Scene parsing with deep neural networks

Ding, Henghui

2020

Ding, H. (2020). Scene parsing with deep neural networks. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/142935>

<https://doi.org/10.32657/10356/142935>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 19:36:23 SGT

Scene Parsing with Deep Neural Networks

DING HENGHUI

School of Electrical & Electronic Engineering

2020

Scene Parsing with Deep Neural Networks

DING HENGHUI

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

2020

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

15-Dec-2019

.....

Date

.....

Ding Henghui

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

6-Jan-2020

.....

Date

.....

Prof. Jiang Xudong

Authorship Attribution Statement

This thesis contains material from 5 paper(s) published in the following peer-reviewed journal(s) / from papers accepted at conferences in which I am listed as an author.

Chapter 3 is published as [Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Context Contrasted Feature and Gated Multi-scale Aggregation for Scene Segmentation. *In Proceedings of the Conference on Computer Vision and Pattern Recognition \(CVPR\)*, IEEE, 2018. DOI: 10.1109/CVPR.2018.00254.](#)

The contributions of the co-authors are as follows:

- Prof. Jiang Xudong supervised me on the overall research direction.
- I developed the algorithm, conducted the experiments and prepared the manuscript drafts.
- Dr. Shuai Bing, Dr. Wang Gang and Prof. Liu Ai Qun provided valuable comments on the manuscript.

Chapter 3 is published as [Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Semantic Segmentation with Context Encoding and Multi-Path Decoding. *IEEE Transactions on Image Processing*, 2020. DOI: 10.1109/TIP.2019.2962685.](#)

The contributions of the co-authors are as follows:

- Prof. Jiang Xudong supervised me on the overall research direction.
- I developed the algorithm, conducted the experiments and prepared the manuscript drafts.
- Dr. Shuai Bing, Dr. Wang Gang and Prof. Liu Ai Qun provided valuable comments on the manuscript.

Chapter 4 is published as [Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Semantic Correlation Promoted Shape-Variant Context for Segmentation. *In Proceedings of the Conference on Computer Vision and Pattern Recognition \(CVPR\)*, IEEE, 2019. DOI: 10.1109/CVPR.2019.00909.](#)

The contributions of the co-authors are as follows:

-
- Prof. Jiang Xudong supervised me on the overall research direction.
 - I developed the algorithm, conducted the experiments and prepared the manuscript drafts.
 - Dr. Shuai Bing, Dr. Wang Gang and Prof. Liu Ai Qun provided valuable comments on the manuscript.

Chapter 5 is published as [Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang. Boundary-Aware Feature Propagation for Scene Segmentation. *In Proceedings of the International Conference on Computer Vision \(ICCV\)*, IEEE, 2019. DOI: 10.1109/ICCV.2019.00692.](#)

The contributions of the co-authors are as follows:

- Prof. Jiang Xudong supervised me on the overall research direction.
- I developed the algorithm, conducted the experiments and prepared the manuscript drafts.
- Dr. Wang Gang, Prof. Nadia Magnenat Thalmann and Prof. Liu Ai Qun provided valuable comments on the manuscript.

Chapter 6 is published as [Henghui Ding, Scott Cohen, Brian Price, and Xudong Jiang. PhraseClick: Toward Achieving Flexible Interactive Segmentation by Phrase and Click. *In European Conference on Computer Vision \(ECCV\)*, Springer, 2020.](#)

The contributions of the co-authors are as follows:

- Dr. Scott Cohen, Dr. Brian Price and Prof. Jiang Xudong supervised me on the overall research direction.
- I developed the algorithm, conducted the experiments and prepared the manuscript drafts.

15-Dec-2019

.....

Date

.....

Ding Henghui

Acknowledgements

First and most, thank you to my three advisors Prof. Jiang Xudong, Dr. Wang Gang and Prof. Liu Ai Qun. They have constantly supported and encouraged me throughout my Ph.D life in Nanyang Technological University. Dr. Wang Gang opened the door for me to enter the fantastic world of deep learning. Prof. Jiang Xudong has helped me a lot in how to do academic research, including problem analyzing, professional thinking, and manuscript writing, he always motivates me to do in-depth research. This thesis is completed under their constant encouragement and support.

Thank you to School of Electrical and Electronic Engineering, Nanyang Technological University, for awarding me the NTU research scholarship and supporting me with wonderful facilities and environment for my Ph.D study.

Thank you to my thesis advisory committee (TAC) members Prof. Yap Kim Hui and Prof. Cai Jianfei for their great support and encouragement.

Thank you to my awesome research collaborators: Dr. Shuai Bing, Dr. Liu Jun, Mr. Mei Jianhan, Ms. Wang Xiaohong. I have learned a lot from them.

Thank you to my colleagues in Rose lab: Dr. Gu Jiuxiang, Mr. Chen Jianda, Mr. Tay Chiat Pin, Dr. Liu Ting, Dr. Zheng Qian, Mr. Yan Ke, Ms. Wang Zhenzhen, Dr. Yu Tan, Dr. Ge Liuhao, Mr. Hu Ping, Dr. Amir Shahroudy, Mr. Jason Kuen, Mr. Wang Xingxing, Dr. Kong Xiangfei, Dr. Chen Zenghai, Dr. Si Jianlou, Ms. Li Ling. They filled my Ph.D life with happiness and sunshine.

Finally, I would like to dedicate this thesis to my parents and my familiars who have generously offered me love, encouragements and support.

Abstract

In this thesis, we address the fundamental and challenging task of scene parsing. Scene parsing (also known as semantic segmentation, scene segmentation, scene labeling) aims to classify every pixel of a given image to one of the predefined semantic categories (e.g., person, car, etc.), including not only countable objects (e.g. person, car, cat) but also uncountable stuff (e.g. road, grass, sky). It is a dense prediction task whose output has the same resolution with input, thus it provides much richer clues for scene understanding. As one of the fundamental tasks in computer vision, scene parsing has been an essential component of computer vision and is in intense demand for many practical applications, such as automation devices, virtual reality, augmented reality, self-driving vehicles and etc. Scene parsing implicitly involves object recognition, object localization and boundary delineation, which requires multi-scale and multi-level visual recognition. A robust segmentation approach needs to perform well at all of these implied tasks.

We address scene parsing based on deep neural networks and explore to enhance scene parsing performance from different aspects. In details, firstly, we discuss how to aggregate customized context information to enhance the high-level feature representation while keep the local discriminations. Then, we explore to make use of multi-scale and multi-level information to address the issue of huge scale variations of objects in scene parsing. Meanwhile, we discuss how to infer and refine the boundary predictions to enhance the spatial details. Furthermore, to segment the unseen/unknown objects that have never appeared in training set, we build an interactive image segmentation model, which greatly speeds up the pixel-level data annotation process and thus can alleviate the issue of lack of large-scale benchmarks.

First of all, we propose CGBNet that employs context encoding and multi-path decoding to enhance the segmentation performance from both encoding and decoding process. In CGBNet, we first propose a context encoding module that generates context-contrasted local feature to make use of the informative context and the discriminative local information. This context encoding module greatly improves the segmentation performance, especially for inconspicuous objects. Furthermore, we propose a scale-selection scheme to selectively fuse the segmentation results from different-scales of features at every spatial position. It adaptively selects appropriate score maps from rich scales of features. To improve the segmentation performance results at boundary, we further propose a boundary delineation module that encourages the location-specific very-low-level features near the boundaries to take part in the final prediction and suppresses them far from the boundaries.

Secondly, to further enhance the contextual modeling, we propose SVCNet that mainly target to build customized contextual model. Due to the diverse shapes of objects and their complex layout in various scene images, the spatial scales and shapes of contexts for different objects have very large variation. It is thus ineffective or inefficient to aggregate various context information from a predefined fixed region. In this work, we propose to generate a scale- and shape-variant semantic mask for each pixel to confine its contextual region. To this end, we first propose a novel paired convolution to infer the semantic correlation of the pair and based on that to generate a shape mask. Using the inferred spatial scope of the contextual region, we propose a shape-variant convolution, of which the receptive field is controlled by the shape mask that varies with the appearance of input. In this way, the proposed network aggregates the context information of a pixel from its semantic-correlated region instead of a predefined fixed region.

Thirdly, we propose a boundary-aware feature propagation module. To increase the feature similarity of the same object while keeping the feature discrimination of different objects, we explore to propagate information throughout the image under the control of objects’ boundaries. To this end, we first propose to learn the boundary as an additional semantic class to enable the network to be aware of

the boundary layout. Then, we propose unidirectional acyclic graphs (UAGs) to model the function of undirected cyclic graphs (UCGs), which structurize the image via building graphic pixel-by-pixel connections, in an efficient and effective way. Furthermore, we propose a boundary-aware feature propagation (BFP) module to harvest and propagate the local features within their regions isolated by the learned boundaries in the UAG-structured image. The proposed BFP is capable of splitting the feature propagation into a set of semantic groups via building strong connections among the same segment region but weak connections between different segment regions.

Finally, we build an interactive image segmentation model, PhraseClick, which is semi-automated and aims to accurately segment the image into foreground and background given a minimal amount of user interactive inputs. Existing interactive object segmentation methods mainly take spatial interactions such as bounding boxes or clicks as input. However, these interactions do not contain information about conspicuous attributes of the target-of-interest and thus cannot quickly specify what is the target area to select. Therefore, excessive user interactions are often required to reach desirable results. We propose to employ language phrases as another interaction input to infer the attributes of the target object. In this way, we can not only leverage spatial clicks to localize the target object but also utilize semantic phrases to qualify the attributes of the target object. Specifically, the phrase input focuses on “what” the target object is and the spatial clicks are in charge of “where” the target object is. Moreover, the proposed approach is flexible in terms of interaction modes and can efficiently handle complex scenarios by leveraging the strengths of each type of input.

In summary, with the four proposed segmentation models (CGBNet, SVCNet, BFP and PhraseClick), we have enhanced the scene parsing from different aspects: contextual modeling, multi-scale and multi-level information aggregation, boundary refinement, and unseen/unknown objects’ segmentation. We have achieved new state-of-the-art segmentation performance on public benchmarks.

Contents

Acknowledgements	ix
Abstract	xi
List of Figures	xix
List of Tables	xxiii
List of Abbreviations	xxv
1 Introduction	1
1.1 Challenges and Motivations	2
1.2 Proposed Segmentation Models	5
1.3 Contributions	7
1.4 Thesis Outline	8
2 Literature Review	9
2.1 Deep Neural Networks	9
2.2 Scene Parsing	11
2.2.1 Contextual Modeling	12
2.2.2 Multi-scale Aggregation	14
2.2.3 Boundary Delineation	14
2.2.4 Graphic models	15
2.2.5 Label variety	15
2.3 Boundary Detection	16
2.4 Interactive Segmentation	17
2.4.1 Spatial Constraints based Interactive Segmentation	17
2.4.2 Referring Expression Comprehension	19
2.4.3 Bringing Clicks and Phrases Together	20
2.5 Scene Segmentation Benchmark Datasets	20
2.6 Summary	22
3 Context Encoding and Multi-Path Decoding	23
3.1 Introduction	24

3.2	The Proposed Segmentation Network	27
3.2.1	Overall Framework	27
3.2.2	Context-Contrasted Local Feature	29
3.2.3	Gated Multi-scale Aggregation	32
3.2.4	Boundary Delineation Refinement	35
3.3	Experiments	38
3.3.1	Implementation Details	38
3.3.2	Multi-scale Context-Contrasted Local Features	39
3.3.3	Embed Gated Sum into Encoder-Decoder Architecture	41
3.3.4	Segmentation result comparisons of Boundary Delineation Refinement	42
3.3.5	Ablation Study and Runtime comparison	44
3.3.6	Comparison with the State-of-the-Arts	46
3.4	Chapter Summary	50
4	Semantic Correlation Promoted Shape-Variant Context	53
4.1	Introduction	54
4.2	The Proposed Approach	56
4.2.1	Semantic Correlation Dependent Context	56
4.2.2	Labeling Denoising	61
4.3	Experiments	64
4.3.1	Ablation Study	65
4.3.2	Visualization of the semantic shape mask	66
4.3.3	Comparison with the State-of-the-Arts	67
4.4	Chapter Summary	70
5	Boundary-Aware Feature Propagation	73
5.1	Introduction	74
5.2	Approach	76
5.2.1	Semantic Boundary Detection	78
5.2.2	Unidirectional Acyclic Graphs	79
5.2.3	Boundary-Aware Feature Propagation	81
5.3	Experiments	83
5.3.1	Implementation Details	83
5.3.2	Efficiency Analysis	84
5.3.3	Ablation Studies	84
5.3.4	Comparison with the State-of-the-Art Works	85
5.4	Chapter Summary	88
6	PhraseClick: Integration of “What” and “Where” for Interactive Segmentation	93
6.1	Introduction	94
6.2	Approach	97
6.2.1	Network Architecture	97

6.2.2	Click Interaction	97
6.2.3	Attribute Feature Attention Module	99
6.2.4	Training Loss	101
6.3	Experiments	102
6.3.1	Implementation Details	102
6.3.2	Ablation Studies	103
6.3.3	Results on Benchmarks	106
6.4	Chapter Summary	109
7	Conclusion and Future Work	111
7.1	Conclusion	111
7.2	Future works	114
	Author's Publications	117
	References	119

List of Figures

1.1	(Best viewed in color) Scene parsing refers to classifying every pixel of a given image to one of the predefined semantic categories (e.g., person, car, etc.), including not only countable objects (e.g. person, car, cat) but also uncountable stuff (e.g. road, grass, sky).	2
2.1	Architecture of VGG-16.	10
2.2	Inception block from GoogleNet [1] architecture and Residual block from the ResNet architecture [2].	10
2.3	Architecture of FCN based on VGG-16.	11
2.4	Eights representative benchmark datasets for scene segmentation. In the demonstration, we show input images and their ground truth.	21
3.1	(Best viewed in color) Scene segmentation refers to labeling every pixel of image including salient objects, inconspicuous objects and stuff. However, the various forms of objects/stuff (e.g salient or inconspicuous), the existence of multi-scale objects (e.g the multi-scale cows in third image) and the capricious boundary shape of objects make it challenging to parsing each pixel using DCNN.	25
3.2	The overall framework of the proposed CGBNet. The proposed context-contrasted local (CCL) model generates multi-level and multi-scale context-aware local features. Gated sum selectively aggregates rich scale features in DCNN and CCL. BDR encourages pixels near the boundaries to be assigned more very-low-level textural information.	28
3.3	(Best viewed in color) Visualization of different feature information. The local information of pixel A could not aggregate useful contexts, such as road and other cars. However, its contexts will be dominated by the features of the men in the both schemes of dense context and coarse context. The context-local scheme injects blur context to local feature of pixel A to make discriminative context-aware local feature.	29

3.4	context-contrasted Local (CCL) scheme is a convolutional network integrating multi-level context-aware local features. Each block of CCL consists of two parallel parts: coarse context convolution and delicate local convolution. The context-aware local features are obtained via making a contrast between the context and local information. Several blocks are chained to make multi-level context-contrasted local features.	31
3.5	Gated Sum controls the information flow of skip layers via its inherent gates. The gates G_p^n adjusts its value for each pixel according to the input images. All the maps in gated sum have the same spatial size of $H \times W$	34
3.6	Boundary Delineation: Although pixel B and Pixel C belong to the same object, their desirable scale features are different due to their distance to the center of the object.	36
3.7	We use isotropic Gaussian filters with different standard deviation ($\sigma^{1,2,3} = 1, 3, 5$) to the hard boundary mask generated by the initial prediction and get a soft boundary masks. Such masks engage the very-low-level features in the regions (near the boundary) with higher SNR and suppress those in the regions (far away from the boundary) with lower SNR.	38
3.8	Example of three 1D profiles b_x^m of the soft boundary masks $\mathcal{B}_p^m, m = 1, 2, 3$. t is the initial textural boundary position.	38
3.9	Visualize of gated weights: here we visualize gated weights of Block 3 in Fig. 3.2, of which the resolution is relatively high and suitable to generate significantly visible weights map. We select the first layer and last layer in the gated sum of Block 3, noted as higher layer and lower layer respectively.	42
3.10	(Best viewed in color) Boundary Delineation Refinement helps to obtain textural boundary information form low-level features but avoid bringing to much noise information.	43
3.11	(Best viewed in color) Qualitative segmentation result comparisons on Pascal Context. Our segmentation network performers well at salient objects, stuff (e.g. road, grass, sky) and inconspicuous objects. Further, our network has a robust adaptability to multi-scale objects.	52
4.1	Most existing methods model spatial-dependent context with predefined window (e.g., the red rectangle region for pixel A in the second image), which may not provide the specific context information and thus weaken the semantic shape layout. In this work, we propose to aggregate the context information from the semantic-dependent region instead of the spatial-dependent one.	54
4.2	(Best viewed in color) We propose a novel semantic correlation dependent shape-variant context, which boosts the semantic-correlated features (magenta color) while suppresses the others (white color).	57

