# 3D Geometry-Aware Semantic Labeling of Outdoor Street Scenes

Yiran Zhong Research School of Engineering, ANU Data61, CSIRO Canberra, Australia

Yuchao Dai School of Electronics and Information Northwestern Polytechnical University Xi'an, China Hongdong Li Research School of Engineering, ANU Australian Centre for Robotic Vision Canberra, Australia

*Abstract*—This paper is concerned with the problem of how to better exploit 3D geometric information for dense semantic image labeling. Existing methods often treat the available 3D geometry information (e.g., 3D depth-map) simply as an additional image channel besides the R-G-B color channels, and apply the same technique for RGB image labeling. In this paper, we demonstrate that directly performing 3D convolution in the framework of a residual connected 3D voxel top-down modulation network can lead to superior results. Specifically, we propose a 3D semantic labeling method to label outdoor street scenes whenever a dense depth map is available. Experiments on the "Synthia" and "Cityscape" datasets show our method outperforms the state-ofthe-art methods, suggesting such a simple 3D representation is effective in incorporating 3D geometric information.

#### I. INTRODUCTION

Semantic labeling (semantic segmentation) aims to assign class labels (e.g., "cars", "road", "building", "pedestrian") to pixels in an image. It is an important task in computer vision and pattern recognition, which has found wide-range applications in the areas such as autonomous driving [1], robot SLAM[2], and augmented reality [3].

Deep Convolutional Neural Networks (CNNs) have gained tremendous success in almost all high-level vision tasks such as image classification, object detection, as well as semantic labeling [4][5][6]. The 2D convolution is defined in the image coordinate, where the filter is applied in the neighborhood defined by image pixel distance. Deep encoder-decoder (SegNet [5], dilated convolution (DeepLab-LargeFOV [6]) have also been proposed under the same framework. The success of these models mainly lies in their general modeling ability for complex unseen visual scenes.

To further improve the performance, deeper and wider networks [7] have been proposed, which require massive labeled data during training. Even though these models have achieved state-of-the-art performance on various benchmarking datasets, they do not harness the full potentials of available depth/3D clues for semantic segmentation. Geometric information provides crucial and discriminative semantic cues for color images. Depth maps generally provide complement information to color images, where the 3D structure of the observed scene has been encoded naturally [8]. Therefore, semantic labeling will benefit from the availability of depth information. For indoor scenes, Hazirbas *et al.*[8] proposed a deep auto-encoder network for semantic labeling, where the encoder consists of two branches of networks that simultaneously extract features from color and depth images and fuse depth features into the color feature maps as the network goes deeper. Furthermore, Ma *et al.*[9] proposed to leverage the consistencies between multi-view semantic labeling.

However, most existing works have focused on indoor scene labeling where the size of the scene is limited. For outdoor street scene semantic labeling using depth information is difficult due to the following reasons: 1) difficulty in accurate depth acquisition for outdoor scenes; 2) large variation in scene scales; and 3) lacking of outdoor training datasets with dense depth information.

In this paper, we advocate the benefit of using 3D information for outdoor labeling, and propose a simple and efficient way to use the 3D information. Specifically, we propose a direct way to represent RGB-D image in its natural 3D space, i.e., the way human sense the surrounding 3D world. Given a color image and the associated depth map (from stereo vision or from LIDAR), we transform the color image into 3D voxel space defined by the 3D position of each pixel, which enables subsequent 3D convolution to cater the 3D geometry in extracting semantic feature maps and thus achieves *3D geometry aware semantic labeling*.

To learn a geometry-aware representation, we propose a light-weight 3D Res-TDM (Residual connected Top-Down Modulation) structure that can squeeze 3D geometric information from depth map and own high resistance to noise and errors. We have performed experiments on the SYNTHIA dataset with ground truth depth map and the Cityscape dataset with computed disparity map. Experimental results demonstrate that our method outperforms the state-of-the-art semantic labeling methods, which indicates the success of our 3D voxel representation in effectively and efficiently encoding 3D geometric information.

