A Hybrid Method of Cascaded Features for RGB-D Semantic Segmentation

Chenxu Wang¹, Jichao Jiao²*, Ning Li³, Zhongliang Deng⁴ and Wei Xu⁵

¹,²,³,⁴School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing, 100876, China
⁵ The 22th research institute of China electronics technology group corporation, Beijing, China
*Corresponding author’s e-mail: jiaojichao@bupt.edu.cn

Abstract. Fully Convolution Network and its following works has achieved the state-of-art performance on the task of RGB semantic segmentation. However, there still lacks an effective method to sufficiently leverage geometric information of depth image to accomplish RGB-D semantic segmentation. To this end, this paper proposed a new method containing two parts: 1) a simple but useful way based on Fast Marching Method to inpaint area of no-measured-depth pixels, which produces better result than standard dataset we used in this paper; 2) a new fusion architecture of CNN in which we fuse RGB and depth features in shallow layers to make geometry information from depth image a better assistant to RGB semantic segmentation. Besides, we add a feature filter architecture to help model choose the most discriminative features and punish useless or repeating features for better hierarchical expression. The original RGB image and inpainted depth image are fed into two feature extracting streams, one for each modality, and get fused in the fusion layer which is consecutively followed by deeper encoder network and decoder network. We evaluate our method on SUN RGB-D and NYUD dataset and experimental result shows that the proposed model has priority on RGB-D semantic segmentation.

1. Introduction

Indoor scene understanding based on RGB-D camera has drawn increasing attention for its key role in the robotic and computer vision. For example, it helps SLAM (Simultaneous Location and Mapping) system equipped with a semantic understanding of surrounding environments. The richer the environment model and the higher its semantic level, the more useful it becomes for a robot in order to perform autonomous tasks. Although 2D semantic segmentation has made a rapid achievement due to many creative works [1-4] using the convolutional neural network, many problems are remained to be solved in semantic segmentation based on RGB-D images, one of the main datatypes of SLAM system.

A few works such as [5, 6] solve RGB-D semantic segmentation from a perspective of three dimensions. X Qi et al. [6] proposed an end-to-end 3D graph neural network that builds a K-nearest neighbor graph directly and learns its representation from 3D points cast from 2D pixels based on depth information. Most existing works [7-9] treat RGB-D semantic segmentation as 2D segmentation problem. Several methods [1, 5] directly feed CNN with depth image as the fourth channel, which has a little improvement of precision. Some researchers [8, 10] find a way to encode depth image into three channels so that a depth image can be input like an RGB image. However, Song et al. [11]
mentioned that a large number of low-level filters are either useless or ignored when finetuning the network on Horizontal Height Angle (HHA) data.

In addition, though depth sensors have been becoming increasingly popular, there are missing values of pixels in the depth image due to kinds of reasons. For example, Fig.1 shows that the hole area destroys complete object structure information. The SUN RGB-D dataset includes in-painted depth images which are obtained by making use of the multi-view fusion technique[12]. However, raw depth images from the aforementioned camera consist of many invalid measurements, therefore in-painted depth images have many false values.

In order to efficiently take advantage of spatial information in depth to assist in RGB-D semantic segmentation, we proposed a hybrid architecture that includes two modules for the RGB-D semantic segmentation. The first module is based on fast marching method (FMM) to fix the missing values using neighboring homologous pixels searched in RGB image. Moreover, we fuse and sift low-level features from RGB and Depth branches in a shallow encoder layer and decode features in following upsampling layer. This part will be discussed in detail in section III.

Fig.1. Depth maps obtained by the Kinect contain holes where the Kinect was unable to obtain a depth reading, referred to as no-measured-depth pixels (NMD pixels). These NMD pixels are randomly present in homogeneous regions of an image, near object boundaries which correspond to depth discontinuities and on materials in the scene which may influence scatter of probe light.

