Graph Guided Context Fusion for Semantic Segmentation

Lingzhi Li[1] Hao Yang[2] Dong Chen[2] Fang Wen[2]
[1]Peking University [2]Microsoft Research
lilingzhi@pku.edu.cn {haya,doch,fangwen}@microsoft.com

Abstract

Self-attention has made remarkable progresses in semantic segmentation by fusing global and dense context recently. However, the self attention aggregates all context without distinction. It may be confused by patches with similar appearances but different labels. In this paper, we propose a novel Graph-Guided Context Fusion (GGCF) module for semantic segmentation. It addresses the problem by fusing information from new local and global structures. The local structures fuse local contexts densely, while the global structures aggregate global contexts sparsely. The proposed GGCF module leverages the long-range context more effectively and efficiently. It also reduces the negative effects of contexts that have similar appearances but different semantic categories. Our approach using only the last layer of a FCN network as feature is able to advance the state-of-the-arts. Comprehensive analysis and extensive experiments have been conducted to show the advantages of the proposed method over traditional self-attention modules. Our method achieves the state-of-the-art results on Cityscapes benchmark.

1. Introduction

Image semantic segmentation is a fundamental topic in computer vision. It aims to classify each pixel into one of several semantic categories, such as person, sky, trees *etc*. The semantic segmentation techniques are useful for a wide range of applications like autonomous driving, image editing and so on.

Deep learning frameworks based on the fully convolutional networks (FCNs) [15] have made remarkable progresses in semantic segmentation. However, its performance is restricted by its limited receptive fields. To better recognize the semantic category of one pixel, it is necessary to also look at other pixels as context. Inspired by the popular self-attention mechanisms [22] from the Natural Language Processing (NLP) field, non-local operators [23] are recently introduced into semantic segmentation field [9, 27] to capture long-range dependency.

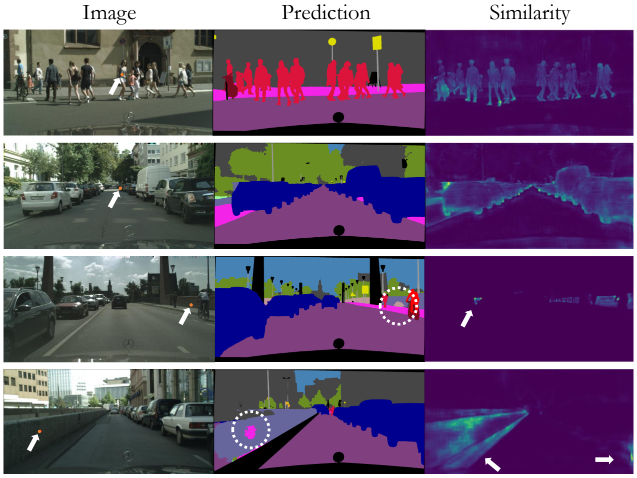


Figure 1: Feature similarity visualization. The second column shows predictions of a non-local network. The third column visualizes how similar backbone features distribute, with respect to the backbone feature whose positions are marked out in the first column. Higher brightnesses correspond to stronger similarities.

The non-local operators rely on pair-wise affinity values for context aggregation, where the new feature vector on each position is computed as a weighted average of all input feature contents. Through this way, a fully connected graph is constructed on top of all feature map positions. Feature contents at different positions will be fused together in the forward pass, as long as they share similar appearances, no matter how far they are from each other. By taking advantage of the context information from similar areas in the image, the non-local network is able to handle some hard cases, such as overlapped persons and small cars, as shown in the first two lines in Figure 1.

However, such a global and dense context fusion may conversely increase the difficulty in training classifiers under some situations. For example, certain instances of wall and fence may share similar appearances and often appear in same images, but their semantic labels are different. In these cases, in the training stage, the gradient signal toward a ground truth annotation of category A will also be backpropagated to other positions where the backbone network is actually seeing category B, if A and B share similar appearances. Thus, the backbone network will be confused and outputs indistinguishable features for these two categories. The last two lines in Figure 1 show samples under this situation.

We would like to fuse the long-range context for more effective training, but we also have to lower the effect of the BAD contexts where different categories have indistinguishable appearances. To address this predicament, our ideal is to treat local contexts and global contexts in different ways. The local contexts nearby are always more reliable than contexts distributed in far distances, since close regions always belong to the same categories. Therefore, we should strengthen the dependencies on more local contexts. As for the far-away global areas which contain both GOOD and BAD contexts, a flexible sampling is required to aggregate more useful information while avoiding the confusing contexts.

In this paper, we propose a novel *Graph-Guided Context Fusion* (GGCF) module to accomplish these two different tasks altogether in a uniform way. Within the GGCF module, we design LOCAL structures with different kernel sizes to densely fuse the local contexts within various ranges. We also design the GRID structures with different strides to sparsely fuse the global contexts following mixed sparsity. All these structures are combined together in GGCF, where trainable weights are applied to balance the importances between different structures. In another perspective, these weights adaptively determines a sparse graph which guides the context fusion.

In addition, we also propose a novel *Context Pruning* method to further improve the inference efficiency and efficacy of our context aggregation module. For each location, it is fused with N features with highest affinity values. The others will be neglected during inference. Through this way, we explicitly leverage more related contexts.

Experiments illustrate the advantages of our GGCF over traditional non-local modules, as well as the importance of the LOCAL and GRID structures. Our network based on GGCF also achieves the state of the art on the challenging Cityscapes benchmark.