4.3	(Best viewed in color) A shape mask is inferred by the Paired Convolution and Gaussian mapping function φ , which are designed to learn the semantic correlation between the target pixel and other pixels within the shape mask. Here we show an example of 4 values in a 11×11 mask of the target pixel (dark), in which 4 mask values are generated by the 4 filters of the same color.	58
4.4	Semantic correlation-dependent shape-variant context aggregates surrounding information according to the semantic correlation and hence customizes an effective contextual region. It helps control the information flow within network via deciding what information to be passed or suppressed.	59
4.5	In-context error, e.g., the first row, refers to incorrect labeling within the label set of the image. Out-context error, e.g., the second row, refers to incorrect labeling outside the label set of the image.	62
4.6	Network Architecture. We use ResNet-101 as our base model for fine-tuning and FCN-4s as the backbone segmentation framework. LD is used in decode process for denoising.	64
4.7	Four visual examples of the shape-variant masks $\mathcal{M}_{m,n}^{i,j}$ generated at four different locations of two testing images by the learned network. The mask center (i, j) is indicated by the small square and its value at (m, n) within the image is shown by the gray level.	66
4.8	Qualitative segmentation examples on COCO-Stuff (1st-4th rows) and PASCAL-Context (5th-7th rows).	71
5.1	(Best viewed in color) The boundary-aware feature propagation module builds stronger connections within the same segment and weaker connections between different segments, which helps to enhance the similarity of features belonging to the same segment while keeping discrimination of features belonging to different segments.	74
5.2	An overview of the proposed approach. We use the ResNet-101 (CNN) with the dilated network strategy [3] as backbone and the proposed boundary-aware feature propagation (BFP) module is placed on the top of CNN. The supervisor of loss 2 is the new ground truth of $N+1$ classes with an additional boundary class generated from the original ground truth of N classes.	77
5.3	Each point of the DAGs has three different directions. Thus, the DAGs have to scan the image pixel by pixel and consumes lots of time with many loops. We decompose the four DAGs to six Unidirectional Acyclic Graphs (UAGs). Each of UAGs propagates information toward a single direction, which deals with pixels of each row in parallel and then deals with pixels of each column in parallel. For example, the UAG_S is in the south direction only and the $UAG_{S,E}$ is in the east direction only (based on UAG_S).	79

5.4	(a) Image; (b) Original ground truth; (c) New generated ground truth: add a boundary class generated from the original ground truth; (d) Boundary confidence map: the probability distribution of the boundary layout; (e) Propagation confidence map: the confidence distribution of propagation.	80
5.5	As our UAGs are unidirectional and in parallel, we show the propagation process in 1D here for clarity. i_t represents the feature of pixel at position t , h_t is output (hidden state), p_t is the propagation confidence.	81
5.6	Qualitative examples of inferred boundary map.	86
5.7	Segmentation performance within band (trimap) around boundaries.	87
5.8	Qualitative segmentation examples on Cityscapes.	89
6.1	(Best viewed in color) Both the click and phrase inputs have their advantages and disadvantages. Here we show ambiguities that exist when using only click or phrase interactions. We propose to utilize both interaction types to make selection flexible and robust to handle complex scenarios.	95
6.2	The mistakes (circled in red) may be difficult to describe precisely to a phrase-based selection method but are easy to click on for a click-based selection method.	96
6.3	The proposed approach takes both clicks and a phrase as interaction input. Clicks are employed to localize the target-of-interest and gradually refine the details, and the phrase is used to infer semantic attributes for quickly discriminating ambiguous candidate objects. It is <i>not</i> necessary to enter both types of interactions at the same time. We use the ResNet-101 based DeepLabv3+ as a segmentation architecture for fine tuning. The <i>Context Encoder</i> is ASPP+Pooling and the <i>Decoder</i> uses skip connections to extract low-level features, as does DeepLabv3+. Graph Cut is employed as a post-processing method during testing.	98
6.4	Attribute Attention: It emphasizes feature channels that have larger response in semantic attribute learning, which is based on the phrase interaction, click interaction and visual patterns.	101
6.5	Examples of our phrase annotations for Grabcut and Berkeley.	103
6.6	We display some segmentation score maps (soft value from 0 to 1) of referential object segmentation. Here the network only takes the phrase as interaction input.	105
6.7	Segmentation accuracy with different number of interaction times.	106
6.8	Different kinds of interaction processes. The proposed approach is more flexible in terms of interaction ways and can better handle complex scenarios via utilizing advantages of these two interactions.	110

List of Tables

3.1	Ablation Experiments of CCL. LA is local aggregation generated by removing the context part of CCL. LA ^d doubles the hidden dimensionality of LA from 512 to 1024, thus its parameter quantity is the same as CCL.	40
3.2	Comparison with State-of-the-art Context Models. Segmentation networks are adapted to encode-decode architecture with rich skip layers, the stride rates (dilation factors) of the four branches in ASP-P are revised to {1, 3, 4, 6} respectively. For fair comparisons, gated sum is not adopted, and they only differentiate each other in terms of context aggregation (CA).	40
3.3	Ablation study of the number of CCL blocks.	41
3.4	Ablation study of BDR in boundary regions	44
3.5	Effect of Using the Very Lower-level Features. The very low-level features (block2 in ResNet) brings detailed textural boundary information but also noise that are not suitable for the high-level tasks. Note the context aggregation model of FCN is replaced by CCL.	44
3.6	Ablation Experiments of Proposed Methods. MS means multi-scale testing. The proposed CCL, gated sum and BDR improve the performance visibly.	45
3.7	Runtime Comparison. Runtime comparison on a single NVIDIA Tesla K80. It shows that the proposed network marginally increase the running time from the baseline.	45
3.8	Performance Comparison on Pascal Context Dataset. The proposed CGBNet outperforms all existing methods by a large margin across all evaluation metrics. Methods trained with extra data are marked with †.	46
3.9	Performance Comparison on SUN-RGBD Dataset. The proposed CGBNet outperforms existing methods consistently across all the three evaluation metrics.	47
3.10	Performance Comparison on Sift Flow Dataset. The proposed CGBNet segmentation networks outperforms state-of-the-arts across all evaluation metrics.	48
3.11	Performance Comparison on COCO Stuff Dataset. The proposed CGBNet outperforms the state-of-the-arts by a large margin across all evaluation metrics. Methods trained with extra data are marked with †.	49

3.12	Parsing performance of different networks on ADE20K dataset. . .	49
3.13	Performance Comparison on Cityscapes Dataset. The proposed CGBNet segmentation networks outperforms state-of-the-arts. . . .	50
4.1	Ablation study of the proposed approach in terms of IoU.	65
4.2	Ablation study of the proposed shape-variant context (SVC) approach by comparing it with the shape-fixed context (SFC) on different kernel size. It also shows that the performance gain is not simply brought by the increased number of parameters.	65
4.3	COCO-Stuff testing accuracies.	67
4.4	SIFT-Flow testing accuracies.	68
4.5	CamVid testing accuracies.	68
4.6	PASCAL-Person-Part testing accuracies.	69
4.7	PASCAL-Context testing accuracies.	69
4.8	Cityscapes testing accuracies.	70
5.1	Inference speed comparison of FCN (baseline), DAGs, and UAGs on dilated ResNet-101 with different resolution inputs.	84
5.2	Ablation Study of Boundary-aware Feature Propagation (BFP) Network on PASCAL-Context. Baseline is dilated FCN and MS means multi-scale testing.	87
5.3	Testing accuracies on PASCAL-Context.	88
5.4	Testing accuracies on CamVid.	88
5.5	Category-wise performance comparison on the Cityscapes test set. Note that the DenseAspp [4] uses stronger backbone DenseNet-161 [5] than Resnet-101 [2] we adopt as backbone.	91
6.1	Ablation study of the proposed approach on testA of RefeCOCO. .	104
6.2	Comparison with previous state-of-the-art interactive object segmentation methods. We demonstrate the number of interactions that each method needed to reach a certain segmentation performance.	107
6.3	Comparison with state-of-the-art referring segmentation methods. .	109

List of Abbreviation

ACA	Average Class-wise Accuracy
ASPP	Atrous Spatial pyramid pooling
BFP	Boundary-aware Feature Propagation
CCL	Context Contrasted Local
CGBNet	Context contrasted, Gated sum and Boundary delineation Network
CNN	Convolutional Neural Network
DAG	Directed Acyclic Graph
DCNN	Deep Convolutional Neural Network
FCN	Fully Convolutional Network
GPA	Global Pixel Accuracy
GPU	Graphical Processing Unit
IOU	mean Intersection Over Union
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
SGD	Stochastic Gradient Descent
SVCNet	Shape-Variant Context Network
UAG	Unidirectional Acyclic Graph
UCG	Undirected Cyclic Graph

Chapter 1

Introduction

Scene Parsing (also known as semantic segmentation, scene segmentation, scene labeling) aims to classify every pixel of a given image to one of the predefined semantic categories (e.g., person, car, etc.), including not only countable objects (e.g. person, car, cat) but also uncountable stuff (e.g. road, grass, sky), as shown in Figure 1.1. The key of scene segmentation refers to parsing and segmenting a scene image into a range of semantic coherent regions. It is a dense prediction task whose output has the same resolution with input, thus it provides much richer clues for scene understanding than other shallower scene understanding tasks, such as scene image recognition that assigns one of categories to the whole image. Based on the plentiful clues provided by scene parsing, further and deeper beneficial interpretations can be deduced, like the implicit bridges among different objects/stuff. Scene parsing implicitly involves object recognition, object localization and boundary delineation, which requires multi-scale and multi-level visual recognition. A robust segmentation approach needs to perform well at all of these implied tasks.

Recently, the great success of deep neural networks has largely improved the performance of computer vision tasks [6], such as image classification [1, 2, 5, 7], and object detection [8, 9]. However, there are still some limitations when directly applying deep neural networks to the dense prediction tasks like scene parsing [3, 10–13]. Originally designed for image classification, the Deep Convolutional Neural Networks (DCNN) learn very abstract feature representation of the whole image [14] and hence extract information of dominant/salient objects of image. This is highly desirable for image classification and object detection. For scene

FIGURE 1.1: (Best viewed in color) Scene parsing refers to classifying every pixel of a given image to one of the predefined semantic categories (e.g., person, car, etc.), including not only countable objects (e.g. person, car, cat) but also uncountable stuff (e.g. road, grass, sky).

parsing that is a dense prediction task, however, spatial information is essential and pixel-level discriminative features are desired. Although most state-of-the-art scene parsing frameworks are based on the image classification networks pre-trained on Imagenet [15], it remains an open question of how to better adopt DCNN on scene parsing.

As one of the fundamental tasks in computer vision, scene parsing has been an essential component of computer vision and is in intense demand for many practical applications, such as automation devices, virtual reality, augmented reality, self-driving vehicles and etc. Furthermore, because of the rich clues offered by scene parsing, it can also provide useful information for other tasks, such as multi-label object recognition [16], pose estimation [17] and image inpainting [18]. Therefore, it is worth exploring scene parsing from both the industrial and academic prospective. In this thesis, we focus on addressing scene parsing with deep neural networks.

1.1 Challenges and Motivations

Scene parsing implicitly involves object recognition, object localization and boundary delineation, which requires multi-scale (i.e., features with receptive fields at different scales) and multi-level (e.g., low-level edges clues and high-level semantic

clues) visual recognition. A robust segmentation approach needs to perform well at all of these implied tasks. In this thesis, we focus on the following inherent challenges in scene parsing,

- 1) Scene parsing is a pixel-level recognition task. For typical object recognition, it is common and beneficial to get global image-level clues to improve performance, for example, global pooling [2], fully connected layer [7] and spatial pyramid pooling [19, 20]. Scene parsing involves object recognition, thus global clues are also advantageous to alleviate local ambiguity [3, 13, 21]. However, different from image-level object recognition, scene parsing requires dense prediction with pixel-level details. To obtain such dense prediction, it is desired to retain the spatial information and discriminative local clues, which are important for pixel-level details. But spatial details and local clues are usually discarded/smoothed in operations that generate global clues. Therefore, it is challenging to perform well on both object recognition and detailed dense prediction.

To address this issue, we explore to build customized contextual model that assemble global clues and local clues together. This contextual modeling is aimed at enhancing the high-level feature representation while keeping the local discriminations.

- 2) There are huge scales of objects in scene parsing, not only in different images (e.g. the different scales of persons in the first and second images in Fig. 1.1) but also in the same image (e.g. the multi-scale cows in the third image in Fig. 1.1). It is irrational to classify all individual pixels based on a single scale of features. For object recognition aiming at image-level labeling, extracting abstract global features for the whole image helps address the issue of huge scale variation in different images. However, scene parsing needs to classify every pixels and the scale variation in the same image should also be considered. Therefore, adaptive scales of features for parsing different pixels are desired, for example, smaller scale of receptive field for pixels belonging to smaller objects while larger scale of receptive field for pixels belonging to larger objects.

This challenge motivates us to explore multi-scale feature aggregation. Different from vanilla multi-scale aggregation that directly fuse multi-scale features, in this thesis, we propose adaptive scale selection scheme to let the framework learn how to decide the desired scales for every pixels.

- 3) Detailed spatial information is essential for scene parsing, especially for the implicit low-level tasks, such as boundary delineation. However, due to the pooling operation or convolution strides of DCNN, lots of spatial information is lost during the encode process. It is necessary and challenging to restore high resolution with spatial details for scene parsing.

To this end, we explore to infer and refine the boundary layout of objects/stuff. Boundary refinement helps to improve the spatial details and also benefit the high-level recognition task. It aims to restore high resolution to achieve dense prediction with fine details.

- 4) Most of scene parsing models cannot correctly segment the unknown/unseen objects, which have never appeared in the training data. For example, Cityscapes [22] contains 19 outdoor labels but no indoor labels, thus scene parsing models trained on Cityscapes cannot deal with indoor scene images. And when comes to real applications, different sets of semantic categories are required for different scenarios, for example, “pillow” and “curtain” for indoor scene analysis, “traffic light” and “traffic sign” for road scene parsing. It is impractical to annotate a benchmark that contains all the possible object categories in the world. Therefore, the ability to segment any unknown/unseen objects is required. Besides, it is well known that large-scale and well-annotated benchmarks are desired for deep-learning-based scene parsing approaches. However, it is time-consuming and resource-expensive to annotate datasets for scene parsing that requires pixel-wise annotations. Therefore, auxiliary annotation tools are demanded to speed up the annotation process.

This motivates us to build interactive image segmentation model, which aims to segment the image according to user interactive inputs. First, with interactive segmentation, user can select any target-of-interest pieces (including unknown or unseen objects) by providing interactions. Second, interactive segmentation can be used as a dataset annotation tool, which greatly speed up the pixel-level annotation process.

1.2 Proposed Segmentation Models

In this thesis, we present four scene parsing models based on the powerful feature extractor DCNN, like ResNet-101 [2], pre-trained on ImageNet [15]. The first three models are fully automated scene parsing networks, which address several challenges that we discussed previously. The fourth model is interactive image segmentation model, which focuses on the fourth challenge.

The first segmentation model employs Context encoding, Gated multi-scale aggregation and Boundary delineation (CGBNet) to enhance the segmentation performance from different levels. In CGBNet, we first propose a context encoding module that generates context-contrasted local feature to make use of the informative context and the discriminative local information. This context encoding module greatly improves the segmentation performance, especially for inconspicuous objects. Furthermore, we propose a scale-selection scheme to selectively fuse the segmentation results from different-scales of features at every spatial position. It adaptively selects appropriate score maps from rich scales of features. To improve the segmentation performance results at boundary, we further propose a boundary delineation module that encourages the location-specific very-low-level features near the boundaries to take part in the final prediction and suppresses them far from the boundaries.

Our second scene parsing model is Shape-Variant Context Network (SVCNet) that mainly target to build customized contextual model. Due to the diverse shapes of objects and their complex layout in various scene images, the spatial scales and shapes of contexts for different objects have very large variation. It is thus ineffective or inefficient to aggregate various context information from a predefined fixed region. In this work, we propose to generate a scale- and shape-variant semantic mask for each pixel to confine its contextual region. To this end, we first propose a novel paired convolution to infer the semantic correlation of the pair and based on that to generate a shape mask. Using the inferred spatial scope of the contextual region, we propose a shape-variant convolution, of which the receptive field is controlled by the shape mask that varies with the appearance of input. In this way, the proposed network aggregates the context information of a pixel from its semantic-correlated region instead of a predefined fixed region.