2. Related work

2.1. RGB-D semantic segmentation

we base our network on FCN which is considered as a milestone since it solves the issue how CNN can be trained end-to-end to generate dense predictions for semantic segmentation. We make an experiment on NYU Depth v2 [13], an order-of-magnitude smaller than modern recognition dataset for RGB images. To reuse filter weights pretrained on the large-scale dataset, several methods [5, 8, 14] encoded the depth image with three channels at each pixel as if it was an RGB image. In [8], the authors proposed a new geocentric embedding HHA for depth images as follows: height above ground, angle with gravity for each pixel and the horizontal disparity. In [14], the authors proposed a computationally inexpensive encoding approach, which maps the distance to color values ranging from red over green to blue. All these methods need much preprocessing work. On the contrary, Fuse Net [15] simultaneously extract features from RGB and raw depth and constantly fuse the features as the network goes deeper.

2.2. Image Inpainting

Telea et al. [16] proposed an inpainting algorithm based on propagating an image smoothness estimator along with the image gradient. It treats the missing regions as level sets and uses the FMM proposed in [17] to propagate the image information. FFM narrows boundary around the inpainting area by processing points closest to the original boundary firstly. We perform this method on depth image with the help of information from RGB image when we need to calculate specific inpainting value.
3. Methods
This section consists of two parts. First, we introduce the mathematical model on which our inpainting bases in Subsection 3.1. Then we detail the architecture of our fusion network to in Section 3.2.

3.1 Depth image hole filling
Our core thought of inpainting is to assign those NMD pixels (no-measured-depth pixel) depth value using those points which have the similar texture information with its corresponding pixel in RGB image. As in Fig.2, we use FFM to narrow boundary, which is an algorithm that solves in (1):

\[ \nabla T = 1 \text{ on } \Omega, \quad T = 0 \text{ on } \partial \Omega \quad (1) \]

Fig.2. FFM method to narrow boundary. it continuously adds new point that has the smallest distance to origin boundary \( \partial \Omega \) and keep a point set \( BAND \). The algorithm stops narrowing boundary as there is no more point to be inpainted.

where \( T \) is a distance map of pixels in inpainting area \( \Omega \), \( \nabla T \) is normal to \( \partial \Omega \). Therefore, we use tags to flag pixels in a boundary and continuously add a new point into a tag set by judging \( T \) and \( \nabla T \) from a small neighborhood \( N_e(p) \) whose radius is \( e \) and center is \( p \). For any pixel \( q \) in the neighborhood \( N_e(p) \), we can calculate a depth value for point \( p \) according to correspondence of two points, which is shown by the following equation:

\[ \text{Depth}_p(q) = D(q) + \nabla D(q)(p - q) \quad (2) \]

where \( D(q) \) is the depth value \( q \), \( \nabla D(q) \) is the gradient value at \( q \).

Then we compute the prediction depth by summing calculations of all points in the divided neighborhood \( N_e(p) \):

\[ \text{Depth}_p(q) = \frac{\sum_{q \in N_e(p)} \omega(p,q)[D(q) + \nabla D(q)(p - q)]}{\sum_{q \in N_e(p)} \omega(p,q)} \quad (3) \]

where \( \omega(p,q) \) is weighting function to measure the correspondence of point \( p \) and \( q \).

The design of weighting function is crucial to propagate depth detail. Since information in the depth image is poor and simple, we consider looking for correspondence in the RGB image:

\[ \text{Corr}(p,q) = \frac{1}{d_e} \cdot \frac{1}{d_i} \quad (4) \]

\[ d_e = \sqrt{(l_p - l_q)^2 + (a_p - a_q)^2 + (b_p - b_q)^2} \quad (5) \]

\[ d_i = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2} \quad (6) \]
(5) gives representation of distance in the Lab color space and (6) gives representation of distance in Euclidean space. We design our weighting function (7).

$$o(p,q_i) = \begin{cases} 1, & \text{Corr}(p,q_i) \geq M(p) \\ 0, & \text{Corr}(p,q_i) < M(p) \end{cases} \quad (7)$$

where $M(p)$ is a threshold we manually chose between max and min value of Corr$(p,q_i)$.