To summarize, our contributions include:

1. We propose a novel GGCF module for semantic segmentation, which applies different strategies for local and global contexts;
2. We propose a novel Context Pruning method, which explicitly leverages more related contexts;
3. The proposed network based on GGCF achieves state-of-the-art performance on the challenging Cityscapes benchmark.

2. Related Work

Semantic Segmentation Semantic segmentation researches have benefited a lot from Fully Convolutional Networks (FCNs) [15]. Variants emerge to address the limited receptive field of FCN and enhance the ability of feature representation, through multi-level layer context fusion [17, 1], large kernel convolutions [16] or multi-scale aggregation via pyramid pooling [28] or dilated convolutions [26, 3, 4, 5, 6].

Instead of relying on local features only, some approaches exploit long-range dependencies using recurrent structures. In particular, Shuai *et al*. [20] designs a Directed Acyclic Graph RNN to embed distant contexts into local features for enhancing representative capability. Liu *et al*. [14] adopts additional convolution layers to approximate the Mean Field algorithm (MF) for capturing high-order relations. Liang *et al*. [13] utilizes a Graph LSTM to deal with general graph-structured data in semantic object parsing.

Inspired by the self-attention [22] structure for Natural Language Processing (NLP) and the non-local structure [23] for video classification. Fu *et al*. [9] and Yuan & Wang [27] explore the usage of self-attention across spatial dimensions and channel dimensions, in order to model long-range visual dependency among image feature maps. Huang *et al*. [12] proposes an crossing-shaped attention structure that is along horizontal and vertical directions to lower the computation cost of full self-attentions. These self-attention based methods all treat context pairs equally, resulting in an indistinct fusion of global contexts. We argue that not all context information are beneficial in improving the representative capability of features. Instances belonging to different categories appeared in same images with very similar appearances will conversely misguide the backbone training. We propose to introduce different guidances to the self-attention for local or global contexts respectively.

Attention Attention modules have attracted lots of favor in NLP owing to its advantage in modeling long-range dependency. The original self-attention [22] injects very implicit positional encodings into feature vectors, which are sine and cosine responses of different frequencies, in order to distinguish contexts at different locations. Instead, Shaw *et al*. [19] explicitly embeds relative positions into self-attention to model the ordering of elements in sequences. Yang *et al*. [25] proposes to model localness for self-attention networks by estimating a Gaussian, which enhances the ability of capturing useful local context. Dai *et al*. [8] extends self-attention by introducing a segment-level recurrent architecture which can encode absolute positions within the segment.

Meanwhile, the attention modules are also increasingly deployed in the image vision field, with more strategies for distinguishing contexts at different positions. For example, Hu *et al*. [11] proposes an object relation module to model the relationships among a set of objects. They distinguish objects with different relative positions by encoding a geometry weight into the relationship representation. Gu *et al*. [10] designs a region feature extraction module, where the attention mechanism is applied to model the geometric relationships between RoI and image positions.

In our work, we do not explicitly model the relative positions of contexts for semantic segmentation. Instead, we find it more effective to distinguish contexts through predefined LOCAL and GRID graph structures, which not only help fuse reliable contexts, but also reduces the computation overhead comparing with other self-attention modules.

3. Graph Guided Context Fusion

In this section, we formally introduce our method. Figure 2 shows an overview of our whole framework. The input image passes through a backbone network first, and generates a rich feature map $\mathbf{x}$. Our proposed *Graph Guided Context Fusion* (GGCF) module then takes $\mathbf{x}$ as input and performs context aggregation to enhance representative capability. The output feature map $\mathbf{y}$ will be fed into a convolution following by a softmax, and finally generates the label prediction.

In following sections, we first briefly introduce non-local operators [23] in Section 3.1. Then we will introduce the general formulation of the proposed GGCF module in Section 3.2. Next, we will explain the network designs in Section 3.3. Finally, we will proposed a new Context Pruning method to further improve the inference efficiency and efficacy in Section 3.4.

3.1. Non-local operators

Let $\mathbf{x}$ be the output of the backbone network, which is also the input feature map of the context fusion module. When deploying a vanilla non-local module, the computation of output feature vector $\mathbf{y}_i$ on 2D position i can be formulated as

$$\mathbf{y}_i = \frac{1}{C_i} \sum_j A(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j), \tag{1}$$

where the 2D index i and j traverses all positions within the feature map. Unary function g is applied to compute new representations of input feature $\mathbf{x}_j$. Binary function $A(\mathbf{x}_i, \mathbf{x}_j)$ calculates the *affinity value* that measures how important one feature vector is for another as a context. It is used to compute a weighted average of all features $g(\mathbf{x}_j)$ for context aggregation on i. In practice, as self-attention is usually adopted, A will be an Embedded Gaussian with $A(\mathbf{x}_i, \mathbf{x}_j) = \exp(\mathbf{x}_i^\intercal W_\theta^\intercal W_\phi \mathbf{x}_j)$, where W_θ, W_ϕ are trainable linear weights. C_i is a normalization factor.

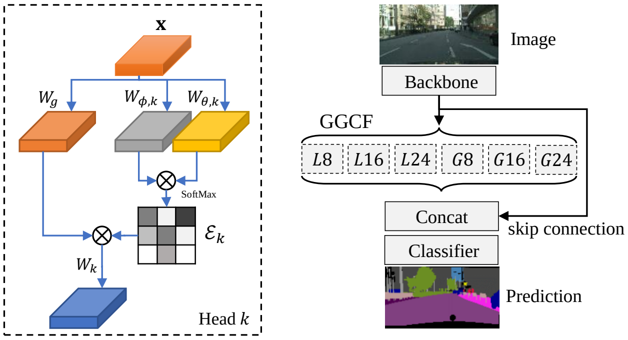


Figure 2: Network architecture of the proposed Graph Guided Context Fusion (GGCF). Structure of each head is shown in the left. Whole network is given in the right.

