = (l^G_{RaD} + l^G_{RaD})/2 \tag{5}$$

3.3.2 Generator Loss. The generator loss consists of three components, (1) the adversarial loss, (2) normal loss and (3) pixel loss. The adversarial loss is already described in Eq. 4. The normal loss,

$$l_{\text{normal}} = \frac{1}{n}\sum_{i=1}^{n}\left(1 - \frac{\langle \nabla_i^{DN}, \hat{\nabla}_i^{DN}\rangle}{\|\nabla_i^{DN}\|\left\|\hat{\nabla}_i^{DN}\right\|}\right) \tag{6}$$

where ∇^{DN}, $\hat{\nabla}^{DN}$ are the corresponding gradient vectors of ground truth P^{DN} and the prediction $\hat{P}^{DN}$. The $\langle\cdot\rangle$ denotes the dot product of the gradient vectors and $\|\cdot\|$ denotes the norm of the corresponding vectors. We choose the normal loss as it is one of the most meaningful intermediate representation of a depth map. Finally, the pixel loss, l_{pixel} is defined as,

$$l_{pixel} = \mathbb{E}_{P^{SP}}||G_{\theta_G}(P^{SP}) - P^{DN}||_1 \tag{7}$$

which leads to the total genereator loss as,

$$L^G = l_{normal} + \lambda_1 * l^G_{RaD} + \lambda_2 * l_{pixel} \tag{8}$$

The λ_1 and λ_2 are weighing factors to balance the different loss components. Generally, training a GAN is a little challenging and having unbalanced loss factors can lead to training instability like mode collapse and non-convergence, two most common issues with GAN training. The scaling factors help to stabilize the training and can lead towards favorable outcomes.

3.4 Network Architecture

Our proposed generator architecture consists of mainly three key components, namely, (1) the transformer encoder structure, (2) a multimodal fusion block and (3) a residual in residual dense block (RRDB). We briefly explain each of these blocks below:

3.4.1 Visual Transformer based Encoder. A visual transformer divides the input image into N patches, each of which is treated as a word. It is then linearly projected, and positional embedding is applied. The transformer encoder receives this input afterwards. To get an output, the first column of the encoder output is passed through an MLP. On large datasets, visual transformers outperform CNNs in visual tasks. The sensor information, either from the camera or LiDAR, can be composed in a 3D volume and treated as a frame. We follow the same approach as ViT [47]. The frame is first divided into patches and then projected into embedding space. Finally, the corresponding positional embedding is added, and the tokens are fed to the transformer-based encoder block.

3.4.2 Multimodal Fusion Block. One of the most important feature of the proposed architecture is its "multimodal fusion block". The transformer based encoder block extracts most meaningful information from its input through its multi-headed attention mechanism and provides the output as a vector embedding. These embedding vectors basically correspond to each of the 2D input patches. In order to have a spatial output from them, these vectors must be converted to 2D patches once again maintaining the original order and locality. In order to do so, we leverage a token folding technique which ensures the correct arrangement of all the patches. Then the folded 3D volume is passed through a convolution block to a higher dimensional projection. Meanwhile, the sparse depth and RGB frames are also passed through two other separate convolution blocks. Finally, the three pipelines are fused inside the multimodal