Next, we propose a boundary-aware feature propagation (BFP) model. To increase the feature similarity of the same object while keeping the feature discrimination of different objects, we explore to propagate information throughout the image under the control of objects’ boundaries. To this end, we first propose to learn the boundary as an additional semantic class to enable the network to be aware of the boundary layout. Then, we propose unidirectional acyclic graphs (UAGs) to model the function of undirected cyclic graphs (UCGs), which structurize the image via building graphic pixel-by-pixel connections, in an efficient and effective way. Furthermore, we propose a boundary-aware feature propagation module to harvest and propagate the local features within their regions isolated by the learned boundaries in the UAG-structured image. The proposed BFP is capable of splitting the feature propagation into a set of semantic groups via building strong connections among the same segment region but weak connections between different segment regions.

In the fourth segmentation model, we build an interactive image segmentation model, PhraseClick, which is semi-automated and aims to accurately segment the image into foreground and background given a minimal amount of user interactive inputs. Existing interactive object segmentation methods mainly take spatial interactions such as bounding boxes [23–26] or clicks [27–32] as input. However, these interactions do not contain information about conspicuous attributes of the target-of-interest and thus cannot quickly specify what is the target area to select. Therefore, excessive user interactions are often required to reach desirable results. We propose to employ language phrases as another interaction input to infer the attributes of the target object. In this way, we can 1) leverage spatial clicks to localize the target object and 2) utilize semantic phrases to qualify the attributes of the target object. Specifically, the phrase input focuses on “what” the target object is and the spatial clicks are in charge of “where” the target object is. Moreover, the proposed approach is flexible in terms of interaction modes and can efficiently handle complex scenarios by leveraging the strengths of each type of input.

Overall, these four proposed models are to address the scene parsing challenges in Section 1.1. The proposed CGBNet, SVCNet and BFP achieve state-of-the-art on eight public scene parsing benchmarks. Our multi-modal phrase+clicks approach achieves new state-of-the-art performance on interactive image segmentation.

1.3 Contributions

In a summary, in this thesis, we makes the following contributions:

- 1) We propose two context modules to customize high-level information for every pixel. Firstly, we propose a novel context-contrasted local feature that is tailored for scene segmentation and propose a context-contrasted local (C-CL) model to aggregate multi-scale context-contrasted local features. The proposed CCL brings 5.8% IoU performance gain on Pascal-Context [33]. Secondly, we propose a novel paired convolution to infer the semantic correlation of two pixels and based on that to generate a semantic-correlated region for each pixel. Based on the semantic-correlated region, we propose a shape-variant context aggregated within the semantic-correlated region to model the diverse shapes and scales of contexts, which greatly enhances the modeling ability of network. The proposed shape-variant context(SVC) brings significant 9.7% and 7.0% IoU performance gain on Pascal-Context [33] and COCO-Stuff [34] respectively, which demonstrates the significance of SVC.
- 2) We propose a gated sum scheme to selectively aggregate appropriate-scale features at each spatial location, which is an efficient and effective way to address the issue of the existence of multi-scale objects. The gated sum scheme significantly boosts the segmentation performance by 3.3% IoU on Pascal-Context [33] And the context module SVC is scale- and shape-variant, it helps to customize appropriate-scale context information for different pixels.
- 3) We propose a boundary delineation model to refine the prediction of boundary, which encourages the location-informative very-low-level features near the boundaries and suppressed them far from the boundaries.
- 4) We show that the boundary can be learned as one of semantic categories, which requires little change on network but obtains essential boundary information. We propose some unidirectional acyclic graphs (UAGs) to propagate information among high-resolution images with a high speed. We propose a boundary-aware feature propagation module to improve the similarity of local features belonging to the same segment region while keeping the discriminative power of features belonging to different segments.

- 5) The proposed three scene parsing models (CGBNet, SCVNet, BFP) consistently achieve state-of-the-art segmentation performance on eight challenging scene segmentation benchmarks including Pascal-Context [33], Cityscapes [22], COCO-Stuff [35] CamVid [36], SUN-RGBD [37], Sift-Flow [38], PASCAL-Person-Part [39], and ADE20K dataset [40].
- 6) We propose a novel interactive segmentation model (PhraseClick) that employ both phrase expressions and clicks. To the best of our knowledge, this is the first work to leverage both clicks and phrases for interactive object segmentation. The proposed approach is more flexible in terms of interaction and can better handle complex scenarios via utilizing advantages of these two interactions. We propose an attribute guided feature attention module to extract discriminative semantic attribute information from phrase expressions and integrate the attribute and location information into the network. Extensive experimental results have demonstrated that phrase expressions are indeed effective at boosting the performance of interactive segmentation, especially in some complex scenarios and with few clicks.

1.4 Thesis Outline

The rest of this thesis is organized as follows:

In Chapter 2, we review and discuss previous works that are related to this thesis. In Chapter 3, we present the motivation, model details of CGBNet, and show its effectiveness on six public benchmarks. Next, Chapter 4 demonstrates the formulation of SVCNet as well as its detailed ablation studies on challenging benchmarks. As follows, in Chapter 5, we presents the details of BPF and demonstrate not only its overall segmentation performance but also its performance within band around boundaries. Then, the motivation, formulation and detailed ablation studies of PhraseClick are elaborated in Chapter 6. Finally, in Chpater 7, we conclude this thesis and discusses some interesting future research works.

Chapter 2

Literature Review

In this chapter, we review and discuss previous works that are related to this thesis from different perspectives. Firstly, we briefly review the Convolutional Neural Networks (CNNs) and Fully Convolutional Network (FCN) that used in the models in this thesis. Secondly, we review and compare the related scene parsing works. Then we discuss the boundary detection and boundary delineation, which are implicit tasks of scene parsing. Next, we review the previous literatures of interactive segmentation, including not only spatial interactions (e.g. clicks and bounding boxes) but language interactions. Finally, we review and demonstrate the representative benchmark datasets for scene segmentation.

2.1 Deep Neural Networks

Recently, Deep Neural Networks has achieved great success in computer vision, here we briefly review the Convolutional Neural Networks (CNNs) and Fully Convolutional Network (FCN) that used in the models in this thesis.

Convolutional Neural Networks (CNNs) is widely employed in deep-learning-based methods and has achieved great success in many of computer vision tasks, like image classification [1, 2, 5, 7], object detection [8, 9, 41–43], and semantic segmentation [3, 10, 12, 13].

FIGURE 2.1: Architecture of VGG-16.

AlexNet [44] is the pioneering framework that first shows the advantages of CNNs in computer vision. AlexNet is very simple and contains only five stacked convolutional layers. Then Simonyan and Zisserman propose VGG-16 [7], which consists of 16 convolutional layers, using very small 3×3 convolutional filters in all of the convolutional layers, as shown in Figure 2.1. Szegedy et al. [1] propose GoogleNet that stacks inception module. Inception block employs parallel convolutional layers and pooling layer to capture different scales of receptive fields and 1×1 convolutional layer to reduce feature dimensionality, as shown in Figure 2.2. He et al. propose a remarkable architecture, ResNet [2], which is much deeper (101 layers or 152 layers) than previous CNNs. In ResNet, residual block is proposed to alleviate the problem of gradient vanish. The residual block contains identity mapping by shortcuts, as shown in Figure 2.2, and this helps to train deeper convolutional neural networks. In this thesis, we mainly employ ResNet-101 [2] as our feature extractor.

FIGURE 2.2: Inception block from GoogleNet [1] architecture and Residual block from the ResNet architecture [2].

FIGURE 2.3: Architecture of FCN based on VGG-16.

Based on the powerful convolutional neural networks, Long et al. [45] introduce the fully convolutional networks (FCN) for semantic segmentation. In FCN, the fully connected layers in standard CNNs (e.g. VGG-16 in Figure 2.1) are transformed to convolutional layers, as shown in Figure 2.3¹. FCN uses deconvolution layer (transposed convolution) to upsample the coarse features to high resolution and thus can generate dense prediction. FCN utilizes the robust features from CNNs pre-trained on image classification and thus outperforms segmentation models based on hand-crafted features. With FCN, CNNs-based segmentation networks can be trained end-to-end under the supervision of dense ground truth having the same resolution with input image. FCN achieves significant performance on scene segmentation and has been widely used as baseline in many computer vision tasks requiring dense prediction.

Generally, both CNNs and FCN achieves great success and demonstrate their powerful learning ability to extract robust features for computer vision tasks. Herein, in this thesis, we employ the FCN architecture with CNNs as backbone network to address scene parsing.

2.2 Scene Parsing

Scene parsing (or scene segmentation, semantic segmentation) is one of the fundamental problems in computer vision and has drawn lots of attentions. Recently, thanks to the great success of Convolutional Neural Networks (CNNs) in computer vision [1, 44, 46, 47], lots of CNNs based segmentation works have been proposed and have achieved great progress. For example, Long et al. [45] introduce the fully

¹For clarity, we do not show the short connections in FCN.

convolutional networks (FCN) in which the fully connected layers in standard CNNs are transformed to convolutional layers. Noh et al. [48] propose deconvolution networks to gradually upsample the coarse features to high resolution. Chen et al. [3] propose to remove some pooling layers (or convolution stride) in CNNs and adopt dilated convolution to retain more spatial information. And some works focus on lightweight network architectures [49, 50] and real-time segmentation [51–53].

2.2.1 Contextual Modeling

Context aggregation is a hot direction in scene parsing and plays an important role. [54] shows that global spatial information helps enforce features consistency and lots of works have contributed to contextual modeling, e.g. [3, 13, 55–57].

One direction is to apply new layers on top of a pre-trained DCNN to enhance high-level contextual aggregation. For example, Chen et al. [3] introduce an atrous spatial pyramid pooling (ASPP) to capture useful context information at multiple scales. DilatedNet [55] appends several dilated convolution layers after the score maps to perform multi-scale context aggregation. Visin et al. [58], Ding et al. [57] and Byeon et al. [59] adopt recurrent neural networks to capture long-range context. Zhao et al. [13] employ multiple parallel pooling layers to exploit global information from different regions. Liu et al. [60] propose to model the mean field algorithm with local convolution layers and incorporate it in deep parsing network (DPN). Yu et al. [55] attach multiple dilated convolution layers after class likelihood maps to exercise multi-scale context aggregation. Zoom-out [61] proposes a feed-forward architecture to extract hierarchical zoom-out features. Sharma et al. [62] propose a recursive context propagation network to disseminate global information to different local regions. Zhang et al. [63] encode semantic context to network and stress class-dependent feature maps. Fu et al. [64] integrate local and global dependencies with both spatial and channel attention.

Another way is to use Conditional Random Fields (CRF) [65] to model the context of score maps [3, 60, 66–68]. For example, Chen et al. [3] adopt CRF to post-process the unary predictions and generate smoother prediction maps. Zheng et al. [67] propose CRF-RNN to jointly train CRF with their segmentation networks.

Different from previous methods, in this thesis, we explore to build customized contextual model that not only assembles global clues to enlarge receptive fields but also preserves local clues to keep discrimination. To this end, in our CGB-Net in Chapter 3, we propose context contrasted local (CCL) feature that consists of two parallel parts: coarse context convolution and delicate local convolution. ASPP [3] is the most related works with the proposed CCL. ASPP[3] aggregates multi-scale contexts via combining score maps generated by different context aggregation branches, each of which uses dilated **Conv** kernels with different stride rates to incorporate different scale contexts. Compared with this type of context model, CCL first contextualizes contrasted features at every block to obtain context aware local features, which combines two different scales in the feature level and take advantage of both context and local information, then further aggregate multi-scale context contrasted local features in score level.

And further in the proposed SVCNet in Chapter 4, we propose a shape-adaptive convolutional layer to learn diverse shaped contexts whose shapes are determined by the object shape, scale and its surrounding support of the input image. The proposed approach is aimed at not only retaining the location identity and layout information but also building the effective semantic correlation shown in the training images. Different from the previous context methods that tend to model statistical average representation within a predefined rectangle region, e.g., [3, 4, 13, 55, 56, 69], the proposed shape-variant context utilizes the semantic correlation and intentionally picks up the relevant information according to the semantic shape mask inferred by the feature appearance of testing image. Thus, it could not only retain the shape and location identity but also effectively build the beneficial connection among correlated features for classification. Comparing with the deformable convolution [70], the objective of the proposed approach is similar to it in terms of collecting the relevant information in the convolution. However, different from [70] that tries to achieve this via deforming the sampling locations, the proposed SVC finds out the semantic correlation to enhance or attenuate the corresponding information, explicitly leading to the shape- and scale-variant modeling. The criteria or the methods to find the relevant information in the two approaches are also different. Furthermore, our method models diverse shapes of semantic-dependent context in a single layer without stacking layers in [70], and avoids the “atrous” in deformable convolution that may lose some detailed information.

2.2.2 Multi-scale Aggregation

Due to the huge scale variation of objects, it is difficult to achieve robust segmentation only based on the single scale of features. Multi-scale aggregation is a crucial way to deliver robust parsing maps. There are several methods to achieve multi-scale aggregation. Farabet et al. [71] and Lin et al. [68] adopt multi-resolution input (image pyramid) approach and fuse the corresponding features from multiple resolution. Liu et al. [72] generate multi-scale patches and aggregated the results. Pinheiro et al. [73] input multi-size images at different layers of a recurrent convolutional neural network. However, the above approaches are computational expensive and consume large GPU memory. Thus, their aggregation ability for multi-scale features is limited due to the limited scales used in practice to avoid heavy computational load of this scheme.

The seminal work FCN [10] introduce the skip layers to locally classify multi-scale feature maps and aggregate their predictions via sum fusion. This is an effective as well as efficient method to integrate different scale features and our work follows this way. Nonetheless, in previous works [10, 61, 66, 69, 74, 75], the score maps of skip layers are fused via a simple sum and hence the different importance of different scales are ignored. To address this issue, we propose a network that facilitates a gated sum to selectively aggregate different scale features. With gated sum fusion, the network can exploit more skip layers from richer scale features in DCNN and customize a suitable integration of different scale features. To the best of our knowledge, our gated sum is the first work to selectively aggregate appropriate scale features in a single network.

2.2.3 Boundary Delineation

Boundary delineation is an implicit task in scene parsing. Due to the diverse shape of objects and complex layout, boundaries are capricious and difficult to be predicted accurately. Conditional Random Fields (CRF) [65] is often adopted as a post-processing method to get a better boundary prediction, such as Deeplab[3]. Apart from CRF, Barron et al. [76] propose bilateral solver to prompt edge-aware smoothness, and Jampani et al. [77] integrated the bilateral solver in DCNN to jointly train them from data. Peng et al. [69] integrate a residual convolutional architecture in their network to increase the accuracy of boundary region.

In this chapter, we propose a boundary delineation refinement (BDR) model, which employs a location-selection scheme for filtration of very-low-level features to refine the spatial boundary. BDR encourages pixels near the boundaries to be assigned more very-low-level information and suppresses it for pixels far from the boundaries. It brings very little computation but enhances the boundary delineation. Furthermore, BDR helps enhance the coalescence of multi-level tasks in a single network.

2.2.4 Graphic models

Graphic models have a long history in scene parsing. Early works construct the graphic model with hand-crafted features [78–81]. Markov Random Fields (MRF) [78, 82, 83] and Conditional Random Fields (CRF) [3, 60, 65, 84] build the dependencies according to the similarities of neighboring pixels. Liang et al. [85] propose to construct graph topology based on superpixel nodes and incorporate long-range context with Graph LSTM. Shuai et al. [75] adopt undirected cyclic graphs (UCGs) to formulate an image and decompose the UCGs with directed acyclic graphs (DAGs). Byeon et al. [59] propose to divide an image into non-overlapping windows and employ a 2d LSTM to construct local and global dependencies. However, most of graph-based method are time-consuming and computationally expensive as they require candidate pre-segments, superpixels, or lots of loops.

In this thesis, we propose unidirectional acyclic graphs (UAGs) in Chapter 5, based on which the local features are quickly propagated in parallel. And to construct strong dependencies within the same segment and weak dependencies among different segments, we exploit the learned boundary information to guide the feature propagation within the UAG-structured image.

2.2.5 Label variety

Label variety is another challenging problem in scene parsing. PSPNet [13] observes the confusion categories and demonstrates that the PSPNet can better address the confusion labels than FCN[45]. Geng et al. [86] propose to infer the discriminative confusing groups from a prior confusion matrix. DFN [87] introduces a smooth

and border network to tackle confusing classes. Davis et al. [88] propose to refine the parsing results using Bayesian strategy with confusion probabilities and label priors. Huang et al. [89] propose a LabelReplacement network to correct the error predictions.

Different from these methods, in Chapter 4, we propose a learnable Labeling Denoising (LD) model to solve the problems of confusion labels by utilizing the robust higher level features to attenuate the noise in lower level features.