To increase the capability of the non-local module, multiple heads are applied according to [22], which becomes

$$\mathbf{y}_i = \sum_k^K \left(\frac{W_k}{C_{i,k}} \sum_{\forall j} A_k(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j) \right), \tag{2}$$

where K is the number of heads. A_k varies among different heads. W_k is the trainable linear weight to fuse the context from the k-th head.

3.2. Graph Guided Context Fusion Module

The problem of the default non-local module is that it treats all context pairs equally. In particular, the affinity $A(\mathbf{x}_i, \mathbf{x}_j)$ is completely feature-determined, it does not consider any other information between indexes i, j. The default non-local module is also computation expansive. It has to consider all position pairs, resulting in a complexity of $O(h^2 w^2)$ with $h \times w$ being the size of feature maps $\mathbf{x}$ and $\mathbf{y}$.

To improve the structural awareness in context fusion, we construct a graph whose vertexes are 2D positions within the feature map, and its vertex adjacency determines which positions to consider during context fusion. From this perspective, it can be regarded that the default non-local module is actually adopting a fully connected graph, where every two vertexes are adjacent, and should thus contribute context information to each other, which seems unnecessary and ineffective.

Instead, we propose to embed different graph structures to each head, where position pairs join context computation only when they are adjacent within the graph. The formulation then becomes

$$\mathbf{y}_i = \sum_k^K \left(\frac{W_k}{C_{i,k}} \sum_{\forall j} \mathcal{E}_k(i,j) A_k(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j) \right), \tag{3}$$

where $\mathcal{E}_k$ represents the *adjacency matrix* of the graph defined in the k-th head. $\mathcal{E}_k(i, j) = 0$ if j is not considered as a context of i in this head. In this way, the distributions of context to aggregate will be more varied than the original multi-head non-local modules. It is hence more capable to capture the subtle differences among contexts from different structures. For convenience, we will refer $\mathcal{E}_k$ as both the adjacency matrix and the graph it represents in all following texts.

3.3. Head Graph

We consider it reasonable to follow the three principles below when designing the graph structure $\mathcal{E}_k$ for each head:
Translation Invariance: $\mathcal{E}_k(i+\delta_i, j+\delta_j) \equiv \mathcal{E}_k(i,j)$ should hold for any valid position pairs (i,j) and $(i+\delta_i, j+\delta_j)$, where δ_i and δ_j are 2D offsets;
Dimension Invariance: It is common practice in semantic segmentation that the input image sizes of a network may vary. As a result, the size of input/output feature maps **x**/**y**, $h \times w$, is not always the same either. The design of $\mathcal{E}_k$ should respect such uncertainty;
Sparsity: We aim to address the inefficiency of fully connected self-attention modules. The sparser the graph $\mathcal{E}_k$ is, the fewer computation will be required for context fusion.

There exist many graph structures that satisfy the above principles. It is however unnecessary to apply many complex graph structures, if they can be approximately represented as linear combinations of simpler graph structures. Based on these assumptions, we propose two simple head structures in this work: the LOCAL structure and the GRID structure. The LOCAL structure is designed as

$$\mathcal{E}_S^{\text{LOCAL}}(i,j) = \begin{cases} 1 & \text{if } \max\{\|i_x - j_x\|, \|i_y - j_y\|\} \le S \\ 0 & \text{otherwise} \end{cases}, \tag{4}$$

which focuses on the *local* context densely, where the parameter S controls the size of the effective window. While the GRID structure is designed as

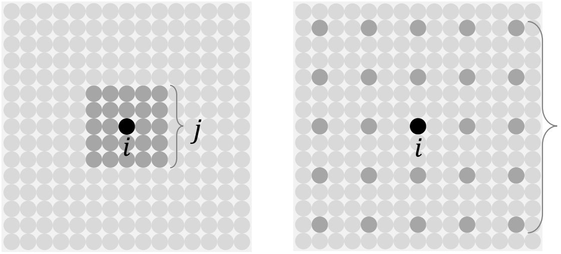
$$\mathcal{E}_T^{\text{GRID}}(i,j) = \begin{cases} 1 & \text{if } i_x \equiv j_x \wedge i_y \equiv j_y \mod T \\ 0 & \text{otherwise} \end{cases}. \tag{5}$$

It covers the *global* context sparsely, its parameter T controls the sparsity of sampling.

Fig. 3 visualizes the adjacency of the two graph structures. We expect that a bunch of the LOCAL heads together can emulate a flexible Gaussian kernel for localness modeling; while a bunch of the GRID heads together can leverage global contexts more flexibly.

3.4. Context Pruning

In addition, we propose the *Context Pruning* method to further improve the inference efficiency and efficacy of our context aggregation module. A sparser subgraph $\mathcal{E}_k^*(N) \subset \mathcal{E}_k$ is adopted in Equation 3 for inference, instead of the graph $\mathcal{E}_k$ used for training. N is an integer that controls the size of $\mathcal{E}_k^*(N)$. In specific, the elements of $\mathcal{E}_k^*(N)$ are the top N valid position pairs (i,j) in $\mathcal{E}_k$ that have high affinity values $A(\mathbf{x}_i, \mathbf{x}_j)$. All other position pairs not belonging to $\mathcal{E}_k^*(N)$ will be pruned, their corresponding context features will be neglected during inference. Through this way, we expect the module to explicitly leverage more related informative contexts.