The main contributions of the paper can be summarized as: 1) A natural and direct 3D representation to encode RGB-D

- data, thus representing the semantic cues in 3D;
- 3D convolution to exploit the geometric constraint for semantic labeling, enabling 3D geometry aware semantic labeling;
- A light weighted 3D res-TDM structure that can squeeze 3D geometric information from depth map and own high resistance to noises and errors.

#### II. RELATED WORK

**Semantic labeling:** Before the era of deep learning, semantic segmentation has been widely formulated as CRF with handcraft features and low-level vision cues. The breakthrough in deep learning has also been brought to semantic labeling to learn the nonlinear mapping from image to dense labeling in an end-to-end manner. The most noticeable deep convolutional network based semantic labeling method is FCN[4], which takes advantage of existing image classification architectures [10] [11] [12]. However, the decoder phase of FCN is relatively simple that makes it difficult to train. SegNet[5] tackles the above weakness by using an auto-encoder structure. Dilated convolution [6] has also been introduced to effectively enlarge the field of view of filters to incorporate larger context without increasing the number of parameters or the amount of computation.

RGB-D semantic labeling: Depth information has been used as an important cue to refine semantic labeling in computer vision [13]. Zhang et al.[14] designed hand-crafted depth features such as surface normals, height above ground and neighboring smoothness and put them into a classifier. Saurabh et al.[15] geocentrically encoded depth into disparity, height and angle as a HHA representation and proposed a 2.5D proposal for object detection and semantic segmentation. Lai et al.[16] utilizes HMP3D features in an MRF framework to label objects in 3D scenes. More recently, Li et al.[17] fused contextual information from RGB and depth channels by stacking convolutional layers with an LSTM layer, which memorizes both short- and long-range spatial dependencies in an image along vertical direction. Another LSTM-F layer has also been used to integrate contexts from different channels and bi-directional propagation is performed to fuse vertical contexts. Hazirbas et al.[8] proposed a simpler network with auto-encoder style, where two encoders are used to extract features from RGB image and depth image individually and one decoder is applied to decode RGB-D channels. Extracted depth features are fused with RGB in every encoder layer. In these works, color features and depth features are coupled in a human-designed way, which may fail to exploit the strong correlation between color image and depth map.

**3D** convolution: Volumetric (i.e., spatially 3D) convolution has been successfully used in video analysis ([18]). VoxNet [19] and 3D ShapeNet [20] are two pioneer works in applying 3D convolution on voxelized 3D shapes. Very recently, Song *et al.*[21] introduced 3D voxel representation of volumetric occupancy and simultaneously performed scene completion and scene parsing for indoor scenes. However, both works only preserve 3D structure information for object recognition and discard color information in 3D convolution. Moreover, the output resolution of [21] is only  $36 \times 60 \times 60$ , which is insufficient for outdoor applications and it requires large labeled data for network training. Multi-view strategy has also been leveraged to exploit 3D geometry information. MVCNN [22] projects 3D point clouds onto different image planes and converts each view image into CNN features. However, this strategy could not be applied to outdoor semantic labeling task straightforwardly due to the difficulty in warping small objects between different views.

By contrast to the above works, we propose to make use of color information as well as 3D structural information for dense semantic labeling under an unified framework. Our lightweight network architecture also allows us to increase the output dimension with a reasonable scale and can be trained from scratch with only thousands of samples.

#### III. OUR APPROACH

Here, we describe our geometry-aware semantic labeling framework by performing 3D convolution in the framework of 3D voxel convolutional neural network. First, by contrast to existing methods that simply treating depth map as an additional channel besides the R-G-B channels, we represent the input RGB-D images in 3D voxel representation, where each voxel is associated with color. Then a top-down module is proposed to exploit the rich 3D geometric information for outdoor scene semantic labeling, where 3D convolution is performed to extract 3D geometry aware features.

