SCN: Switchable Context Network for Semantic Segmentation of RGB-D Images

Di Lin, *Member, IEEE*, Ruimao Zhang, *Member, IEEE*, Yuanfeng Ji, *Student Member, IEEE*, Ping Li, *Member, IEEE*, and Hui Huang, *Senior Member, IEEE*

***Abstract*—Context representations have been widely used to profit semantic image segmentation. The emergence of depth data provides additional information to construct more discriminating context representations. Depth data preserves the geometric relationship of objects in a scene, which is generally hard to be inferred from RGB images. While deep convolutional neural networks (CNNs) have been successful in solving semantic segmentation, we encounter the problem of optimizing CNN training for the informative context using depth data to enhance the segmentation accuracy. In this paper, we present a novel switchable context network (SCN) to facilitate semantic segmentation of RGB-D images. Depth data is used to identify objects existing in multiple image regions. The network analyzes the information in the image regions to identify different characteristics, which are then used selectively through switching network branches. With the content extracted from the inherent image structure, we are able to generate effective context representations that are aware of both image structures and object relationships, leading to a more coherent learning of semantic segmentation network. We demonstrate that our SCN outperforms state-of-the-art methods on two public datasets.**

***Index Terms*—Context representation, convolutional neural network (CNN), RGB-D images, semantic segmentation.**

I. Introduction

SEMANTIC segmentation has been extensively studied in computer vision and graphics. Semantic image segmentation is challenging, as it requires per-pixel categorizations of objects [1]–[4] in images. Correctly understanding the complex relationship of objects in an image is critical. Using the powerful capabilities of deep convolutional neural networks (CNNs) [5]–[19] and large-scale image datasets [20], [21] available for model pretraining, we are able to create useful image representations that greatly improve semantic segmentation.

Recently, the employment of depth data has been found to benefit the analysis of image representations. With the availability of low-priced, high-performance sensors, depth data is easily obtained. Depth provides necessary geometric information, that is, the spatial layout of objects in 3-D scene, which is not held by RGB images and thus enriches the image representations. Most of the prior work [22]–[26] provides the depth as an additional channel, alongside color, of input to CNNs. In principle, depth separates objects with respect to their spatial layout. But the existing works miss the opportunity to well utilize the depth data to model the information propagation between network layers [27]–[29], which is vital to produce useful image representations. Additionally, compared to color data, depth provides much less of the crucial semantic information needed for segmentation. Given this, using depth as an input channel similar to color would likely create inconsistencies between network inputs that mislead the network training.

In this paper, we use a different but consistent combination of color and depth to improve the learning of the segmentation network. We make joint use of the color image and depth data as cues to update the network, which produces more accurate context for image segmentation. Specifically here we refer to context as the co-existence of objects[1] in multiple image regions. In semantic segmentation, aggregating the convolutional features of image regions [27], [28], [30]–[35] has become a *de facto* core process for generating context representation. Our idea is to better utilize the image structure and depth data to guide the aggregation of features by considering object co-existence, which is introduced below.

A. Our Findings

We can observe, e.g., from Fig. 1, the correlation between depth and object co-existence. In far regions that have high depth, the objects in general densely co-exist. Such complexity yields rich yet distracting information. It is thus critical to reduce the clutter from diversities when constructing stably useful context. The near regions that have low depth usually

Manuscript received January 17, 2018; revised April 26, 2018, July 29, 2018, and October 6, 2018; accepted December 1, 2018. Date of publication December 20, 2018; date of current version January 21, 2020. This work was supported in part by NSFC under Grant 61702338, Grant 61522213, Grant 61761146002, and Grant 61861130365, in part by the National 973 Program under Grant 2015CB352501, in part by the Guangdong Science and Technology Program under Grant 2015A030312015, in part by the Shenzhen Innovation Program under Grant KQJSCX20170727101233642, and in part by the Macau Science and Technology Development Fund under Grant 0027/2018/A1. This paper was recommended by Associate Editor Q. M. J. Wu. *(Corresponding author: Hui Huang.)*

D. Lin, Y. Ji, and H. Huang are with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: ande.lin1988@gmail.com; jyuanfeng8@gmail.com; hhzhiyan@gmail.com).

R. Zhang is with the Department of Electronic Engineering, Chinese University of Hong Kong, Hong Kong (e-mail: ruimao.zhang@ieee.org).

P. Li is with the Faculty of Information Technology, Macau University of Science and Technology, Macau 999078, China (e-mail: pli@must.edu.mo).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2018.2885062

[1] The object co-existence is generally represented by the object categories that are coherently present.

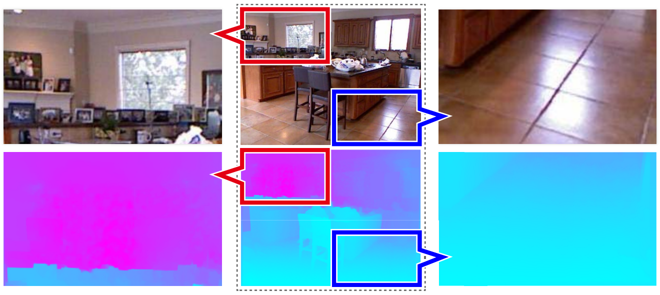


Fig. 1. Correlation observed between the depth and the object co-existence: in common the near regions, e.g., highlighted in blue rectangles, have relatively simple object co-existences, while very likely different objects co-exist densely in the far regions, such as highlighted in red rectangles.

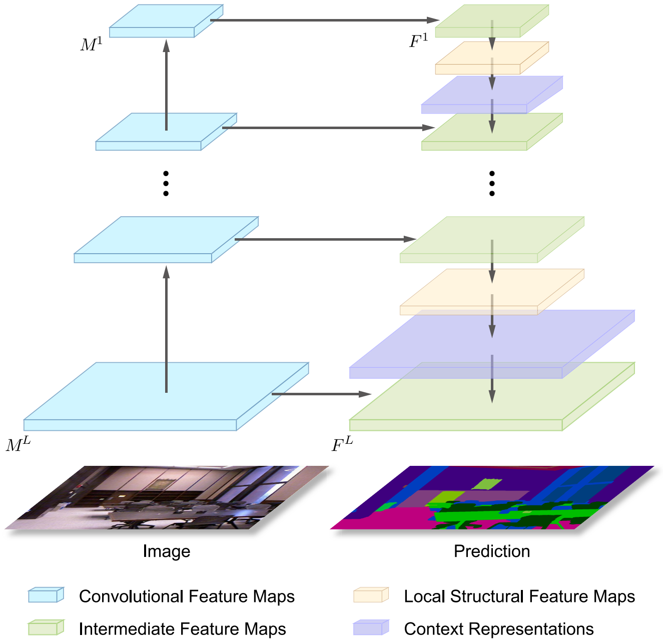

M^1
F^1
M^L
F^L
Image
Prediction
Convolutional Feature Maps
Local Structural Feature Maps
Intermediate Feature Maps
Context Representations

Fig. 2. Overview of our SCN. Given an RGB image, we produce the convolutional feature maps layer-by-layer in a resolution-descending order. Our SCN first produces the local structural feature maps, which are used to compute the context representations in top-down switchable information propagation. The context representations are combined with the convolutional features to form the intermediate feature maps, which are used for the final semantic segmentation.

contain fewer objects and variants. A wider range of context shall hence be included to alleviate the problem caused by the lack of object diversities. It requires an adaptive way to construct and switch between different contexts, which are dependent upon different levels of object co-existences closely associated with the depth.

B. Our Contributions

Based on our findings, two contributions are made here. First, we newly model the information propagation between network layers. Unlike the structure-insensitive information propagation proposed in [27]–[29], our local structural information propagation utilizes the super-pixels, which are defined by the latent image structure, to capture more relevant content of the image regions. Second, by sensing different levels of object co-existences, the features can be flexibly switched through network branches to capture useful contextual information on demand. After combining the structure of image regions and the object co-existences together consistently to serve as the guidance, our SCN excels in learning with effective context representations. Therefore, as shown in Fig. 2, we introduce SCN with the following features.

1) A new model of contextual information propagation, which includes: a) *local structural* information propagation and b) *top-down switchable* information propagation. All propagations are guided by super-pixels [36], [37] that are defined by the underlying image structure.
2) Two switchable branches enabling the network to process different image regions accordingly: a) one branch with a *compression* architecture to reject over-cluttered information of the image regions and b) the other branch with an *expansion* architecture to broaden the receptive fields of the less informative image regions presented in convolutional feature maps.

We evaluate our method on two public datasets. The results indicate that our SCN achieves the mean intersection-over-union (IoU) of 49.6 on the NYUDv2 dataset [38] and 50.7 on the SUN-RGBD dataset [21], both of which outperform state-of-the-art methods.