(a) LOCAL (b) GRID

Figure 3: The two graph structures we propose. Contextual information will be fused from j (dark gray dots) to i (black dot) in both structures.

3.5. Implementation Details

In detail, the input feature map $\mathbf{x}$ is forwarded into multiple heads in the module. Within each head, we follow self-attention and utilize the Embedded Gaussian for affinity calculation. Specifically, we apply two different 1×1 convolutions $W_{\theta,k}, W_{\phi,k}$ for computing the *query* features and the *key* features of the attention module. Another 1×1 convolution W_g is applied to generate the *value* features. Then for each 2D position index i, we aggregate all the input feature vectors $\mathbf{x}_j$ in its neighboring set $\{\forall j \text{ s.t } \mathcal{E}_k(i,j) > 0\}$ using Equation 3, with $g(\mathbf{x}_i) = W_g\mathbf{x}_i$, $A(\mathbf{x}_i, \mathbf{x}_j) = \exp(\mathbf{x}_i^\intercal W_\theta^\intercal W_\phi \mathbf{x}_j)$. The normalization factor $C_{i,k} = \sum_j \mathcal{E}_k(i,j) A_k(\mathbf{x}_i, \mathbf{x}_j)$. When Context Pruning is applied during inference, we shall pick the top-N positions within the neighboring set of i according to the affinity $A(\mathbf{x}_i, \mathbf{x}_j)$, and use only these N feature vectors for context aggregation.

We then apply different 1×1 convolutions W_k to the output feature maps of each head, and sum them together as the context fusing feature map $\mathbf{y}$. It is concatenated with a skip connection from $\mathbf{x}$. Finally, the result passes through a last 1×1 convolution and a softmax for classification, and generates the label prediction.

4. Experiments

To evaluate the proposed method, comprehensive experiments are performed on two common datasets of semantic segmentation: the Cityscapes dataset [7] and the ADE20K dataset [31].

The Cityscapes consists of 5,000 street-view images captured from 50 different cities. The resolution of each image is 2048×1024, which is annotated in high quality pixel-level labels from 19 semantic classes. There are 2,979 images in training set, 500 images in validation set and 1,525 images in test set.

The ADE20K contains over 22K images. The types of scenes in ADE20K are much more diverse than those in Cityscapes. Each image is annotated with pixel-level labels from 150 semantic classes. There are 20,210 images in the training set and 2,000 images in the validation set.

4.1. Training Setting

Following [27, 6], we employ the poly learning rate policy: $lr = (1 - \frac{iter}{total\_iter})^p \times init\_lr$. For Cityscapes training, we set $p = 2$, $init\_lr = 0.02$, $total\_iter = 50,000$. Momentum and weight decay are set as 0.9 and 5×10^{-4} respectively. Input images are randomly cropped as 769×769 patches with random rescaling (from 0.5 to 1.5) and random horizontal flipping. By default, we use the 2,975 fine-annotated images for training unless otherwise stated. Auxiliary loss [15] and Synchronized InPlace BatchNorm [18] are deployed for training all models, with a batch size of 8. As for training on ADE20K, we set $p = 0.9$ to update the poly learning rate, and the total iteration is $total\_iter = 900,000$.

4.2. Ablation Study

To prove the effectiveness of different components in GGCF architecture, ablation experiments are conducted on the validation sets of Cityscapes and ADE20K for comparison. Cityscapes results are reported in Table 1. Per-label results are given in Table 3. Results on ADE20K are reported in Table 2. Following the common literature [4, 5, 9], we also use mean IoU percentage for quantitative evaluation.

For convenience, we use Full to represent the fully connected graph structure, *i.e.* $\mathcal{E}_k(i,j) \equiv 1$, which is equivalent to a self-attention. We also use subscripts to represent the parameters of Local and Global heads. For example, Local$_{8,16,24}$ represents three Local heads with $S = 8, 16, 24$, respectively. The Atrous Spatial Pyramid Pooling (ASPP) [5] module is also included for comparison in Table 1.

It is observable from Tables 1 and 3 that a single Full structure *i.e.* self-attention, brings improvement over both baseline (no context fusion performed) and ASPP, which proves the effectiveness of long-range context fusion. More importantly, results from Tables 1, 3 and 2 also show that, by combining our proposed Local or Grid head structures , we can achieve remarkably better results than a single Full structure.

Table 1: Ablation results on Cityscapes validation set. All models adopt ResNet-101 as backbone.

Module	mean IoU
Baseline	76.0
ASPP	77.8
Full	78.8
Full + ContextPruning	79.1
Full, Full	78.7
Full, Local$_{10}$	78.9
Full, Local$_{8,16,24}$	79.3
Full, Grid$_{10}$	78.9
Full, Grid$_{8,16,24}$	78.3
Local$_{10}$	76.2
Local$_{8,16,24}$	76.2
Grid$_{10}$	77.6
Grid$_{10,10,10}$	78.7
Grid$_{8,16,24}$	79.4
$\{$Local, Grid$\}_{8,16,24}$	79.6
$\{$Local, Grid$\}_{8,16,24}$ + ContextPruning	**79.9**
Full, $\{$Local, Grid$\}_{8,16,24}$	79.0
Trainable $\mathcal{E}$	78.9

Table 2: Ablation results on ADE20K validation set. All models adopt ResNet-50 as backbone.

