# Supplementary Material for Exploit Visual Dependency Relations for Semantic Segmentation

Mingyuan Liu, Dan Schonfeld and Wei Tang*
University of Illinois at Chicago
{mingyuan, dans, tangw}@uic.edu

## 1. Network Architecture

In this section, we describe more details of the proposed DependencyNet in Figure 1. The kernel sizes, operation types, and output channels are displayed in the boxes. The three dependency reasoning modules, *i.e*. intra-class, inter-class, and global, are distinguished by different colors in the figure.

The global reasoning (green) branch and the $L_{seg}$ (gray) branch consist of only group convolutions. As a result, there is no information exchange among different groups. Moreover, the $k^{th}$ group of features are only supervised by the ground truth of category $k$ so that they are guided to encode the spatial and semantic representations of objects that belong to the $k^{th}$ category. The intra-class dependence module (blue) updates the class-specific representations by two group convolutions. The inter-class dependency module (orange) conducts spatial and semantic reasoning among different classes via two group weighted convolutions (gwConv). The interactions among different categories are based on prior knowledge about the inter-class dependency relations, which is extracted from training annotations. Finally, the representation of each category is further refined by multiplying it with the probability that objects of the corresponding category exist in the whole scene.

For all group-based operations, *i.e*., group convolutions and group weighted convolutions, the number of groups equals the number of categories. For example, the number of groups is 20 for the Cityscapes dataset, which includes 19 foreground classes and a background class.

As for the complexity of our design, given a batch size of 4, an input size of 768x768 and a Resnet-101 baseline, the model size increases from 47.9M to 52.6M and the GPU memory consumption increases from 21.1G to 22.3G. In testing, the computational complexity increases from 1497 GFLOPS to 1617 GFLOPS. In our ablation study, we built the baselines by replacing our modules with conventional convolutions to retain the network depth and model size and showed that the performance gain was not caused by an unfair advantage of model complexity.

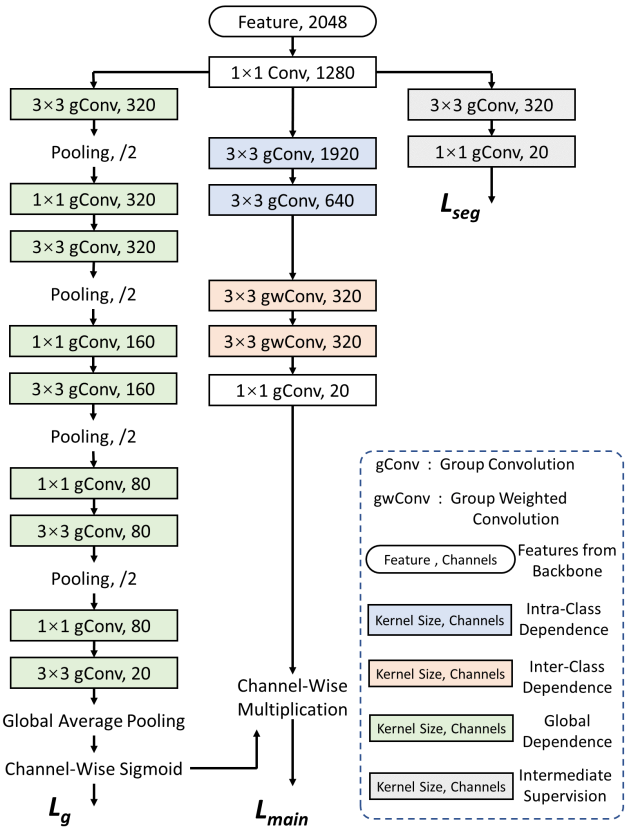


Figure 1. Detailed architecture of the proposed DependencyNet. All convolutions, except for those directly linked to the loss functions, are followed by a batch normalization and a ReLU activation.

## 2. Difference with Dependency Modeling in Pre-Deep Era

In this section, we detail the difference between the proposed DependencyNet and the dependency modeling in the pre-deep era.

*Corresponding author.

**Intra-class dependency.** The relations among parts and an object are conventionally modeled via deformable part-based models (DPMs) [3], compositional models [8, 18], and grammar models [19]. They explicitly model the displacement of each part w.r.t. the object. By contrast, we perform group convolutions on category-specific representations. More recently, [5] generates part proposals via a network and constrains them via the object for semantic part detection. They need part annotations during training. However, we do not use part proposals nor require part supervision, and our goal is pixel labeling.

