Data Augmented 3D Semantic Scene Completion with 2D Segmentation Priors

Aloisio Dourado, Frederico Guth   and   Teofilo de Campos

University of Brasilia

Campus Darcy Ribeiro. Asa Norte, Brasilia, DF - 70910-900, Brazil

aloisio.dourado.bh@gmail.com   fredguth@fredguth.com   t.decampos@oxfordalumni.org

Abstract

Semantic scene completion (SSC) is a challenging Computer Vision task with many practical applications, from robotics to assistive computing. Its goal is to infer the 3D geometry in a field of view of a scene and the semantic labels of voxels, including occluded regions. In this work, we present SPAwN, a novel lightweight multimodal 3D deep CNN that seamlessly fuses structural data from the depth component of RGB-D images with semantic priors from a bimodal 2D segmentation network. A crucial difficulty in this field is the lack of fully labeled real-world 3D datasets which are large enough to train the current data-hungry deep 3D CNNs. In 2D computer vision tasks, many data augmentation strategies have been proposed to improve the generalization ability of CNNs. However those approaches cannot be directly applied to the RGB-D input and output volume of SSC solutions. In this paper, we introduce the use of a 3D data augmentation strategy that can be applied to multimodal SSC networks. We validate our contributions with a comprehensive and reproducible ablation study. Our solution consistently surpasses previous works with a similar level of complexity.

1. Introduction

Reasoning about scenes in 3D is a natural human ability that remains a challenge for Computer Vision. In the past, the two most common scene understanding tasks were scene completion [6] and semantic labeling of visible surfaces [11, 26, 27]. Noticing that these are intertwined tasks, in 2017, Song et al. [31] introduced the Semantic Scene Completion (SSC) task for simultaneously completing occluded voxels and inferring their semantic labels and proposed SSCNet, achieving better results than dealing with these tasks separately. Early approaches only used depth information, ignoring the RGB channels [31, 10]. The use of color channels was introduced later [8].

We present a new approach for exploring information from the RGB-D input (explained in section 3), as shown in Figure 1. The solution uses 2D prior probabilities from a bimodal 2D segmentation network as semantic guidance to the depth map's structural data. The proposed multimodal 3D network, *SPAwN*, uses a new memory-saving batch-normalized dimensional decomposition residual building block (BN-DDR) and can be trained on a single 10Gb GPU with a 4 scene mini-batch.

To overcome the limitations imposed by the lack of sizeable real-world datasets, we are the first to apply 3D data augmentation for the SSC task. Data augmentation is widely used in the training of 2D deep CNNs [17, 14] and its goal is to reduce overfitting by artificially increasing the variety of samples in the training dataset using transformations like flipping, cropping, rotation and color transforms. However, those transformations can not naïvely be used in 3D applications like semantic Scene completion because of the difference in the number of dimensions of the input (2D) and output (3D). In this paper, we propose to apply data augmentation to inner 3D volumes of the solution with three fast 3D transformations in voxel space that preserve the main characteristics of the scene. Our proposed data augmentation approach reduces overfitting and achieves unprecedented levels of semantic completion when compared to previous works of similar memory footprint and complexity.

We evaluated our contributions with and without pre-training on synthetic data and observed that our method surpasses, by far, all previous state-of-the-art results on both scenarios. We demonstrate the benefits of the proposed architecture and the data augmentation approach separately, with several experiments in a comprehensive and reproducible ablation study. Regarding the proposed augmentation scheme, we evaluate it for training (regular data augmentation) and test (test-time data augmentation).

Supplementary material provides additional graphs and data regarding all experiments. All models and training code necessary to reproduce our results and the ablation experiments are publicly available[1] .

Our contributions are listed below.

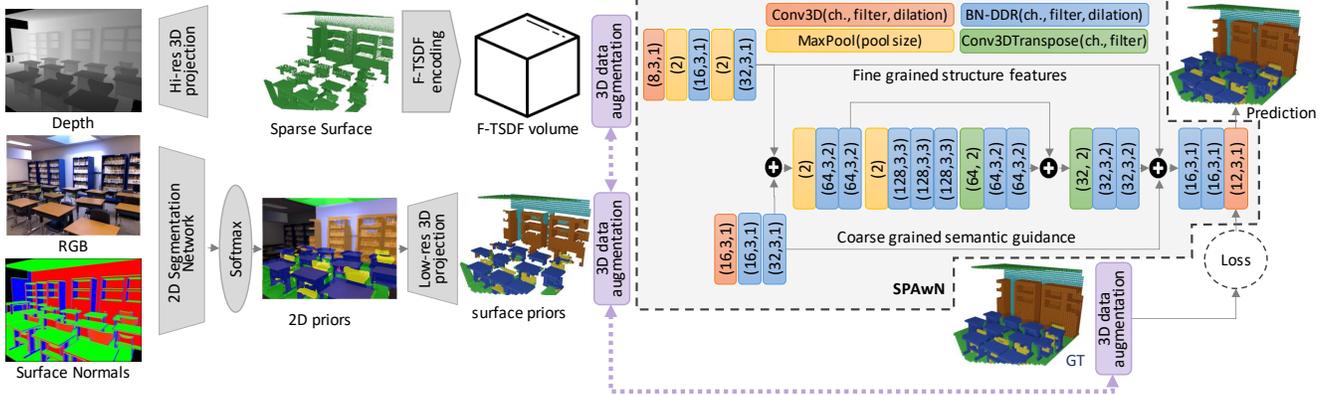[1]Source code: https://cic.unb.br/~teodecampos/aloisio/

Figure 1: **Overview of our solution**. Our system comprises SPAwN, a 3D CNN that uses 2D priors as semantic guidance, and a novel 3D data augmentation approach for regularization and overfitting reduction. The 2D segmentation network is multimodal, combining RGB and surface normals. (Best viewed in color.)