II. Related Work

A. Resolution Recovery of Convolutional Features

The fully convolutional network (FCN) [23], [32], [39]–[41] is the prevalent architecture for semantic segmentation. Given input images, FCN has stacked down-sample operations to compute the features having high-level semantics. However, such down-sampling progressively reduces the resolution of features, leading to the loss of object detail. To address this issue, Chen *et al.* [39] and Yu and Koltun [42] applied atrous convolution to preserve the feature resolutions, at the significant cost of memory space. Badrinarayanan *et al.* [43], Noh *et al.* [44], and Ghiasi and Fowlkes [45] exploited deconvolution and unpooling to increase the resolution of the last convolutional feature map. These methods, however, still fail to reuse the high-resolution feature maps computed by the preceding convolutional layers, thus ignoring the richer information [32], [46] they may provide.

Other recent works [27]–[29], [46] propose to use a top-down cascaded network to preserve semantic meanings and object details simultaneously. With the lateral connection of top-down cascaded architectures, the semantic content of low-resolution features can be propagated as contextual information to strengthen the high-resolution features. The top-down cascaded architecture eventually yields high-resolution features that result in compelling performance on semantic segmentation. The common method for propagating the contextual information uses interpolation and deconvolution operations to match the resolutions of different feature maps. However, these operations propagate the information regardless of the image structure, indiscriminately propagating the information from low-resolution features to the regions regularly predefined in high-resolution features. In the scenario of semantic segmentation of RGB-D images, such a common top-down network inevitably limits the usable information for better learning segmentation features. In our top-down cascaded network, we employ super-pixels to guide the information propagation, which makes the inherent image structure available to all context representations.

B. Contextual Information for Semantic Segmentation

Contextual information has been intensively used to benefit semantic segmentation. Previous works [32], [39]–[41], [47] use traditional graphical models [48] to capture the interaction between adjacent image regions as part of the contextual information. Similarly, aggregation of the convolutional features of image regions is a popular and effective way to construct contextual information, as evidenced in [27], [28], [31], [34], [42], and [46]. Yu and Koltun [42] applied multiscale atrous convolutions to compute segmentation scores. Chen *et al.* [39], [46] used different atrous convolutions to extract contextual information from convolutional feature maps. Lin *et al.* [28] and Zhao *et al.* [34] used different combinations of convolutional and pooling kernels to enrich the contextual information. Peng *et al.* [27] and Mostajabi *et al.* [31] grouped a wide range of convolutional features to embed global context in segmentation features. All of these methods ignore image structures.

Recent works [24], [30], [31] use super-pixels as cues to aggregate the convolutional features and form context representations. Lin *et al.* [35] considered the relationship of super-pixels when constructing context representations. As super-pixels represent image structures in some sense, context representations generated using these methods are associated closely with the image structure.

Nonetheless, to construct the context representation for an image region, existing works aforementioned aggregate the convolutional features computed by canonical upstream network. We suggest a different approach, using network branching architectures to construct context representations adaptively and selectively, allowing better handling of object co-existences that look different in different image regions.

C. Semantic Segmentation of RGB-D Image

The use of depth data to assist semantic image segmentation has been studied in many prior works [22], [24], [25], [38], [49]. Depth data provides a better understanding of the geometric relationships between objects and benefits image segmentation. For instance, Silberman *et al.* [38] used color along with depth to compute the support relations of objects, while Gupta *et al.* [49] constructed geometric contours using depth data as a cue.

Because deep neural networks [5]–[7] perform well at image recognition, many researchers apply CNNs to help the segmentation of RGB-D images. Couprie *et al.* [50] used color and depth from images as a unified input to train CNNs. However, the output from the CNNs loses much of the useful information contained in the depth data. To better exploit the depth content, Gupta *et al.* [22] and He *et al.* [24] encoded depth images as horizontal disparity, height above ground, angle of pixel's local surface normal (HHA) images [49], storing the horizontal disparity, the height above ground, and the angle of the local surface normal for each pixel. Long *et al.* [23] and Wang *et al.* [25] used color and HHA images as inputs to train CNNs to compute more effective segmentation features. Though the enriched features do improve the segmentation accuracy, there is a lack of the semantic relationship between color and HHA images. Directly combining them [22]–[25] inevitably introduces inconsistencies into the network learning.

TABLE I
LIST OF SYMBOLS WITH DESCRIPTION

symbol	description
r_i	the i^{th} region in the feature maps
y	ground-truth labels
S_n	the n^{th} super-pixel
F^l	the l^{th} intermediate feature maps
M^l	the l^{th} convolutional feature maps
D^l	the l^{th} local structural feature maps
D^l_c	the l^{th} compression feature maps
D^l_e	the l^{th} expansion feature maps
$D^{l\rightarrow(l+1)}$	context representation produced by top-down switchable information propagation
$\mathcal{H}(:)$	local structural mapping
$\mathcal{C}(:)$	compression network architecture
$\mathcal{E}(:)$	expansion network architecture
$L(:)$	softmax loss function
$J(:)$	training objective function of SCN

In this paper, depth data plays a more important role in guiding the construction of context representations. We use the depth to identify the object co-existences within image regions. A switchable feature aggregation method is then proposed to produce appropriate context representations adaptively for image regions that contain different levels of object co-existences. The framework thus facilitates a more coherent learning of segmentation features from the color and depth data, yielding better segmentation results.

III. SWITCHABLE CONTEXT NETWORK

We present here a switchable context network (SCN), where the propagation of contextual information is adaptively guided by the image structure and object co-existence. In addition to generic convolutional features produced by an FCN from the same resolution, our SCN propagates contextual information from low-resolution feature maps to high-resolution ones in a top-down manner with a switchable feature aggregation scheme. Instead of regular propagation strategies [27]–[29], [46], we employ super-pixels that are defined according to the inherent image structure. We list the critical symbols and their descriptions in Table I for reference.

The architecture of our SCN is shown in Fig. 2. Given an input image I, the SCN produces L feature maps $\{F^l\}$ that

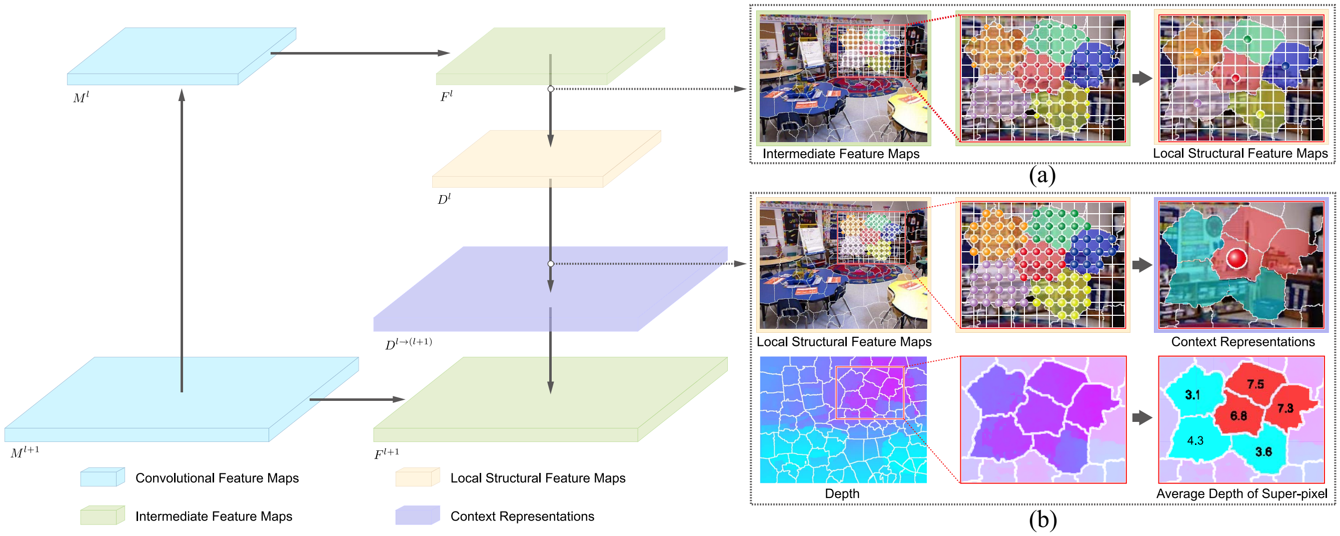

M^l
F^l
D^l
$D^{l\to(l+1)}$
M^{l+1}
F^{l+1}
Convolutional Feature Maps
Local Structural Feature Maps
Intermediate Feature Maps
Context Representations
Intermediate Feature Maps
Local Structural Feature Maps
(a)
Local Structural Feature Maps
Context Representations
Depth
Average Depth of Super-pixel
3.1
7.5
6.8
7.3
4.3
3.6
(b)

Fig. 3. Construction of our contextual representation undergoes two information propagations. (a) Local structural information propagation. In this stage, each region (a color node of the regular grid in the intermediate feature maps) receives the information from the regions located in the same super-pixel. The regions (the enlarged node of the regular grid) having richer information constitute the local structural feature maps. (b) Top-down switchable information propagations. We compute the average depth value for each super-pixel. In the last column, the super-pixels highlighted in red and blue contain the regions that provide the information output by compression and expansion architectures, respectively. Each region (a color node of the regular grid in the local structural feature maps) receives the information from the regions located in the adjacent super-pixels, and form the region (the highlighted red node in the context representations) having accurate contextual information. For illustration, the context representations are shown in the same size with the local structural feature maps. Actually, the context representations have larger resolution than the local structural feature maps do.

have different resolutions. The feature maps are depicted in a top-down shape, where F^1 has the lowest resolution and F^L (given to the classifier for pixel-wise categorization) has the highest resolution. We formulate the feature map F^{l+1} as

$$F^{l+1} = M^{l+1} + D^{l\to(l+1)}, \quad l = 0, \ldots, L-1 \tag{1}$$

where $D^{l\to(l+1)}$ denotes the context representation that is used to enhance the generic convolutional feature M^{l+1} produced by the FCN; $D^{0\to 1} = \mathbf{0}$.