2.3 Boundary Detection

Boundary detection is a fundamental component for many computer vision tasks and has a long history [90–93]. For example, Lim et al. [94] propose sketch tokens (ST) and Dollár [95] et al. propose structured edges (SE) based on fast random forests to deal with boundary detection as local classification problem. Recently the success of CNNs have great improve the performance of boundary detection [96–100]. Xie et al. [100] employ features from the middle layers of CNNs for boundary detection. Shen et al. [101] propose multi-stage fully convolutional networks for boundary detection of electron microscopy images. These methods target at optimizing the accuracy of boundary detection instead of generating semantic boundary information for high-level tasks.

Boundary information could be used for improving segmentation performance. For example, Bertasius et al. [97], Hayder et al. [102], Chen et al. [103] and Kokkinos [104] employ the binary edges to improve the segmentation performance. However, they all employ an additional branch of network for edge detection, which requires more resources and deal with segmentation and boundary detection as two detached tasks. Different from [97, 102, 103], our goal is not to detect the clearly binary boundaries, but to infer a boundary confidence map that represents the probability distributions of high-level boundary layout. We only focus on the boundaries of semantic objects that are predefined in segmentation datasets. We have observed that boundaries have the property of dramatic changes in RGB and feature information. And the boundary label is easy to be generated from the existing ground truth. Consequently, we assume that the boundary could be viewed as an additional semantic category and simultaneously learned with other

existing object categories. As in Chapter 5, we obtain a new ground truth (Loss 2, $N+1$ classes) from the original ground truth (Loss 1, N classes) and utilize the new ground truth for supervising the network to learn and infer the boundary layout. And different from previous boundary detection works that aim to boundary delineation only [96–100] or deal with segmentation and boundary detection as two detached tasks [97, 102, 103], our proposed semantic boundary detection is embedded with semantic object parsing. These two tasks are combined into one, and they benefit each other. By training them together, the scene parsing classes help suppress the edges within objects that are not semantic boundary of objects, e.g., edges of eyes. Scene segmentation helps boundary detection to filter out noise and delineate well-directed boundaries that are contours of semantic objects predefined in dataset. While boundary detection makes the scene parsing be aware of the important boundary layout information and provide more spatial details.

2.4 Interactive Segmentation

Many interactive segmentation methods have been proposed in the past decades using many different interactive approaches such as bounding boxes [23–26], contours [105, 106], strokes [31, 32, 107, 108] and clicks [27–30, 109]. Some language-based segmentation methods [110–112] have been proposed to segment objects, but these methods only provide an initial result and cannot further refine the result to correct mistakes.

2.4.1 Spatial Constraints based Interactive Segmentation

The click interaction is commonly used in interactive object segmentation and has shown significant success [27, 28, 30, 113]. Click based approaches can well locate the target-of-interest with sufficient and accurate interactive clicks. However, early methods rely on low-level features, such as color similarity or boundary properties [105, 106], to segment foreground and background. For example, [23, 32, 107] adopt graphical models, [114] employs random walker and [115–117] are based on geodesic approaches. However, low-level features are not robust and excessive user interactions are required for these methods to achieve desirable segmentation

results, especially in case of complex illumination conditions and undistinguished foreground/background appearances.

Recently, thanks to the great success of convolution neural networks (CNN), CNN-based interactive segmentation methods have achieved exciting progress. For example, Xu et al. [27] concatenate Euclidean distance maps to user foreground and background clicks with the image as input to an FCN [45]. Acuan et al. [118] and Castrejon et al. [119] propose Polygon-RNN which uses recurrent neural network to address formulate interactive object selection as a sequential polygon prediction task. Liew et al. [28] leverage the local information and global prior to refine local prediction. Mahadevan et al. [120] employ a two-stream network to deal with image and user interaction separately to make the user interaction impact the result more directly. Instead of requiring clicks in the interior of the target regions like the above methods, Le et al. [30] introduce boundary clicks to perform object selection. Each of these methods provides only spatial constraints regarding the location of the object and do not directly provide any higher level attribute information.

Bounding boxes or extreme points have also been used as input to interactive selection methods. An early example is GrabCut [23, 26] which optimizes an objective function given a bounding box constraint to select an object. Neural networks have also been trained to take bounding boxes as input [24]. As a variant to using bounding boxes, Papadopoulos et al. [121] propose extreme points (the top, bottom, left- and right-most points) for object annotation and Maninis et al. [29] use extreme points to generate Gaussian heatmap and crop the image to achieve instance segmentation. Agustsson et al. [113] further segment all regions jointly with extreme points and scribbles. While these methods provide a different kind of spatial constraint than clicks, they still do not provide any attribute information.

As these methods receive only spatial constraints, they cannot provide high level attribute information to the method and thus may require additional user input to overcome this drawback in challenging cases. For example, since existing methods are based on the assumption that the target-of-interest is a single object, they demand excessive user interaction inputs to reach desirable performance when the assumption is not true. For one click, there are often several potential interested areas with different scale. Therefore, the performance of existing methods drops a lot with only one or few clicks under complex scenarios. While recent work has looked at addressing this [122], these methods are leveraging spatial constraints and

still may not target the user desired region. Additionally, on some mobile devices, such as mobile phones and tablets, user cannot provide precise clicks as with a mouse but only rough clicks by finger touching. It still remains a big challenge to accurately segment the target-of-interest with one or few rough clicks.

Semantic segmentation and instance segmentation are closely related with interactive object segmentation. Driven by the significant success of CNN, many deep-learning-based works have been proposed for semantic segmentation [3, 13, 63, 64, 67, 123] and instance segmentation [124–126]. For example, Long et al. [45] propose the fully convolutional networks (FCN) to train the segmentation network end-to-end. Chen et al. [3] propose to discard some downsample operations (e.g., pooling) in CNN and employ dilated convolution to enlarge receptive field. He et al. [124] propose to add instance-level segmentation mask branch on the top of Faster R-CNN [8]. Chen et al. [127] utilize position-sensitive scores to refine instance segmentation results. However, it is not reasonable to directly transform semantic/instance segmentation to interactive segmentation [27]. Interactive object segmentation methods respond to user’s inputs instead of predefined labels, thus the ability to segment any unseen objects is required. In this work, we modify the semantic segmentation framework [128] to have it accept click and phrase interactions, and re-train the modified framework on interactive object segmentation datasets.

2.4.2 Referring Expression Comprehension

Referring expression comprehension methods [110, 111, 129–132] detect a specific object in an image given a referring expression. Some comprehension methods can be used to segment the referential object [110–112]. However, these methods only compute an initial segmentation and cannot further correct the segmentation mask, which is required for interactive segmentation if any mistakes are made. For example, MAttNet [111] depends on the instance segmentation results of Mask R-CNN [124]. MAttNet first scores and ranks the similarity between phrase embedding and instance objects generated by Mask R-CNN, and then outputs the mask of best matched object. These methods cannot further improve the segmentation mask even with new additional phrase input. Thus, they cannot meet the requirement of practical application for interactive segmentation. In contrast,

our approach enables users to add interactive information until the segmentation results meets the users' requirements.

2.4.3 Bringing Clicks and Phrases Together

Spatial interactions such as clicks have been proven to be effective at selecting objects in images as they give spatial constraints that a network can use to identify potential objects [27, 28, 30, 113]. Click information is usually introduced at the beginning of a network after which the network computes low-level features such as corners and edges. Meanwhile, higher-level features in CNN represent abstract semantic concepts, such as attributes and category-level evidence. The connection between image features and phrase expression has been proven in referring expression comprehension [111, 129–132]. In this work, motivated by the observation that a phrase expression can infer the attributes of an object [111, 133] and such attribute information is well preserved at the feature level of CNN [134], we explore using phrase expressions as another interaction input to express the semantic attributes of a target-of-interest and quickly narrow the range of candidates, which can assist the click interaction process and decrease the user interaction times. For regions that are difficult to articulate, such as small mistakes that need correction or objects that do not have easily identifiable attributes that will separate them from surrounding objects, clicks provide a strong spatial constraint to supplement the attribute information.

To the best of our knowledge, the proposed PhraseClick in Chapter 6 is the first work to bring clicks and phrases together for interactive segmentation.

2.5 Scene Segmentation Benchmark Datasets

In this section, we demonstrate the popular benchmark datasets for scene segmentation. Some examples are shown in Figure 2.4. In the Figure, we show images and their corresponding ground truth. As clearly shown, images in scene segmentation datasets are richly and densely annotated in pixel-level.

Pascal Context [33] contains 10103 images from Pascal VOC 2010 [135], and these images are re-annotated as pixel-wise segmentation maps. There are 4998

FIGURE 2.4: Eight representative benchmark datasets for scene segmentation. In the demonstration, we show input images and their ground truth.

images for training and 5105 images for testing in Pascal Context. We use the most common 59 categories in this dataset for evaluation.

SUN-RGBD [37] provides pixel-wise labeling for 37 categories. It has 10335 indoor images which are from SUN3D [136], NYUDv2 [137], Berkeley B3DO [138] and the newly captured images. The training set has 5285 images and the test set contains 5050 images. We only use the RGB modality as input for training and testing.

SIFT-Flow [38] contains 2688 images labeled by 33 classes. We use the standard split protocol thus 2488 images are used for training and 200 images are used for testing.

COCO-Stuff [35] contains 10000 images from Microsoft COCO dataset [139], out of which 9000 images are for training and 1000 images for testing. The unlabeled stuff pixels in original images of Microsoft COCO are further annotated with additional 91 classes in COCO Stuff. Herein, this dataset contains 171 categories including objects and stuff annotated to each pixel.

PASCAL-Person-Part [39] provides pixel-level labels for six person parts. There are 1717 training/validation images and 1818 testing images.

CamVid [36] is a road scene image segmentation dataset which provides dense pixel-wise annotations for 11 semantic categories. There are 367 training images, 101 validation images and 233 testing images.

ADE20K [140] is a recently released dataset for scene parsing which has 20210 training, 2000 validation and 3352 test images. This dataset contains 150 categories including object and stuff annotated to each pixel.

Cityscapes [22] is a recent street scene dataset which contains 5000 images with pixel-level fine annotations. There are 2975 training images, 500 validation images and 1525 testing images. And 19 classes (e.g. roads, bicycles and cars) are considered for evaluation on the testing sever provided by the organizers.

2.6 Summary

In a nutshell, in this chapter, we discuss major research directions in scene parsing in depth. We review and discuss the scene parsing works from contextual modeling, multi-scale aggregation, graphic models, and label variety. Next, we briefly go through the related boundary detection literatures and discuss how to enhance scene parsing performance with boundary information. Then, we discuss and compare different interaction ways for interactive segmentation and propose a novel interactive segmentation model that brings clicks and phrases together. Finally, we demonstrate the representative benchmark datasets for scene segmentation.

Starting from next chapter, we present and elaborate the technical details of the three proposed scene parsing models and one interactive segmentation model.

Chapter 3

Context Encoding and Multi-Path Decoding

Scene parsing implicitly involves object recognition, localization, and boundary delineation, a robust segmentation network needs to perform well at all of these implied tasks. In this chapter, we propose a segmentation network called CGBNet to enhance the segmentation performance by context encoding and multi-path decoding. We first propose a context encoding module that generates context-contrasted local feature to make use of the informative context and the discriminative local information. This context encoding module greatly improves the segmentation performance, especially for inconspicuous objects. Furthermore, we propose a scale-selection scheme to selectively fuse the segmentation results from different-scales of features at every spatial position. It adaptively selects appropriate score maps from rich scales of features. To improve the segmentation performance results at boundary, we further propose a boundary delineation module that encourages the location-specific very-low-level features near the boundaries to take part in the final prediction and suppresses them far from the boundaries. The proposed segmentation network achieves very competitive performance in terms of all three different evaluation metrics consistently on the six popular scene segmentation datasets, Pascal Context, SUN-RGBD, Sift Flow, COCO Stuff, ADE20K, and Cityscapes.

3.1 Introduction

Semantic segmentation aims to classify every pixel of a given image to one of semantic classes including not only objects (e.g. person, car, cat) but also stuff (e.g. road, grass, sky). It is a dense prediction task whose output has the same resolution with input. Semantic segmentation has been an essential component of computer vision and is in intense demand for practical applications, such as automation devices, virtual reality, self-driving vehicles and etc. It implicitly involves object recognition, object localization and boundary delineation, which requires multi-scale and multi-level visual recognition. A robust segmentation network needs to perform well at all of these implied tasks.

Recently, the great success of Deep Convolutional Neural Networks (DCNN) has largely improved the performance of computer vision tasks [6], such as image classification [1, 2, 5, 7], image deblurring [141] and object detection [8, 9, 41–43]. However, there are still some limitations when directly applying DCNN to the dense prediction tasks like semantic segmentation [3, 10, 12, 13, 142–144]. DCNN learns very abstract feature representation of the whole image [14] and hence extracts information of dominant/salient objects of image. This is highly desirable for image classification and object detection. For semantic segmentation, however, spatial information is essential and pixel-level discriminative features are desired. Although most state-of-the-art semantic segmentation frameworks are based on the image classification networks pre-trained on Imagenet [15], it remains an open question of how to better adopt DCNN on semantic segmentation. Herein, we mainly consider three issues when applying DCNN to dense prediction tasks: the various forms of objects/stuff (e.g. salient or inconspicuous), the existence of multi-scale objects and the loss of accurate spatial boundary information. To address these issues, we propose a new segmentation model with context encoding, multi-path decoding and boundary delineation to enhance the segmentation performance from different levels.

First, semantic segmentation aims at labeling every pixel, which is different from image classification and object detection that target at image-level or region-level labeling. It requires dense prediction with high resolution. Moreover, not only the dominant salient objects but also the stuff and inconspicuous objects should be well parsed. DCNN pretrained on Imagenet [15] prefers image-level abstract

FIGURE 3.1: (Best viewed in color) Scene segmentation refers to labeling every pixel of image including salient objects, inconspicuous objects and stuff. However, the various forms of objects/stuff (e.g salient or inconspicuous), the existence of multi-scale objects (e.g the multi-scale cows in third image) and the capricious boundary shape of objects make it challenging to parsing each pixel using DCNN.

features that is not equally discriminative at every spatial position but dominated by salient objects. Therefore, when directly applying DCNN on semantic segmentation, inconspicuous objects are easy to be overwhelmed by salient objects and their information will be weakened or even disregarded. This is undesirable for the scene segmentation. To address this issue, locally discriminative features together with their contexts are essential for scene segmentation. A lot of work devotes to get informative context, e.g. [3, 55, 60, 75]. However, contexts often have smooth representation and are dominated by features of salient objects, which is harmful for labeling inconspicuous objects. Better features for scene segmentation are discriminative context-aware local features, in which each pixel remains its discrimination and also gets information from surroundings. For this purpose, we propose a context-contrasted local feature, which benefits from both context and local information. The proposed context-contrasted local feature not only exploits the informative context but also spotlights the discriminative local information in contrast to the context. Further, we use a context-contrasted local (CCL) model to aggregate multi-scale and multi-level context-contrasted local features.

Second, object recognition extracts features for recognizing the whole image, but for segmentation, it is irrational to classify all individual pixels based on a single scale of features. Due to the huge scale variation of objects (e.g. the multi-scale cows in Fig. 1), adaptive scales of features for parsing different pixels are desired, for example, smaller scale of receptive field for pixels belonging to smaller objects. Previous works address this issue with several different ways. One way is to resize the input image to multiple resolutions and feed them to different (or a shared) networks, then fuse their features, such as approaches in [68, 71, 73, 145]. The aggregation ability of this strategy is limited due to the limited scales of input images used in practice to avoid heavy computational load of this scheme. Another way makes use of multiple levels of features from the middle layers of network, such as approaches in [10, 74, 146, 147]. The intention of this strategy is to exploit multi-scale features. We follow the way of FCN [10] to adopt skip layers to utilize multi-scale features, which is effective and economic. However, in previous works, such as [10, 61, 66, 69, 74, 75], the score maps of skip layers are integrated via a simple sum fusion that ignores different importance of different scales. To address this problem and find an optimal integration choice, we propose a network that controls the information flow of different scale features. It generates control signals to perform a gated sum of the score maps to aggregate multi-scale features selectively. As a selection mechanism is embedded in the multi-scale fusion, more skip layers can join the aggregation to provide richer information for selection. This enhances the aggregation ability of multi-scale features.

Third, detailed spatial information is essential for scene segmentation, especially for the implicit low-level tasks, such as boundary delineation. However, due to the pooling operation or convolution strides of DCNN, lots of spatial information is lost during the encode process. To address this issue, one direction is to retain the spatial information using dilated DCNN, such as DeepLab [3] and PSPNet [13] that remove some of the pooling layers or convolution strides. The dilated DCNN, though performs well, is computationally expensive due to retaining numerous relatively high-resolution feature maps. Another direction is to recover the spatial information during the decode process, such as FCN [10] and RefineNet [123] that employ low-level features. These features contain sufficient low-level visual information that provides textural information and helps make the final prediction finer. However, incorporating very-low-level features (e.g. `block2` in ResNet [2]) brings some noise, which adversely affects the high-level tasks such as the segmentation of

coherent semantic regions of large-scale objects. In other word, chaotic very-low-level information is not always helpful for every pixels. It should be exploited only for pixels near the textural boundary for accurate boundary delineation while be stifled in the homogenous region. To find a way to appropriately exploit the very-low-level features, we propose a refinement method that gets the textural boundary information from an initial prediction of segmentation network. Then, to refine the initial prediction, the boundary information is fed to the network to incorporate the very-low-level features in the aggregation for pixels near the boundary while suppressing them for other pixels. This method brings very little computation but improves the boundary delineation.