3.2. Network Architecture

3.2.1. Depth feature analysis

Normally in CNN, shallow layers respond to corners/edges or colors while deeper layers show more complexity and are more class-specific. The conclusion, however, is more likely appliable to RGB images since the model is trained on RGB dataset.

In Fig.4, we selected feature maps among 512 channels in the fourth conv block, most RGB features are local-activated while depth features are either global-activated or non-activated. The global-activated maps don’t contain any advanced feature and the non-activated maps are useless to the feature extraction. In Fig.5, we chose some feature maps among 64 channels in the first conv block that extracts edges and color information. On the one hand, these features are obviously complementary to each other. On the other hand, the edges show high similarity in two modalities though it is more complicated in RGB images. In Fig.3, according to the analysis above, we chose to fuse low-level features in pool3 layer with a feature sifting architecture to allow the network automatically choose the most discriminative features and eliminate similar features.

3.2.2. Fusion method

we define the activation of the kth feature map for lth layer as $x^l_i$ with a bias parameter $b^l_i$ and denote the weight matrix as $w^l_i$. Each layer is computed as mapping of input $X = [x_1, x_2, x_3...x_n]$ with a function $f$ and the inference process is formulated as follows,

$$x^{l+1}_i = f(X^l; w^l_i) = \text{relu}(\langle w^l_i, X^l \rangle + b^l_i) \quad (8)$$

In Block3 of VGG16, there are 256 channels in feature map which is corresponded to the weight scalar $W = [w_1, w_2, w_3...w_{256}]$. For fusion, a simple way is to apply an element-wise summation on the output of corresponding layers in two streams. However, it increases activation and reduces feature diversity. To this end, we directly concatenate the feature maps from two branches and feed the output
Fig.4. feature maps among 512 channels in conv block 4, left is RGB feature maps, right is depth feature maps.

Fig.5. feature maps among 64 channels in conv block 1, left is RGB feature maps, right is depth feature maps.

to the following network where RGB and depth features will be automatically filtered and merged into a higher-level feature.

3.2.3. Feature sifting
Followed by fusion method in the last section, we stack low-level feature maps from two different modalities in a shallow layer. As described, an architecture of feature sifting with the aim of decreasing redundancy of feature maps from RGB and Depth image as well as making network generate more discriminative features is proposed. The architecture uses a group of learnable factors which can change the activation value of feature maps according to their contribution. More specifically, as shown in Fig.6, an average pooling layer in which the feature maps are down-sampled by a kernel is utilized. It is noted that the size of the kernel is the channel number of feature maps. Then, a fully-connection layer that is the same size as pooling layer is used.

Fig.6. the architecture of our network which contains two pre-processing streams in low-level and one fusion line as network goes deeper. The model is the same as FCN-8s based on VGG16.
4. Experiment

4.1. NYUD v2

We made an evaluation of our proposed algorithm on the NYU Depth v2 dataset which contains 1449 pixel-annotated images. We used three metrics to perform the quantitative evaluation. Let $TP$, $FP$, $FN$ denote pixel number of true positive, false positive, false negative, respectively, and let $N$ be the total number of pixels of class $K$.