## A. 3D Representation

Given RGB-D images, existing methods either represent the generic 3D point clouds with volumetric or multi-view representation. The volumetric representation encodes a 3D shape as a 3D tensor of binary or real values while the multiview representation encodes a 3D shape as a collection of renderings from multiple viewpoints. However these representations are mainly designed for indoor applications and cannot cope with outdoor scenarios for the following reasons : 1) difficulty in accurate depth acquisition; 2) large variation in scene scales; and 3) lack of outdoor training dataset with dense depth information. Furthermore, for a typical driving scene, the depth ranges from 0.5 meters to infinity (i.e., the sky), which makes it impossible to discretize depth values into a certain range. Therefore direct voxelizing in 3D space for outdoor street scene is infeasible.

To cater the above difficulties, we propose a new and yet direct 3D voxel representation for outdoor street scenes. Specifically, instead of resorting to the XYZ space for 3D point clouds, we propose to combine the image coordinate and the disparity directly, thus UVD space, where (U, V) index the 2D image coordinate while D indexes the discrete disparity. At a first glance, this 3D voxel representation may introduce severe distortion in 3D representation. Here we demonstrate that while providing simplicity in representation, the UVD 3D voxel representation also owns much desired geometric property as in the original XYZ space.

**Theorem 1.** Any order curve in the XYZ 3D space corresponds to a 3D curve with the same order in the UVD 3D space.

*Proof.* Without loss of generality, we take the second order surface in 3D as an example. A second order surface in the XYZ 3D space is defined by the following equation:

$$[X_i, Y_i, Z_i, 1] \mathbf{A} [X_i, Y_i, Z_i, 1]^T = 0,$$
(1)

where  $(X_i, Y_i, Z_i)$  defines the 3D points on the surface and  $\mathbf{A} \in \mathbb{R}^{4 \times 4}$  indexes the 3D surface. The *UVD* space and the *XYZ* space are connected via perspective projection:

$$X_{i} = \frac{(u_{i} - u_{0})}{f_{x}} Z_{i}, Y_{i} = \frac{(v_{i} - v_{0})}{f_{y}} Z_{i}, Z_{i} = \frac{fb}{d_{i}}, \quad (2)$$

where  $f_x, f_y, u_0, v_0$  are the intrinsic parameters of the camera while f, b define the transformation from disparity  $d_i$  to 3D coordinate  $Z_i$ . By substituting these relations into the 3D surface and re-organizing the equation, we have

$$\begin{bmatrix} u_{i} - u_{0} \\ v_{i} - v_{0} \\ 1, \\ d_{i} \end{bmatrix}^{T} \begin{bmatrix} \frac{fb}{f_{x}} & 0 & 0 & 0 \\ 0 & \frac{fb}{f_{y}} & 0 & 0 \\ 0 & 0 & fb & 0 \\ 0 & 0 & \frac{fb}{f_{y}} & 0 & 0 \\ 0 & \frac{fb}{f_{y}} & 0 & 0 \\ 0 & 0 & fb & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{i} - u_{0} \\ v_{i} - v_{0} \\ 1 \\ d_{i} \end{bmatrix} = 0.$$
(3)

It is thus clear that a second order surface in XYZ space has been transformed to another second order surface in UVDspace. The above proof could be extended to any order 3D surface directly.

Therefore, we can conclude that any order parametric surfaces defined in the XYZ space have a corresponding surface of the same order in the UVD space. In other words, the transformation from XYZ space to UVD space is curve order preserving.

### B. 2D convolution VS 3D convolution

State-of-the-art semantic labeling methods use deep convolutional network to learn the nonlinear mapping from image to dense semantic labeling, where the convolution is conducted in a 2D manner. As the neighboring relation is defined on the image plane, the 2D convolution may fail to extract feature with 3D geometry aware. Instead, 3D convolution in the 3D voxel space could integrate the appearance cues in 3D geometry aware manner, *i.e.*, the 3D distance has been catered in convolution.