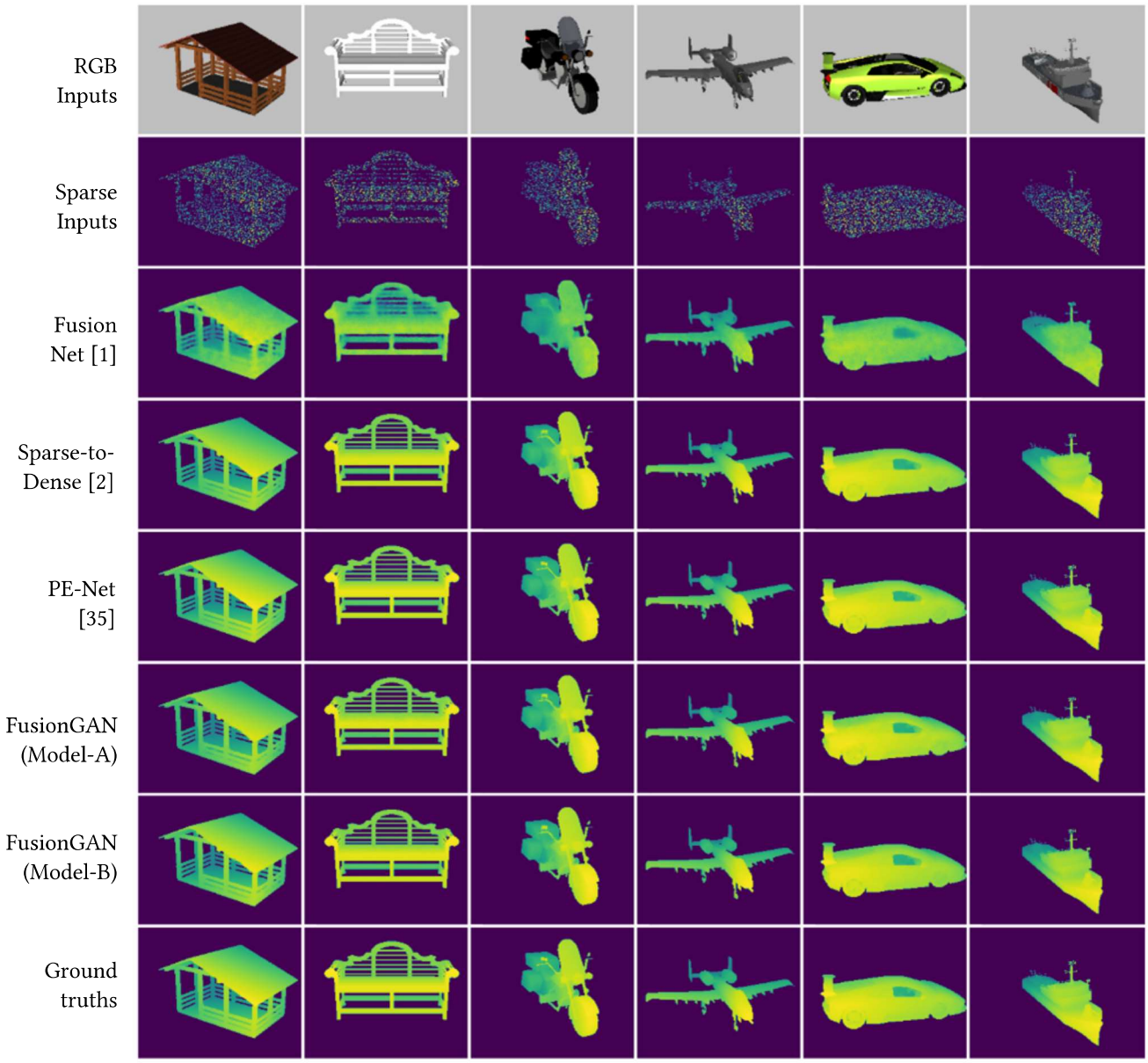

Figure 3: The figure shows the qualitative comparison among the depth reconstruction studies. Both the proposed FusionGAN Model A & B provides better dense depth reconstruction outputs with much finer details compared to the existing studies.

Table 1: Model Variation

FusionGAN Variants	RGB	Sparse Depth	Early Fusion	Late Fusion
Model-A	✓	✓	X	✓
Model-B	✓	✓	✓	✓

fusion block as shown in Fig. 2. The transformer based encoder extracts the relevant guidance for depth and the other two pipelines enforce that with the multimodal fusion.

3.4.3 Residual in Residual Dense Block (RRDB):. RRDB [15] blocks are inspired by Dense Net architecture which connects all the layers in the residual block directly with each other. By increasing the number of connections, it enhances performance. Due to the strong representation to capture the semantic information, a deeper model using RRDB aids in further improving the reconstruction of finer details. Since our task is to reconstruct dense depth in fine granularity, we choose an RRDB block before the final output generation.

3.4.4 Model Variation: In this work, we also propose another variant of our generator model. In the basic model, we have only the RGB image going to the transformer encoder and the two separate pipelines of RGB and sparse depth are combined later within the fusion block. We name this late fusion based basic variant as

Table 2: Depth Completion Accuracy (ShapeNet)

Architecture	RMSE	MAE	iRMSE	iMAE
FusionNet [1]	0.12	0.03	36.24	0.15
Sparse-to-Dense [2]	0.023	0.01	**0.42**	0.03
PENet [35]	0.022	0.004	1.99	0.03
FusionGAN (Model A)	0.021	0.003	8.99	0.007
FusionGAN (Model B)	**0.017**	**0.002**	1.02	**0.004**

"Model-A". In the other variant, we leverage both late and early fusion. The whole pipeline remains same except we introduce the sparse depth to the transformer as well i.e. we feed in both the RGB and sparse depth to the transformer encoder. Later the encoder output is then fused with the other two late fusion pipelines. We call this variant as **"Model-B"**. The model variants are presented in Table 1.