In summary, this chapter makes the following contributions: 1) We propose a novel context-contrasted local feature that is tailored for scene segmentation and propose a context-contrasted local (CCL) model to aggregate multi-scale context-contrasted local features. 2) We propose a gated sum scheme to selectively aggregate appropriate-scale features at each spatial location, which is an efficient and effective way to address the issue of the existence of multi-scale objects. 3) We propose a boundary delineation model to refine the prediction of boundary, which encourages the location-informative very-low-level features near the boundaries and suppressed them far from the boundaries. 4) We achieve very competitive performance consistently on the six popular semantic segmentation datasets.

3.2 The Proposed Segmentation Network

3.2.1 Overall Framework

In this chapter, we propose a segmentation network called CGBNet to enhance the segmentation performance by context encoding, multi-path decoding and boundary delineation.

The overall framework of the proposed segmentation network, named as CGBNet (CCL, Gated sum and BDR), is shown in Fig. 3.2. It contains context-contrasted feature extraction for context encoding, gated multi-scale aggregation and boundary delineation refinement for multi-path decoding. The baseline is a FCN-like architecture with ResNet-101 as backbone network. The proposed context-contrasted

FIGURE 3.2: The overall framework of the proposed CGBNet. The proposed context-contrasted local (CCL) model generates multi-level and multi-scale context-aware local features. Gated sum selectively aggregates rich scale features in DCNN and CCL. BDR encourages pixels near the boundaries to be assigned more very-low-level textural information.

local (CCL) model in Fig. 3.2 generates multi-level and multi-scale context-aware local features. The proposed gated sum denoted by $g+$ in Fig. 3.2 selectively aggregates rich scale features¹ extracted in DCNN and CCL. Furthermore, to generate finer boundary delineation in the final prediction, a boundary delineation refinement model (BDR) is proposed to filter the very-low-level features. The proposed CCL, Gated Sum and BDR are presented in details in the following sections.

¹Our skip connections extract features from last layer of each residual block in ResNet-101, the residual block is shown in Figure 2.2.

FIGURE 3.3: (Best viewed in color) Visualization of different feature information. The local information of pixel $\mathbb{A}$ could not aggregate useful contexts, such as road and other cars. However, its contexts will be dominated by the features of the men in the both schemes of dense context and coarse context. The context-local scheme injects blur context to local feature of pixel $\mathbb{A}$ to make discriminative context-aware local feature.

3.2.2 Context-Contrasted Local Feature

Context aims at collecting surrounding information and enlarging the effective receptive field, which can greatly improve the performance of semantic segmentation. DCNN trained for object recognition has already generated relatively high-level context features [2, 7], but its high-level features are learnt for overall abstract representation of the whole image, which focus on the dominant parts and cannot ensure useful context for inconspicuous objects and stuff. Also, they may not be discriminative at some spatial positions as they are trained to discriminate the whole image collectively. Therefore, the context features for object recognition are not directly favorable for scene segmentation that aims at classifying every pixel [56]. Lots of previous works devote to obtain context for semantic segmentation. For example, [3] and [55] use dilated convolutions to aggregate coarse context while [75] and [59] adopt a recurrent neural network to capture long-range dense context. However, either dense context or coarse context is easy to be dominated by the features of salient objects, resulting in weak signals of inconspicuous objects. Comparing to object segmentation that focuses on salient objects, there are richer

categories and complex conjunctions between categories in scene segmentation [75]. Due to the complexity of objects and stuff in scene segmentation, indiscriminately collecting context information may bring harmful interference, especially under clutter surroundings. For example, in Fig. 3.3, compared with the two persons, the cars behind them are inconspicuous objects. The detailed local features collect information around pixel A and are discriminative to other pixels, but they are not aware of global information such as road and building, thus could not obtain robust high level information for pixel A. However, aggregating context for pixel A brings features of salient objects like the persons and hence will be dominated by the features of the persons. Some information of cars may be ignored in the final prediction, resulting in wrong labeling for pixels at that location. Also, contexts at different positions are apt to a consistency representation of dominant features, thus are lacking in spatial discrimination. Therefore, it's very hard to collect appropriate and discriminative high level features for pixel A.

Herein, it is significant to design tailored context features for scene segmentation. To this end, a context-contrasted local feature is proposed in this chapter to perform high-level feature modeling, which injects blur context to local feature to make discriminative context-aware local feature. Specifically, we propose to generate local information and context separately and then fuse them via making contrast between these two:

$$\mathbb{CL} = \mathcal{F}_l(\mathbb{F}, \Theta_l) - \mathcal{F}_c(\mathbb{F}, \Theta_c) \quad (1)$$

where $\mathbb{F}$ is the input features, $\mathcal{F}_l$ is the function of local **Conv**, $\mathcal{F}_c$ is the function of context **Conv**, Θ_l and Θ_c are respective parameters, and $\mathbb{CL}$ is our designed context-contrasted local features. They make a contrast between the separated context and local information, thus not only exploit useful context but also foreground the local information in contrast to the context. The function of context-local forces the networks generating tailored features for scene segmentation. Context shows solicitude for each pixel but brings some unnecessary information while local focuses on vicinal information but ignores other parts, resulting in losing some essential information. Different from the above two, context-local pays more attention to local while collecting coarse context to aggregate pivotal information, thus could get robust high-level features.

It is a mechanism that imitates the human visual behavior. When our human beings look at an object, we always collect the context for that object in a way

FIGURE 3.4: **context-contrasted Local (CCL)** scheme is a convolutional network integrating multi-level context-aware local features. Each block of CCL consists of two parallel parts: coarse context convolution and delicate local convolution. The context-aware local features are obtained via making a contrast between the context and local information. Several blocks are chained to make multi-level context-contrasted local features.

that our eyes focus on that object in contrast to the blurred surroundings [148]. In other word, we concentrate on that object while we are aware of its surroundings. The architecture of CCL is shown in Fig. 3.4. It consists of several chained context-local blocks to make multi-level context-contrasted local features. Gated sum (presented in the next section) is adopted in CCL to selectively aggregate different levels of context-contrasted local features.

Comparison with Existing Context Models. DAG-RNN [75] performs contextual modeling by propagating local information in feature maps to encode long-range context. Different from DAG-RNN, CCL exploits multi-scale features for segmentation, and the context-aware local features of CCL are different from those in DAG-RNN. ASPP [3] aggregates multi-scale contexts via combining score maps generated by different context aggregation branches, each of which uses dilated

Conv kernels with different stride rates to generate different scale contexts in parallel. Compared with this type of context model, CCL first contextualizes contrasted features at every block to obtain context-aware local features, which combines two different scales in the feature level and takes advantage of both context and local information, then further aggregates multi-scale context-contrasted local features in score level. Moreover, the score maps of CCL are fused via gated sum instead of the simple sum. CRF [65] is ordinarily applied to score maps and boosts consistency of low-level information, while CCL aims at discriminative high-level features. In fact, CRF can also be used as a post-processing step to promote performance of our segmentation network.

3.2.3 Gated Multi-scale Aggregation

Due to the existence of objects at multiple scales, it is difficult to directly apply DCNN to scene segmentation to obtain appropriate information for all pixels [56]. In this section, we discuss how to extract different scale of features from DCNN. An efficient and effective way is to add skip layers from the middle layers of DCNN. Based on the encoder-decoder architecture of FCN [10], skip layers being as classifiers are used to exploit multi-scale features in DCNN to generate corresponding segmentation score maps. However, in previous works such as [10, 61, 66, 74, 75], the classification score maps of skip layers are non-selectively fused by sum, which does not take into account the individual differences of these score maps for different pixels. Some pixels prefer scores from features with a larger receptive field, e.g., pixels belonging larger objects while others desire features with smaller receptive field, e.g., pixels belonging to smaller objects. Moreover, some pixels of complex structures may need to aggregate multi-scale features for better information collection. If these score maps are aggregated indiscriminately, the inapposite or incorrect scores may harm the final prediction. It is better to provide scale selections for every pixel.

To this end, we propose an selection scheme called gated sum to adaptively select the appropriate receptive field for each pixel based on its scale or representation support. This scheme contains inherent position-wise gates to select outputs from skip layers and control the information flow of DCNN. As skip layers are aimed at capturing multi-scale features, a by-pass and simple solution is to pick different

skip layers for different pixels in FCN framework. With the gated sum fusion, the network customizes a suitable aggregation choice of score maps according to the information of images, corresponding to choosing which scales of features are better and more desirable for each pixel. Moreover, with gated sum fusion, we can add more skip layers to extract richer scale information without posing the problem of inapposite results.

In order to obtain the information to control the gates, such as scale and contextual support, info-skip layers consisting of **Conv+Sigmoid** are introduced to extract the information from corresponding feature maps and generate information maps with size of $H \times W$, where $H \times W$ is the spatial size of feature maps. Since these information maps, $\mathbb{I}_p^n$, and score maps, $\mathbb{S}_p^{c,n}$, of skip layers are generated from a same DCNN or CCL, the sequence relationship, e.g. from low level to high level, among feature maps of DCNN and CCL should also be considered. Recurrent Neural Networks (RNN) [47, 149, 150] are effective and efficient to learn such sequence relationship, thus all of the information maps are fed to RNN in sequence to learn the relationship of these information maps. Based on RNN, these information maps can be aware of neighbourhood maps and acquire the sequence relationship among all of the information maps. The proposed scheme of a general gated sum for feature maps of the same size is shown in Fig. 3.5.

In details, we hypothesize that the information maps of higher layers have already grasped the information of lower layers due to the effect of DCNN, thus the RNN begins with information map of the last layer of our segmentation network. Suppose a gated sum has N score maps in a block that have the same spatial size $H \times W$, $\mathbb{S}_p^{c,n}$, generated by N skip layers from different scale features $\mathbb{F}_p^n$. We have $\mathbb{S}_p^{c,n} = \mathcal{F}_s^n(\mathbb{F}_p^n, \Theta_s^n)$, where p is the spatial position, $n \in 1, 2, \dots, N$, $c \in 1, 2, \dots, C$ and C is the number of class labels. $\mathcal{F}_s^n$ is the classification function of n th skip layer, Θ_s^n is its parameters and $\mathbb{F}_p^n$ denotes the input feature maps with the dimensionality of $H \times W \times \#channels$. For each skip layer, we first generate an information map $\mathbb{I}_p^n$ of size $H \times W \times 1$ from the corresponding feature maps:

$$\mathbb{I}_p^n = \mathcal{F}_i^n(\mathbb{F}_p^n, \Theta_i^n) \quad (2)$$

where $\mathcal{F}_i^n$ is the function of n th info-skip layer **Conv+Sigmoid** and Θ_i^n is its parameters. Then these information maps $\mathbb{I}_p^n$ are inputted to a RNN in sequence to learn

FIGURE 3.5: **Gated Sum** controls the information flow of skip layers via its inherent gates. The gates G_p^n adjusts its value for each pixel according to the input images. All the maps in gated sum have the same spatial size of $H \times W$.

their relationships:

$$h_p^n = \tanh \left(W^n \begin{pmatrix} h_p^{n-1} \\ \mathbb{I}_p^n \end{pmatrix} \right) \quad (3)$$

where h_p^n is the n th output of RNN. To make our network efficient, all positions are processed parallelly and W^n is shared for all spatial positions. To ensure that every information map be aware of global information, the outputs of RNN are concatenated, $H_p = (h_p^1 \dots h_p^N)^T$, and refined with global information:

$$\bar{H}_p = \mathcal{F}_g(H_p, \Theta_g) + H_p \quad (4)$$

where $\mathcal{F}_g$ is a 1×1 Conv and Θ_g is its parameters. Next, $\bar{H}_p$ is split, $\bar{H}_p = (\bar{h}_p^1 \dots \bar{h}_p^N)^T$, and used to generate the gates G_p^n for gated sum:

$$G_p^n = N \frac{e^{\bar{h}_p^n}}{\sum_{ik=1}^N e^{\bar{h}_p^i k}} \quad (5)$$

The sum of G_p^n for each position p is normalized to N . Finally, N score maps are selectively fused via gated sum:

$$\bar{\mathbb{S}}_p^c = \sum_{n=1}^N G_p^n \circledast \mathbb{S}_p^{c,n} \quad (6)$$

where $\circledast$ denotes element-wise multiplication and $\bar{\mathbb{S}}_p^c$ is the output of the gated sum. The embedding of this gated sum (Fig. 3.5) into the encoder-decoder architecture (Fig. 3.2) will be discussed in section 3.3.3.

The gates of gated sum control the information flow of skip layers, i.e. how much the $\mathbb{S}_p^{c,n}$ can pass the gates depends on the value of G_p^n . A larger G_p^n means a better feature n th skip layer for labeling of position p . While a smaller G_p^n means that for position p , the segmentation result generated by the n th skip layer is not desirable and should be inhibited. More importantly, G_p^n is neither fixed value nor directly learned from training data. It is generated from the testing image by the proposed network learned from the training data. Thus, G_p^n is adaptive to different pixels of testing image. The values of G_p^n not only depend on the training data, but also depend on the testing input images and vary according to the feature maps. Therefore, we call them “gates” to differentiate them from the simple fixed or learned “weights”. With gated sum, the network adaptively (to different testing images) selects appropriate score maps from rich scales of features.

3.2.4 Boundary Delineation Refinement

Semantic segmentation requires multi-level tasks, including low-level tasks such as boundary delineation, high-level tasks such as object recognition. A well-performed semantic segmentation architecture should be able to effectively deal with different tasks. In this section, we discuss how to recover the boundary information and refine the boundary delineation of segmentation.

Due to the pooling operations and convolution strides, detailed spatial information is lost during the encode process, which does not conform to the goal of scene segmentation. To address this issue, up-sampling and decoding the output of D-CNN through learnable deconvlutional filters [48] is feasible and efficient. But the deconvolution alone cannot recover the textual spatial information that is already lost due to the pooling operations and convolution strides. To recover the spatial knowledge, it is necessary to exploit the very-low-level feature information that contains sufficient textural visual information (e.g. corners, edges, etc), and

FIGURE 3.6: **Boundary Delineation:** Although pixel B and Pixel C belong to the same object, their desirable scale features are different due to their distance to the center of the object.

supplement these information for the accurate boundary prediction. Compared with the gated sum that mainly selects different scale features for the task of segmenting different scales of objects, boundary delineation requires higher resolution (lower-level) features to provide finer informative textural boundary knowledge. For example, in Fig. 3.6, to delineate the boundary layout of object 1, the parsing of pixel B is essential. Although the gated sum can select different scales of feature for pixel A and pixel C, as the processing of upsample, the spatial size is enlarged and the very-low-level information (e.g. corners, edges, etc) becomes more and more important for getting finer prediction around pixel B. Therefore, providing meticulous edge information for parsing pixel B is necessary. However, the very-low-level features are not suitable for high-level tasks such as object recognition as incorporating very-low-level features brings some noise information for the high-level tasks, especially for the positions relatively far from boundary.

To get meticulous boundary information, it is necessary to make use of the very-low-level features that embody textural visual information and have higher signal-to-noise ratio (SNR) near the boundary. On the other hand, the very-low-level features contain noise information and have lower SNR far away from the boundary, which are not robust for object recognition. Therefore, we propose a boundary delineation refinement (BDR) model that suppresses the very-low-level features far away from the boundary where their SNR is lower, and engages them near the boundary where the SNR is higher.