- *SPAwN*, a novel lightweight multimodal 3D SSC CNN architecture that uses 2D prior probabilities from a 2D segmentation network. These priors are used as semantic guidance to the structural data from the depth part of the RGB-D input. This architecture can be efficiently trained on a single 10Gb GPU and achieves state-of-the-art results on both real and synthetic data.

- *BN-DDR*, a memory-saving batch-normalized dimensional decomposition residual building block for 3D CNNs. It preserves previous approaches' regularization characteristics while consuming much less memory during training.

- We are the first to apply a data augmentation technique for 3D semantic scene completion. Our method uses three 3D data transformations which operate on batches directly in GPU tensors.

- *SPAwN* surpasses by far all known previous works with similar memory footprint and pipeline complexity on both real and synthetic data. When trained with the proposed data augmentation strategy, our solution achieves unprecedented levels of SSC scores.

2. Related Work

2D RGB-D Semantic Segmentation extends regular 2D RGB semantic segmentation with the addition of 3D depth maps obtained from devices such as structured light sensors (*e.g.* Kinect), time-of-flight IR sensors, or stereo cameras. Early works [27, 11, 29] relied on handcrafted features further fed into a classification model. Later, following the boom of CNNs, models started using the depth map as a fourth input channel [5], eventually encoded in HHA (horizontal disparity, height above ground, and angle with gravity) [12], before feeding into the neural net-

work [23, 26, 32]. Lin et al. [21] introduced the RefineNet module to fuse skip-connections from different scales of the segmentation encoder, and [25] extended it to deal with multiple modes introducing the Multimodal Fusion (MMF) module. Later, [15] proposed an encoder-decoder network that uses depth maps as ground truth to extract depth embeddings that are later fused to semantic features from the RGB image, thus eliminating the need for depth maps during inference and also achieving the current state-of-the-art in RGB-D semantic segmentation. Our 2D network (explained in detail in section 3.1) is a simplified version of [25] which did not have the ambition of being state-of-the-art for this auxiliary task. Its design prioritized keeping a low memory footprint.

Semantic Scene Completion was introduced relatively recently [31] and is already an active field of study in computer vision. Early solutions [31, 35, 10] used only the depth maps of the RGB-D images, neglecting the RGB channels. Guedes *et al*. [8] reported preliminary results obtained by adding color to an SSCNet [31]-like architecture. After that, many solutions [7, 22, 18, 20] started using RGB channels as input to a 2D segmentation network, whose generated features were fed into a 3D CNN. Current state-of-the-art solutions include using RGB edges to improve detection of hard-to-detect classes [3], using 360° panoramic images [4], fusing multi-scale context information [36], multiple separate generators [33], and semi-supervision from depth maps [2]. All previous methods use a straightforward training pipeline, where the input flows through the network in a single direction, without any loops. More recently, [1] introduced the scene-instance-scene pipeline, which includes a sequence of semantic segmentation and instance completion networks. This pipeline is iteratively executed multiple times, achieving great results at the cost of requiring much more computational

power.

In this paper we present a lightweight and straightforward solution that, like some other methods, explores 2D segmentation. However, we propose a completely novel way of extracting knowledge from the RGB-D channels. Our results are a significant improvement w.r.t. the state-of-the-art on benchmark datasets for this kind of pipeline and are comparable with the much more expensive iterative solution.

Data Augmentation is a regularization technique that is vastly used in 2D segmentation problems [17, 14, 21, 15] and other 2D computer vision applications. Regularization and consequent reduction in overfitting is achieved by artificially enlarging the training dataset randomly applying transformations like flipping and cropping to the input images [17]. More recently introduced data augmentation strategies include blocking-based approaches like Blockout [24] and Random Erasing [37]. Applying the transformations to the input images is enough for image classification tasks since they do not affect the corresponding labels. However, for semantic segmentation tasks, if the transformations affects objects' position in the input image, it is necessary to apply the same transformations to the ground truth maps to preserve the pixel-to-pixel correspondence [17]. The need for applying the same transformations to the input and the ground truth is an extra difficulty for using data augmentation in RGB-D to voxel segmentation tasks like semantic scene completion. Keeping the correspondence between the input and ground truth representation is necessary, but there is no direct pixel to voxel mapping from the input to the output. When the solution uses some kind of Truncated Sign Distance Function like TSDF or F-TSDF [31] the problem is exacerbated because changing one pixel of the input would propagate in 3D space through the TSDF-encoded volume in a pyramid shape, affecting a large region.

In this paper, we overcome those side-effects by applying the data augmentation transformations to the inner 3D volumes of the input and the ground truth volume, rather than the RGB-D channels. They, therefore, preserve the TSDF or F-TSDF encoding and lead to impressive results.

3. Proposed Solution

The overall multimodal solution is shown in Figure 1. Initially, we feed two modes of the input RGB-D image (RGB and surface normals) into a 2D segmentation CNN. Then, we submit the output of the 2D network to a Softmax function and obtain the prior probabilities that will be further projected to a low-resolution 3D voxel volume. The depth map, a third input mode, is projected to a high-resolution 3D volume and encoded with F-TSDF [31]. Data augmentation is applied directly to the 3D volumes, including ground truth, before feeding our SPAwN CNN. The in-
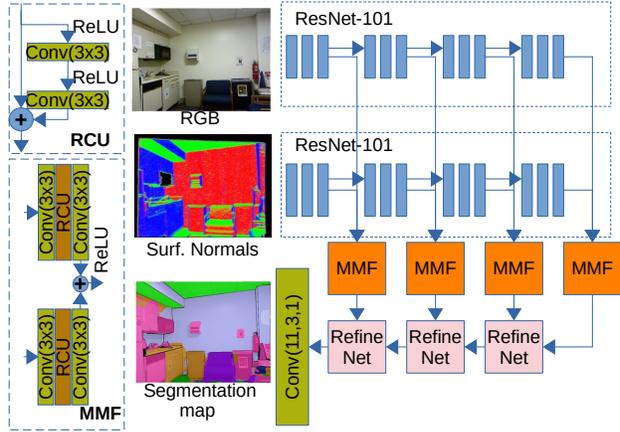

Figure 2: **2D bimodal segmentation network architecture.** The Residual Convolution Unit (RCU) and the RefineNet module were first defined in [21]. Here, we use a simplified MMF block [25]. (Best viewed in color.)

put branches of SPAwN match the scale and the volumes are fused with an early/late fusion network to produce the final predictions.