4 EXPERIMENTAL SETUP

In this section, we briefly explain the datasets and training methodology. We also expand upon our proposed novel training recipe for model robustness. We conduct our experiments primarily on the ShapeNet [48] and NYU-Depth-v2 [49] datasets. The ShapeNet dataset contains 128K randomly chosen training samples and 1.2K validation samples. For NYU-Depth-v2 dataset, we use the official split of training (around 47.5K samples) and testing (654 samples). The depth images are first down-sampled to half and then center-cropped to the size of 304×228. Finally, the sparse depth samples are obtained by uniformly sampling the dense depth maps. For NYU-Depth-v2, we only keep 5% valid depth points and remove the rest.

Following previous studies [29], we train the model using only pixel loss for initial 200-300 iterations and then add the remaining two losses. The warm up session helps stabilizing the training by not sending extreme false samples to the discriminator, thus avoiding probable chances of local minima. We use Adam [50] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ as our learning rule. We initially start with a learning rate of $2e-4$ and gradually decrease it to $1.25e-5$ within a training span of 23 epochs.

For the transformer based encoder block, we use the original ViT-base architecture with 12 being the no of layers and heads. The patch embedding vector is of length 768 as well. We train our model with and without the pretrained transformer model. We find negligible difference in final accuracy. However, using the pretrained model speeds up the training time.

4.1 Sensor Resilient Training Recipe

In order make our model robust against possible sensor failures, occlusion, noise and uncertainties, we introduce a novel training recipe for our multimodal fusion GAN. We simulate the sensor unavailability scenario during our GAN training. While training, we randomly select 20% of the training batches and instead of feeding those batches to the network, we input a zero or null matrix of the same shape. The number of such batches have been chosen empirically. We find that by training in such way, the model becomes more robust to any kind of incoming sensor failures. We further discuss it in section 5.2.

Table 3: Depth Completion Accuracy (NYU-Depth-v2)

Architecture	RMSE (m)	REL	MAE (m)
FusionNet [1]	0.12	0.014	0.035
Sparse-to-Dense [2]	0.2	0.046	0.12
CSPN [37]	0.087	0.013	0.037
FusionGAN (Model A)	0.079	0.011	**0.032**
FusionGAN (Model B)	**0.076**	**0.011**	0.033

4.2 Evaluation Metrics

In order to compare the performances, we use the following set of evaluation metrics, (1) Root Mean Square Error (RMSE), (2) Mean Absolute Error (MAE), (3) Inverse RMSE (iRMSE), (4) Inverse MAE (iMAE) and (5) Mean Absolute Relative Error (REL). Let $k = 1, 2, ..., K$ be the total samples in the validation set. Then the evaluation metrics are defined as:

$$\text{RMSE} : \sqrt{\frac{1}{K}\sum_k \left(\hat{P}_{DN} - P_{DN}\right)^2}$$

$$\text{MAE} : \frac{1}{K}\sum_k \left|\hat{P}_{DN} - P_{DN}\right|$$

$$\text{iRMSE} : \sqrt{\frac{1}{K}\sum_k \left(\frac{1}{\hat{P}_{DN}} - \frac{1}{P_{DN}}\right)^2}$$

$$\text{iMAE} : \frac{1}{K}\sum_k \left|\frac{1}{\hat{P}_{DN}} - \frac{1}{P_{DN}}\right|$$

$$\text{REL} : \frac{1}{K}\sum_k \left|\frac{\hat{P}_{DN} - P_{DN}}{P_{DN}}\right|$$

One thing must be mentioned here is that, for the reported numbers on ShapeNet dataset, all the evaluation metrics operate upon normalized output data. On the other hand, the NYU-Depth-v2 dataset result metrics are computed from actual depth values.