In detail, we propose a refinement method that first gets the boundary map $\mathcal{B}_p^0$ from

the initial prediction of segmentation network, $\mathcal{B}_p^0 = 1$ for pixels at the boundary and $\mathcal{B}_p^0 = 0$ otherwise. Since the boundary generated from the initial prediction is not perfectly aligned with the real edges, directly using this hard mask may repeat mistakes of the initial prediction. To address this issue, isotropic low-pass Gaussian filters with different standard deviation σ^m are adopted to diffuse the initial hard boundary map, which yields soft boundary masks $\mathcal{B}_p^m$, where m is the layer index of block 2 in Fig. 3.2 and $m \in \{1, 2, 3\}$. As shown in Fig. 3.7 for $m = 3$, wider boundary with different soft values is produced. The 1D profile b_x^m of the soft boundary masks $\mathcal{B}_p^m$ for a single line initial boundary can be expressed as

$$b_x^m = \frac{1}{\sigma^m \sqrt{2\pi}} e^{-\frac{(x-t)^2}{2\sigma^m}} \quad (7)$$

where t is the position of the initial textural boundary, σ^m represents the σ value for m th layer and $\sigma^m \in \{1, 3, 5\}$, $m \in \{1, 2, 3\}$, sequentially corresponding to three very-low-level features (from lower to higher) used in BDR. Fig. 3.8 plots these three 1D profiles b_x^m of the soft boundary masks $\mathcal{B}_p^m$. It shows that all these three soft boundary masks decrease their values with the increase of the distance from the initial boundary. Moreover, as shown in Fig. 3.8 at $x = t, t + \Delta_1, t + \Delta_2$, nearer the initial boundary, lower-level features have higher mask values than higher-level features while farther away from the initial boundary, higher-level features have higher mask values than lower-level features. These masks are then used to select the very-low-level features that contain textural boundary information as a supplement to the up-sampled high-level score maps from the gated sum:

$$\tilde{\mathbb{S}}_p^c = \sum_{m=1}^M \mathcal{B}_p^m \circledast \dot{\mathbb{S}}_p^{c,m} + \bar{\mathbb{S}}_p^c \quad (8)$$

where $\mathcal{B}_p^m$ is the m th soft boundary mask, $\dot{\mathbb{S}}_p^{c,m}$ is the score maps from m th very-low-level features, $\tilde{\mathbb{S}}_p^c$ is the boundary refined score maps, $\bar{\mathbb{S}}_p^c$ is the upsampled output of gated sum. To remove the possible isolated noise introduced by using the very-low-level features, a simple median filter is adopted in testing process to smooth the final prediction.

The proposed BDR encourages pixels near the boundaries to be assigned more very-low-level information and suppresses it far from the boundaries. It brings very little computation yet enhances the boundary delineation.

FIGURE 3.7: We use isotropic Gaussian filters with different standard deviation ($\sigma^{1,2,3} = 1, 3, 5$) to the hard boundary mask generated by the initial prediction and get a soft boundary masks. Such masks engage the very-low-level features in the regions (near the boundary) with higher SNR and suppress those in the regions (far away from the boundary) with lower SNR.

FIGURE 3.8: Example of three 1D profiles b_x^m of the soft boundary masks $\mathcal{B}_p^m$, $m = 1, 2, 3$. t is the initial textural boundary position.

3.3 Experiments

We evaluate our segmentation framework on six public scene segmentation datasets, Pascal Context, SUN-RGBD, Sift Flow, COCO Stuff, ADE20K and Cityscapes.

3.3.1 Implementation Details

We use truncated ResNet-101 (pre-trained on ImageNet [15]) as our backbone model. In detail, pool5 and layers after it are discarded and a convolutional adaption layer that decrease the feature channels from 2048 to 512 is placed on the top of truncated ResNet-101 to reduce parameters. The number of layers in CCL

can be modified according to inputs. All the experiments of this chapter have six. We upsample the score maps with deconvolution (transpose convolution).

Our Network is trained end-to-end with SGD with fixed momentum 0.9 and weight decay 0.0005. Following [3], we employ the "poly" learning rate, $Lr_c = Lr_i \times (1 - \frac{iter}{max.iter})^{power}$, where the Lr_c is current learning rate and Lr_i is the initial learning rate. The initial learning rate is set to be 10^{-3} and the power is set to 0.9. The iteration number is set to 15K for Pascal Context, 13K for SUN-RGBD and 20K for COCO Stuff. The batch size is 10 during training and the statistics of batch normalization layer is updated after the final iteration. The parameters of new layers are randomly initialized with Gaussian distribution (variance 10^{-2}) and trained with higher learning rate ($\times 3$). For batch processing, all images are resized to have maximum extent of 512 pixels and padded with zeros to 512×512 pixels during training. We randomly flip the images horizontally to augment the training data. Batch Normalization (BN) [151] is used to accelerate training.

We evaluate our network with three performance metrics: Global Pixel Accuracy (**GPA**), Average Class Accuracy (**ACA**) and Mean Intersection-over-Union (**IoU**). Mathematical definitions please refer to [10].

3.3.2 Multi-scale Context-Contrasted Local Features

In section 3.2.2 we introduced context-contrasted local (CCL) model to integrate multi-level context-aware local features. To evaluate the key principle (i.e. multi-scale context-contrasted local features) of CCL, we simplify our context-local network architecture CCL to LA, and LA^d . LA is local aggregation which abandons the context parts (dilated **Conv**) of CCL and LA^d doubles the hidden dimensionality of LA. The performance of these models are listed in Table 3.6. Their performance gap clearly demonstrates the benefit brought by the proposed CCL model.

First, compared with LA that is conventional convolutional feature, CCL aggregates specialized context-contrasted local features that not only leverages the informative context but also exploits the discriminative local information in contrast to the context. In consequence, CCL outperforms LA by a noticeable margin, which clearly shows the significance of the context-contrasted local features for scene segmentation.

Method	GPA	ACA	IoU
Baseline	73.5%	53.9%	42.5%
Baseline+LA	75.8%	57.6%	45.9%
Baseline+LA ^d	75.7%	56.6%	45.4%
Baseline+CCL	76.6%	61.1%	48.3%

TABLE 3.1: Ablation Experiments of CCL. LA is local aggregation generated by removing the context part of CCL. LA^d doubles the hidden dimensionality of LA from 512 to 1024, thus its parameter quantity is the same as CCL.

Networks	CA	IoU
Baseline	None	42.5%
Baseline + CRF[65]	CRF	43.2%
Baseline + DAG-RNN [75]	DAG-RNN	44.1%
Baseline + ASPP [3]	ASPP	44.9%
Baseline + CCL	CCL	48.3%

TABLE 3.2: Comparison with State-of-the-art Context Models. Segmentation networks are adapted to encode-decode architecture with rich skip layers, the stride rates (dilation factors) of the four branches in ASPP are revised to {1, 3, 4, 6} respectively. For fair comparisons, gated sum is not adopted, and they only differentiate each other in terms of context aggregation (CA).

It’s crucial to introduce new parameters that fill the domain gap during the fine-tuning of segmentation networks from classification networks. However, we believe that the network architecture outweighs the magnitude of parameters for boosting the performance. To validate this, we increase the hidden dimension of LA from 512 to 1024, which is denoted by LA^d in Table 3.6. The parameter quantity of LA^d is then the same as CCL, but LA^d does not improve the performance of LA and even slightly make it worse. This convinces us that the noticeable performance boost is mainly contributed by the architecture of the context-contrasted local features, not from the simple increase of network parameters. And we study the effect of the number of CCL blocks, as shown in TABLE 3.3, the CCL-6 achieve the best performance and thus we adopt 6 blocks in CCL. Further tests of CCL can be seen in TABLE 3.6.

We compare with other context models in a controlled experiment and summarize their performance in Table 3.2. The proposed CCL noticeably outperforms others,

CCL-#	CCL-1	CCL-3	CCL-5	CCL-6	CCL-7	CCL-8
IoU	44.3%	46.8%	48.1%	48.3%	48.3%	48.2%

TABLE 3.3: Ablation study of the number of CCL blocks.

which demonstrates the significance of CCL.

3.3.3 Embed Gated Sum into Encoder-Decoder Architecture

Gated sum is a selection mechanism to pick appropriate scale of features. But for the encoder-decoder architecture, the spatial sizes of distinct blocks are not the same, e.g. 16×16 for block 5 and 64×64 for block 3 in Fig. 3.2. This causes difficulty of aggregating all the score maps in one run. The most straightforward solution is upsampling all the score maps to the same resolution before the aggregation. However, this consumes a large amount of resources. Therefore, in this work, we first propose a general scheme of gated sum shown in Fig. 3.5 and then embed it into the encoder-decoder architecture in Fig. 3.2. For the embedding, we adopt the gated sum within each block where the feature maps possess the same spatial resolution. Then the output of gated sum is upsampled to higher resolution to participate in the gated sum in block with higher resolution. Meanwhile, to pass the information maps from block to block, the last output of RNN is also upsampled to generate the gates for the upsampled score map and the inputs of the next RNN.

We visualize the learned weights of gated sum in Fig. 3.9. As shown in Fig. 3.9, for the higher layer, the gated weights are relatively larger in some larger-scale area, such as the background stuff in first two images and the bigger cow in third image. For the lower layer, the gated weights are weaker in the large-scale area but stronger in smaller-scale area, such as the smaller aeroplane and boundary of larger aeroplane in the first image.

We also present an ablation experiment of the gated sum in Table 3.6. As shown in Table 3.6, the gated sum improves the performance visibly. Comparing ResNet-101 to ResNet-50, we see that the performance gain brought by the gated sum will be higher if there are more score maps for selecting.

FIGURE 3.9: **Visualize of gated weights:** here we visualize gated weights of Block 3 in Fig. 3.2, of which the resolution is relatively high and suitable to generate significantly visible weights map. We select the first layer and last layer in the gated sum of Block 3, noted as higher layer and lower layer respectively.

3.3.4 Segmentation result comparisons of Boundary Delineation Refinement

In Table. 3.5, we show the effect of using the very-lower-level features (e.g. block2 in ResNet, Pool2 in VGG16). As shown in Table. 3.5, although FCN-4s utilizes more skip connections from the very lower-level features, its segmentation results are inferior than FCN-8s [10]. The very low-level features contain informative

FIGURE 3.10: (Best viewed in color) **Boundary Delineation Refinement** helps to obtain textural boundary information from low-level features but avoid bringing too much noise information.

BDR	Boundary Region IoU	Overall IoU
X	40.2%	51.6%
✓	41.6%	52.2%

TABLE 3.4: Ablation study of BDR in boundary regions

Methods	Very low-level	BDR	IoU
FCN-8s	X	n/a	48.1%
FCN-4s	✓	X	47.8%
FCN-4s	✓	✓	48.5%

TABLE 3.5: Effect of Using the Very Lower-level Features. The very low-level features (`block2` in ResNet) brings detailed textural boundary information but also noise that are not suitable for the high-level tasks. Note the context aggregation model of FCN is replaced by CCL.

textural layout but also noise. Using them at all pixels in the multiple scale aggregation adversely affects the segmentation performance. To address this issue, we propose a boundary delineation refinement model to selectively utilize the low-level features: encourages the very low-level features near the boundaries but suppress them far from the boundaries.

We also present qualitative segmentation result comparisons of Boundary Delineation Refinement in Fig. 3.10. As shown in Fig. 3.10, the boundary delineation refinement model improves the details of boundary visibly. To verify the effect of BDR among boundaries, we also evaluate our model only in boundary regions, which locate close to boundaries (distance ≤ 10 pixels). As shown in TABLE 3.4, although the overall performance brought by BDR is only 0.6%, the performance gain in the boundary regions reaches 1.4%. This agree with the goal of BDR and also confirms its effectiveness.

3.3.5 Ablation Study and Runtime comparison

An ablation experiment of the proposed models is presented in Table 3.6. We provide an insight into the effect of the proposed methods by a detailed experimental

Baseline Model	MS	BDR	Gated Sum	CCL	IoU
ResNet-50					40.7%
ResNet-50				✓	46.3%
ResNet-50		✓		✓	47.0%
ResNet-50			✓	✓	48.1%
ResNet-50		✓	✓	✓	48.5%
ResNet-50	✓	✓	✓	✓	49.2%
ResNet-101					42.5%
ResNet-101				✓	48.3%
ResNet-101		✓		✓	49.4%
ResNet-101			✓	✓	51.6%
ResNet-101		✓	✓	✓	52.2%
ResNet-101	✓	✓	✓	✓	53.4%

TABLE 3.6: Ablation Experiments of Proposed Methods. MS means multi-scale testing. The proposed CCL, gated sum and BDR improve the performance visibly.

Model	BDR	Gated Sum	CCL	Time(ms)
ResNet-101				138.9
ResNet-101			✓	140.8
ResNet-101		✓	✓	158.7
ResNet-101	✓	✓	✓	159.1

TABLE 3.7: Runtime Comparison. Runtime comparison on a single NVIDIA Tesla K80. It shows that the proposed network marginally increase the running time from the baseline.

study that illustrates how the segmentation results are affected by each of the proposed methods as well as their combinations. As shown in Table 3.6, the proposed CCL, gated sum and BDR improve the performance respectively, which shows the significance of this work.

The average inference time of proposed segmentation networks on Pascal Context images are listed in Table 3.7 (on a single NVIDIA Tesla K80). It shows that the proposed network marginally increase the running time from the baseline.

Methods	GPA	ACA	IoU
O2P[152]	-	-	18.1%
CFM [153]	-	-	34.4%
FCN-8s [10]	67.5%	52.3%	39.1%
CRF-RNN [67]	-	-	39.3%
ParseNet [21]	-	-	40.4%
BoxSup [154]	-	-	40.5%
ConvPP-8 [155]	-	-	41.0%
HO-CRF [156]	-	-	41.3%
PixelNet [157]	-	51.5%	41.4%
Context-CRF [68]	71.5%	53.9%	43.3%
DAG-RNN + CRF [75]	73.6%	55.8%	43.7%
FCRN [158]	72.9%	54.8%	44.5%
EFCN [159]	74.5%	57.7%	45.0%
DeepLab-v2+CRF [†] [3]	-	-	45.7%
Hu et al.[160]	73.5%	56.7%	45.8%
Global-Context[161]	73.8%	-	46.5%
RefineNet-Res152 [123]	-	-	47.3%
DenseDecoder [162]	-	-	47.8%
EncNet [63]	81.2%	-	52.6%
CGBNet	79.6%	64.7%	53.4%

TABLE 3.8: Performance Comparison on Pascal Context Dataset. The proposed CGBNet outperforms all existing methods by a large margin across all evaluation metrics. Methods trained with extra data are marked with [†].

3.3.6 Comparison with the State-of-the-Arts

Pascal Context [33] contains 10103 images from Pascal VOC 2010, and these images are re-annotated as pixel-wise segmentation maps. There are 4998 images for training and 5105 images for testing in Pascal Context. We use the most common 59 categories in this dataset for evaluation. A few examples on validation set of Pascal Context are shown in Fig. 3.11. Compared with the baseline, our segmentation network performs better at global information, salient objects, stuff and inconspicuous objects and has a robust adaptability to multi-scale objects. Quantitative results of Pascal Context are shown in Table 3.8. It shows that our segmentation network outperforms the state-of-the-arts by a large margin for all the three evaluation metrics.

SUN-RGBD [37] provides pixel-wise labeling for 37 categories. It has 10335 indoor images which are from SUN3D [136], NYUDv2 [137], Berkeley B3DO [138]

Methods	GPA	ACA	IoU
Liu et al. [79]	-	10.0%	-
Ren et al. [163]	-	36.3%	-
FCN-8s [10]	68.2%	38.4%	27.4%
DeconvNet [48]	66.1%	33.3%	22.6%
Kendall et al. [164]	71.2%	45.9%	30.7%
SegNet [165]	72.6%	44.8%	31.8%
DeepLab [3]	71.9%	42.2%	32.1%
Context-CRF [68]	78.4%	53.4%	42.3%
RefineNet-Res152 [123]	80.6%	58.5%	45.9%
CGBNet	82.3%	61.3%	48.2%

TABLE 3.9: Performance Comparison on SUN-RGBD Dataset. The proposed CGBNet outperforms existing methods consistently across all the three evaluation metrics.

and the newly captured images. The training set has 5285 images and the test set contains 5050 images. We only use the RGB modality as input for training and testing. Quantitative results of SUN-RGBD are reported in Table 3.9. It shows that our segmentation network outperforms the previous state-of-the-arts consistently across all evaluation metrics.

Sift Flow [38] contains 2688 images labeled by 33 classes. We use the standard split protocol thus 2488 images are used for training and 200 images are used for testing. Quantitative results of Sift Flow are shown in Table 3.10. Our segmentation networks outperforms state-of-the-arts across all evaluation metrics.