3.1. 2D Segmentation Network

To acquire high-quality 2D priors while keeping the memory footprint low, we designed a tailor-made version of RDFNet [25]. We use a bimodal encoder-decoder 2D RGB-D segmentation network with two ImageNet pre-trained ResNet-101 [14] backbones, one for each input mode, as presented in Figure 2. The main adjustments to the original RDFNet are simplifications to the MMF module reducing the number of convolution layers; the usage of 3 RefineNet modules, instead of 4; and the modification of the number of channels of the last layer since we need a classifier for 11 classes (in 2D, the empty or void class is ignored and the original RDFNet was trained for 40 classes).

This customized RDFNet takes two input images: the color channels of the RGB-D input, as in the original version, and surface normals, instead of the HHA encoded image. The surface normals are obtained from the depth map after aligning the scene to the Cartesian axes following the Manhattan assumption, as in [29]. Each axis is mapped to one RGB channel, and the absolute normal values are normalized from 0 to 255. Our goal with this simplified 2D network is only to test the hypothesis that 2D segmentation priors would improve the overall result.

3.2. SPAwN Semantic Scene Completion Network

Our Segmentation Priors Aware Network (*SPAwN*) is a novel 3D CNN architecture that uses 2D prior probabilities from a semantic segmentation network to guide the depth map's structural information.

Depth map projection and encoding. Following previous works on SSC, we align the scene to the Cartesian axes considering the Manhattan assumption, project the depth map to 3D and then apply F-TSDF [31] to reduce data sparsity. The dimensions of the 3D space are set to 4.8m horizontally, 2.88m vertically, and 4.8m in depth. Voxel grid size is 0.02m, and the dimension of the resulting volume is $240 \times 144 \times 240$. The F-TSDF truncation value is 0.24m. Examples of a projected depth map are shown in Figure 5b.

2D prior probabilities projection and ensemble. Projecting the output of the 2D network to 3D is a delicate task. One could be tempted to project the features to 3D at the same resolution of the structural volume. However, it would lead to 11 volumes of $240 \times 144 \times 240$ voxels, consuming too much GPU memory. Our solution, instead, consists in projecting the data from 2D at a lower resolution. We use $60 \times 36 \times 60$, the same as the output of the network. We compensate the reduction in details, which come from the structural branch, with a boost in accuracy during downsampling by using a simple yet effective classifier ensemble method known as the "sum rule" [16], as follows. Firstly we obtain the probabilities for the 11-class output by applying a Softmax function, resulting in 11 planes with the exact resolution as the input image. Then, we project each pixel to 3D at low resolution (voxel size = 0,08m). When more than one pixel falls into the same voxel, we sum the probabilities of each class and, to normalize the resulting priors, we divide the probabilities by the number of pixels that fell into the voxel.

The previous approach only provides information for the surface voxels. To provide information to non-surface voxels, we add an extra channel for the empty class, and for all non-surface voxels, we set the probability 1 for voxels belonging to the class "empty" and 0 to the other 11 classes. An example of projected prior volume compared to the ground truth is shown in Figures 5c and 5f.

Batch-normalized DDR. The fundamental building block of our 3D network is a batch normalized version of the Dimensional Decomposition Residual (DDR) [19] block, named *BN-DDR*. The DDR block is a lightweight alternative to the ResNet block [14] when facing the vanish gradient problem in deep neural networks. Our preliminary experiments showed that adding batch normalization layers to the DDR block produces better results. However, adding a batch normalization layer after each convolutional layer of the block as in [20], consumes too much memory, making the network difficult to train with larger mini-batches, reducing the overall benefits of the batch normalization, and making the training slower. Our solution eliminates the batch normalization layers between the dimensional decomposition layers resulting in our proposed *BN-DDR* block as presented in Figure 3. Due to its reduced memory footprint, we keep the same number of channels of the outer
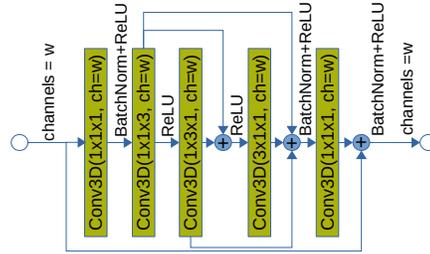

Figure 3: **Proposed *BN-DDR* module**. Our arrangement presents good discrimination and regularization properties while keeping memory consumption manageable.

layers in the inner layers, while keeping the mini-batch with 4 scenes. For example, previous DDR-based solutions like DDR-Net [19] and AMFNet [20] use a batch size of 2 and 1, respectively. SketchAware [2] uses a batch size of 4, as well, however it requires 2 11GB GPUs for training.

Fusing structure and semantics. We use an early/late fusion strategy to fuse the detailed structure information from the F-TSDF encoded high-resolution volume to the semantic information from the surface prior volume, as presented in Figure 1. The two initial branches are used to match the resolution and number of channels of both inputs. Then, both signals are early fused in an encoder-decoder 3D CNN with a skip connection in the mid-resolution stage, inspired by the U-Net architecture [28]. This fusion helps to preserve the details of the higher resolution level. The outer skip connections provide additional structural and semantic guidance, and the output branch performs the final late fusion and classification into the desired number of classes.