5 RESULTS

5.1 Depth Completion with Fusion

We first compare the depth reconstruction accuracy among the contemporary studies. Table 2 and Table 3 present the quantitative results and comparison against other baselines for the ShapeNet [48] and NYU-Depth-v2 [49] datasets respectively. Clearly, our proposed FusionGAN models (A & B) achieve the best results among all. In our model B variant, we give the transformer encoder both the input RGB image and sparse depth data and it outperforms all the other prior works. The multi-headed attention mechanism plays a crucial role in finding the most useful context from both of the modalities and hence, yielding the best performance. The best value in each category of evaluation metrics has been marked in bold. Fig. 3 shows the qualitative comparison among the studies for the ShapeNet dataset. Both of our proposed models achieve much detailed and accurate dense depth maps across the different depth completion datasets.

5.2 Robustness to Sensor Failures

In order to make our model robust to sensor failures, we adopt the training recipe described in section 4.1. The quantitative results are displayed in Table 4. The first row shows the baseline accuracy for normally trained FusionGAN (model A). Now we run another inference by randomly dropping one of the input modalities in 20% samples from the total validation set. As we can observe

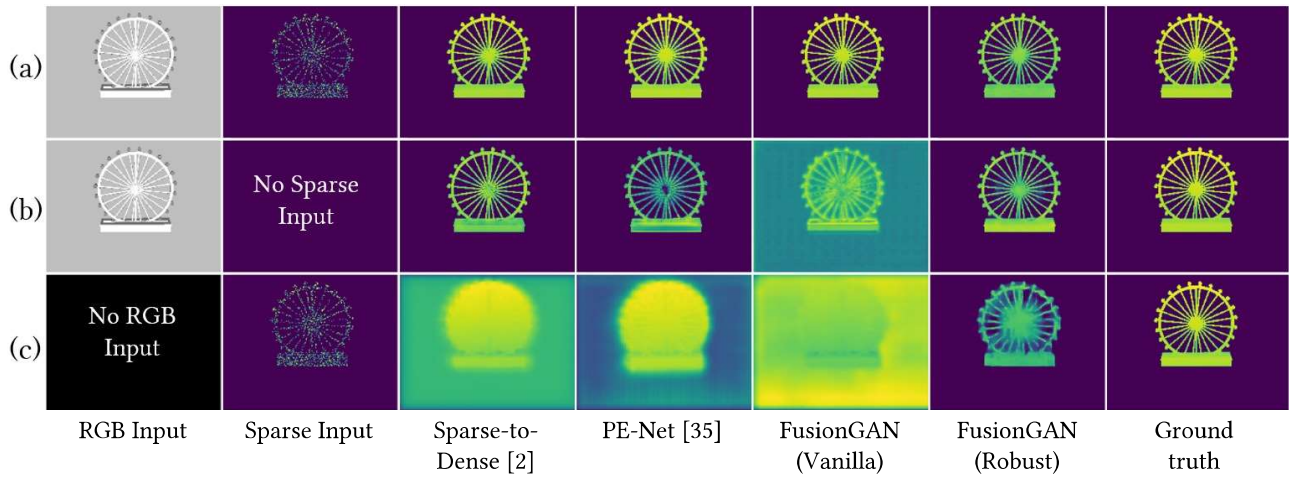

Figure 4: The figure shows the reconstruction results simulating a sensor failure. (a) denotes the normal case when both the input modalities are present. (b) and (c) represent the failure in any of the two modalities. Compared to all the model trained normally, our robust FusionGAN model produces the best depth completion outputs whenever one such failure occurs. Moreover, it is also capable of reconstructing depth maps on par with baseline models when both the modalities are present.

Table 4: Model Robustness Comparison

Architecture	RMSE	MAE	iRMSE	iMAE
FusionGAN (Vanilla)	0.02	0.003	8.99	0.007
With Failure	0.8	0.36	inf	inf
FusionGAN (Robust)	0.08	0.02	30.18	0.05
With Full Info	0.03	0.01	6.88	0.03

that when this model is presented with a missing input modality, the reconstruction accuracy gets severely degraded presented in row 2. The third row shows the same model accuracy but trained with simulated sensor failure in the loop. Obviously, the overall reconstruction accuracy is not as good as the combined multimodal approach but it is significantly better compared to the vanilla base model. Moreover, the training scheme does not necessarily affect the reconstruction capability of the model when both of the modalities are present. As shown in the last row, the robust model is still capable of producing outputs as good as the base model when both of the sensor inputs are available. The Fig. 4 shows the reconstruction quality for all the cases. As we can see that, the models trained in normal fashion perform terribly when one of the sensor modalities is absent. In contrast to that, our robust FusionGAN model is capable of producing both better and meaningful results.