As illustrated in Fig. 3, the construction of our contextual representation $D^{l\to(l+1)}$ has two stages: local structural and top-down switchable information propagations. The first stage processes a region using the contextual information from the regions belonging to the same super-pixel. Contextual information is propagated between the regions in the same convolutional feature map [Fig. 3(a)]. The stage computes the new feature D^l where each region respects the local property of a super-pixel.

Next, the top-down switchable stage [Fig. 3(b)] generates $D^{l\to(l+1)}$. Each region in F^{l+1} receives the information propagated from F^l that has a lower resolution. In handling each region, the process uses information from super-pixels in adjacent regions to capture richer object relationships. For a given receiver region, our network selects a proper branch to process the contextual information propagated from different regions. For the regions having complex object co-existences, the branch equipped with the compression architecture is used to reduce excessive diverse information (clutter) while retaining the critical context information. For the regions lacking discriminative information, the expansion architecture includes more global object content to enrich contextual information.

IV. Contextual Information Propagation

We detail in this section how we propagate information from the two stages.

A. Local Structural Information Propagation

Given an input image I, we generate a set of nonoverlapping super-pixels $\{S_n\}$. In the feature maps, the region[2] r_i uniquely defines a receptive field in image I. $\Phi(S_n)$ defines a set of centers of regular receptive fields that are located within the super-pixel S_n. For r_i, the local structural information propagation step produces the feature as

$$D^l(r_i) = F^l(r_i) + \sum_{r_i \neq r_j} \mathcal{H}\Big(F^l(r_j)\Big)$$
$$\text{s.t.} \quad r_i, r_j \in \Phi(S_n) \tag{2}$$

where $D^l \in \mathbb{R}^{C\times H\times W}$ is the local structural feature maps. $F^l(r_i) \in \mathbb{R}^C$ is the lth intermediate feature maps computed by the SCN [see also (1)]. For the region r_i, the feature $F^l(r_i)$ directly contributes to the new feature $D^l(r_i) \in \mathbb{R}^C$. Thus, the new feature $D^l(r_i)$ retains the local content of the region r_i. In addition, the other regions located in the same super-pixel S_n are processed by a local structural mapping $\mathcal{H}(:)$. In our implementation, we model the mapping $\mathcal{H}(:)$ as two sequential convolutions as shown in Fig. 4(a). Each convolution has 1×1 local kernels without changing the receptive fields of the corresponding regions. In this way, the local information propagation allows the local pattern of the super-pixel S_n to be adaptively encoded in the feature $D^l(r_i)$. In addition, the 1×1 convolutional kernels are manipulated on each local region, leading to a fast computation of the local structural feature

[2] Here, we refer the region as a node in the feature maps.

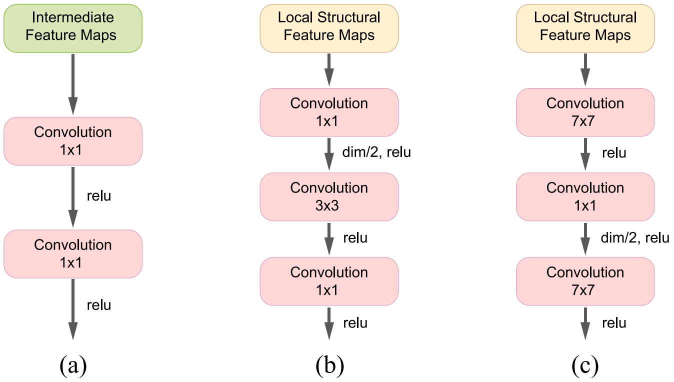


Fig. 4. Illustrations for (a) local structural mapping, (b) compression architecture, and (c) expansion architecture, respectively.

maps D^l. We empirically find that larger kernels (e.g., 3×3 and 5×5 kernels) lead to a drop of the segmentation accuracy. Also, the larger kernels and more convolutional layers (e.g., 3–5 layers) inevitably increase the computational complexity of the network.

We emphasize that the local structural information propagation plays an important role. As formulated in (2), this propagation uses all of the regions resident in the same super-pixel to produce new features in which the structural information of the whole super-pixel is embedded. The structural information strengthens the relationship between the regions in the same super-pixel, providing better features that are used in subsequent steps of the top-down propagation.

B. Top-Down Switchable Information Propagation

Compared to the local information propagation that processes each local super-pixel, the top-down information propagation accounts for the interaction between distant regions in adjacent super-pixels, which may have various object co-existences. As we have observed, there is the correlation between depth and object co-existences. In the top-down information propagation, we propose a switchable feature aggregation scheme guided by depth data. This feature produces context representations that reflect the different object co-existences of the regions in multiple super-pixels.

In (1), we have denoted $D^{l \to (l+1)}$ as the context representation that is used to enhance the feature F^l. Given $r_i \in S_n$, the context representation $D^{l \to (l+1)}(r_i)$ is formulated as

$$\begin{aligned} &D^{l \to (l+1)}(r_i) \\ &\quad = \sum_{S_m \in \mathcal{N}(S_n)} \sum_{r_j \in \Phi(S_m)} \left(\lambda_c D_c^l(r_j) + \lambda_e D_e^l(r_j) \right) \\ &\text{s.t. } \ \lambda_c = \mathbb{1}(d(S_n) < d(S_m)), \ \lambda_e = \mathbb{1}(d(S_n) \geq d(S_m)). \quad (3) \end{aligned}$$

We define $S_m \in \mathcal{N}(S_n)$ if the super-pixels S_n and S_m are adjacent. Equation (3) models the top-down information propagation from the region r_j, which resides in the super-pixel S_m, to the given region r_i. $D_c^l \in \mathbb{R}^{C \times H \times W}$ represents the compression feature maps while $D_e^l \in \mathbb{R}^{C \times H \times W}$ represents the expansion feature maps, and $d(S_n)$ denotes the average depth of the super-pixel S_n. We use the indicator function $\mathbb{1}(:)$ to switch between the compression and expansion features. Given the condition $d(S_n) < d(S_m)$ that means the super-pixel S_n is in a nearer field comparing to S_m, we activate the compression feature to refine the information of the region r_j. If the super-pixel S_n is in a more distant field than S_m is, we use the expansion feature to enrich the information of r_j.

The compression feature maps D_c^l is computed by the compression architecture $\mathcal{C}$. To reduce the excessively diverse information within source regions, the compression architecture learns to reweight the corresponding regional features. Given a source region r_j in (3), the compression architecture $\mathcal{C}$ outputs the refined feature maps $D_c^l \in \mathbb{R}^{C \times H \times W}$, that is,

$$D_c^l(r_j) = D^l(r_j) \odot \mathcal{C}\left(D^l(r_j)\right). \quad (4)$$

The compression architecture takes the local structural feature maps D^l in (2) as input. The details of C is shown in Fig. 4(b). The compression architecture consists of 1×1, 3×3, and 1×1 convolutional layers. The first 1×1 convolution is used to reduce the dimension of the feature $D^l(r_i)$ by half. The dimension reduction procedure can filter over-diverse information out while preserving useful information well. After the dimension reduction, we reconstruct the information of in the features by using 3×3 convolution. The last 1×1 convolution produces a reweighting vector $\mathcal{C}(D^l(r_j)) \in \mathbb{R}^C$ that selects useful information from the feature $D^l(r_j)$ to produce $D_c^l(r_j)$, similar to the attention model in [51].