Data balancing and loss function. There are two main sources of data unbalancing in the 3D SSC problem: the first is the unbalance between occluded empty and occupied voxels; the second is the unbalance between the several classes. In this work, we face both unbalances in our loss functions. For the first one, we follow [3] and, for each training mini-batch, we randomly sample the occluded voxels to balance occupied and empty voxels, while ignoring empty voxels in the visible space. For the class unbalance, we use a class-weighted cross-entropy loss function, where the weight $w_c = 2$ for less frequent classes like TVs, objects, table and chair and $w_c = 1$ for all other classes. Being V the set of voxels of the mini-batch (lmb) selected for evaluation, $v \in V$, n the number of classes, $y_{v,c}$ a binary indicator if the voxel v belongs to class c and $P_{v,c}$ the predicted probability of the voxel v related to class c, the loss L is given by equation 1.

$$L = \frac{1}{|V|} \sum_v \left(-\frac{\sum_{c=1}^{n}[w_c \cdot y_{v,c} \cdot \log(P_{v,c})]}{\sum_{c=1}^{n} w_c} \right) \quad (1)$$

3.3. Data Augmentation for SSC

Data augmentation techniques have proven to be very effective in 2D segmentation problems, however, there are obstacles that make it difficult to apply them for the problem of semantic scene completion, where the goal is to map from RGB-D images to 3D labeled voxels. To overcome the lack of pixel-to-voxel correspondence in those applications, we introduce the use of data augmentation applied directly to the projected 3D volumes of the SSC solutions. We randomly rotate the scene in 45-degree steps and randomly flip along the horizontal axes (X and Z), to avoid generating upside-down scenes. This precaution is usually taken in 2D domains. For instance, vertical flipping and mirroring are usually avoided in 2D. We achieve this augmentation with 3 simple and fast 3D transformations. The first two transformations are random X-axis flipping and random Z-axis flipping. The third transformation is random X $\leftrightarrow$ Z axes swapping. The use of this strategy during training is equivalent to augment eight times the size of the dataset. Figure 4 illustrates all possible augmented volumes from a single scene. All operations are done in the GPU tensors and can be easily executed in parallel, with almost no impact on training time. The proposed data augmentation strategy is mode and resolution agnostic, thus, it may be readily applied to multi-modal or to multi-resolution SSC setups. For instance, SPAwN 3D mini-batches contain the F-TSDF encoded high resolution surface volume and the 12 channels of low resolution priors.

Training-time data augmentation. During training, for each training mini-batch, we randomly choose to apply or not each one of the 3 transformations. The chosen transformations are then applied to the whole 3D input mini-batch at once and also to the ground truth.

Test-time data augmentation. We also evaluate the use of the transformations in test-time. In this case, for each test mini-batch, we apply all the eight possible combinations of transformation to each input volumes, generate the predicted volumes and, unlike in training-time, we apply the inverse transformation to the output volumes instead of the same transformation. In this way, the eight output volumes share the same orientation as the original mini-batch. The aligned predictions are then ensembled. For that, we apply the "sum rule" [16], generating a single and more accurate output.

4. Experiments

4.1. Datasets

We executed our experiments on the three most important SSC benchmark datasets: NYUDv2 [29], NYUCAD [6] and SUNCG [31].

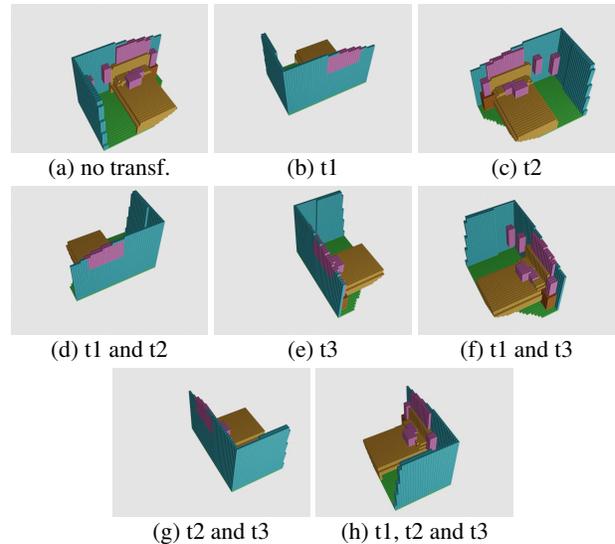**NYUDv2** includes depth and RGB images captured by the Kinect depth sensor gathered from commercial and res-

(a) no transf.   (b) t1   (c) t2

(d) t1 and t2   (e) t3   (f) t1 and t3

(g) t2 and t3   (h) t1, t2 and t3

Figure 4: **All augmented volumes generated from a single scene**. Each image caption indicates which transformations were applied. The transformations are: t1 (X-axis flipping), t2 (Z-axis flipping) and t3 (X $\leftrightarrow$ Z axes swapping).

idential buildings, comprising 464 different indoor scenes, divided into 795 samples for training and 654 for testing. We generated ground truth by voxelizing the 3D mesh annotations from [9] and mapped object categories based on [13] to label occupied voxels with semantic classes. NYUDv2 is a challenging dataset due to the small number of images for training and the misalignment between ground truth generated from 3D objects meshes and the depth maps.

NYUCAD is a dataset generated from NYUD-v2 where the depth maps are synthesized from the 3D objects meshes, eliminating the misalignment between the depth map and the 3D ground truth. The RGB images and 2D segmentation ground truth are the original ones from NYUDv2. Hence, they present misalignment when compared to the generated depth map and the 3D ground truth. To reduce the effect of this remaining misalignment on 2D modes, when training our 2D segmentation network, we use surface normals extracted from the synthetic depth maps, and the 2D ground truth is re-generated back-projecting the 3D ground truth to 2D.