Given a color image, the 2D convolution is expressed as Eq 4. The value of an unit at position (u, v) in the  $i^{th}$  feature map is denoted as  $p_i^{u,v}$ ,

$$p_i^{uv} = \sum_k \sum_{m=0}^{M_i - 1} \sum_{n=0}^{N_i - 1} w_{ik}^{mn} p_{(i-1)k}^{(u+m)(v+n)},$$
(4)

where  $w_{ik}^{mn}$  is the coefficient at the position (m, n) of the kernel connected to the  $k^{th}$  feature map. M, N are the height and width of the kernel. When the convolution is conducted in

3D, the value of an unit at position (u, v, d) in the  $i^{th}$  feature map denoted as  $p_i^{u,v,d}$  is given by

$$p_i^{uvd} = \sum_k \sum_{m=0}^{M_i - 1} \sum_{n=0}^{N_i - 1} \sum_{l=0}^{L_i - 1} w_{ik}^{mnl} p_{(i-1)k}^{(u+m)(v+n)(d+l)}, \quad (5)$$

where d is the third dimension of the feature map and  $L_i$  is the size of the 3D kernel along the third dimension. 3D convolution can extract features from both spatial and disparity dimensions.

In Fig. 1, we compare 2D convolution and 3D convolution for an outdoor street scene. As observed from the illustration, the 2D convolution extracts feature in neighborhood defined on the image plane, which could involve points far away in 3D space. By contrast, 3D convolution in the *UVD* space succeeds to extract features in a 3D geometry aware manner.



Fig. 1: Illustration of 2D convolution and 3D convolution for semantic labeling. The left image demonstrates the widely used 2D convolution in extracting feature maps while the right image illustrates the corresponding 3D convolution conducted in the 3D voxel space. Note that the natural neighborhood relation is not preserved in the projection from 3D to 2D.

#### C. Network Architecture

Our goal is to assign each 3D point with a class label. A natural solution is to do 3D convolution on these point clouds. Also, since small objects such as traffic lights and signs play equally important role in semantic labeling, we adopt the idea of Top-Down Modulation (TDM) [23]. We not only convert it to 3D, but also modify it to better suit for our case. Note that in our 3D representation, most voxelized 3D labels are 0s. For these points, identity mappings are optimal. Therefore we swap the lateral connection between Bottom-up features and Top-down features with residual connection, and let the solvers simply drive the weights of the multiple nonlinear layers toward zero to approach identity mappings. Formally, we define a block from Top-down path:

$$\mathbf{y} = \mathcal{F}(\mathbf{x}_{C_i}, \{\mathbf{W}_i\}) + \mathbf{x}_{DC_i},\tag{6}$$

where y denotes the output of residual connection,  $\mathbf{x}_{C_i}$  is the output from the *i*-th convolution layer and  $\mathbf{x}_{DC_i}$  is output from the *i*-th deconvolution layer. The function  $\mathcal{F}(\mathbf{x}, {\mathbf{W}_i})$  is the residual mapping to learn. + represents element-wise addition. Thus the dimensions of  $\mathbf{x}_{R_i}$  and  $\mathbf{x}_{DC_i}$  must be equal as in Eq. 6.



Fig. 2: A nutshell of our 3D geometry-aware semantic labeling framework. The input RGB-D images are converted to the 3D voxel representation.  $C_i$  denotes the 3D convolution layer that encodes geometric and contextual information,  $R_i$  is residual module that connects low level features to the top-down pathway.  $DC_i$  is the 3D deconvolution layer to decode geometric and contextual information. We achieve a 3D semantic labeling, which could be projected to 2D for the sake of comparison.

In Fig. 2, we present a nutshell of our overall architecture of the proposed network. Given a color image and its corresponding disparity map, we first represent the RGB-D in the 3D UVD space as defined in Section 3.1. In the Bottom-up phase, the 3D volume  $(H \times W \times (D+1) \times Ch)$  passes through a series of 3D convolutional layers  $(C_i)$  with the same kernel size  $3 \times 3 \times 3$  and a stride 2 until achieving an encoded feature volume with dimension  $(1/16)H \times (1/16)W \times (1/16)(D+1) \times F$ , where H, W, D, Ch, F represent the height, width, disparity levels, and number of channels and features respectively. In the Top-down phase, a mirrored process scales up the encoded feature volume back to the original size by swapping the 3D convolution with 3D deconvolution. For each scale, we apply our Res-TDM with a residual module  $R_i$ . Each  $R_i$  consists of two 3D convolution layers with the same kernel size  $3 \times 3 \times 3$ and stride 1.