**Inter-class dependency.** Some prior approaches [14, 4, 2, 1] build a CRF to model the co-occurrence statistics of objects and their spatial arrangements in an image. [7] divides objects into things and stuff, and explicitly models their spatial relations. Instead of using a graphical model, we incorporate the dependency graph in a CNN and perform reasoning via our novel group weighted convolutions, which have never been studied before.

**Global dependency.** [13, 11] predict the presence of each object via global image features and then use it to turn on/off local detectors in a graphical model. Several other works [15, 10, 16, 9, 17] retrieve the best matches of an input image from an annotated image database via global descriptors and transfer their labels via dense pixel or superpixel correspondence. [12] extends the DPM with potential functions modeling the presence of objects in the global image and local neighborhood. The work most related to ours is [6], which refines the detection score of a window by multiplying it with the probability of object presence in the image. Our global dependency module differs in that (1) it predicts the presence of each class via their respective category-specific representations, (2) it passes the global information to each pixel for semantic segmentation, and (3) it is built in a neural network and learned end-to-end with the backbone and other reasoning modules.

## 3. Ablative Study

**The Impact of Weights in Loss Function**. Recall the final loss function $L$ is

$$L = L_{main} + \lambda_1 \times L_g + \lambda_2 \times L_{seg} \quad (1)$$

$L_{main}$ supervises the learning of the final segmentation output. $L_g$ is designed to help the network to learn a global scene representation. $L_{seg}$ is an intermediate supervision to spatially supervise the category-specific representations. $\lambda_1$ and $\lambda_2$ are weights of $L_g$ and $L_{seg}$, respectively.

The influence of different values of $\lambda_1$ and $\lambda_2$ on the performance is demonstrated in Table 1 and Table 2, respectively. In the experiments, we use all the three dependency reasoning modules and only change the weights. The optimal performance is achieved when both $\lambda_1$ and $\lambda_2$ are set to 0.1.

| Backbone | $\lambda_1$ | mIoU% |
|---|---|---|
| ResNet50 | 0.05 | 77.14 |
| ResNet50 | 0.1 | 77.66 |
| ResNet50 | 0.3 | 76.37 |

Table 1. Ablative studies of weight $\lambda_1$ on the Cityscapes validation dataset. $\lambda_2$ is set to 0.1 in these experiments.

| Backbone | $\lambda_2$ | mIoU% |
|---|---|---|
| ResNet50 | 0.05 | 76.98 |
| ResNet50 | 0.1 | 77.66 |
| ResNet50 | 0.3 | 76.29 |

Table 2. Ablative studies of weight $\lambda_2$ on the Cityscapes validation dataset. $\lambda_1$ is set to 0.1 in these experiments.

| Backbone | Inter | mIoU% |
|---|---|---|
| ResNet50 | gwConv×1 | 74.50 |
| ResNet50 | gwConv×2 | 77.66 |
| ResNet50 | gwConv×3 | 76.75 |

Table 3. Ablative studies of the number of group weighted convolutions (gwConv) on the Cityscapes validation dataset.

| Backbone | Inter | mIoU% |
|---|---|---|
| ResNet50 | $\mathcal{G}_{conn}$ | 77.66 |
| ResNet50 | $\mathcal{G}_{edge}$ | 77.41 |
| ResNet50 | $\mathcal{G}_{mean-ari}$ | 76.85 |
| ResNet50 | $\mathcal{G}_{mean-geo}$ | 75.86 |
| ResNet50 | $\mathcal{G}_{mean-qua}$ | 77.05 |

Table 4. Ablative studies of different graph integration strategies. The graphs $\mathcal{G}_{mean-ari}$, $\mathcal{G}_{mean-geo}$ and $\mathcal{G}_{mean-qua}$ are respectively calculated by taking the arithmetic mean, geometric mean, and quadratic mean of $\mathcal{G}_{conn}$ and $\mathcal{G}_{edge}$.

**The Impact of the Number of gwConv**. The group weighted convolution (gwConv) is designed to exploit inter-class dependency relations for spatial and semantic reasoning. We use all three dependency modules in this ablative study and change the number of gwConv in the model. As shown in Table 3, the optimal number of gwConv is 2.

**The Impact of Different Graph Integration Strategies**. We have investigated two different strategies to discover the dependency graph from training annotations, *i.e.*, $\mathcal{G}_{conn}$ and $\mathcal{G}_{edge}$. They have their respective pros and cons. Here we study whether their integration can further improve the performance. Specifically, three different *averaging* methods are compared.