We meanwhile use an expansion architecture $\mathcal{E}(:)$ to enrich the feature of the source region r_j. The expansion architecture is shown in Fig. 4(c). We use the term of "expansion" because the architecture contains relatively large convolutional kernels that enlarge the receptive field of the region r_j. As already investigated in [27] and [34], larger receptive fields help to produce richer context. The expansion architecture $\mathcal{E}(:)$ produces the feature map $D_e^l \in \mathbb{R}^{C \times H \times W}$ where

$$D_e^l(r_j) = D^l(r_j) + \mathcal{E}\left(D^l(r_j)\right). \quad (5)$$

Again, D^l is considered as the input for the expansion architecture. As shown in Fig. 4(c), the expansion architecture consists of 7×7, 1×1, and 7×7 convolutional layers. The first 7×7 convolutional layer employs relatively large kernels to broaden receptive field and learn relevant context. The following 1×1 is used for dimensional reduction. In our implementation, the 1×1 convolution reduces the feature dimension by half. The reduction operation removes the redundant information that may be included by the upstream large kernels, which produces compact feature for the consequent process. Then we have another 7×7 convolution to recover the feature dimension. This last 7×7 convolution adapts the outcome feature $\mathcal{E}(D^l(r_j)) \in \mathbb{R}^C$ to match with the dimension of $D^l(r_j)$. With the expansion architecture, the produced feature $D_e^l(r_j)$ includes more context with less redundant information, as it is based on the relatively clean feature output by the reduction operation.

V. Network Training

The high-resolution feature map F^L produced by the SCN (1) is fed to the pixel-wise classifier for semantic segmentation. The pixel-wise classifier outputs a set of class labels y

for the pixels of the input I. The label set y is defined as

$$y = f\big(F^L\big) \tag{6}$$

where the function $f(:)$ is a softmax regressor that gives pixel-wise categorization. We denote y^* as the ground-truth annotation of the image I. We predict the pixel-wise class labels of I using (6).

We train the SCN with the following objective function:

$$J\big(F^L\big) = \sum_{r_i \in I} L\big(y^*(r_i), y(r_i)\big) \tag{7}$$

where the function $L(:)$ is softmax loss that widely used for penalizing pixel-wise classification error. For the region r_i, we denote $y(r_i)$ as its predicted class label. We minimize the training objective of (7) to optimize the parameter set of the SCN. The standard stochastic gradient descend (SGD) method [5] is applied as the solver.

Using (1)–(3), we can update all features of the regions in the feature maps produced by the SCN. For the feature $F^l(r_i)$ of the region $r_i \in \Phi(S_n)$ in the feature map F^l, we compute the gradient as

$$\frac{\partial J}{\partial F^l(r_i)} = \sum_{S_m \in \mathcal{N}(S_n)} \sum_{r_j \in \Phi(S_m)} \frac{\partial J}{\partial F^{l+1}\big(r_j\big)} \sum_{r_k \in \Phi(S_n)} \frac{\partial F^{l+1}\big(r_j\big)}{\partial D^l(r_k)} \times \frac{\partial D^l(r_k)}{\partial F^l(r_i)} \tag{8}$$

where

$$\frac{\partial F^{l+1}\big(r_j\big)}{\partial D^l(r_k)} = \lambda_c \frac{\partial D^l_c(r_k)}{\partial D^l(r_k)} + \lambda_e \frac{\partial D^l_e(r_k)}{\partial D^l(r_k)}$$
$$\lambda_c = \mathbb{1}(d(S_n) < d(S_m)),\ \ \lambda_e = \mathbb{1}(d(S_n) \geq d(S_m)). \tag{9}$$

By using (8) to optimize the feature $F^l(r_i)$, we find that the region $r_i \in \Phi(S_n)$ receives the update signal $[(\partial F^{l+1}(r_j))/(\partial D^l(r_k))]$ from the region r_k that also resides in the same super-pixel S_n. This update signal adaptively adjusts the features of the regions located in the same super-pixel, in order to make those regions consistently represent the overall property of a super-pixel. As shown in (8), the region r_i also receives the update signal from the region r_j in F^{l+1} that resides in the adjacent super-pixel S_m. As the relationship of r_i, r_j and r_k are defined by super-pixels, one can see that the feature update here well preserves the image structure.

Besides image structure, the depth simultaneously guides the update of the features. In (8), the update signal from the region r_j is denoted as $[\partial J/(\partial F^{l+1}(r_j))]$. When propagated to the region r_i, the update signal $[\partial J/(\partial F^{l+1}(r_j))]$ is influenced by the signal $[(\partial F^{l+1}(r_j))/(\partial D^l(r_k))]$, which can be expanded as in (9). The parameters λ_c and λ_e are then used as switches, determined by the average depths of S_n and S_m. With (4) and (5), the signals $[(\partial D^l_c(r_k))/(\partial D^l(r_k))]$ and $[(\partial D^l_e(r_k))/(\partial D^l(r_k))]$ are expanded as

$$\frac{\partial D^l_c(r_k)}{\partial D^l(r_k)} = \mathcal{C}\Big(D^l(r_k)\Big) + D^l(r_k) \odot \frac{\partial \mathcal{C}\big(D^l(r_k)\big)}{\partial D^l(r_k)} \tag{10}$$

and

$$\frac{\partial D^l_e(r_k)}{\partial D^l(r_k)} = 1 + \frac{\partial \mathcal{E}\big(D^l(r_k)\big)}{\partial D^l(r_k)}. \tag{11}$$

If $d(S_n) < d(S_m)$, the signal $[(\partial D^l_c(r_k))/(\partial D^l(r_k))]$ (10) makes an impact on the signal $[\partial J/(\partial F^{l+1}(r_j))]$ that is propagated from the region r_j. In (10), the compression architecture $\mathcal{C}(:)$ can be optimized by back propagation. More importantly, the reweighting vector $\mathcal{C}(D^l(r_k))$ also takes part in the update process. As modeled in (4), the vector $\mathcal{C}(D^l(r_k))$ is used to select the important information of the feature $D^l(r_k)$ to construct the feature $F^{l+1}(r_j)$. Reversely, the reweighting vector $\mathcal{C}(D^l(r_k))$ adjusts the back-propagated signal $[\partial J/(\partial F^{l+1}(r_j))]$ in the training stage. With the reweighting vector $\mathcal{C}(D^l(r_k))$, useful signal from the region r_j can be passed to better update the feature of the region r_i.

If $d(S_n) \geq d(S_m)$, the signal $[(\partial D^l_e(r_k))/(\partial D^l(r_k))]$ (11) affects the signal $[\partial J/(\partial F^{l+1}(r_j))]$. In (11), we find a factor of 1 that forms a skip connection between the regions r_i and r_j. That means the back-propagated signal from r_j to r_i does not need to be processed by the expansion architecture. We emphasize the importance of this fact. Though the expansion architecture employs more context of the feature by broadening the receptive field, the large convolutional kernels of the expansion architecture may distract the back-propagated signal from r_j to r_i during the training. Using the skip connection between the regions r_i and r_j, we allow the back-propagated signal to be propagated directly from r_j to r_i.

In summary, our SCN wisely takes image structure and depth data as cues to guide the training procedure, which yields optimized features for a better segmentation.

VI. Implementation Details

A. Data Preparation

Given an RGB image, we apply the toolkit [37] to compute the super-pixels in it. The size of the super-pixels is tunable by setting a scale parameter. In our SCN, the super-pixels are used to guide the construction of context representations. Each RGB image is given along with a depth image. The single-channel depth images are used in the top-down switchable information propagation, as described in Section IV.

In our experiments, the original RGB images were used to train the segmentation network. Afterward, we compute a three-channel HHA image [22], [49] based on the corresponding depth image. The HHA images contain rich geometric information of pixels used to train another segmentation network to enhance segmentation accuracy.

For an RGB image, the generation of super-pixels and an HHA image takes about 0.5 s, which is completed before the network training. We can simply use GPUs to accelerate this generation process for handling more data.

B. Network Construction

We use the Caffe platform [52] for network construction. Our SCN is applicable to different FCN variants [23], [27], [28], [34], [39], [45]. In our implementation, we choose ResNet-101 [7] pretrained on ImageNet [20] to serve as

TABLE II
SENSITIVITIES TO THE SIZE OF SUPER-PIXELS. THE PERFORMANCE IS EVALUATED ON THE NYUDv2 VALIDATION SET. EACH SEGMENTATION ACCURACY IS REPORTED IN TERMS OF MEAN IoU (%)

scale	500	1000	2000	4000	8000	12000
mean IoU	42.7	43.5	**45.6**	43.6	44.2	42.9

TABLE III
STRATEGIES OF PROPAGATING LOCAL INFORMATION PROPAGATION, EVALUATED ON THE NYUDv2 VALIDATION SET. EACH SEGMENTATION ACCURACY IS REPORTED IN TERMS OF MEAN IoU (%)

strategy	**local information propagation**	**mIoU**
w/o super-pixel	global identity mappings	40.3
	Noh et al. [44]	41.8
	Chen et al. [39]	43.0
w/ super-pixel	Lin et al. [35]	43.8
	3 × 3 kernels	44.5
	5 × 5 kernels	42.1
	SCN	**45.6**

the network architecture. We apply atrous convolution [39] to augment the ResNet-101 architecture as an eight-stride network, which produces high-quality segmentation results. The ResNet-101 network is mainly used for internal study of our SCN. When we compare our SCN to state-of-the-art methods, we utilize deeper ResNet-152 architectures [7] to further improve segmentation results.