Module	mean IoU
Baseline	33.3
Full	40.3
$\{$Local, Grid$\}_{8,16,24}$	41.2
$\{$Local, Grid$\}_{8,16,24}$ + ContextPruning	**41.3**

Effectiveness of Local structure As shown in Figure 1, we attach additional Local heads to an existing Full head. Although the effect of one additional Local$_{10}$ (Full, Local$_{10}$) is marginal (78.9 v.s 78.8), attaching more local heads (Full, Local$_{8,16,24}$) brings obviously higher improvements (79.3 v.s 78.8). On the other hand, after attaching additional Local heads to the Grid heads, the $\{$Local, Grid$\}_{8,16,24}$ is also slightly better than Grid$_{8,16,24}$ (79.6 v.s 79.4). It verifies the effectiveness of Local heads.

Although localness is important, but it will bring no good result if only applying single or multiple Local heads without fusing any long-range global context. For example, both the Local$_{10}$ and the Local$_{8,16,24}$ results are just slightly better than the baseline (76.2 v.s 76.0) and significantly worse than the Full (76.2 v.s 78.8).

Effectiveness of Grid structure The Grid structure captures the global context in a more efficient and structure-aware way. It can be observed from Table 1 that: i) directly

Table 3: Detailed ablation results on Cityscapes validation set.

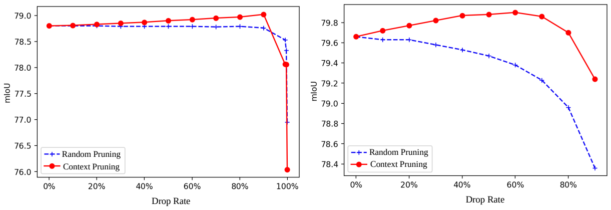
Module	mean IoU	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle
Baseline	76.3	98.3	85.9	92.8	52.3	59.7	66.7	72.8	80.4	92.7	64.6	94.8	83.5	65.4	95.0	67.6	80.9	59.0	59.3	78.8
FULL	78.8	98.4	86.5	92.8	54.2	60.2	66.4	72.7	**80.7**	92.7	65.5	94.9	**83.6**	**65.6**	95.5	78.9	89.6	75.3	64.5	79.1
$\{$LOCAL,GRID$\}_{8,16,24}$	79.6	98.4	86.6	93.0	58.4	60.7	**66.7**	**72.9**	80.5	92.9	66.2	94.8	83.2	65.0	95.6	82.7	90.6	79.8	65.5	**79.1**
↳ + ContextPruning	**79.9**	**98.4**	**86.7**	**93.1**	**60.1**	**60.7**	66.4	72.7	80.1	**92.9**	**66.6**	**95.0**	83.2	65.0	**95.6**	**83.6**	**90.7**	**79.9**	**67.8**	78.9

sparsifying the FULL into one GRID$_{10}$ deteriorates the overall performance significantly (77.6 v.s 78.8); ii) when combining multiple GRID as GRID$_{8,16,24}$, the prediction accuracy comes from behind, and outperforms the FULL (79.4 v.s 78.8). It can also be assured by ablation results that the structural difference within GRID heads plays an important role for their good performance. For example, combining three same GRID heads (GRID$_{10,10,10}$) does not bring comparable improvement (78.7 v.s 79.4); There is neither any goodness in combining two FULL heads (FULL,FULL) over a single FULL head (78.7 v.s. 78.8). On the other hand, we find that combining GRID with FULL (*e.g.* FULL, GRID$_{8,16,24}$) deteriorates performance comparing with either FULL (78.3 v.s 78.8) or GRID$_{8,16,24}$ (78.3 v.s 79.4).

Effectiveness of Context Pruning We compare context pruning on two different models: FULL and $\{$LOCAL, GRID$\}_{8,16,24}$, where unreliable contexts with low affinity values are dropped during inference. We also conduct a random context pruning on these two models for comparison, where the dropped contexts are chosen randomly. The mean IoU values with dropping rates ranging from 0 to 90% are recorded in Figure 4.

It shows an obvious advantage of our proposed context pruning over random context dropping: randomly dropping context in inference will always deteriorates the performances of both two models. On the contrary, the context pruning with appropriate dropping rates improves the performances of both two models. Pruning too much context, however, will eventually damages both the performances. Note that the amount of contexts in FULL is much larger than that of $\{$LOCAL, GRID$\}_{8,16,24}$ (approximately 300K v.s 0.5K). This explains why the performance changes in FULL is more gentle than $\{$LOCAL, GRID$\}_{8,16,24}$ for most drop rates, but more steeper near the 100% drop rate.

As reported in Table 1, Context Pruning with a best drop rate brings a gain of 0.3 for both the FULL structure (79.1 v.s 78.8) and the $\{$LOCAL, GRID$\}_{8,16,24}$ structure (79.9 v.s 79.6) on the Cityscapes validation set. The performance gain of $\{$LOCAL, GRID$\}_{8,16,24}$ on the ADE20K validation set is relatively small (41.3 v.s 41.2), as reported by Table 2.



(a) FULL (b) $\{$LOCAL, GRID$\}_{8,16,24}$

Figure 4: Context Pruning v.s Random Pruning.

Trainable Graph Guidance Our proposed module adopts multiple predefined graph structures $\mathcal{E}_k$ to guide context fusion. However, a more straight-forward idea for graph guidance is directly making $\mathcal{E}_k(i,j)$ a trainable function, implemented by an additional neural network.