5.3 Memory Comparison

In this section, we do a comparative study on model size, training memory requirement and the inference latency. Even though most of the studies to date compare the model sizes, we believe that the training memory requirement is also an important aspect, especially, when we think retraining or tuning the model on the fly using limited compute/memory resources. Having a smaller size definitely helps since it creates a small static memory footprint. However, the model sizes are not themselves reflective of the compute memory requirement of the models. The reason is the model

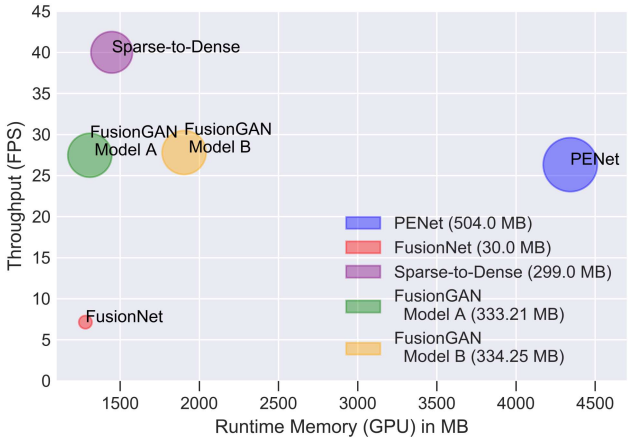

Figure 5: Comparative study on runtime memory and throughput. The bubbles are in proportion with corresponding model sizes. The legend contains the parameter size for each model. Our proposed networks deliver competitive throughput requiring comparatively less runtime memory.

architecture. Fig. 5 is a comparative study between the model runtime memory and throughput. The model radius is proportional to its size. As we can see, the model size is not representative of its runtime memory requirement. We attribute the main reason to the residual connections. Since a lot of intermediate activation maps are needed to be stored for the skip connections, even though the parameter memory footprint is not much, the heavily residual connection based models contribute to the larger memory needs. All the runtime memory numbers are calculated for a batch size of 2. Both model A and B variants provide competitive performance in terms of both runtime memory needs and overall throughput.

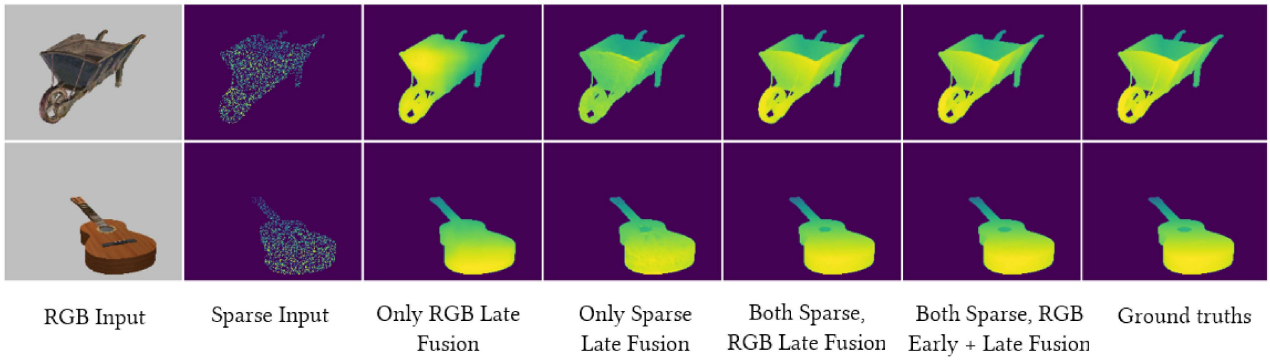

Figure 6: The figure presents the reconstruction accuracy for different sensor fusion techniques. Only RGB based late fusion performs worst since, it is a unimodal variant. The sparse depth fusion performs better. The RGB and sparse depth fused at both earlier and later stages performs the best.

6 ABLATION STUDY

In this section, we perform an analysis on the different fusion strategies and their effect on the reconstruction accuracy. It is already evident that multimodal fusion is essential for a better depth reconstruction. In order to understand the individual impact of the sensors and their corresponding fusion strategy, we perform an ablation study. We add or remove each of the fusion pipeline and see how it affects the reconstruction accuracy.