COCO Stuff [35] contains 10000 images from Microsoft COCO dataset [139], out of which 9000 images are for training and 1000 images for testing. The unlabeled stuff pixels in original images of Microsoft COCO are further annotated with additional 91 classes in COCO Stuff. Herein, this dataset contains 171 categories including objects and stuff annotated to each pixel. Quantitative results of COCO Stuff are shown in TABLE 3.11. On this more challenging dataset with much more number of classes, our proposed scene segmentation network outperforms the state-of-the-arts by a large margin across all evaluation metrics, which demonstrates the capacity of the proposed CGBNet to handle complex dataset. Recently, COCO Stuff is updated to 2017 version, in which more annotated images for segmentation are provided and the split is adjusted to 40000 training images

Networks	GPA	ACA	IoU
Liu et al. [79]	76.7%	-	-
Tighe et al. [81]	75.6%	41.1%	-
Farabet et al. [71]	78.5%	29.6%	-
Pinheiro et al. [73]	77.7%	29.8%	-
Sharma et al. [62]	79.6%	33.6%	-
Yang et al. [80]	79.8%	48.7%	-
FCN-8s [10]	85.9%	53.9%	41.2%
DAG-RNN + CRF[75]	87.8%	57.8%	44.8%
Context + CRF [68]	88.1%	53.4%	44.9%
CGBNet	89.7%	58.5%	46.8%

TABLE 3.10: Performance Comparison on Sift Flow Dataset. The proposed CGBNet segmentation networks outperforms state-of-the-arts across all evaluation metrics.

and 5000 validation images. We also test our segmentation network on this new challenging dataset, shown in TABLE 3.11. There is no other published result on this newly enlarged dataset for comparison. The results of DSSPN that trained with extra data (marked with † in TABLE 3.11) are better than ours. DSSPN builds a semantic neuron graph by explicitly incorporating the semantic concept hierarchy into network construction. In such a way, DSSPN could exploit intrinsic taxonomy and semantic hierarchy of segmentation datasets and overcome the issue of the label discrepancy among different datasets, which effectively arms the segmentation network with the ability of training different datasets within a unified segmentation model. DSSPN is orthogonal to our methods. They can be applied to the segmentation network together. We believe that, if not considering about the GPU memory limitation, our approach could be further improved with such semantic concept hierarchy architecture in term of classification. We could train our model more effectively across several segmentation datasets using DSSPN.

ADE20K [140] is a recently released dataset for scene parsing which has 20210 training, 2000 validation and 3352 test images. This dataset contains 150 categories including object and stuff annotated to each pixel. We report our results on 2000 validation images in TABLE 3.12. We achieve state-of-the-art results on this dataset.

Cityscapes [22] is a recent street scene dataset which contains 5000 images with

Networks	GPA	ACA	IoU
FCN [35]	52.0%	34.0%	22.7%
DeepLab [66]	57.8%	38.1%	26.9%
DAG-RNN[75]	62.2%	42.3%	30.4%
RefineNet-Res101 [123]	65.2%	45.3%	33.6%
DSSPN-Res101 [166]	68.5%	47.0%	36.2%
CGBNet	67.4%	49.9%	36.9%
Networks	COCO Stuff 2017		
CGBNet	64.0%	52.3%	37.7%

TABLE 3.11: Performance Comparison on COCO Stuff Dataset. The proposed CGBNet outperforms the state-of-the-arts by a large margin across all evaluation metrics. Methods trained with extra data are marked with †.

Networks	GPA	ACA	IoU
SegNet[165]	71.0%	31.1%	21.6%
Cascaded-SegNet [140]	71.8%	37.9%	27.5%
FCN [10]	71.3%	40.3%	29.4%
DilatedNet [55]	74.1%	47.9%	32.3%
DAG-RNN [75]	73.9%	49.0%	33.5%
Cascaded-DilatedNet [140]	74.5%	45.4%	34.9%
RefineNet-Res152 [123]	-	-	40.7%
PSPNet-Res101 [13]	80.6%	-	42.0%
DSSPN-Res101-Universal† [166]	81.1%	-	43.7%
EncNet [63]	81.7%	-	44.7%
CGBNet	82.1%	58.2%	44.9%

TABLE 3.12: Parsing performance of different networks on ADE20K dataset.

pixel-level fine annotations. There are 2975 training images, 500 validation images and 1525 testing images. And 19 classes (e.g. roads, bicycles and cars) are considered for evaluation on the testing sever provided by the organizers. The test results are shown in TABLE 3.13.

Networks	IoU
CRF-RNN [67]	62.5%
FCN [10]	65.3%
SiCNN [167]	66.3%
DPN [60]	66.8%
DilatedNet [55]	66.8%
LRR [147]	69.7%
Deeplab-v2+CRF [3]	70.4%
Context + CRF [68]	71.6%
RefineNet-Res101 [123]	73.6%
DUC [168]	77.6%
PSPNet [13]	78.4%
DFN [87]	79.3%
DenseASPP(DenseNet-161) [4]	80.6%
CGBNet	81.2%

TABLE 3.13: Performance Comparison on Cityscapes Dataset. The proposed CGBNet segmentation networks outperforms state-of-the-arts.

3.4 Chapter Summary

In this chapter, we address the challenging task of scene segmentation. Scene segmentation aims at parsing an image into a set of coherent semantic regions and classifying each pixel to one of classes. Therefore, the context and multi-scale aggregation are crucial to achieve good segmentation. However, DCNN designed and trained for image classification tends to extract abstract features of dominated objects, which weakens or even disregards some essentially discriminative information for inconspicuous objects and stuff. To address this issue, we propose a novel context-contrasted local feature to leverage the useful context and spotlight the local information in contrast to the context. The proposed context-contrasted local feature greatly improves the image segmentation performance, especially for inconspicuous objects and stuff. Adding skip layers is a common effective and efficient way to exploit multi-scale features, but the existing approaches indiscriminately fuse the score maps of skip layers via a simple summation. To achieve an optimal multi-scale aggregation, we propose a scheme of gated sum to selectively aggregate multi-scale features for each pixel. The parameters of gates are generated from the testing image by the proposed networks learnt from the training data. Thus, they are adaptive not only to the training data, but also to the specific testing image. We

further propose a boundary refinement model to delineate the prediction of boundary. It encourages the very low-level features near the boundary and suppresses them far from it because the very-low-level features in terms of the signal-to-noise ratio often tend to be informative near the boundary but noisy far from it. Without bells and whistles, our segmentation network achieves state-of-the-art performance consistently on the six popular scene segmentation datasets used in the evaluation, Pascal Context, SUN-RGBD, Sift Flow, COCO Stuff, ADE20K and Cityscapes.

FIGURE 3.11: (Best viewed in color) Qualitative segmentation result comparisons on Pascal Context. Our segmentation network performs well at salient objects, stuff (e.g. road, grass, sky) and inconspicuous objects. Further, our network has a robust adaptability to multi-scale objects.

Chapter 4

Semantic Correlation Promoted Shape-Variant Context

Context is essential for semantic segmentation. In this chapter, we explore to further build stronger context module. Due to the diverse shapes of objects and their complex layout in various scene images, the spatial scales and shapes of contexts for different objects have very large variation. It is thus ineffective or inefficient to aggregate various context information from a predefined fixed region. In this work, we propose to generate a scale- and shape-variant semantic mask for each pixel to confine its contextual region. To this end, we first propose a novel paired convolution to infer the semantic correlation of the pair and based on that to generate a shape mask. Using the inferred spatial scope of the contextual region, we propose a shape-variant convolution, of which the receptive field is controlled by the shape mask that varies with the appearance of input. In this way, the proposed network aggregates the context information of a pixel from its semantic-correlated region instead of a predefined fixed region. Furthermore, this work also proposes a labeling denoising model to reduce wrong predictions caused by the noisy low-level features. Without bells and whistles, the proposed segmentation network achieves new state-of-the-arts consistently on the six public segmentation datasets.

FIGURE 4.1: Most existing methods model spatial-dependent context with pre-defined window (e.g., the red rectangle region for pixel A in the second image), which may not provide the specific context information and thus weaken the semantic shape layout. In this work, we propose to aggregate the context information from the semantic-dependent region instead of the spatial-dependent one.

4.1 Introduction

Semantic segmentation or scene parsing is aimed at classifying every pixel in scene images to one of the predefined categories (e.g., person, car, etc.). It has been a critical element in artificial intelligence and can be applied in many practice applications, such as automatic parking system. The recent success of Deep Neural Networks has greatly improved the performance of semantic segmentation [3, 13, 123, 169, 170]. Most of state-of-the-arts segmentation networks are based on Convolutional Neural Networks (CNNs) [1, 2, 5, 7, 44] pre-trained on ImageNet [15], in which CNNs are employed as the local feature extractor. To achieve robust semantic segmentation, informative high-level context is necessary. Context provides surrounding information of the object, which helps a better discrimination of the object.

However, due to the diverse shapes (including contours, scales, etc) and the complex layout of objects in scene images, the common context aggregated within a

predefined fixed region weakens the semantic shape layout and might bring unnecessary irrelevant information. For example in Figure 4.1, contexts of pixel A (lake) and pixel B (train) should be different but the predefined receptive fields to collect their contexts largely overlap, which decrease their discriminative power. Meanwhile, not all information in a predefined surrounding region (rectangle region in the second image) is beneficial for their final parsing. Information collected in the semantic correlated region is more helpful to identify the object while that in an irrelevant region, though spatially close, may result in parsing error and hence should be suppressed or even disregarded. For pixel A in Figure 4.1, information of pixels belonging to the lake and its shore, which are semantic correlated, might be more beneficial than information of other pixels in the predefined fixed window. And for pixel B, the desirable shape of context would align with the train and railroad track. Besides, uniformly integrating smooth global information would degrade the location identity and the local discriminative features [171]. Therefore, for better scene parsing, diverse shapes of semantic consistent context should be customized. Most existing methods tend to model statistical average representation among a fixed rectangle region [3, 4, 55, 56] or the global region [63, 87, 161]. In this work, by taking into account the semantic correlation and the shape layout of objects, we propose a shape-variant context model to aggregate the surrounding information of each pixel from their semantic-correlated region inferred according to the appearance of input image.

To this end, we propose first to learn the relation between a target pixel and others by a novel paired convolution followed by a Gaussian mapping function. The learnt network produces higher value for two pixels with stronger semantic correlation and lower value for weaker correlation. Thus, the proposed network will generate a shape mask indicating a semantic correlated region for each pixel. With the shape mask specifying the size and shape of desirable receptive field, we further propose a shape-variant convolution to aggregate context from semantic-dependent region. The shape-variant convolution is specified by a set of learnable location-invariant convolution parameters and the location-variant shape masks. Thus, the parameters are dependent variables of the semantic correlated region of input image, which change with varying shapes and scales of objects. Furthermore, since the shape-variant context is implicitly scale-variant, we can model not only multi-shape but also multi-scale information in a single layer instead of paralleled [3, 4, 13] or stacked [55, 56] multi-layers. From a macro perspective, the proposed

approach helps control the information flow within network through learning the semantic and spatial relationship of features and determine the information passing or suppression.

The proposed scale- and shape-variant context model enhances the discriminative power of the high level features. Higher level features are more robust to noise than lower level features at a price of lower spatial position sensitivity. Thus, many segmentation networks also aggregate low-level features to improve the position accuracy of the segmentation [45, 56, 74, 169]. However, aggregating low-level features, though helps to recover spatial information, may bring some debatable noise sensitive information that leads to a wrong classification of some pixels. Thus, we propose a model that utilizes the higher level features to attenuate the noisy information of the low-level features before aggregating them, i.e. signals denoising. In such a way, the network could better exploit the advantages of low-level features by alleviating their problems.

In summary, this chapter makes the following contributions: 1) we propose a novel paired convolution to infer the semantic correlation of two pixels and based on that to generate a semantic-correlated region for each pixel; 2) we propose a shape-variant context aggregated within the semantic-correlated region to model the diverse shapes and scales of contexts, which greatly enhances the modeling ability of network; 3) we propose a labeling denoising model to reduce the labeling errors caused by the noisy low-level features; 4) we achieve new state-of-the-art performance consistently on six public semantic segmentation datasets.

4.2 The Proposed Approach

4.2.1 Semantic Correlation Dependent Context

Semantic Segmentation needs to simultaneously deal with object recognition and localization, and hence should build the dense feature connections among large region as well as retain the location identity. Meanwhile, due to the diverse shapes and complicated layout of objects in scene images, the scales and shapes of contexts for different objects are supposed to have very large variation.

FIGURE 4.2: (Best viewed in color) We propose a novel semantic correlation dependent shape-variant context, which boosts the semantic-correlated features (magenta color) while suppresses the others (white color).

Many existing context modeling methods tend to aggregate surrounding information with a fixed size of rectangle window across all locations, which weaken the location identity and might not be able to effectively represent the diverse shapes and scales of objects in scene images. Different from previous works, we put forward that the more desirable context region should be shape-variant according to the shape of the object and its background that support the object. For example, for pixels belong to the train in Figure 4.2, the more beneficial context should be the surrounding information along the railroad track (magenta color), which are closer in semantics than in space. In a word, for different pixel positions, surrounding information should be collected from semantic correlated region that supports the existence of the correct class of this pixel. Therefore, in this work, we propose a semantic correlation dependent shape-variant context (SVC) to model diverse shape/scale contexts with location identity. In the SVC, context aggregation is controlled by a semantic correlation mask, specifying where the information should be collected with what extent. With the semantic mask, features in semantic correlated regions are boosted and other irrelevant ones are suppressed. Thus, better context information for parsing of each pixel is aggregated within the specific shape region that supports the existence of the correct class of the object.

Learn the Semantic Correlation

The shape masks that represent semantic correlation trim the context shapes and decide where the information should be collected with what extent. Next we discuss how to learn the semantic correlation, i.e. how to generate the semantic shape

FIGURE 4.3: (Best viewed in color) A shape mask is inferred by the Paired Convolution and Gaussian mapping function φ , which are designed to learn the semantic correlation between the target pixel and other pixels within the shape mask. Here we show an example of 4 values in a 11×11 mask of the target pixel (dark), in which 4 mask values are generated by the 4 filters of the same color.

masks. Each value in a shape mask represents the correlation of the corresponding pixel to the target pixel (the center pixel of the mask). Thus, the semantic relationship of each pixel with the target pixel need be learned and injected to the corresponding position in the shape mask. To this end, we introduce a paired convolution, as shown in Figure 4.3, in which a pair of 3×3 local convolution with specific relative position are employed to learn the semantic and spatial correlation of the corresponding pixel with the target pixel (the central dark pixel in Figure 4.3). In each filter of the paired convolution, there are a central convolution for the target pixel and another convolution whose position corresponds to the position in shape mask for the corresponding pixel. We have observed and hence assumed that the feature appearances of pixels belonging to the same object and its context will show strong correlation because they frequently coexist in the training images. Therefore, the difference of the two convolution outputs $\mathcal{D}_{m,n}^{i,j}$ can be minimized for two pixels belonging to the same object and its context by learning the convolution parameters from the training images.

$$\mathcal{D}_{m,n}^{i,j} = \mathcal{F}^{i,j} * \Theta_{0,0}^{m,n} - \mathcal{F}^{i-m,j-n} * \Theta_{m,n}^{m,n} \quad (4.1)$$

where $*$ donates the local convolution operator, $\mathcal{D}_{m,n}^{i,j}$ represents the convolution output discrepancy of $(i-m, j-n)$ to target position (i, j) , $\mathcal{F}^{i,j}$ and $\mathcal{F}^{i-m,j-n}$ are local features at position (i, j) and $(i-m, j-m)$, $\Theta_{0,0}^{m,n}$ and $\Theta_{m,n}^{m,n}$ are corresponding parameters of two local kernels in the paired convolution. As there might be negative or positive difference, we map it to the value of our context shape mask by a Gaussian function:

$$\mathcal{M}_{m,n}^{i,j} = \varphi(\mathcal{D}_{m,n}^{i,j}) \quad (4.2)$$

FIGURE 4.4: Semantic correlation-dependent shape-variant context aggregates surrounding information according to the semantic correlation and hence customizes an effective contextual region. It helps control the information flow within network via deciding what information to be passed or suppressed.

where $\varphi(a) = \exp(-\frac{a^2}{\sigma^2})$, which maps the convolution output discrepancy to a semantic correlation value. A smaller discrepancy generates a higher semantic correlation value. $\mathcal{M}_{m,n}^{i,j}$ is the mask value at position (m, n) in the semantic shape mask of pixel (i, j) . Note that the result is not sensitive to σ as the parameters of the two convolutions are learnt. We use $\sigma = 3$ in our experiments.

Shape-Variant Context

The goal of shape-variant context is to customize a desirable shape/scale of context for each pixel instead of a simple smooth context information. To achieve this, we further propose a shape-variant convolution (SV Conv) to adaptively collect the surrounding information. The parameters of shape-variant convolution consists of location-invariant learnable convolution parameters and semantic shape mask inferred by the proposed paired convolution. The shape mask is used to control the receptive field of the convolution process for each position according to the semantic correlation. Such shape mask crops the convolution kernels into different shapes/scales and leads to a shape-variant convolution operation. In such a way, the proposed method greatly enhances the network modeling ability of diverse shape context.