We employ the cross-entropy loss given by Eq. 7 as our loss function for training the network.

$$L(\mathbf{w}) = -\frac{1}{N+1} \sum_{n} [y_n \log \hat{y}_n + (1-y_n) \log(1-\hat{y}_n)]$$
(7)

where  $\hat{y}_n = g(\mathbf{w}\dot{\mathbf{x}}_n)$  with logistic function g(z). w is the vector of weights and each sample is labeled by n = 0, 1, 2, ..., N - 1. Note that there is a large variation in the number of pixels for each class. Despite of the imbalance distribution of valid labels, we only have 1/D valid labels in total. In other words, if the network predict all zeros, it can still achieve a training accuracy higher than 95%. In order to avoid our network drop to this local minimum, we apply a residual module R directly from the input that impose the network only to learn parameters around the areas with non-zero input.

Training our end-to-end pixel-wise semantic labeling network is very straightforward, which can be trained under the supervision of ground-truth semantic labels. Supervision is applied on the volumized 3D predictions and labels. All void 3D points (e.g., points before an object or behind an object) cannot be ignored and should be also given a void label 0. In the prediction phase, we perform max pooling along the disparity dimension to convert the 3D volume back to a 2D image and calculate the errors. This strategy can crease the robustness when dealing with noises and errors on disparity maps. For example, there is no guarantee that our network will predict the right label at the exact disparity level. When the disparity map is noisy, points with the same labels may have very different disparity levels. In this case, the network may predict the right label on the similar disparity level rather than the noise one.

#### IV. EVALUATION

In this section, we evaluate our proposed method with a comparison to alternative approaches and present an ablation study to better understand the proposed framework. Our method is evaluated on both synthetic and real datasets.

#### A. Dataset

For synthetic data, we use the SYNTHetic Collection of Imagery and Annotations (SYNTHIA) dataset [24], which contains 3 subsets: synthia-rand-cvpr16, synthia-rand-cityscapes and synthia-video-sequence. We choose the synthia-rand-cityscapes subset for experiments. It consists of 23 classes and a total of 9400 frames of outdoor scene with different weather and lighting conditions as well as randomly generated viewing angles. Since the dataset does not provide training and testing split, we randomly select 6000 frames for training, 1900 frames for validation and the remaining 1500 frames for testing. We manually convert the given ground truth depth maps to scaled inverse depth map in the range of [0, 191] and resize the input image to  $80 \times 128$ .

For real data experiment, we use the Cityscape dataset [25], which contains 5,000 stereo frames of fine annotated ground truth semantic labels. We choose 2975 frames for training and use 500 frames for testing. In experiments, we compute the disparity map by using state-of-the-art stereo matching method, which is truncated to the range [0, 111]. The input images are resized to the resolution of  $256 \times 512$ .

**Data augmentation** We employ the mirror manipulation to augment the training examples for both datasets, since it maintains the geometry relationships.

## B. Optimization

The proposed network architecture was implemented with Tensorflow [26]. We employed the RMSProp [27] with a constant learning rate of  $1 \times 10^{-3}$  to optimize all models in end-to-end manner. For the "Synthia" dataset, we normalized input images' RGB values to [-1, 1] and trained our network from a random initialization for 50 epochs, which took 50 hours to converge by using a single NVIDIA Pascal Titan-X GPU and 1 second per frame in testing phase. However, the testing global accuracy climbs up to over 80% within one epoch. For the Cityscape dataset, we trained our network (S3D) with color input for 30 epochs. In order to fit the 12G memory, we reduce the number of disparity levels to 48. We also trained our S3D network with feature input. The features were extracted from Resnet-38[7] with the same input dimension. The input feature dimension of our network is  $32 \times 64 \times 512$ . We added 3 extra upsampling layers in order to match the output resolution. The network converged quickly within 14 epochs. Note we did not use any post-processing to refine the results.