Late RGB and Sparse Depth Fusion: In our Fusion GAN Model-A, we feed in the RGB image to the transformer-based encoder. We also have another RGB pipeline going for late fusion via a convolution block. Since, the sparse depth already has some accurate depth values to it, we do not perform much computation on it. The sparse depth is also sent for late fusion via another convolution block. We call this setting as late RGB and sparse depth fusion.

Only RGB Late Fusion: In this setting, we keep all the settings mentioned above unchanged, except for the sparse depth pipeline. We do not feed the sparse depth input at all. Hence, this variant becomes a unimodal depth completion network with RGB as its primary input. Later the same RGB input is fused through a convolution block.

Only Sparse Depth Late Fusion: We repeat the methodology as described previously. But this time, we do not have any RGB late fusion. Rather, we feed in the Sparse depth information though a simple convolution block after the encoder stage. This setting takes both the RGB and sparse depth and tries to recover the dense depth.

Both RGB and Sparse Depth Early+Late Fusion: In this setting, we have both the RGB and sparse depth late fusion pipeline. However, we change the initial input to the transformer-based encoder block. We feed in both the RGB and sparse depth to the transformer encoder so that can take a look at both of the sensor modalities at the same time. This is basically our FusionGAN variant, Model-B. In this case, early and late fusion are happening for the RGB and sparse depth data.

Table 5: Sensor Fusion Ablation Study

FusionGAN	Fusion	RMSE	MAE	iRMSE	iMAE
RGB+Sparse	Late	0.021	0.003	8.99	0.007
RGB only	Late	0.11	0.03	26.72	0.04
Sparse only	Late	0.11	0.02	5.04	0.02
Both RGB & Sparse	Early + Late	0.017	0.002	1.02	0.004

The Table 5 compares the fusion strategies quantitatively and the Fig. 6 shows the qualitative results. The best results are obtained with both RGB and sparse depth fused at early and later stages. The late RGB fusion variant performs the worst since, it is basically a unimodal setting. The late sparse depth variant performs comparatively better because it unifies both the sensor modalities. In short, more involved fusion strategy with different modalities yields most accurate dense depth image.

7 CONCLUSION

In this paper, we propose a multimodal sensor fusion strategy using transformer-based self-attention models. We train the network in a generative setting to obtain the best results. Our proposed models outperform existing studies in terms of reconstruction accuracy and are competitive in terms of throughput. We also showcase that the model parameter size and runtime memory requirement are not analogous. A small model can occupy larger compute memory than a bigger model just because of its internal architecture. The runtime memory requirement by the proposed models is comparatively less than that of other contemporary works. Furthermore, we suggest a novel training recipe to make the model robust to certain sensor failure scenerios. The models trained in such strategy deliver reasonably good outputs even if one input modality is unavailable.

ACKNOWLEDGMENTS

This work was supported in part by Center for Brain-inspired Computing (C-BRIC), one of the six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA & National Science Foundation (NSF) SOPHIA (CCF-1822923).

REFERENCES

[1] Wouter Van Gansbeke, Davy Neven, Bert De Brabandere, and Luc Van Gool. 2019. Sparse and noisy lidar completion with rgb guidance and uncertainty. In *2019 16th International Conference on Machine Vision Applications (MVA)*, 1–6. DOI: 10.23919/MVA.2019.8757939.

[2] Fangchang Ma and Sertac Karaman. 2018. Sparse-to-dense: depth prediction from sparse depth samples and a single image. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 4796–4803. DOI: 10.1109/ICRA.2018.8460184.

[3] Amir Atapour-Abarghouei and Toby P. Breckon. 2019. Veritatem dies aperit - temporally consistent depth prediction enabled by a multi-task geometric and semantic scene understanding approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. (June 2019).

[4] Minseok Kim, Sung Ho Choi, Kyeong-Beom Park, and Jae Yeol Lee. 2021. A hybrid approach to industrial augmented reality using deep learning-based facility segmentation and depth prediction. *Sensors*, 21, 1. ISSN: 1424-8220. DOI: 10.3390/s21010307.

[5] MO Kadry. 2022. [online]. Available: https://www.cubictelecom.com/blog/self-driving-cars-future-of-autonomous-vehicles-automotive-vehicles-2030/. [Accessed: Apr-04-2022]. (2022).

[6] Radu Horaud, Miles Hansard, Georgios Evangelidis, and Clément Ménier. 2016. An overview of depth cameras and range scanners based on time-of-flight technologies. *Machine vision and applications*, 27, 7, 1005–1020.