- **Pixel accuracy**: $\frac{1}{N} \sum_{k \in \{1,2...K\}} TP_k$
- **Mean accuracy**: $\frac{1}{K} \sum_{k \in \{1,2...K\}} \frac{TP_k}{TP_k + FP_k}$
- **Intersection-over-union (IoU)**: $IoU = \frac{1}{K} \sum_{k \in \{1,2...K\}} \frac{TP_k}{TP_k + FP_k + FN_k}$

|                          | Pixel acc. | Mean acc. | IoU   |
|--------------------------|------------|-----------|-------|
| FCN-32s RGBD             | 61.5       | 32.4      | 30.5  |
| FCN-16s RGB-HHA          | 65.4       | 46.1      | 34.8  |
| Ours with raw depth      | 68.5       | 48.4      | 36.2  |
| Ours with inpainting     | **69.7**   | **49.8**  | **39.1** |

We trained our network on the standard split of 795 training images and 654 testing images. As shown in TABLE 1, our model produces better result compared with our baseline FCN. It is noted that our model is an end-to-end without any preprocessing work (e.g. HHA) to extract information. Our segmentation results are given in Fig.7.

![Fig.7. Our result on NYU Depth v2 test split. From left to right is RGB image, depth image, ground truth, our result.](image-url)
5. Conclusion
In this paper, we proposed a simple but efficient method for inpainting depth image with the help of the color information and a fusion architecture for RGB-D semantic segmentation based on FCN. Experimental result on NYU Depth v2 demonstrate that the proposed model can efficiently fuse the geometry depth information into texture information of RGB. As we use FCN as our baseline, it is an obstacle to keep local detailed information. In the future, we intend to cooperate with more advanced architecture to improve localization property in the deep network. We also plan to evaluate the proposed hybrid architecture with more datasets and introduce transfer learning to enable our model to learn better on small-sample data.

Acknowledgement
This work was supported by the National Key Research and Development Program under Grant 2017YFB0503701.

References
[1] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 39, no. 4, pp. 640-651, 2014.
[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Trans Pattern Anal Mach Intell, vol. 39, no. 6, pp. 1137-1149, 2017.
[3] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 234-241.
[4] S. Zheng et al., "Conditional Random Fields as Recurrent Neural Networks," pp. 1529-1537, 2015.
[5] A. Hermans, G. Floros, and B. Leibe, "Dense 3D semantic mapping of indoor scenes from RGB-D images," in IEEE International Conference on Robotics and Automation, 2014, pp. 2631-2638.
[6] X. Qi, R. Liao, J. Jia, S. Fidler, and R. Urtasun, "3D Graph Neural Networks for RGBD Semantic Segmentation," in IEEE International Conference on Computer Vision, 2017, pp. 5209-5218.
[7] D. Eigen and R. Fergus, "Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-scale Convolutional Architecture," pp. 2650-2658, 2014.
[8] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning Rich Features from RGB-D Images for Object Detection and Segmentation," vol. 8695, pp. 345-360, 2014.
[9] Z. Li, Y. Gan, X. Liang, Y. Yu, H. Cheng, and L. Lin, "LSTM-CF: Unifying Context Modeling and Fusion with LSTMs for RGB-D Scene Labeling," pp. 541-557, 2016.
[10] S. Gupta, J. Hoffman, and J. Malik, "Cross Modal Distillation for Supervision Transfer," pp. 2827-2836, 2015.
[11] X. Song, L. Herranz, and S. Jiang, "Depth CNNs for RGB-D scene recognition: learning from scratch better than transferring from RGB-CNNs," in AAAI Conference on Artificial Intelligence, 2017.
[12] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suite," in CVPR, 2015, vol. 5, p. 6.
[13] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in European Conference on Computer Vision, 2012, pp. 746-760: Springer.
[14] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal deep learning for robust RGB-D object recognition," pp. 681-687, 2015.
[15] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "FuseNet: Incorporating Depth into Semantic Segmentation via Fusion-Based CNN Architecture," pp. 213-228, 2016.
[16] A. Telea, "An Image Inpainting Technique Based on the Fast Marching Method," Journal of Graphics Tools, vol. 9, no. 1, pp. 23-34, 2004.
[17] J. A. Sethian, "A fast marching level set method for monotonically advancing fronts," Proceedings of the National Academy of Sciences of the United States of America, vol. 93, no. 4, p. 1591, 1996.