SUNCG dataset consists of about 45K completely synthetic house models from which were extracted more than 130K 3D snapshots with corresponding depth maps and ground truths divided into train and test datasets. As SUNCG neither includes the RGB images nor the surface normals, we render the RGB images using the provided camera positions for each snapshot as specified in [3] and extract the surface normals from the depth map.

4.2. Training Details

All models of the experiments conducted in this paper were optimized with SGD using the *one cycle learning rate policy* [30] with the maximum and minimum learning rates (LR) multipliers set to 25 and 1×10^{-4}, respectively, cosine annealing, 1×10^{-5} weight decay, and mini-batches with four scenes.

2D network training. 2D models were trained in two data augmented stages. In the first stage, the ImageNet pretrained ResNet-101 backbones weights were frozen, and the base LR was set to 2×10^{-4}. In the second stage, the base LR was set to 8×10^{-5}, and all weights were unfrozen, but the ResNet backbones LR was set to 1/10 of the running LR. For NYUDv2 and NYUCAD, each stage comprises 150 epochs, and for SUNCG, 10 epochs. The data augmentation transforms were the conventional 2D random resize, random crop, and random horizontal flip.

3D network training. 3D models were trained in a single stage, with base LR set to 4×10^{-4}. Our 3D data augmentation strategy reduces overfitting and thus, allows us to train for more epochs. For NYUDv2 and NYUCAD, the models were trained for 80 epochs when not using data augmentation and for 120 epochs when augmented. As SUNCG is a very large dataset, the benefits of data augmentation is expected to be inexpressive, so we do not apply data augmentation on SUNCG and train for only 10 epochs.

Fine-tuning from SUNCG. When fine-tuning from SUNCG, both 2D and 3D models were first trained on SUNCG then fine-tuned to the desired target dataset. The same protocols described previously were applied.

Metrics. We follow the previous works and report scores for the completion task and the semantic scene completion task. For completion we report precision, recall and Intersection over Union (IoU) considering the prediction of occupancy for each occluded voxel. For the semantic scene completion task we report the IoU of each object class on both the observed and occluded voxels and the averaged result over all classes except the void class (mIoU). Voxels outside the view and visible empty voxels are ignored.

4.3. Ablation Study

We evaluate the importance of each aspect of the proposed solution with a comprehensive set of experiments using only real images from NYUDv2 without pretraining on synthetic images. Table 1 presents the progressive contribution of the proposed *BN-DDR* module, the use of the proposed class balancing strategy, and the data augmentation training and test approaches, considering one, two, and three input modes. All models were trained and evaluated under the same protocols except for the number of epochs when using data augmentation, as explained in previous section.

input modes	DDR type	class bal.	DA	TTDA	comp. IoU	SSC mIoU
depth	Regular	no	no	no	55.5	24.5
	BN-DDR	no	no	no	60.8	31.8
	BN-DDR	yes	no	no	60.8	32.2
depth rgb	Regular	no	no	no	60.9	38.6
	BN-DDR	no	no	no	63.0	41.0
	BN-DDR	yes	no	no	64.4	42.2
depth rgb sn	Regular	no	no	no	61.3	39.2
	BN-DDR	no	no	no	63.4	41.4
	BN-DDR	yes	no	no	63.8	43.4
	BN-DDR	yes	yes	no	65.7	47.7
	BN-DDR	yes	yes	yes	66.2	48.0
oracle test	*BN-DDR*	yes	no	no	76.7	67.9

Table 1: **Progressive impact of SPAwN components on NYUDv2.** No pretraining was performed. "sn" means surface normals, DA means data augmentation, and TTDA means test-time data augmentation.

We observed positive contributions for each of the evaluated aspects. Using RGB and surface normals as inputs consistently produced positive impacts. The model itself surpassed state-of-the-art-results with regular not augmented training. Moreover, combining the proposed network with the data augmentation approaches enhances results to unprecedented levels.

Table 1 also evaluates the model's theoretical upper bound limit in an Oracle Test, supposing we have predicted perfect semantic 2D priors. To this matter, we replace the output of the 2D network with the 2D ground truth. The Oracle experiment shows there is still room for improvements by enhancing 2D predictions. Future works can exploit this.

We provide an investigation on the benefits of data augmented training with real images (NYUDv2) in more detail in supplementary material.

4.4. Comparison to the State-of-the-Art

In Tables 2, 3, and 4, for each training scenario, the best scores for straightforward solutions are presented in bold, while the second-best scores are underlined. We only show the best two or three competing models in each category. Further results are in the supplementary material.

SUNCG. Table 2 shows the results on SUNCG synthetic dataset. As the SUNCG training set is large, the benefits of data augmentation are not expected to be significant. Therefore, we only evaluate the standard training approach. Our proposed 3D CNN surpassed CCPNet, the previous best straightforward solution, by a 6.3% margin (4.7 p.p.) and got similar results to the iterative models.

NYUDv2. Table 3 presents the results on real images.

model	pipeline type	scene completion			semantic scene completion (IoU, in percentages)											
		prec.	rec.	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
SISNet-BiSeNet [1]	iterative	93.3	96.1	89.9	85.2	90.0	83.7	80.8	60.0	83.5	80.8	68.6	77.3	86.7	70.1	78.8
SISNet-DeepLabv3 [1]		92.6	96.3	89.3	85.4	90.6	82.6	80.9	62.9	84.5	82.6	71.6	72.6	85.6	69.7	79.0
EdgeNet[3]	straight-forward	93.3	90.6	85.1	97.2	95.3	78.2	57.5	51.4	80.7	74.1	54.5	52.6	70.3	60.1	70.2
ESSC[34]		92.6	90.4	84.5	96.6	83.7	74.9	59.0	55.1	83.3	78.0	61.5	47.4	73.5	62.9	70.5
CCPNet[36]		98.2	96.8	91.4	99.2	89.3	76.2	63.3	58.2	86.1	82.6	65.6	53.2	76.8	65.2	74.2
SPAwN (ours)		91.9	88.7	82.3	99.3	96.1	84.4	75.1	59.2	81.5	78.1	67.3	80.1	76.3	70.4	78.9