In (1), we use the generic convolutional feature maps computed by FCN to construct context representations. We note that the ResNet-101 and ResNet-152 architectures [7] have similar five-stage convolutions, where each stage has several intermediate convolutional feature maps. For a stage of convolutions, we select the last convolutional feature map that is relatively stronger than other feature maps computed from the same stage, as following [29].

Here, we apply the standard SGD solver [5] to optimize the network parameters. With a multiple-GPU technique, we set the size of each mini-batch as 16. We enable the update of batch normalization to achieve better optimization. The network is fine-tuned with a learning rate of 1e-10 for 60K mini-batches. After that, we decay the learning rate to 1e-11 for the next 40K mini-batches. In our experiments, the network is trained for recognizing about 40 categories. With the transferring capacity of deep network, the trained network can be fine-tuned on more complex datasets. It saves the training time and increases the segmentation accuracy.

VII. EXPERIMENTAL RESULTS

We test our SCN on two public benchmarks for semantic segmentation of RGB-D images, which are NYUDv2 [38] and SUN-RGBD [21]. The NYUDv2 [38] dataset has been widely used for evaluating segmentation performance. It has 1,449 RGB-D images. In this dataset, 795 images are split for training and 654 images are for testing. As suggested in [22], we select a validation set that comprises of 414 images from the original training set. We use the pixel-wise annotations provided in [49], where all pixels are labeled by 40 categories. We use the NYUDv2 [38] dataset for the main evaluation of our method. We further use the SUN-RGBD [21] dataset for extensive comparison with state-of-the-art methods.

We follow the widely used multiscale testing [28], [34] to compute the segmentation results. That is, we use four scales (i.e., 0.6, 0.8, 1, and 1.1) to resize the testing image before providing it to the network. The output segmentation scores of the rescaled images are then averaged for the post-processing of dense CRF [48]. All segmentation performances in this paper are reported in terms of mean IoU [22], [23], [28].

A. Experiments on NYUDv2 Dataset

1) Sensitivity to the Number of Super-Pixels: In our SCN, the control of contextual information is in part subject to the size of super-pixels. Here, we investigate how sensitive our SCN is regarding to different super-pixel sizes. Using the toolkit [37], we adjust the size of super-pixels by a scale parameter. We empirically select different scales, which are 500, 1000, 2000, 4000, 8000, and 12 000. For each scale, we train our SCN based on ResNet-101 model [23]. The inputs to SCN are RGB image for segmentation and depth image for switching the features. The segmentation accuracies on the validation set of the NYUDv2 [38] are listed in Table II.

Among all of the cases in Table II, we find that the scale of 500 leads to the lowest segmentation accuracy. This occurs because the super-pixels are too small and contain too little contextual information. As the scale increases, we observe that the segmentation performance improves. Empirically, we observe that our SCN performs the best when the scale is set to 2000. We find super-pixels that are too large reduce performance. This is because too large super-pixels may include excess objects which limit the stages preservation of local properties of the super-pixels. In subsequent experiments, we continued using the scale of 2000 to construct our network.

2) Strategies of Propagating Local Structural Information: The local structural information propagation produces features that have stronger relationship with the regions. Here, we conduct an ablation analysis in Table III, where the local structural information propagation is replaced with other strategies using structural information.

The first experiment measures the performance of other approaches that do not use local structural information. We apply the full version of an SCN that achieves the segmentation score of 45.6 on the NYUDv2 validation set. Then we retrain our SCN without propagating the local structural information of super-pixels. Equivalently, all of the intermediate features are processed by global identity mappings. In this way, we achieve the accuracy of 40.3. We also apply interpolation [39] and deconvolution [44] to produce new features, where each region contains information of wider but regular receptive field. These methods produce structure-insensitive features that achieve lower scores than our SCN does.

We note that there are several alternatives to propagate the local structural information of super-pixels. One simple way is proposed by Lin *et al.* [35], where the information is

TABLE IV
ABLATION EXPERIMENT ON TOP-DOWN SWITCHABLE INFORMATION PROPAGATION. THE RESULTS ARE EVALUATED ON THE NYUDV2 VALIDATION SET

super-pixel	depth	method	mean IoU
no	no	Noh et al. [44]	42.4
		Chen et al. [39]	42.7
no	yes	Lin et al. [29]	43.4
yes	no	expansion	41.6
		compression	43.0
		Lin et al. [35]	44.1
yes	yes	SCN	**45.6**

TABLE V
STRATEGIES OF INFORMATION COMPRESSION. THE RESULTS ARE EVALUATED ON THE NYUDV2 VALIDATION SET

compression architecture	$[1 \times 1\ conv, relu]$ $[1 \times 1\ conv, relu]$	$[1 \times 1\ conv, relu]$ $[3 \times 3\ conv, relu]$	$[1 \times 1\ conv, relu]$ $[3 \times 3\ conv, relu]$ $[1 \times 1\ conv, relu]$
mean IoU	42.3	41.7	**45.6**

TABLE VI
STRATEGIES OF INFORMATION EXPANSION. THE RESULTS ARE EVALUATED ON THE NYUDV2 VALIDATION SET

expansion architecture	$[7 \times 7\ conv, relu]$	$[7 \times 7\ conv, relu]$ $[7 \times 7\ conv, relu]$	$[7 \times 7\ conv, relu]$ $[1 \times 1\ conv, relu]$ $[7 \times 7\ conv, relu]$
mean IoU	43.8	44.2	**45.6**

computed by averaging the features of the regions in the same super-pixel. It means that the local structural mappings are implemented with identity kernels. With this, we achieve the segmentation score of 43.8. As identity kernels do not contain learnable parameters, they miss the flexibility to select useful information. We also investigate different convolutional kernels, which have the sizes of 3×3 and 5×5. Compared with 1×1 kernels that capture the finer structure of super-pixels, the larger kernels yield inferior results.

3) Evaluation on Top-down Switchable Propagation: Given local structural features, we apply the top-down switchable information propagation to yield context representations. This scheme is guided by super-pixels and depth, which are explored for their efficacy in this experiment.

In Table IV, we measure our top-down propagation without using both super-pixels and depth. Instead, we just apply deconvolution [44] and interpolation [39] to construct the context representations. The obtained segmentation accuracies are lower than our SCN.

In the next test, we disable the guidance of super-pixels only, followed by top-down information propagation [29]. Without super-pixels, we perform switchable process propagation on the compression and expansion feature maps as [30] and [53], where the information propagation is defined by regular kernels. Compared to this setting, our complete SCN has better performance. In addition to the fact that super-pixels provide more natural information propagation, the average depth computed on each super-pixel enables more stable feature switching by avoiding noisy depth of isolated regions.

We also study the case where depth is not used in the top-down switchable information propagation. In this case, we, respectively, regard the compression and expansion feature maps as context representations, as shown in Table IV. We also compare our method with the previous best method [35] that uses super-pixels without the depth. They underperform the switchable construction of context representations driven by the depth. This is because the depth data is not used to differentiate the far and near regions, for which the compression and expansion feature maps cannot be switched accordingly.

4) Compact Features for Adjusting Contextual Information: The top-down switchable information propagation consists of compression and expansion architectures, which provides different contextual information. These architectures use compact features to generate context representations. In this experiment, we study our compression and expansion architectures, and show that they can achieve effective compact features to adjust the contextual information.

In Table V, we provide a comparison on different designs of compression architectures. We note that a naive way to conduct information compression is to apply a 1×1 convolution for learning the compact features and a consequent 1×1 convolution for restoring the feature dimension, which yields lower accuracy than our compression architecture. Compared to the simple alternative that uses two sequential 1×1 convolutions, our compression architecture involve a 3×3 convolution [Fig. 4(b)] between two 1×1 convolutions. To some extent, the 3×3 convolution achieve wider range of contextual information, complementing the compact features resulted by dimension reduction that may lead to the loss of information. We note the features attained by 3×3 convolution of our compression architecture are still compact. When we remove the last 1×1 convolution used for restoring the feature dimension, and directly use the 3×3 convolution to produce the relatively high dimension features, we find that the performance is lower than our compression architecture. It shows the importance of the compact features generated by our 3×3 convolution.