#### C. Evaluation metric

We measure the semantic labeling performance of our network with three metrics. Denote the total number of classes as k,  $p_{ij}$  as the amount of pixels belonging to class i which are predicted to be class j, the Global accuracy  $G = \frac{\sum_i p_{ii}}{\sum_i \sum_j p_{ij}}$  measures the percentage of pixels correctly classified in the dataset. The Class Average Accuracy  $C = \frac{1}{k+1} \sum_i \frac{p_{ii}}{\sum_j p_{ij}}$  normalizes the accuracy over the classes, therefore all classes share the same weight under this metric. Mean intersection over union  $mIoU = \frac{1}{k+1} \sum_i \frac{p_{ii}}{\sum_j p_{ij} + \sum_j p_{ij} - p_{ij}}$  is used in the Cityscapes benchmark [25]. It is a more strict metric than class average accuracy since it penalizes false positive predictions.

### D. Experimental results

**Results on Synthia.** In Table I, we quantitatively compare our method (S3D) with state-of-the-art RGB semantic segmentation approach "SegNet" [5] and RGB-D based approach "FuseNet" [8]. For SegNet and FuseNet, we use the same input size and initialize the network parameters from the VGG model pre-trained on ImageNet. We train SegNet for 790 epochs and 230 epochs for FuseNet. For FuseNet, we use the same scaled inverse depth maps to train our network. Our method significantly outperforms competing methods with a notable margin under all three metrics: **18.6**% on class average accuracy, **7.6**% on global accuracy and **17.8**% on mIoU. Note that there are 4 classes never show up in testing set, so we remove them from the table and during the error calculation. In Fig. 3, we present qualitative comparison between our method and state-of-the-art methods on the Synthia dataset, which clearly demonstrates the superior performance of our method.

Results on Cityscapes. Quantitative comparison with stateof-the-art semantic labeling methods on the Cityscapes dataset is shown in Table II. The weight of FuseNet and SegNet are initialized from the VGG model trained on ImageNet. We also compare our method with the top performing one on the Cityscapes benchmark: ResNet-38[7]. Given RGB-D pair as input, we achieve similar performance with FuseNet. However, by swapping RGB image with trained features, our method outperforms all competing methods with a margin 1.2%, 0.6%, 1.0% for class average accuracy, global accuracy and mIoU respectively. The margin is not as clear as previous one is due to the noise and errors in the disparity map. However, our algorithm still successfully squeezed useful information from it and increased the performance. Advanced disparity recovery algorithm [28] should lead to better performance. In Fig. 4, we present qualitative comparison between our framework and state-of-the-art methods on the Cityscapes dataset, which proves the superiority of our method.

Ablation study To better understand the effectiveness of our 3D voxel representation, we perform ablation analysis and present the results in Table III. S2D is the 2D version of our algorithm that replaces all 3D convolutions with 2D ones, where we stack the RGB image and disparity map into 4 channel input and plug into the S2D. S3D (Depth only) is the one with colorless "point clouds" which only provides shape information. According to this study, 3D voxel representation significantly improves the performance by 20.9% in mIoU.

# V. CONCLUSION

In this paper, we have proposed a 3D voxel representation to integrate both appearance and depth information and a corresponding light-weight 3D Res-TDM network architecture for 3D geometry aware semantic segmentation. Our method provides an efficient and effective way to use geometric information to achieve better semantic labeling. Experiments on the "Synthia" and "Cityscape" datasets demonstrate that direct 3D convolution with our light-weight Res-TDM network can lead to superior performance, suggesting that such a simple 3D representation with Res-TDM is effective in incorporating 3D geometric information.

#### ACKNOWLEDGMENT

We gratefully acknowledge the support of NVIDIA Corporation with donation of TITAN Xp GPU used for this research, as well a NVIDIA Drive-PX2 platform for an autonomous driving project. YZ's PhD scholarship is funded by CSIRO Data61. Y. Dai was supported in part by National 1000 Young Talents Plan of China, Natural Science Foundation of China (61420106007, 61671387), and ARC grant (DE140100180). H. Li's work is funded in part by Australia ARC Centre of Excellence for Robotic Vision (CE140100016).