[7] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and koray kavukcuoglu koray. 2015. Spatial transformer networks. In *Advances in Neural Information Processing Systems*. C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors. Volume 28. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2015/file/33ceb07bf4eeb3da587e268d663aba1a-Paper.pdf.

[8] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning* (Proceedings of Machine Learning Research). Francis Bach and David Blei, editors. Volume 37. PMLR, Lille, France, (July 2015), 2048–2057. https://proceedings.mlr.press/v37/xuc15.html.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors. Volume 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[10] Dana Lahat, Tülay Adali, and Christian Jutten. 2015. Multimodal data fusion: an overview of methods, challenges, and prospects. *Proceedings of the IEEE*, 103, 9, 1449–1477. DOI: 10.1109/JPROC.2015.2460697.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

[12] Xudong Mao and Qing Li. 2021. *Generative Adversarial Networks for Image Generation*. Springer.

[13] Ming-Yu Liu, Xun Huang, Jiahui Yu, Ting-Chun Wang, and Arun Mallya. 2020. Generative adversarial networks for image and video synthesis: algorithms and applications. *CoRR*, abs/2008.02793.

[14] Qi Wei, Chao Yuan, and Amit Chakraborty. 2021. Generative adversarial networks for time series. US Patent App. 17/271,646. (November 2021).

[15] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. 2018. Esrgan: enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*. (September 2018).

[16] Abdul Jabbar, Xi Li, and Bourahla Omar. 2021. A survey on generative adversarial networks: variants, applications, and training. *ACM Comput. Surv.*, 54, 8, Article 157, (October 2021), 49 pages. ISSN: 0360-0300. DOI: 10.1145/3463475.

[17] David Eigen and Rob Fergus. 2015. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. (December 2015).

[18] David Eigen, Christian Puhrsch, and Rob Fergus. 2014. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*. Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors. Volume 27. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2014/file/7bccfde7714a1ebadf06c5f4cea752c1-Paper.pdf.

[19] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. 2016. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38, 10, 2024–2039. DOI: 10.1109/TPAMI.2015.2505283.

[20] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. 2016. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth International Conference on 3D Vision (3DV)*, 239–248. DOI: 10.1109/3DV.2016.32.

[21] Lee-Kang Liu, Stanley H. Chan, and Truong Q. Nguyen. 2015. Depth reconstruction from sparse samples: representation, algorithm, and sampling. *IEEE Transactions on Image Processing*, 24, 6, 1983–1996. DOI: 10.1109/TIP.2015.2409551.

[22] Simon Hawe, Martin Kleinsteuber, and Klaus Diepold. 2011. Dense disparity maps from sparse disparity measurements. In *2011 International Conference on Computer Vision*, 2126–2133. DOI: 10.1109/ICCV.2011.6126488.

[23] Abdelrahman Eldesokey, Michael Felsberg, and Fahad Shahbaz Khan. 2018. Propagating confidences through cnns for sparse data regression. *arXiv preprint arXiv:1805.11913*.

[24] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. 2017. Sparsity invariant cnns. *CoRR*, abs/1708.06500.

[25] Zixuan Huang, Junming Fan, Shenggan Cheng, Shuai Yi, Xiaogang Wang, and Hongsheng Li. 2020. Hms-net: hierarchical multi-scale sparsity-invariant network for sparse depth completion. *IEEE Transactions on Image Processing*, 29, 3429–3441. DOI: 10.1109/TIP.2019.2960589.

[26] Kaiyue Lu, Nick Barnes, Saeed Anwar, and Liang Zheng. 2022. Depth completion auto-encoder. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 63–73. DOI: 10.1109/WACVW54805.2022.00012.

[27] Maximilian Jaritz, Raoul De Charette, Emilie Wirbel, Xavier Perrotton, and Fawzi Nashashibi. 2018. Sparse and dense data with cnns: depth completion and semantic segmentation. In *2018 International Conference on 3D Vision (3DV)*, 52–60. DOI: 10.1109/3DV.2018.00017.

[28] Jason Ku, Ali Harakeh, and Steven L. Waslander. 2018. In defense of classical image processing: fast depth completion on the cpu. In *2018 15th Conference on Computer and Robot Vision (CRV)*, 16–22. DOI: 10.1109/CRV.2018.00013.