In specific, we design a new module with *trainable graph guidance*. For each position pair (i,j), we feed a 3-vector $(i_x - j_x, i_y - j_y, \|i-j\|_2)$ into an additional network consisting of two fully connected layers. The output is a scalar and serves as the graph adjacency $\mathcal{E}_k(i,j)$ in Equation 3. We train this additional network together with the segmentation pipeline in an end-to-end fashion. Such a single headed trainable graph guided module just achieves a marginally better result over the FULL structure (78.9 v.s 78.8), as shown by the bottom row of Table 1.

4.3. Analysis

Qualitative Comparison In the Cityscapes dataset, some semantic categories are more likely to confuse with each other due to their similar appearances, *e.g. wall* and *building*, *vegetation* and *terrain*, *bus*, *truck* and *train*. However, self-attention modules can not address such issues well, since the similarity values they used for context pooling are all feature determined, which can not distinguish similar appearances either. This is verified in Figure 5, where some qualitative results are present for comparison. From these results, we can observe that the FULL structure is easy to

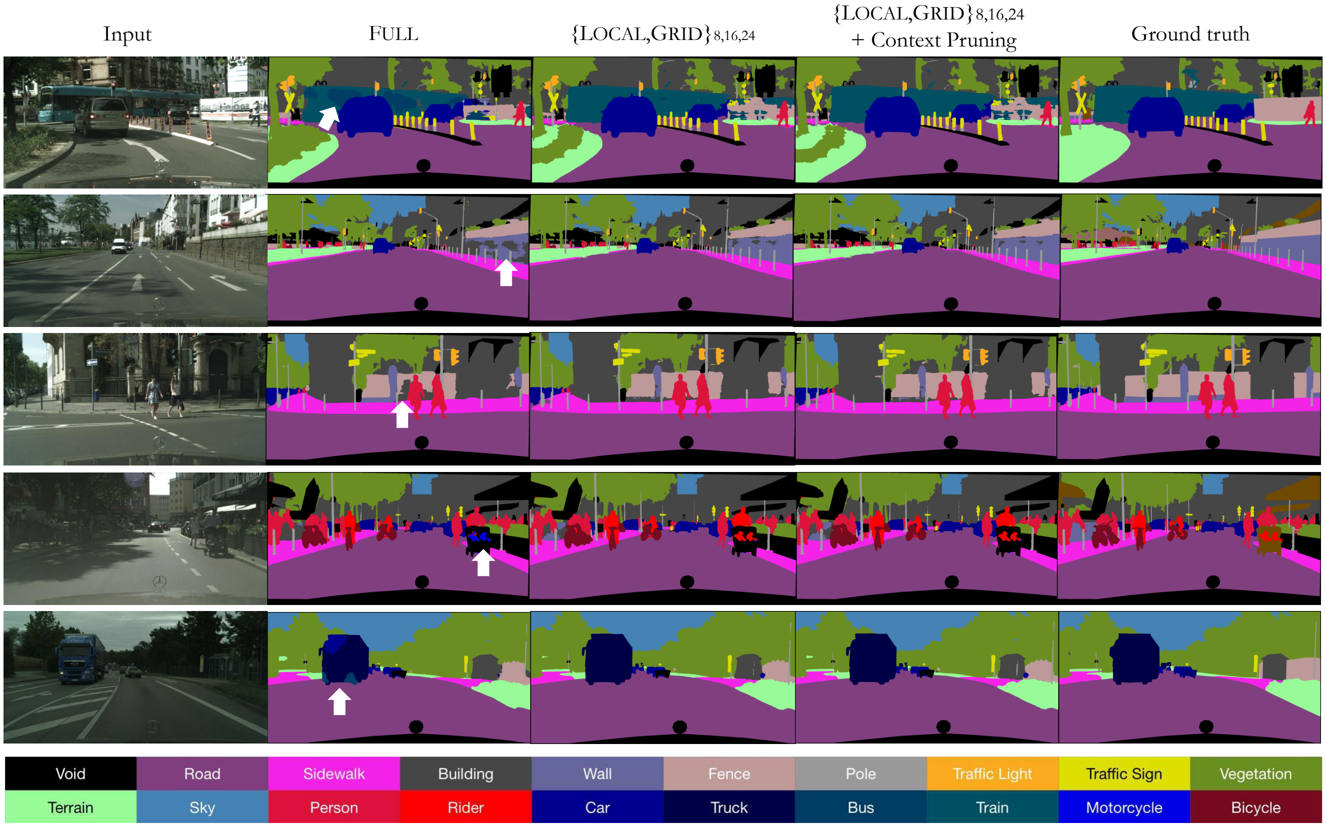


Figure 5: Qualitative comparison of different context fusion modules on Cityscapes validation set.

be misled by similar appearances. For example, parts of the *train* shown in the first row are mis-classified as *bus* by the FULL structure. In other samples, the FULL structure incorrectly recognizes *wall* or *fence* as *building*.

On the contrary, our proposed $\{\text{LOCAL}, \text{GRID}\}_{8,16,24}$ structure is able to conduct more accurate predictions for these confusing labels, as shown in the third column of Fig. 5. This can also be verified by the per-label results reported in Table 3, where we can see obvious advantages of the proposed $\{\text{LOCAL}, \text{GRID}\}_{8,16,24}$ over FULL on labels including *truck*, *bus* and *train*; *wall*, *fence* and *building*; *vegetation* and *terrain etc*. Finally, our context pruning technique achieves further polished predictions, as reported by Table 3 and shown in the forth column of Fig. 5.