Fig. 3: Quality comparison on the SYNTHIA dataset We select images with different lighting and weather conditions as well as different viewing angles. Our method (d) shows superior performance, particularly it generates sharp boundaries for small objects. FuseNet (e) and SegNet (f) achieve similar performance but with the help of disparity map, FuseNet (e) captures more small objects such as pedestrians and poles.

TABLE	I:	Performance	evaluation	on	the	SYNTHIA	dataset
-------	----	-------------	------------	----	-----	---------	---------

Method	sky	Building	Road	Sidewalk	Fence	Vegetation	Pole	Car	Traffic sign	Pedestrian	Bicycle	Motorcycle	Road-work	Traffic light	Rider	Bus	Wall	Lanemarking	Class avg.	Global avg.	mloU
SegNet[5]	95.5	93.3	85.0	87.2	24.9	79.9	16.9	60.8	0.2	50.2	1.4	10.6	40.3	0.0	11.2	65.5	18.6	45.9	43.7	82.6	36.7
FuseNet[8]	92.4	94.5	79.9	70.2	35.6	73.0	29.9	64.4	2.5	57.5	2.8	9.4	46.4	2.6	16.4	60.9	13.7	20.9	42.9	78.1	35.9
S3D(ours)	97.4	97.1	91.8	91.2	59.6	90.7	47.8	87.6	15.5	72.9	13.5	36.4	72.24	31.7	32.6	83.3	58.2	42.1	62.3	90.2	54.5

TABLE II: Performance evaluation on Cityscapes validation set

Method	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic light	Traffic sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	Class avg.	Global avg.	mIoU
SegNet[5]	97.4	82.0	92.5	35.2	35.8	40.9	8.2	40.5	93.2	62.6	96.2	71.9	11.8	92.7	46.3	41.6	27.2	9.9	61.2	55.1	90.5	45.8
ResNet-38[7]	97.9	81.1	93.6	62.0	58.4	41.9	55.4	62.5	94.2	63.8	92.3	79.1	54.9	94.7	74.5	76.5	66.1	61.4	75.1	72.9	92.2	63.3
FuseNet[8]	88.6	82.1	93.3	25.4	48.2	42.7	0.5	47.0	94.5	38.5	96.8	75.8	0.5	93.6	56.9	2.2	12.4	0.0	60.2	50.5	87.4	39.1
S3D(RGB-d)	94.5	72.2	86.1	15.7	17.7	34.0	38.1	52.3	91.1	65.9	96.2	64.3	8.0	86.9	22.0	20.5	14.6	3.5	28.2	48.2	86.9	39.1
S3D(feature-d)	98.0	87.4	93.3	66.2	71.3	46.8	60.2	62.3	95.2	67.4	92.8	78.0	41.8	91.8	78.7	81.2	73.3	47.5	75.0	74.1	92.8	64.3

TABLE III: Ablation study on the SYNTHIA dataset

Method	sky	Building	Road	Sidewalk	Fence	Vegetation	Pole	Car	Traffic sign	Pedestrian	Bicycle	Motorcycle	Road-work	Traffic light	Rider	Bus	Wall	Lanemarking	Class avg.	Global avg.	mloU
S2D(RGB-d)	98.0	92.0	62.6	82.2	41.6	80.9	33.4	68.6	2.5	66.9	0.3	8.9	60.2	0.9	1.6	37.0	0.0	16.6	41.9	77.9	33.6
S3D(d only)	100.0	98.2	91.4	91.7	59.5	92.6	52.5	83.8	0.3	80.8	12.6	2.8	46.4	0.7	25.9	76.0	38.0	0.1	52.9	89.9	47.0
S3D(RGB-d)	97.4	97.1	91.8	91.2	59.6	90.7	47.8	87.6	15.5	72.9	13.5	36.4	72.24	31.7	32.6	83.3	58.2	42.1	62.3	90.2	54.5

## REFERENCES

 Nick Schneider, Lukas Schneider, Peter Pinggera, Uwe Franke, Marc Pollefeys, and Christoph Stiller. Semantically guided depth upsampling. In *German Conference on Pattern Recognition*, 2016. [2] Javier Civera, Dorian Gálvez-López, L. Riazuelo, Juan D. Tardós, and J. M. M. Montiel. Towards semantic slam using a monocular camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1277–1284, Sept 2011.