Table 2: **Results on SUNCG test set**. Our SPAwN semantic scene completion overall results surpass by far all known previous straight-forward solutions on SUNCG synthetic images, and are comparable to both SISNet models, even though they have a much higher parameter count and operate with a complext iterative pipeline for both training and inference.

model	pipeline type	train	scene compl.			semantic scene completion (IoU, in percentages)											
			prec.	rec.	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
SISNet-BiSeNet[1]	iterative	NYU	90.7	84.6	77.8	53.9	93.2	51.3	38.0	38.7	65.0	56.3	37.8	25.9	51.3	36.0	49.8
SISNet-DLabv3[1]			92.1	83.8	78.2	54.7	93.8	53.2	41.9	43.6	66.2	61.4	38.1	29.8	53.9	40.3	52.4
TS3D[7]	straight-forward	NYU	-	-	60.0	9.7	93.4	25.5	21.0	17.4	55.9	49.2	17.0	27.5	39.4	19.3	34.1
SketchAware[2]			85.0	81.6	71.3	43.1	93.6	40.5	24.3	30.0	57.1	49.3	29.2	14.3	42.5	28.6	41.1
SPAwN (ours)			82.3	77.2	66.2	41.5	94.3	38.2	30.3	41.0	70.6	57.7	29.7	40.9	49.2	34.6	48.0
TNetFuse[22]	straight-forward	NYU + SUNCG	67.3	85.8	60.6	17.3	92.1	28.0	16.6	19.3	57.5	53.8	17.7	18.5	38.4	18.9	34.4
ForkNet[33]			-	-	63.4	36.2	93.8	29.2	18.9	17.7	61.6	52.9	23.3	19.5	45.4	20.0	37.1
CCPNet[36]			91.3	92.6	82.4	25.5	98.5	38.8	27.1	27.3	64.8	58.4	21.5	30.1	38.4	23.8	41.3
SPAwN (ours)			81.2	80.4	67.8	44.2	94.2	40.9	33.5	42.5	69.3	58.4	32.4	44.3	53.4	36.3	49.9

Table 3: **Results on NYUDv2 test set**. SUNCG + NYU means trained on SUNCG and fine-tuned on NYUDv2. Our SPAwN models hold the best and second-best overall semantic scene completion results for real-world images, on both training scenarios, when compared to previous straight-forward solutions.

We evaluated two training scenarios: training from scratch on NYUDv2 and training on SUNCG, then fine-tuning on NYUDv2. Data augmented *SPAwN* presented the best overall results in all scenarios. Without fine-tuning, the boost over SketchAware was 16.7% (6.9 p.p.). With fine-tuning, the bost over CCPNet was 20.8% (8.6 p.p.) and the result is comparable to the more expensive SISNet models.

NYUCAD. Table 4 confirms our expectation of good results due to the better quality of the ground truth related to surface volume and 2D priors. The observed mIoU boost of our model over SketchAware and CCPNet, the best previous solution in each training scenario, is 12.7% (7.0 p.p.) and 19.8% (10.9 p.p.), respectively. Our fine-tuned model even surpassed both expensive SISNet models.

4.5. Qualitative Analysis

Figure 5 presents a qualitative analysis on NYUCAD due to its better alignment between depth map and ground truth compared to NYUDv2, making it easier to perceive our predictions' high quality visually. We generated predictions with *SPAwN* trained on SUNCG and fine-tuned on NYU-

CAD, using both training-time and test-time data augmentation. We also qualitatively compare our results to a baseline SSCNet model, pretrained on SUNCG and fine-tuned to NYUCAD. This SSCNet model was trained with our one cycle learning rate protocol and achieves better results than the presented in the original paper [31] (53.3% vs 40.0% avg. IoU, respectively). *SPAwN* overall results are perceptible better than SSCNet. In column (c), it is possible to see that our projection and ensemble methods provide first-rate priors to the visible surface, with minimal prediction errors. *SPAwN* fusion strategy can complete the predictions and fix errors from priors, achieving remarkable final results. Flat objects on flat surfaces are difficult to be detected by depth-only approaches like SSCNet. Notice in the third row of figure 5 that SSCNet was unable to identify the window, while *SPAwN* predictions are almost perfect. Supplementary material presents qualitative results on the other datasets.

5. Conclusions

We introduce *SPAwN*, a novel 3D SSC network that explicitly fuses semantic priors with high-resolution structural

model	pipeline type	train	scene compl.			semantic scene completion (IoU, in percentages)											
			prec.	rec.	IoU	ceil.	floor	wall	win.	chair	bed	sofa	table	tvs	furn.	objs.	avg.
SISNet-BiSeNet[1]	iterative	NYUCAD	94.2	91.3	86.5	65.6	94.4	67.1	45.2	57.2	75.5	66.4	50.9	31.1	62.5	42.9	59.9
SISNet-DLabv3[1]			94.1	91.2	86.3	63.4	94.4	67.2	52.4	59.2	77.9	71.1	58.1	46.2	65.8	48.8	63.5
CCPNet[36]	straight-forward	NYUCAD	**91.3**	**92.6**	<u>82.4</u>	56.2	**96.6**	58.7	<u>35.1</u>	44.8	68.6	65.3	37.6	<u>35.5</u>	53.1	35.2	53.2
SketchAware[2]			<u>90.6</u>	<u>92.2</u>	**84.2**	<u>59.7</u>	94.3	**64.3**	32.6	<u>51.7</u>	<u>72.0</u>	<u>68.7</u>	<u>45.9</u>	19.0	<u>60.5</u>	<u>38.5</u>	<u>55.2</u>
SPAwN (ours)			84.5	87.8	75.6	**65.3**	<u>94.7</u>	<u>61.9</u>	**36.9**	**69.6**	**82.2**	**72.8**	**49.1**	**43.6**	**63.4**	**44.4**	**62.2**
SSCNet[31]	straight-forward	NYUCAD + SUNCG	75.4	**96.3**	73.2	32.5	92.6	40.2	8.9	40.0	60.0	62.5	34.0	9.4	49.2	26.5	40.0
CCPNet[36]			**93.4**	<u>91.2</u>	**85.1**	<u>58.1</u>	**95.1**	<u>60.5</u>	<u>36.8</u>	<u>47.2</u>	<u>69.3</u>	<u>67.7</u>	<u>39.8</u>	<u>37.6</u>	<u>55.4</u>	<u>37.6</u>	<u>55.0</u>
SPAwN (ours)			86.3	90.1	<u>78.9</u>	**77.6**	<u>95.0</u>	**68.0**	**38.1**	**67.9**	**82.2**	**77.1**	**56.8**	**50.0**	**65.7**	**46.5**	**65.9**