Effective Receptive Field The two proposed LOCAL and GRID are both very sparse structures comparing with the original self-attention. In order to investigate whether such sparsity affects the empirical receptive fields of the network, we conduct an effective receptive field test [30] on the two models: FULL and $\{\text{LOCAL},\text{GRID}\}_{8,16,24}$.

In specific, we apply a 64 × 64 window onto the input image, all pixels within this window are replaced by the mean color of the image. As we slide this window within the image rectangle, responses at some positions in the last convolution output will change accordingly. We record all these changes measured by Euclidean distances as a heatmap. It reflects which part of the input image these pixel-wise classifiers are actually concerning about.

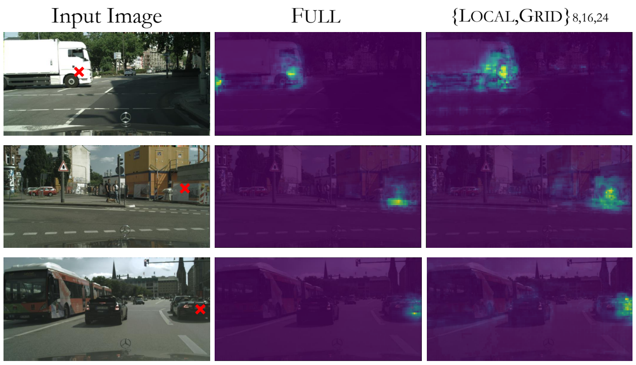


Figure 6: Comparing the receptive fields of the two models: FULL and the $\{\text{LOCAL},\text{GRID}\}_{8,16,24}$ with respect to the red × in the input image. Larger values are visualized in higher brightnesses.

Fig. 6 shows receptive fields of two models FULL and

Table 4: Comparison with the state-of-the-arts on Cityscapes testing set. All models we compare are trained using the fine and coarse annotations from Cityscapes training set.

Method	mean IoU	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle
PSANet [29]	80.1																			
PSPNet [28]	81.2	98.6	86.9	93.4	58.3	63.6	67.6	76.1	80.4	93.6	72.2	95.2	86.8	71.9	96.2	77.6	91.5	83.6	70.8	77.5
OCNet [27]	81.2	98.7	87.1	93.7	59.4	62.3	69.6	78.0	80.8	94.0	72.6	95.8	87.5	73.5	96.4	73.6	88.2	80.6	71.9	78.3
DeepLab v3 [5]	81.3																			
CCNet [12]	81.4																			
DANet [9]	81.5	98.6	86.1	93.5	56.1	63.3	69.7	77.3	81.3	93.9	72.9	95.7	87.3	72.9	96.2	76.8	89.4	86.5	72.2	78.2
InPlace-ABN [18]	82.0	98.4	85.0	93.6	61.7	63.8	67.6	77.4	80.8	93.7	71.8	95.6	86.7	72.7	95.7	79.9	93.0	89.7	72.5	78.2
DeepLab v3+ [6]	82.1	98.6	87.0	93.9	59.4	63.7	**71.3**	78.1	82.1	93.9	73.0	95.8	87.9	73.2	96.4	78.0	90.9	83.9	73.8	78.8
SSMA [21]	82.3	98.6	86.8	93.6	57.8	63.4	68.9	77.1	91.1	93.8	73.0	95.3	87.4	73.7	96.3	81.1	**93.4**	89.9	73.5	78.4
RelationNet [32]	82.4	98.8	87.8	94.0	**67.6**	64.3	70.2	77.0	81.1	93.9	73.5	95.8	87.8	73.3	96.4	75.3	89.4	88.1	72.0	78.2
DPC [2]	82.7	98.6	87.1	93.7	57.7	63.5	71.0	78.0	82.0	94.0	73.3	95.4	**88.2**	74.4	96.4	**81.1**	93.3	89.0	**74.1**	**78.9**
DRN-CRL [33]	82.8	98.8	87.7	93.9	65.0	64.1	70.0	77.3	81.5	93.9	73.4	**95.8**	88.0	**74.9**	96.4	80.8	92.1	88.4	72.0	78.7
Ours	**83.2**	**98.8**	**87.8**	**94.1**	66.0	**66.1**	71.1	**78.4**	**82.2**	**94.0**	**74.5**	95.7	88.0	74.0	**96.4**	79.9	92.4	**90.8**	71.7	78.3

the $\{\text{Local},\text{Grid}\}_{8,16,24}$, as well as the input images and the classifier positions (marked by a red crossing). Though sparser, the $\{\text{Local},\text{Grid}\}_{8,16,24}$ has effective receptive fields larger than the self-attention. Besides, the heat-map of the $\{\text{Local},\text{Grid}\}_{8,16,24}$ model in the first image obviously depicts the key regions of a truck better than the other one. For example, it covers the *headstock* and the *chassis* of the truck more compactly, while the other heat-map show that the FULL model focuses mostly on the appearance of the back wheel. This suggests an advantage of the proposed $\{\text{Local},\text{Grid}\}_{8,16,24}$ model over the self-attention in learning structural information.

Efficiency Under inference mode with the backbone feature size of 128×256, the proposed $\{\text{Local},\text{Grid}\}_{8,16,24}$ module has a computation overhead of 63 GFLOPs (excluding backbone). It is much smaller than the overhead of the self-attention structure FULL (520 GFLOPs), due to the sparse nature of both LOCAL and GRID.