In Table VI, we study our expansion architecture, and compare it with different ways of information expansion. Again, we simply use a single convolution that has 7×7 kernels to enlarge the receptive files, which produces the segmentation score of 43.8. We suppose that adding extra convolution having large kernels is able to improve the performance further. Thus, we use two 7×7 convolutions to achieve a better score of 44.2. But we empirically observe that adding more convolutional layer yields negligible improvement. We note that the segmentation scores produced by the above convolutions are lower than our expansion architecture, which use 1×1 convolution to compute compact features.

5) Comparisons With State-of-the-Art Methods: In Table VII, we compare our SCN with state-of-the-art methods. We follow the comparison rule in [35] by dividing all of the methods into two groups. We note that all of the methods are evaluated on the NYUDv2 test set.

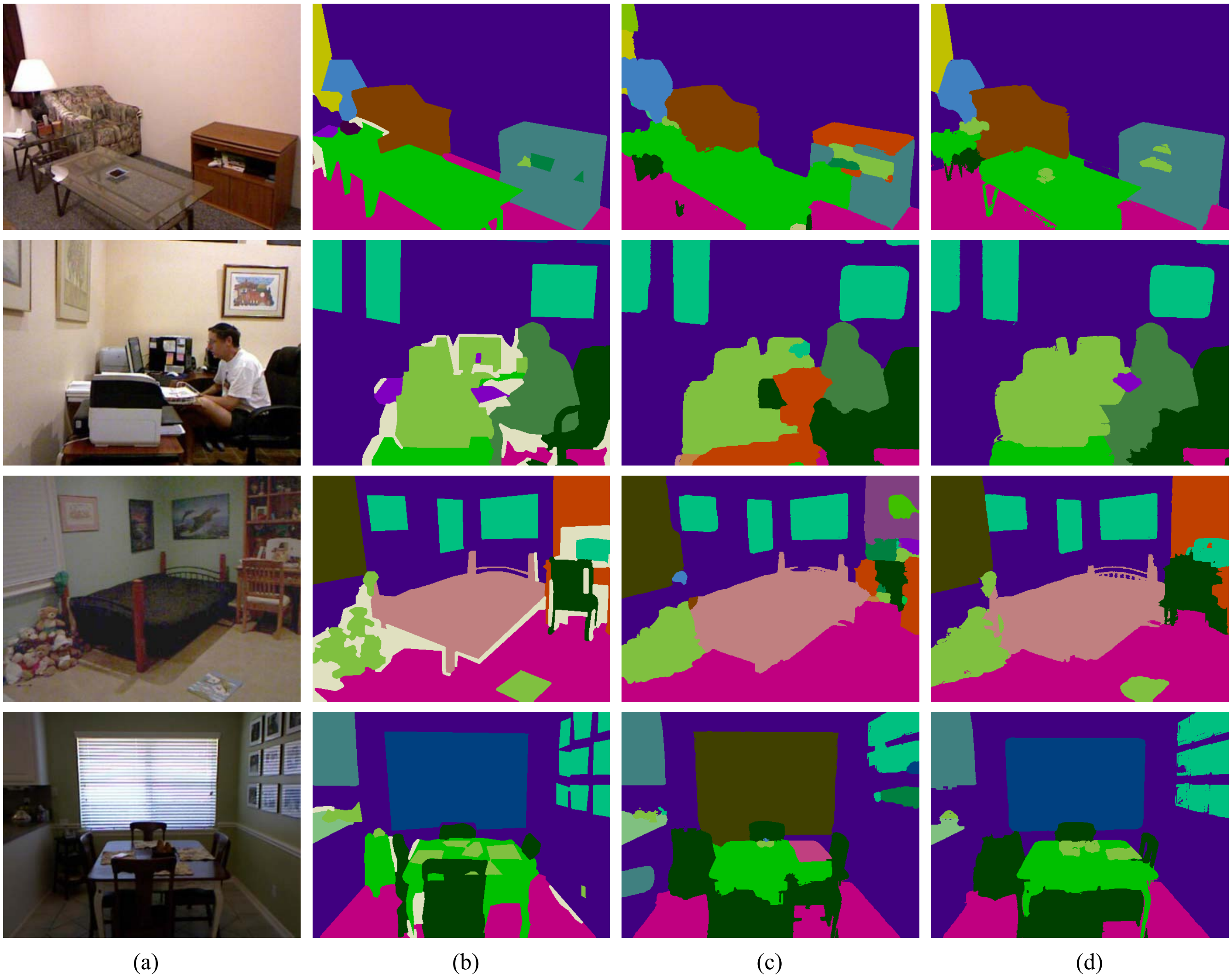


Fig. 5. Sample of the comparison to the state-of-the-art model [35] and our SCN. The images are selected from NYUDv2 [38] dataset.

The first group consists of the methods that use only RGB images for segmentation. We list their performances in the column *RGB-input* of Table VII. Typically, the deep networks proposed by Lin *et al.* [28] have top-down information propagation, yielding high-quality segmentation features. The accuracy reported in [28] is the highest in this group.

We also compare our SCN with the second group of methods, which take RGB-D images as input. The performances are reported in the column *RGB-D-input* of Table VII. We encode each depth image into an HHA image with 3 channels for maintaining richer geometric information [22], [49]. Following Long *et al.* [23], we use HHA images to train an independent segmentation network in place of RGB images. The trained network is tested on the HHA images for segmentation score map, which is combined with the score map computed by the network trained on RGB images. Using this combination strategy, the previous best result is 47.7 achieved by Lin *et al.* [35]. Compared to the network in, we find that using both RGB and HHA images improves the segmentation accuracy.

TABLE VII
COMPARISONS WITH OTHER STATE-OF-THE-ART METHODS ON THE NYUDv2 TEST SET. EACH SEGMENTATION ACCURACY IS REPORTED IN TERMS OF MEAN IOU (%)

RGB-input	mean IoU	RGB-D-input	mean IoU
		Gupta et al. [22]	28.6
		Fayyaz et al. [54]	30.9
		Deng et al. [33]	31.5
Long et al. [23]	29.2	Long et al. [23]	34.0
Kendall et al. [55]	32.4	Eigen et al. [56]	34.1
Lin et al. [32]	40.6	He et al. [24]	40.1
Zhao et al. [34]	45.2	Lin et al. [28]	47.0
Lin et al. [28]	46.5	Lin et al. [35]	47.7
		SCN (ResNet-101)	48.3
		SCN (ResNet-152)	**49.6**

We also use RGB and HHA images as training and testing data. Based on ResNet-101, our SCN achieves the score of 48.3. This score is better than the results reported in [28] and [35], where the deeper ResNet-152 architecture is used. We further employ the ResNet-152 to construct our