Table 4: **Results on NYUDCAD**. Our SPAwN models hold the best and second-best overall results on both training scenarios, when compared to previous straight-forward solutions. When fine-tuned from SUNCG, SPAwN surpasses both SISNet models, which are much more complex than ours.
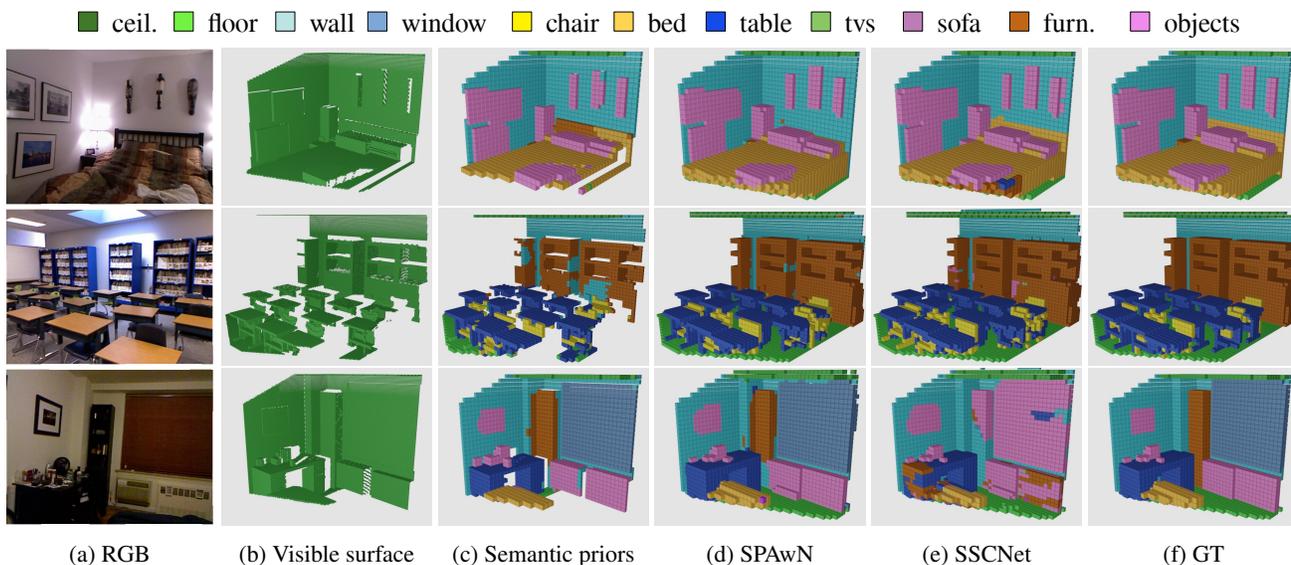

ceil. floor wall window chair bed table tvs sofa furn. objects

(a) RGB   (b) Visible surface   (c) Semantic priors   (d) SPAwN   (e) SSCNet   (f) GT

Figure 5: **SPAwN qualitative results on NYUCAD.** 2D segmentation priors projected to 3D provide good semantic guidance while SPAwN complete and refine the predictions, achieving results visually close to perfection. Compared to baseline SSCNet [31], results are much more accurate. (Best viewed in color).

information from depth maps. *SPAwN* uses as fundamental building block a novel lightweight batch normalized DDR module with higher discrimination power than its predecessors. We also introduce a the use of 3D data augmentation to the SSC task. The proposed data augmentation strategy is mode and resolution agnostic and may be applied to other SSC solutions. An ablation study with a comprehensive set of experiments shows the effectiveness of each one of our contributions. That study also includes an oracle test, which showed that the proposed solution can be further enhanced using better sources of semantic priors.

Data augmented *SPAwN* surpasses by far all previous state-of-the-art solutions with similar complexity in SSC benchmarks, in all training scenarios, achieving a boost of 19.8% (10.9 p.p.) over the best previously reported result on real images. Comparing to the recently introduced and much more expense iterative solution, the improvement is of 3.8% (2.4 p.p.).