4.4. Comparing with State-of-the-art

We train the proposed $\{\text{Local},\text{Grid}\}_{8,16,24}$ model on both the fine and coarse annotations from the Cityscapes training set. Following [18], we also adopt the WideResNet-38 pre-trained on Mapillary as the backbone. During training, we augment the image with a random rescaling chosen from $[0.7, 2]$ and a random aspect ratio adjustment by $[-0.1, +0.1]$. Then we randomly crop a 1024×1024 patch from the image and feed the patch into the network. We follow [9] and adopt both OHEM [24] and multi-grid for training. During inference, we apply multi-scale inputs and Context Pruning to further improve the performance.

The model is finally compared with current state-of-the-art methods on the Cityscapes test benchmark. We list our results in Table 4, as well as existing state-of-the-art methods. It shows that our best model, $\{\text{Local}, \text{Grid}\}_{8,16,24}+$ ContextPruning, achieves a mIoU of 83.2, surpassing all the state-of-the-arts in most categories.

5. Conclusions

In semantic segmentation, existing self-attention-based methods conduct context fusion in a dense and indistinct way, making it more likely to confuse semantic categories sharing very similar appearances. In this paper, we presented a Graph-Guided Context Fusion (GGCF) module to address this problem. The GGCF module consists of multiple LOCAL heads for capturing local contexts densely, as well as multiple GRID heads for aggregating global contexts sparsely. These head structures are combined together through trainable weights, in order to adaptively guide the context fusion. In addition, we propose the context pruning, which brings further improvements to GGCF by explicitly leveraging more informative contexts. Experiments show that our GGCF module is advantageous over original self-attention methods in semantic segmentation. Our method also surpasses the state-of-the-arts on the challenging Cityscapes benchmark.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation (segnet). *The Computing Research Repository (CoRR), abs/1511.00561*, 2015. 2

[2] L.-C. Chen, M. Collins, Y. Zhu, G. Papandreou, B. Zoph, F. Schroff, H. Adam, and J. Shlens. Searching for effi-

cient multi-scale architectures for dense image prediction. In *Advances in Neural Information Processing Systems*, pages 8713–8724, 2018. 8

[3] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014. 2

[4] 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 transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018. 2, 5

[5] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2, 5, 8

[6] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 801–818, 2018. 2, 5, 8

[7] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 4

[8] Z. Dai, Z. Yang, Y. Yang, W. W. Cohen, J. Carbonell, Q. V. Le, and R. Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019. 2

[9] J. Fu, J. Liu, H. Tian, Z. Fang, and H. Lu. Dual attention network for scene segmentation. *arXiv preprint arXiv:1809.02983*, 2018. 1, 2, 5, 8

[10] J. Gu, H. Hu, L. Wang, Y. Wei, and J. Dai. Learning region features for object detection. *arXiv preprint arXiv:1803.07066*, 2018. 3

[11] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3588–3597, 2018. 3

[12] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu. Ccnet: Criss-cross attention for semantic segmentation. *arXiv preprint arXiv:1811.11721*, 2018. 2, 8

[13] X. Liang, X. Shen, J. Feng, L. Lin, and S. Yan. Semantic object parsing with graph lstm. In *European Conference on Computer Vision*, pages 125–143. Springer, 2016. 2

[14] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang. Semantic image segmentation via deep parsing network. In *Proceedings of the IEEE international conference on computer vision*, pages 1377–1385, 2015. 2

[15] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1, 2, 5

[16] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun. Large kernel matters–improve semantic segmentation by global convolutional network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361, 2017. 2

[17] 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*, pages 234–241. Springer, 2015. 2

[18] S. Rota Bulò, L. Porzi, and P. Kontschieder. In-place activated batchnorm for memory-optimized training of dnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5639–5647, 2018. 5, 8

[19] P. Shaw, J. Uszkoreit, and A. Vaswani. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018. 2

[20] B. Shuai, Z. Zuo, B. Wang, and G. Wang. Dag-recurrent neural networks for scene labeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3620–3629, 2016. 2

[21] A. Valada, R. Mohan, and W. Burgard. Self-supervised model adaptation for multimodal semantic segmentation. *arXiv preprint arXiv:1808.03833*, 2018. 8

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 1, 2, 3

[23] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018. 1, 2, 3

[24] Z. Wu, C. Shen, and A. v. d. Hengel. High-performance semantic segmentation using very deep fully convolutional networks. *arXiv preprint arXiv:1604.04339*, 2016. 8

[25] B. Yang, Z. Tu, D. F. Wong, F. Meng, L. S. Chao, and T. Zhang. Modeling localness for self-attention networks. *arXiv preprint arXiv:1810.10182*, 2018. 2

[26] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 2

[27] Y. Yuan and J. Wang. OCNet: Object context network for scene parsing. *arXiv preprint arXiv:1809.00916*, 2018. 1, 2, 5, 8

[28] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 2, 8

[29] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, and J. Jia. Psanet: Point-wise spatial attention network for scene parsing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 267–283, 2018. 8

[30] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014. 7

[31] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 4

[32] Y. Zhuang, L. Tao, F. Yang, C. Ma, Z. Zhang, H. Jia, and X. Xie. Relationnet: Learning deep-aligned representation for semantic image segmentation. In *2018 24th Inter-*

national Conference on Pattern Recognition (ICPR), pages 1506–1511. IEEE, 2018. 8

[33] Y. Zhuang, F. Yang, L. Tao, C. Ma, Z. Zhang, Y. Li, H. Jia, X. Xie, and W. Gao. Dense relation network: Learning consistent and context-aware representation for semantic image segmentation. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 3698–3702. IEEE, 2018. 8