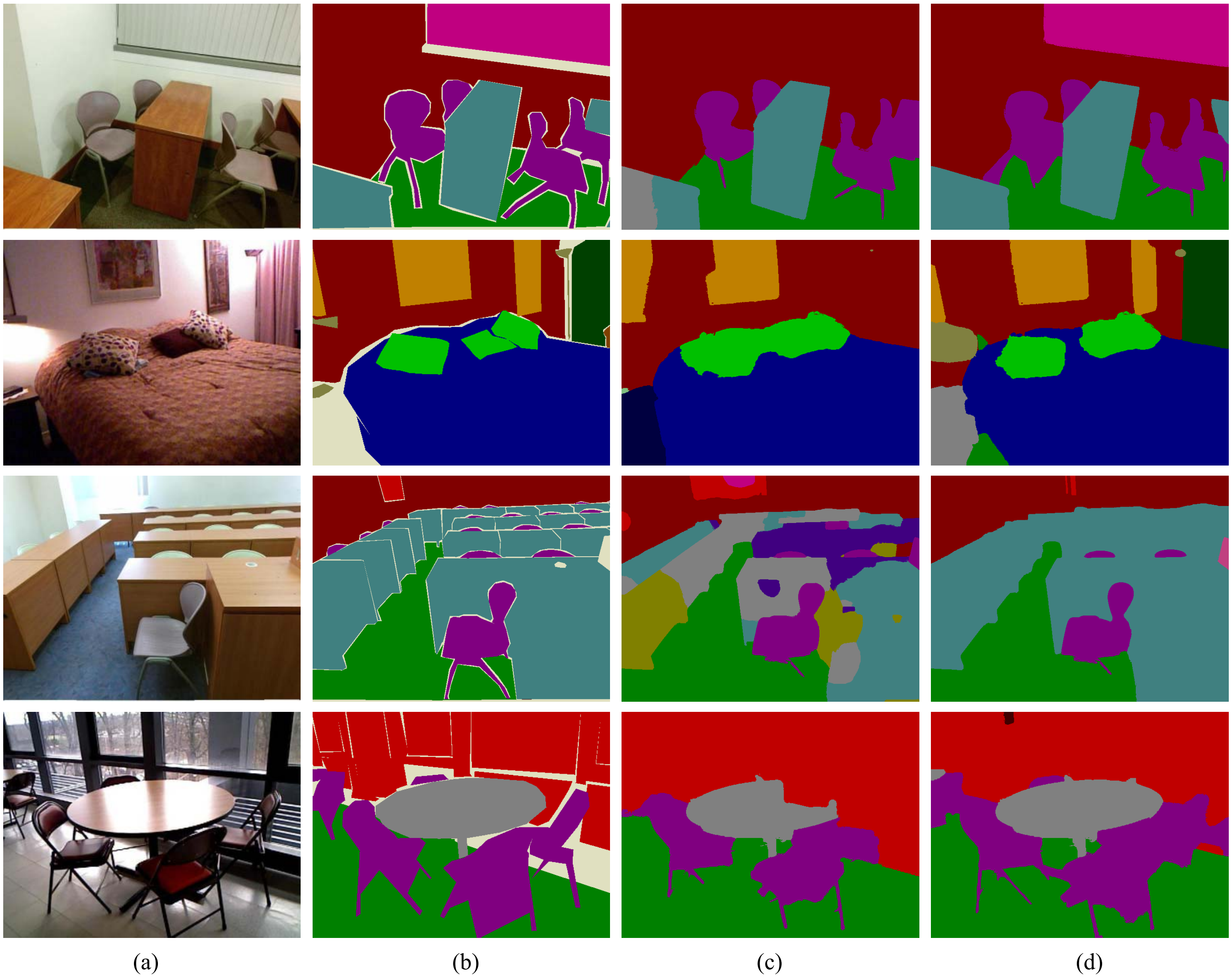


Fig. 6. Sample of the comparison to the state-of-the-art model [35] and our SCN. The images are selected from SUN-RGBD [21] dataset.

SCN, which enhances the segmentation score to 49.6. This result is better than state-of-the-art methods by about 2%. It demonstrates that our SCN can use different networks to improve the segmentation result in general. The network proposed by Lin *et al.* [35] also uses super-pixels and depth to construct context representations. Nonetheless, the switchable information propagation is not developed in [35]. In Fig. 5, we show the visual improvement against over state-of-the-art network of [28]. We also remove the local structural information propagation and top-down switchable propagation from the full model, resulting in 4.2% performance drop. By adaptively using the depth and super-pixel information, our method substantially reduces the segmentation error in image regions.

B. Experiments on SUN-RGBD Dataset

We also evaluate our method on the SUN-RGBD dataset [21] as well, which contains 10 335 images labeled with 37 classes. Compared to the NYUDv2 [38] dataset, the SUN-RGBD [21] dataset has more complex scene and depth conditions, which are probably more suitable to measure the generality of our method. From this dataset, we select 5285 images for training and the rest for testing.

TABLE VIII
ABLATION EXPERIMENT ON LOCAL STRUCTURE AND TOP-DOWN SWITCHABLE INFORMATION PROPAGATION. THE RESULTS ARE EVALUATED ON THE SUN-RGBD TEST SET

super-pixel	depth	method	mean IoU
no	no	Noh et al. [44]	43.8
		Chen et al. [39]	44.2
no	yes	Lin et al. [29]	44.5
yes	no	expansion	45.3
		compression	47.6
		Lin et al. [35]	46.2
yes	yes	SCN	**49.5**

In Table VIII, we experiment with different methods of using the super-pixel and depth information. By using the super-pixel and depth information, we equip SCN with the local structural and top-down switchable information propagation to tackle the complicated scene structure, achieving better result than the compared approaches. We further report

TABLE IX
COMPARISONS WITH OTHER STATE-OF-THE-ART METHODS ON THE SUN-RGBD TEST SET. EACH SEGMENTATION ACCURACY IS REPORTED IN TERMS OF MEAN IoU (%)

RGB-input	**mean IoU**	**RGB-D-input**	**mean IoU**
Noh et al. [44]	22.6		
Long et al. [23]	24.1		
Chen et al. [39]	27.4	Long et al. [23]	35.1
Kendall et al. [55]	30.7	Hazirbas et al. [57]	37.8
Lin et al. [32]	42.3	Lin et al. [28]	47.3
Lin et al. [28]	45.9	Lin et al. [35]	48.1
		SCN (ResNet-101)	49.5
		SCN (ResNet-152)	**50.7**

the performances of our SCN and state-of-the-art methods in Table IX. In this experiment, we again compare our SCN with the methods that take both RGB and HHA images as input. The previous best performance on the SUN-RGBD dataset is produced by the method of Lin *et al.* [35]. We note that the model there is based on the ResNet-152 architecture. Thanks to the better handling of the information propagation, we can use the simpler ResNet-101 architecture to achieve better result than that of Lin *et al.* [35]. With a deeper ResNet-152, we obtain the segmentation accuracy of 50.7, which outperforms all of the compared methods. The visualization results of our SCN on SUN-RGBD [21] dataset can be viewed in Fig. 6.

VIII. CONCLUSION

The CNNs, which are trained on large-scale image data, have been quite helpful in semantic segmentation. In this paper, we show that the training of segmentation network can further benefit from the depth data. We present an SCN that is trained with the consistent guidance of super-pixels and the depth. Our network uses super-pixels to provide spatially variant information propagation, which augment the relationship between image regions in the context representations. In addition, our SCN takes advantage of the depth data for network branching, which adaptively selects useful contextual information for image regions that have different object co-existences. With intensive studies and careful comparisons, we demonstrate that our SCN outperforms state-of-the-art methods on two public datasets by about 2% segmentation accuracy.

On the other hand, our SCN uses the structured edge detection toolbox [37] to generate super-pixels. This process, however, heavily relies on the low-level image features, e.g., pixel values and gradients. In low-contrast input images that do not show obvious structure, the super-pixels may not separate objects accurately. In the future, we plan to make the generation of super-pixels sensitive to the semantic segmentation results. It would hopefully provide an adaptive way to adjust the super-pixels to further minimize the segmentation error.

ACKNOWLEDGMENT

The authors would like to thank the reviewers and the editors for their valuable comments.

REFERENCES

[1] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The PASCAL visual object classes (VOC) challenge,” *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.

[2] T.-Y. Lin *et al.*, “Microsoft COCO: Common objects in context,” in *Proc. ECCV*, 2014, pp. 740–755.

[3] R. Mottaghi *et al.*, “The role of context for object detection and semantic segmentation in the wild,” in *Proc. CVPR*, 2014, pp. 891–898.

[4] M. Cordts *et al.*, “The cityscapes dataset for semantic urban scene understanding,” in *Proc. CVPR*, 2016, pp. 3213–3223.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Proc. NIPS*, 2012, pp. 1097–1105.

[6] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. CVPR*, 2016, pp. 770–778.

[8] R. Ren, T. Hung, and K. C. Tan, “A generic deep-learning-based approach for automated surface inspection,” *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 929–940, Mar. 2018.

[9] J. Han, H. Chen, N. Liu, C. Yan, and X. Li, “CNNS-based RGB-D saliency detection via cross-view transfer and multiview fusion,” *IEEE Trans. Cybern.*, vol. 48, no. 11, pp. 3171–3183, Nov. 2018.

[10] Y. Chen, Y. Hu, L. Zhang, P. Li, and C. Zhang, “Engineering deep representations for modeling aesthetic perception,” *IEEE Trans. Cybern.*, vol. 48, no. 11, pp. 3092–3104, Nov. 2018.

[11] C. Cao, Y. Zhang, C. Zhang, and H. Lu, “Body joint guided 3D deep convolutional descriptors for action recognition,” *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 1095–1108, Aug. 2018.

[12] G. Cao, A. Iosifidis, K. Chen, and M. Gabbouj, “Generalized multi-view embedding for visual recognition and cross-modal retrieval,” *IEEE Trans. Cybern.*, vol. 48, no. 9, pp. 2542–2555, Sep. 2017.

[13] D. Lin, X. Shen, C. Lu, and J. Jia, “Deep LAC: Deep localization, alignment and classification for fine-grained recognition,” in *Proc. CVPR*, 2015, pp. 1666–1674.

[14] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, “ScribbleSup: Scribble-supervised convolutional networks for semantic segmentation,” in *Proc. CVPR*, 2016, pp. 3159–3167.

[15] P. Rodriguez *et al.*, “Deep pain: Exploiting long short-term memory networks for facial expression classification,” *IEEE Trans. Cybern.*, to be published.

[16] B. Xue and N. Tong, “DIOD: Fast and efficient weakly semi-supervised deep complex ISAR object detection,” *IEEE Trans. Cybern.*, to be published.