First, $\mathcal{G}_{mean-ari}$ is the arithmetic mean of the two graphs, whose edges are

$$\{e_{i,j}^{mean-ari} = (e_{i,j}^{conn} + e_{i,j}^{edge})/2 : \forall i, j\} \quad (2)$$

where $e_{i,j}^{conn}$ and $e_{i,j}^{edge}$ denote edges in $\mathcal{G}_{conn}$ and $\mathcal{G}_{edge}$ respectively, indicating the degree of category $i$'s dependency on category $j$.

Second, $\mathcal{G}_{mean-geo}$ takes the geometric mean of the two graphs:

$$\{e_{i,j}^{mean-geo} = \sqrt{e_{i,j}^{conn} \times e_{i,j}^{edge}} : \forall i, j\} \quad (3)$$

Third, $\mathcal{G}_{mean-qua}$ uses the quadratic mean, also known as the root mean square, of the two graphs:

$$\{e_{i,j}^{mean-qua} = \sqrt{((e_{i,j}^{conn})^2 + (e_{i,j}^{edge})^2)/2} : \forall i, j\} \quad (4)$$

In this ablative study, we use all intra, inter, and global reasoning modules and validate the effectiveness of different dependency graphs in the inter-class module. Results are displayed in Table 4. The three integrated graphs do not perform as well as the two individual graphs $\mathcal{G}_{conn}$ and $\mathcal{G}_{edge}$. One possible reason is that these integration methods have damaged some important relations encoded in the original graphs.

## References


[1] Myung Jin Choi, Joseph J Lim, Antonio Torralba, and Alan S Willsky. Exploiting hierarchical context on a large database of object categories. In *CVPR*, pages 129–136, 2010. 2

[2] Chaitanya Desai, Deva Ramanan, and Charless Fowlkes. Discriminative models for static human-object interactions. In *CVPR*, pages 9–16, 2010. 2

[3] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 32(9):1627–1645, 2009. 2

[4] Carolina Galleguillos, Andrew Rabinovich, and Serge Belongie. Object categorization using co-occurrence, location and appearance. In *CVPR*, pages 1–8, 2008. 2

[5] Abel Gonzalez-Garcia, Davide Modolo, and Vittorio Ferrari. Objects as context for detecting their semantic parts. In *CVPR*, pages 6907–6916, 2018. 2

[6] Hedi Harzallah, Frédéric Jurie, and Cordelia Schmid. Combining efficient object localization and image classification. In *ICCV*, pages 237–244, 2009. 2

[7] Geremy Heitz and Daphne Koller. Learning spatial context: Using stuff to find things. In *ECCV*, pages 30–43, 2008. 2

[8] Ya Jin and Stuart Geman. Context and hierarchy in a probabilistic image model. In *CVPR*, volume 2, pages 2145–2152, 2006. 2

[9] Jaechul Kim and Kristen Grauman. Shape sharing for object segmentation. In *ECCV*, pages 444–458, 2012. 2

[10] Ce Liu, Jenny Yuen, and Antonio Torralba. Nonparametric scene parsing: Label transfer via dense scene alignment. In *CVPR*, pages 1972–1979, 2009. 2

[11] Davide Modolo, Alexander Vezhnevets, and Vittorio Ferrari. Context forest for object class detection. In *BMVC*, volume 1, page 6, 2015. 2

[12] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, pages 891–898, 2014. 2

[13] Kevin P Murphy, Antonio Torralba, and William T Freeman. Graphical model for recognizing scenes and objects. In *NeurIPS*, pages 1499–1506, 2003. 2

[14] Andrew Rabinovich, Andrea Vedaldi, Carolina Galleguillos, Eric Wiewiora, and Serge Belongie. Objects in context. In *ICCV*, pages 1–8, 2007. 2

[15] Bryan Russell, Antonio Torralba, Ce Liu, Rob Fergus, and William Freeman. Object recognition by scene alignment. *NeurIPS*, 20:1241–1248, 2007. 2

[16] Joseph Tighe and Svetlana Lazebnik. Superparsing: Scalable nonparametric image parsing with superpixels. In *ECCV*, pages 352–365, 2010. 2

[17] Jimei Yang, Brian Price, Scott Cohen, and Ming-Hsuan Yang. Context driven scene parsing with attention to rare classes. In *CVPR*, pages 3294–3301, 2014. 2

[18] Long Zhu, Yuanhao Chen, Antonio Torralba, William Freeman, and Alan Yuille. Part and appearance sharing: Recursive compositional models for multi-view. In *CVPR*, pages 1919–1926, 2010. 2

[19] S. Zhu and D. Mumford. A stochastic grammar of images. *Foundations and Trends in Comp. Graphics and Vision*, 2(4):259–362, 2006. 2