Fig. 4: Qualitative evaluation on the Cityscapes dataset. Our method with feature and disparity inputs (d) clearly outperforms the competing methods. The shape of pedestrians and poles are well preserved in our predictions. We shall see that FuseNet is effected by the noisy disparity map that has worse performance than SegNet. Note the invalid label is colored with black.

- [3] Tamás Matuszka, Gergő Gombos, and Attila Kiss. A new approach for indoor navigation using semantic webtechnologies and augmented reality. In Virtual Augmented and Mixed Reality. Designing and Developing Augmented and Virtual Environments, pages 202–210, 2013.
- [4] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional models for semantic segmentation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 3431–3440, 2015.
- [5] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(12):2481–2495, Dec 2017.
- [6] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.*, PP(99):1–1, 2017.
- [7] Z. Wu, C. Shen, and A. van den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. arXiv:1611.10080, 2016.
- [8] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In *Proc. Asian Conf. Comp. Vis.*, pages 213–228, 2017.
- [9] L. Ma, J. Stückler, C. Kerl, and D. Cremers. Multi-view deep learning for consistent semantic mapping with rgb-d cameras. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 598–605, Sept 2017.
- [10] 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, *Proc. Adv. Neural Inf. Process. Syst.*, pages 1097–1105. Curran Associates, Inc., 2012.
- [11] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556, 2014.
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385, 2015.
- [13] A. C. Müller and S. Behnke. Learning depth-sensitive conditional random fields for semantic segmentation of RGB-D images. In *IEEE International Conference on Robotics and Automation*, pages 6232–6237, May 2014.
- [14] C. Zhang, L. Wang, and R. Yang. Semantic segmentation of urban scenes using dense depth maps. In *Proc. Eur. Conf. Comp. Vis.*, pages 708–721, 2010.
- [15] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik. Learning rich features from rgb-d images for object detection and segmentation. In *Proc. Eur. Conf. Comp. Vis.*, pages 345–360, 2014.

- [16] K. Lai, L. Bo, and D. Fox. Unsupervised feature learning for 3d scene labeling. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3050–3057, 2014.
  [17] Z. Li, Y. Gan, X. Liang, Y. Yu, H. Cheng, and L. Lin. RGB-D
- [17] Z. Li, Y. Gan, X. Liang, Y. Yu, H. Cheng, and L. Lin. RGB-D scene labeling with long short-term memorized fusion model. *CoRR*, abs/1604.05000, 2016.
- [18] Abhijit Kundu, Vibhav Vineet, and Vladlen Koltun. Feature space optimization for semantic video segmentation. In Proc. IEEE Conf. Comp. Vis. Patt. Recogn., pages 3168–3175, June 2016.
- [19] D. Maturana and S. Scherer. VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 922–928, 2015.
- [20] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 1912–1920, 2015.
- [21] S. Song, F.er Yu, A. Zeng, A. X Chang, M. Savva, and T. Funkhouser. Semantic scene completion from a single depth image. *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 190–198, 2017.
- [22] H. Su, S. Maji, E. Kalogerakis, and E. G. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proc. IEEE Int. Conf. Comp. Vis.*, pages 945–953, 2015.
- [23] A. Shrivastava, R. Sukthankar, J. Malik, and A. Gupta. Beyond skip connections: Top-down modulation for object detection. *CoRR*, abs/1612.06851, 2016.
- [24] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and An. Lopez. The SYNTHIA Dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 3234–3243, 2016.
- [25] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The Cityscapes dataset for semantic urban scene understanding. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 3213–3223, 2016.
- [26] Martín Abadi, Ashish Agarwal, and Paul Barham *et al.*. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467, 2016.
- [27] T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning, 4:26–31, 2012.
- [28] Y. Zhong, Y. Dai, and H. Li. Self-supervised learning for stereo matching with self-improving ability. *CoRR*, abs/1709.00930, 2017.