References

[1] Yingjie Cai, Xuesong Chen, Chao Zhang, Kwan-Yee Lin, Xiaogang Wang, and Hongsheng Li. Semantic scene completion via integrating instances and scene in-the-loop. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 324–333, June 2021. 2, 7, 8

[2] Xiaokang Chen, Kwan-Yee Lin, Chen Qian, Gang Zeng, and Hongsheng Li. 3D Sketch-Aware Semantic Scene Completion via Semi-Supervised Structure Prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 4, 7, 8

[3] Aloisio Dourado, Teofilo Emidio de Campos, Hansung Kim, and Adrian Hilton. EdgeNet: Semantic Scene Completion from a single RGB-D image. In *International Conference on Pattern Recognition (ICPR)*, 2020. 2, 4, 5, 7

[4] Aloisio Dourado, Hansung Kim, Teofilo Emidio de Campos, and Adrian Hilton. Semantic scene completion from a single 360-degree image and depth map. In 15th *International Conference on Computer Vision Theory and Applications (VISAPP) - part of VISIGRAPP*, Valletta, Malta, February 27-29 2020. 2

[5] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, December 2015. 2

[6] M. Firman, O. M. Aodha, S. Julier, and G. J. Brostow. Structured prediction of unobserved voxels from a single depth image. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5431–5440, June 2016. 1, 5

[7] Martin Garbade, Yueh-Tung Chen, Johann Sawatzky, and Juergen Gall. Two Stream 3D Semantic Scene Completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 2, 7

[8] Andre Bernardes Soares Guedes, Teófilo Emídio de Campos, and Adrian Hilton. Semantic Scene Completion combining colour and depth: preliminary experiments. *CoRR arXiv*, 1802.04735, 2018. 1, 2

[9] Ruiqi Guo, Chuhang Zou, and Derek Hoiem. Predicting complete 3D models of indoor scenes. *CoRR arXiv*, 1504.02437, 2015. 5

[10] Yuxiao Guo and Xin Tong. View-Volume Network for Semantic Scene Completion from a Single Depth Image. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI)*, pages 726–732, Stockholm, Sweden, July 2018. 1, 2

[11] Saurabh Gupta, Pablo Arbeláez, and Jitendra Malik. Perceptual organization and recognition of indoor scenes from RGB-D images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 564–571, June 2013. 1, 2

[12] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from RGB-D images for object detection and segmentation. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – (ECCV)*, pages 345–360, Cham, 2014. Springer International Publishing. 2

[13] Ankur Handa, Viorica Patraucean, Vijay Badrinarayanan, Simon Stent, and Roberto Cipolla. SceneNet: Understanding real world indoor scenes with synthetic data. *CoRR arXiv*, 1511.07041, 2015. 5

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016. 1, 3, 4

[15] Jianbo Jiao, Yunchao Wei, Zequn Jie, Honghui Shi, Rynson W.H. Lau, and Thomas S. Huang. Geometry-aware distillation for indoor semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 3

[16] Josef Kittler, Mohamad Hatef, Robert P. W. Duin, and Jiri Matas. On Combining Classifiers. *PAMI*, 20(3):226–239, Mar. 1998. 4, 5

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 1, 3

[18] Jie Li, Kai Han, Peng Wang, Yu Liu, and Xia Yuan. Anisotropic Convolutional Networks for 3D Semantic Scene Completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[19] Jie Li, Yu Liu, Dong Gong, Qinfeng Shi, Xia Yuan, Chunxia Zhao, and Ian Reid. RGB-D based dimensional decomposition residual network for 3D Semantic Scene Completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 4

[20] Siqi Li, Changqing Zou, Yipeng Li, Xibin Zhao, and Yue Gao. Attention-based Multi-Modal Fusion Network for Semantic Scene Completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11402–11409, Apr. 2020. 2, 4

[21] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 3

[22] Shice Liu, YU HU, Yiming Zeng, Qiankun Tang, Beibei Jin, Yinhe Han, and Xiaowei Li. See and Think: Disentangling Semantic Scene Completion. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Conference on Neural Information Processing Systems (NeurIPS)*, pages 263–274. Curran Associates, Inc., 2018. 2, 7

[23] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, June 2015. 2

[24] Calvin Murdock, Zhen Li, Howard Zhou, and Tom Duerig. Blockout: Dynamic model selection for hierarchical deep

networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2583–2591, 2016. 3

[25] Seong-Jin Park, Ki-Sang Hong, and Seungyong Lee. RDFNet: RGB-D multi-level residual feature fusion for indoor semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2017. 2, 3

[26] Xiaojuan Qi, Renjie Liao, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. 3D graph neural networks for RGB-D semantic segmentation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5209–5218, 10 2017. 1, 2

[27] X. Ren, L. Bo, and D. Fox. RGB-(d) scene labeling: Features and algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2759–2766, June 2012. 1, 2

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, Cham, 2015. Springer International Publishing. 4

[29] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGB-D images. In *European Conference in Computer Vision (ECCV)*, pages 746–760, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. 2, 3, 5

[30] Leslie N. Smith. A disciplined approach to neural network hyper-parameters: Part 1 - learning rate, batch size, momentum, and weight decay. *CoRR arXiv*, 1803.09820, 2018. 6

[31] Shuran Song, Fisher Yu, Andy Zeng, Angel X. Chang, Manolis Savva, and Thomas Funkhouser. Semantic Scene Completion from a Single Depth Image. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 3, 4, 5, 7, 8

[32] Weiyue Wang and Ulrich Neumann. Depth-aware CNN for RGB-D segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2

[33] Y. Wang, D. J. Tan, N. Navab, and F. Tombari. Forknet: Multi-branch volumetric semantic completion from a single depth image. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8607–8616, 2019. 2, 7

[34] Jiahui Zhang, Hao Zhao, Anbang Yao, Yurong Chen, Li Zhang, and Hongen Liao. Efficient Semantic Scene Completion network with spatial group convolution. In *The European Conference on Computer Vision (ECCV)*, September 2018. 7

[35] Liang Zhang, Le Wang, Xiangdong Zhang, Peiyi Shen, Mohammed Bennamoun, Guangming Zhu, Syed Afaq Ali Shah, and Juan Song. Semantic scene completion with dense CRF from a single depth image. *Neurocomputing*, 318:182–195, Nov. 2018. 2

[36] Pingping Zhang, Wei Liu, Yinjie Lei, Huchuan Lu, and Xiaoyun Yang. Cascaded context pyramid for full-resolution 3D Semantic Scene Completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2, 7, 8

[37] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI Conference on Artificial Intelligence*, volume 34, pages 13001–13008, April 2020. 3