[29] 2021. *Sparse to dense depth completion using a generative adversarial network with intelligent sampling strategies. Proceedings of the 29th ACM International Conference on Multimedia*. Association for Computing Machinery, New York, NY, USA, 5528–5536. ISBN: 9781450386517. https://doi.org/10.1145/3474085.3475688.

[30] Jiashen Hua and Xiaojin Gong. 2018. A normalized convolutional neural network for guided sparse depth upsampling. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*. International Joint Conferences on Artificial Intelligence Organization, (July 2018), 2283–2290. DOI: 10.24963/ijcai.2018/316.

[31] Maximilian Jaritz, Raoul de Charette, Émilie Wirbel, Xavier Perrotton, and Fawzi Nashashibi. 2018. Sparse and dense data with cnns: depth completion and semantic segmentation. *CoRR*, abs/1808.00769.

[32] Jie Tang, Fei-Peng Tian, Wei Feng, Jian Li, and Ping Tan. 2021. Learning guided convolutional network for depth completion. *IEEE Transactions on Image Processing*, 30, 1116–1129. DOI: 10.1109/TIP.2020.3040528.

[33] Jiaxiong Qiu, Zhaopeng Cui, Yinda Zhang, Xingdi Zhang, Shuaicheng Liu, Bing Zeng, and Marc Pollefeys. 2019. Deeplidar: deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. (June 2019).

[34] Zhiqiang Yan, Kun Wang, Xiang Li, Zhenyu Zhang, Baobei Xu, Jun Li, and Jian Yang. 2021. Rignet: repetitive image guided network for depth completion. *CoRR*, abs/2107.13802.

[35] Mu Hu, Shuling Wang, Bin Li, Shiyu Ning, Li Fan, and Xiaojin Gong. 2021. Penet: towards precise and efficient image guided depth completion. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 13656–13662. DOI: 10.1109/ICRA48506.2021.9561035.

[36] Xinjing Cheng, Peng Wang, Chenye Guan, and Ruigang Yang. 2020. Cspn++: learning context and resource aware convolutional spatial propagation networks for depth completion. 34, (April 2020), 10615–10622. DOI: 10.1609/aaai.v34i07.6635.

[37] Xinjing Cheng, Peng Wang, and Ruigang Yang. 2020. Learning depth with convolutional spatial propagation network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42, 10, 2361–2379. DOI: 10.1109/TPAMI.2019.2947374.

[38] Chongzhen Zhang, Yang Tang, Chaoqiang Zhao, Qiyu Sun, Zhencheng Ye, and Jürgen Kurths. 2021. Multitask gans for semantic segmentation and depth completion with cycle consistency. *IEEE Transactions on Neural Networks and Learning Systems*, 32, 12, 5404–5415. DOI: 10.1109/TNNLS.2021.3072883.

[39] Kranti Kumar Parida, Siddharth Srivastava, and Gaurav Sharma. 2021. Beyond image to depth: improving depth prediction using echoes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8268–8277.

[40] Aditya Prakash, Kashyap Chitta, and Andreas Geiger. 2021. Multi-modal fusion transformer for end-to-end autonomous driving. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

[41] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. 2021. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. (October 2021), 12179–12188.

[42] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. 2020. Seeing through fog without seeing fog: deep multimodal sensor fusion in unseen adverse weather. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (June 2020).

[43] Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

[44] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. Generative adversarial text to image synthesis. In *International conference on machine learning*. PMLR, 1060–1069.

[45] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1125–1134.

[46] Alexia Jolicoeur-Martineau. 2019. The relativistic discriminator: a key element missing from standard GAN. In *International Conference on Learning Representations*. https://openreview.net/forum?id=S1erHoR5t7.

[47] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: transformers for image recognition at scale. In *International Conference on Learning Representations*. https://openreview.net/forum?id=YicbFdNTTy.

[48] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. 2015. ShapeNet: An Information-Rich 3D Model Repository. Technical report arXiv:1512.03012 [cs.GR]. Stanford University — Princeton University — Toyota Technological Institute at Chicago.

[49] Pushmeet Kohli Nathan Silberman Derek Hoiem and Rob Fergus. 2012. Indoor segmentation and support inference from rgbd images. In *ECCV*.

[50] Diederik P. Kingma and Jimmy Ba. 2014. Adam: a method for stochastic optimization. (2014). eprint: arXiv:1412.6980.