[17] M. Sun, Z. Zhou, Q. Hu, Z. Wang, and J. Jiang, “SG-FCN: A motion and memory-based deep learning model for video saliency detection,” *IEEE Trans. Cybern.*, to be published.

[18] D. Lin, C. Lu, H. Huang, and J. Jia, “RSCM: Region selection and concurrency model for multi-class weather recognition,” *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4154–4167, Sep. 2017.

[19] C. Lu, D. Lin, J. Jia, and C. Tang, “Two-class weather classification,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2510–2524, Dec. 2017.

[20] J. Deng *et al.*, “ImageNet: A large-scale hierarchical image database,” in *Proc. CVPR*, 2009, pp. 248–255.

[21] S. Song, S. P. Lichtenberg, and J. Xiao, “Sun RGB-D: A RGB-D scene understanding benchmark suite,” in *Proc. CVPR*, 2015, pp. 567–576.

[22] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, “Learning rich features from RGB-D images for object detection and segmentation,” in *Proc. ECCV*, 2014, pp. 345–360.

[23] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proc. CVPR*, 2015, pp. 1–10.

[24] Y. He, W.-C. Chiu, M. Keuper, and M. Fritz, “RGBD semantic segmentation using spatio-temporal data-driven pooling,” *CoRR*, vol. abs/1604.02388, 2016. [Online]. Available: http://arxiv.org/abs/1604.02388

[25] J. Wang, Z. Wang, D. Tao, S. See, and G. Wang, “Learning common and specific features for RGB-D semantic segmentation with deconvolutional networks,” in *Proc. ECCV*, 2016, pp. 664–679.

[26] F. Husain, H. Schulz, B. Dellen, C. Torras, and S. Behnke, “Combining semantic and geometric features for object class segmentation of indoor scenes,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 1, pp. 49–55, Jan. 2017.

[27] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, “Large kernel matters–improve semantic segmentation by global convolutional network,” in *Proc. CVPR*, 2017, pp. 1743–1751.

[28] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks with identity mappings for high-resolution semantic segmentation," in *Proc. CVPR*, 2016, pp. 5168–5177.

[29] T.-Y. Lin *et al.*, "Feature pyramid networks for object detection," in *Proc. CVPR*, 2017, pp. 1–9.

[30] X. Liang, X. Shen, J. Feng, L. Lin, and S. Yan, "Semantic object parsing with graph LSTM," in *Proc. ECCV*, 2016, pp. 125–143.

[31] M. Mostajabi, P. Yadollahpour, and G. Shakhnarovich, "FeedForward semantic segmentation with zoom-out features," in *Proc. CVPR*, 2015, pp. 3376–3385.

[32] G. Lin, C. Shen, A. van den Hengel, and I. Reid, "Efficient piecewise training of deep structured models for semantic segmentation," in *Proc. CVPR*, 2016, pp. 3194–3203.

[33] Z. Deng, S. Todorovic, and L. J. Latecki, "Semantic segmentation of RGBD images with Mutex constraints," in *Proc. ICCV*, 2015, pp. 1733–1741.

[34] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. CVPR*, 2017, pp. 6230–6239.

[35] D. Lin, G. Chen, D. Cohen-Or, P.-A. Heng, and H. Huang, "Cascaded feature network for semantic segmentation of RGB-D images," in *Proc. ICCV*, 2017, pp. 1320–1328.

[36] X. Li, K. Liu, and Y. Dong, "Superpixel-based foreground extraction with fast adaptive trimaps," *IEEE Trans. Cybern.*, vol. 48, no. 9, pp. 2609–2619, Sep. 2017.

[37] P. Dollár and C. L. Zitnick, "Structured forests for fast edge detection," in *Proc. ICCV*, 2013, pp. 1841–1848.

[38] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. ECCV*, 2012, pp. 746–760.

[39] 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," *CoRR*, vol. abs/1606.00915, 2016. [Online]. Available: http://arxiv.org/abs/1606.00915

[40] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang, "Semantic image segmentation via deep parsing network," in *Proc. ICCV*, 2015, pp. 1377–1385.

[41] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. ICCV*, 2015, pp. 1529–1537.

[42] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *CoRR*, vol. abs/1511.07122, 2015. [Online]. Available: http://arxiv.org/abs/1511.07122

[43] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *PAMI*, vol. 39, no. 12, pp. 2481–2495, 2017.

[44] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. ICCV*, 2015, pp. 1520–1528.

[45] G. Ghiasi and C. C. Fowlkes, "Laplacian pyramid reconstruction and refinement for semantic segmentation," in *Proc. ECCV*, 2016, p. 55.

[46] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, vol. abs/1706.05587, 2017. [Online]. Available: http://arxiv.org/abs/1706.05587

[47] D. Lin, C. Lu, R. Liao, and J. Jia, "Learning important spatial pooling regions for scene classification," in *Proc. CVPR*, 2014, pp. 3726–3733.

[48] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with Gaussian edge potentials," in *Proc. NIPS*, 2011, pp. 1–9.

[49] S. Gupta, P. Arbelaez, and J. Malik, "Perceptual organization and recognition of indoor scenes from RGB-D images," in *Proc. CVPR*, 2013, pp. 1–8.

[50] C. Couprie, C. Farabet, L. Najman, and Y. LeCun, "Indoor semantic segmentation using depth information," *CoRR*, vol. abs/1301.3572, 2013. [Online]. Available: http://arxiv.org/abs/1301.3572

[51] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *Proc. CVPR*, 2016, pp. 3640–3649.

[52] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[53] X. Liang *et al.*, "Semantic object parsing with local-global long short-term memory," in *Proc. CVPR*, 2016, pp. 3185–3193.

[54] M. Fayyaz *et al.*, "STFCN: Spatio-temporal FCN for semantic video segmentation," *CoRR*, vol. abs/1608.05971, 2016. [Online]. Available: http://arxiv.org/abs/1608.05971

[55] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SEGNET: Model uncertainty in deep convolutional encoder–decoder architectures for scene understanding," *CoRR*, vol. abs/1511.02680, 2015. [Online]. Available: http://arxiv.org/abs/1511.02680

[56] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. ICCV*, 2015, pp. 2650–2658.

[57] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "FuseNet: Incorporating depth into semantic segmentation via fusion-based CNN architecture," in *Proc. ACCV*, 2016, pp. 213–228.

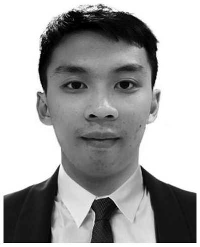
Di Lin (M'17) received the bachelor's degree in software engineering from Sun Yat-sen University, Guangzhou, China, in 2012, and the Ph.D. degree from the Chinese University of Hong Kong, Hong Kong, in 2016.

He is currently an Assistant Professor with the Visual Computing Research Center, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. His current research interests include computer vision and machine learning.

Ruimao Zhang (M'16) received the B.E. and Ph.D. degrees from Sun Yat-sen University, Guangzhou, China, in 2011 and 2016, respectively.

He is currently a Post-Doctoral Research Fellow with the Department of Electronic Engineering, Chinese University of Hong Kong, Hong Kong. From 2013 to 2014, he was a visiting Ph.D. student with the Department of Computing, Hong Kong Polytechnic University, Hong Kong. His current research interests include computer vision, deep learning, and related multimedia applications.

Dr. Zhang currently serves as a Reviewer for several academic journals, including the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, *Pattern Recognition*, and *Neurocomputing*.



Yuanfeng Ji (S'17) received the bachelor's degree in electronic information from Shenzhen University, Shenzhen, China, in 2018.

He is currently a Research Assistant with the Visual Computing Research Center, College of Computer Science and Software Engineering, Shenzhen University. His current research interest includes computer vision.

Ping Li (M'14) received the Ph.D. degree from the Chinese University of Hong Kong, Hong Kong.

He is currently an Assistant Professor with the Macau University of Science and Technology, Macau, China. He holds one image/video processing national invention patent, and has excellent research project reported worldwide by ACM TechNews. His current research interests include image/video stylization, big data visualization, GPU acceleration, and creative media.

Hui Huang (SM'18) received the Ph.D. degree in applied mathematics from the University of British Columbia, Vancouver, BC, Canada, in 2008, and the second Ph.D. degree in computational mathematics from Wuhan University, Wuhan, China, in 2006.

She is currently a Distinguished Professor with Shenzhen University, Shenzhen, China, where she directs the Visual Computing Research Center, College of Computer Science and Software Engineering. Her current research interests include computer graphics and computer vision.

Dr. Huang was a recipient of the NSFC Excellent Young Researcher Program Award and the Guangdong Technology Innovation Leading Talent Award. She is an Associate Editor-in-Chief of *Visual Computer* and is on the Editorial Board of *Computers and Graphics*. She is a Senior Member of ACM and a Distinguished Member of CCF.