The proposed shape-variant context is shown in Figure 4.4. There are two branches, the bypass is designed to learn the semantic correlation, whose outputs are then input to the shape-variant convolution (SV Conv) to provide the semantic shape mask. In detail, the side branch employs paired convolution to learn from the local feature $\mathbb{F}$ from a pre-trained CNN the correlations of each pixel with each of all other pixels within the kernel of size $K \times K$ centered at this pixel, as described in Eq. (4.1) and Eq. (4.2). The number of the output channels is $S=K \times K$ where $K \times K$ is the kernel size of the proposed shape-variant convolution. The semantic shape masks inferred from input features are employed to weight the normal learnable convolutions (F filters) parameterized by $\theta_{m,n}^{d,f}$ of the main branch by:

$$\hat{\theta}_{m,n}^{i,j,d,f} = \mathcal{M}_{m,n}^{i,j} \theta_{m,n}^{d,f} \quad (4.3)$$

where $\theta_{m,n}^{d,f} \in \Theta$ is the convolution parameter for d^{th} input channel at position (m, n) of the f^{th} normal location-invariant learnable filter, $d \in (1, 2, \dots, D)$ and $f \in (1, 2, \dots, F)$. The filter kernel size is $K \times K$ and (i, j) is the index of feature map position across all the $H \times W$ positions. By Eq. (3), the receptive field of the F normal learnable convolutions parameterized by $\theta_{m,n}^{d,f}$ is transformed from a fixed size of $K \times K$ to effectively different sizes and shapes for different pixels (i, j) determined by the proposed semantic shape masks $\mathcal{M}_{m,n}^{i,j}$. The resulting F shape-variant filters are employed to generate variegated shape context for each spatial position (i, j) :

$$\hat{\mathcal{F}}^{i,j,f} = \sum_{d=1}^D \sum_{m=-K_1}^{K_1} \sum_{n=-K_1}^{K_1} \hat{\theta}_{m,n}^{i,j,d,f} \mathcal{F}^{i-m,j-n,d} \quad (4.4)$$

where $K_1 = (K - 1)/2$ and $\mathcal{F}^{i-m,j-n,d} \in \mathbb{F}$. $\hat{\mathcal{F}}^{i,j,f} \in \hat{\mathbb{F}}$ is one target contextual feature map. In such a way, $\mathcal{M}_{m,n}^{i,j}$ instructs how to collect $\mathcal{F}^{i-m,j-n}$ for $\hat{\mathcal{F}}^{i,j}$ during convolution. All of these functions are differentiable and its back-propagation is easy to derive.

The standard convolutional operation is location invariant and do not vary with testing images after training. Thus, it could not customize different shapes/scales of context information for different objects of input images. The proposed SV Conv consists of a learnable location-invariant convolution and a location-variant semantic shape mask inferred from the input image. The former is to model the statistical average of the spatial-channel distribution and the latter is to determine

the size and shape of the convolution receptive field. They together function as a shape-variant operator to better model the shaped context.

Modeling Diverse Shapes in a Single Layer

Due to the diverse shapes/scales of contextual regions and the shape constraint of convolutional kernels, it is difficult to use a single normal convolutional layer to effectively model shape-variant context because the scope of the context, including its scale and contour, dramatically changes for different objects of input images. With the proposed shape-variant semantic correlation masks, convolution regions with diverse shapes and scales are specified and thus we could model multi-shape and multi-scale information in a single layer.

Comparison with State-of-the-art Context Models

Different from the previous context methods that tend to model statistical average representation within a predefined rectangle region, e.g., [3, 4, 13, 55, 56, 69], the proposed approach utilizes the semantic correlation and intentionally picks up the relevant information according to the semantic shape mask inferred by the feature appearance of testing image. Thus, it could not only retain the shape and location identity but also effectively build the beneficial connection among correlated features for classification. Comparing with the deformable convolution [70], the objective of the proposed approach is similar to it in terms of collecting the relevant information in the convolution. However, different from [70] that tries to achieve this via deforming the sampling locations, the proposed SVC finds out the semantic correlation to enhance or attenuate the corresponding information, explicitly leading to the shape- and scale-variant modeling. The criteria or the methods to find the relevant information in the two approaches are also different. Furthermore, our method models diverse shapes of semantic-dependent context in a single layer without stacking layers in [70], and avoids the “atrous” in deformable convolution that may lose some detailed information.

4.2.2 Labeling Denoising

Due to the label variety and the complicated correlation among labels in segmentation datasets, regular errors can be found in the results of most state-of-the-arts

FIGURE 4.5: In-context error, e.g., the first row, refers to incorrect labeling within the label set of the image. Out-context error, e.g., the second row, refers to incorrect labeling outside the label set of the image.

segmentation networks [3, 13, 88, 123, 169]. These regular errors could be categorized into “in-context” error and “out-context” error, as show in Figure 4.5. The “in-context” error is mainly caused by inaccurate positioning and inter-context influence while the “out-context” error is mainly caused by inaccurate classification. The proposed shape-variant context aggregates information from specific semantic correlated region, which helps mitigate the in-context and out-context labeling errors. To get elaborate spatial information, lower-level features from middle layers of CNNs are important in the encoder-decoder architecture [45, 56, 74, 169] as they contain more information about where these objects are [45, 147]. But these low-level features also bring debatable noisy information that results in out-context errors. In contrast, the high-level features, e.g., the shape-variant context in this work, though less sensitive to the spatial location, are more robust to noise and more aware of what categories existing in a scene image. To better combine “what” and “where”, we propose a labeling denoising (LD) model in this work that attenuates the noise information when extracting low-level spatial information from middle layers.

The labeling denoising model first infers the existence potential (i.e., the existence probability) of each category from a higher-level block and learns penalty scores from the existence potentials. Then, the score maps generated from a lower-level block are charged by the penalty scores. Using the penalty scores learnt from the

higher-level block, the scores of nonexistent categories of an input image generated by the lower-level block are greatly suppressed. First, the existence potential is inferred by the score maps from a higher-level block by:

$$\mathcal{E}_k = F_g(F_{sf}(\mathcal{S}_k)) \quad (4.5)$$

where $\mathcal{S}_k$ is the score maps from a higher-level (level k) block, F_{sf} is softmax and F_g is global max pooling. $\mathcal{E}_k = (e_k^1, \dots, e_k^c, \dots, e_k^C)$ and e_k^c is the existent potential for class c inferred by level k . Then, the penalty $\mathcal{P}^c$ is learnt by:

$$\mathcal{P}_k^c = \text{ReLU}(\mathcal{T} - e_k^c) \Delta_k^c \quad (4.6)$$

where $\mathcal{T}$ is a penalty threshold and Δ_k^c is a learnable penalty parameter. The penalty threshold and function ReLU are used to keep the score distribution of existent classes unchanged. The penalty $\mathcal{P}_k^c$ is used to modify the score map of its next lower-level block before aggregating it into the upsampled score map of its next higher-level block as:

$$\mathcal{S}_{k-1}^c = \text{ReLU}(\hat{\mathcal{S}}_{k-1}^c - \mathcal{P}_k^c) + \mathcal{S}_k^c \quad (4.7)$$

where $\hat{\mathcal{S}}_{k-1}^c$ is the score map of class c directly from the lower-level block. $\mathcal{S}_{k-1}^c$ is the denoised and aggregated score map from the highest level to the level $k-1$, which is further used to modify and be aggregated to the score maps of the lower-level blocks as Eqs (5), (6), and (7). The proposed labeling denoising (LD) model is shown in Figure 4.6. In such a way the network could take advantage of both high-level features and low-level features, i.e. better combine “what” and “where”. For the skip layers from low-level features, the scores of nonexistent categories in an input image are attenuated and those of the existent categories are retained and supplemented to score maps for positioning enhancement. As this mechanism is included in the end-to-end training process, less noisy scores are taken into account during training, and gradients for training such noisy information could be saved for other things such as positioning. The proposed approach could also be viewed as some kind of dropout, which applies dropout to connections that reach some conditions.

Methods	PASCAL-Context	COCO-Stuff
Baseline	42.7	31.5
Baseline+SVC	52.4	38.5
Baseline+SVC+LD	53.2	39.6

TABLE 4.1: Ablation study of the proposed approach in terms of IoU.

Kernel Size	SFC	SVC
0×0	42.7	42.7
7×7	45.6	48.5
11×11	46.6	49.4
15×15	47.1	51.2
19×19	47.0	52.1
23×23	46.7	52.4
27×27	46.5	52.3

TABLE 4.2: Ablation study of the proposed shape-variant context (SVC) approach by comparing it with the shape-fixed context (SFC) on different kernel size. It also shows that the performance gain is not simply brought by the increased number of parameters.

4.3.1 Ablation Study

In this section we do ablation studies of the proposed shape-variant context (SVC) and labeling denoising (LD). As shown in Table 4.1, comparing the performance gain brought by the proposed LD on PASCAL-Context (59 classes) and COCO-Stuff (171 classes), we can conclude that the LD could mitigate noisy prediction and it works better on datasets with more semantic categories. This is not a surprise because more categories cause heavier prediction noise and hence the LD works more effectively. Table 4.1 shows the significant performance gain (7 percent on COCO-Stuff and almost 10 percent on PASCAL-Context) from the baseline by applying the proposed SVC.

To further study where the performance gain of the proposed SVC comes from, we compare it with the shape-fixed context (SFC) that is implemented by setting a constant shape mask, i.e., $\mathcal{M}_{m,n}^{i,j}=1$ in Eq. (4.3). We compare them on PASCAL-Context with different kernel sizes shown in Table 4.2, where the 0×0 means the baseline without the context layer. As we employ just a single layer to capture the context information, all kernels used in Table 4.2 are larger than convolutional

FIGURE 4.7: Four visual examples of the shape-variant masks $\mathcal{M}_{m,n}^{i,j}$ generated at four different locations of two testing images by the learned network. The mask center (i, j) is indicated by the small square and its value at (m, n) within the image is shown by the gray level.

kernels of most other work so that diverse shapes and scales of contextual information could be modelled in a single layer. Table 4.2 shows that with the increase of kernel size, the segmentation performance improves up to a certain level then slightly drops with further increase of the kernel size. This is because the network lose too much locality information in overlarge kernel situation. It also shows that a simple increase of the network parameters may not always improve the performance. Table 4.2 shows that the proposed SVC significantly outperforms the SFC at all different kernel sizes. It is also not a surprise that the best performance of the SVC is achieved at the kernel size (23×23) , much larger than that of the SFC (15×15) because the proposed SVC provides diverse shape context, which is implicitly multi-scale with better location identity.

4.3.2 Visualization of the semantic shape mask

As Table 4.1 and Table 4.2 show the significant performance gains brought by the proposed shape-variant context (SVC) that is determined by the proposed shape mask $\mathcal{M}_{m,n}^{i,j}$, it is worthwhile to further study how the mask captures the shape of the context by visualizing it. Four examples of the shape mask $\mathcal{M}_{m,n}^{i,j}$ generated at four different locations of two testing images by the learned network are shown

Methods	pixel acc.	mean acc.	mean IoU
FCN [34]	52.0	34.0	22.7
DeepLab [66]	57.8	38.1	26.9
FCN-8s [45]	60.4	38.5	27.2
DAG-RNN+CRF [75]	63.0	42.8	31.2
DC+FCN+ [160]	65.5	44.6	33.6
Deeplab-V2 [3]	65.1	45.5	34.4
CCL-ResNet101 [56]	66.3	48.8	35.7
DSSPN [166]	68.5	48.1	36.2
SVCNet (ours)	69.2	51.5	39.6

TABLE 4.3: COCO-Stuff testing accuracies.

in Figure 4.7. The first is the mask of the center of a car. It has higher values at pixels of cars and road as they contains the context information of the center of the car. The second is the mask of some pixel of the cow in the middle of a road. It has higher values at pixels of cows and grass, though they are far away from the target cow and separated by the road. Values of the second mask are low in the area of road as it does not show correlation with the cow in the training database. Consistently, the semantic correlation of the context is also shown by the third and fourth masks respectively for a train and a railway in the second testing image.

4.3.3 Comparison with the State-of-the-Arts

The proposed semantic segmentation network is named as SVCNet and we compare it with the state-of-the-arts on six public benchmark, COCO-Stuff, SIFT-Flow, CamVid, PASCAL-Person-Part, PASCAL-Context, and Cityscapes. Before the quantitative comparison, some qualitative results of the proposed SVCNet are shown in Figure 4.8.

COCO-Stuff [34] provides dense pixel-wise annotations for 171 semantic categories. There are 9000 images used for training and 1000 images used for testing. Quantitative results of COCO-Stuff are shown in Table 4.3. The proposed SVCNet outperforms the previous state-of-the-arts across all evaluation metrics.

SIFT-Flow [38] contains 2688 images annotated with 33 semantic classes. There are 2488 training images and 200 testing images. Quantitative results are shown

Methods	pixel acc.	mean acc.	mean IoU
Liu et al. [79]	76.7	-	-
Tighe et al. [81]	75.6	41.1	-
Farabet et al. [71]	78.5	29.6	-
Pinheiro et al. [73]	77.7	29.8	-
Sharma et al. [62]	79.6	33.6	-
Yang et al. [80]	79.8	48.7	-
FCN-8s [10]	85.9	53.9	41.2
DAG-RNN+CRF [75]	87.8	57.8	44.8
Piecewise [68]	88.1	53.4	44.9
SVCNet (ours)	89.1	58.2	46.3

TABLE 4.4: SIFT-Flow testing accuracies.

Methods	mean IoU
DeconvNet [48]	48.9
SegNet [49]	50.2
FCN-8s [45]	52.0
DeepLab [66]	54.7
DilatedNet [55]	65.3
Dilation+FSO [173]	66.1
FC-DenseNet103 [174]	66.9
G-FRNet [175]	68.0
Dense Decoder [176]	70.9
SVCNet (ours)	73.4

TABLE 4.5: CamVid testing accuracies.

in Table 4.4. The proposed SVCNet outperforms previous state-of-the-arts across all evaluation metrics.

CamVid [36] is a road scene image segmentation dataset which provides dense pixel-wise annotations for 11 semantic categories. There are 367 training images, 101 validation images and 233 testing images. The testing results are shown in Table 4.5. It shows that the proposed SVCNet outperforms previous state-of-the-arts by a large margin.

PASCAL-Person-Part [39] provides pixel-level labels for six person parts. There are 1717 training/validation images and 1818 testing images. Quantitative results of PASCAL-Person-Part are reported in Table 4.6. It shows that the proposed SVCNet outperforms the previous state-of-the-arts by a large margin on this small

Methods	mean IoU
Attention [145]	56.4
HAZN [177]	57.5
LG-LSTM [85]	58.0
DeepLab [66]	62.8
DeepLab-V2 [3]	64.9
RefineNet-ResNet101 [123]	68.6
Dense Decoder [176]	68.6
SVCNet (ours)	73.9

TABLE 4.6: PASCAL-Person-Part testing accuracies.

Methods	mean IoU
FCN-8s [10]	39.1
CRF-RNN [67]	39.3
BoxSup [154]	40.5
HO-CRF [156]	41.3
Piecewise [68]	43.3
FCRN [158]	44.5
EFCN [159]	45.0
DeepLab-V2[3]	45.7
Global-Context[161]	46.5
RefineNet-ResNet152 [123]	47.3
DenseDecoder [176]	47.8
MSCI [178]	50.3
CCL-ResNet101 [56]	51.6
EncNet [63]	51.7
SVCNet (ours)	53.2

TABLE 4.7: PASCAL-Context testing accuracies.

dataset, which indicates that the the proposed approach could be trained very well even on small dataset.

PASCAL-Context [33] provides pixel-wise segmentation annotation for 59 classes. There are 4998 training images and 5105 testing images. Quantitative results of Pascal Context are shown in Table 4.7. It shows that the proposed SVCNet outperforms the state-of-the-arts by a large margin.

Cityscapes [22] contains 5000 street scene images with pixel-level fine annotations and 19 classes are considered for evaluation. There are 2975 training images, 500

Methods	mean IoU
Deeplab-v2 [3]	70.4
RefineNet-Res101 [123]	73.6
DSSPN-Universal [166]	76.6
GCN [69]	76.9
DepthSet [179]	78.2
PSPNet [13]	78.4
AAF [180]	79.1
DFN [87]	79.3
PSANet [181]	80.1
DenseASPP-DenseNet161 [4]	80.6
SVCNet (ours)	81.0

TABLE 4.8: Cityscapes testing accuracies.

validation images and 1525 testing images. The test results are shown in Table 4.8.

4.4 Chapter Summary

In this work, we propose to aggregate the context information based on the semantic correlation rather than the predefined spatial-dependent window to collect more effective and discriminative surrounding information for semantic segmentation. The semantic-correlated information even at a far away spatial location will be enhanced and the semantic-uncorrelated information even at a close spatial location will be suppressed in collecting the context information. To this end, we first propose a novel paired convolution to learn the feature semantic-correlation from the training images and to infer it of the query image. This generates a semantic shape mask at each position of the image. Based on it, we propose a shape-variant convolution, in which the receptive field of the convolution is specified by different semantic shape masks at different positions of different query images. The semantic shape masks form diverse scales and shapes of the convolution receptive field to aggregate discriminative context information effectively. Furthermore, to ease the labeling errors, we propose a labeling denoising model, which utilizes more robust higher-level features to attenuate the prediction errors caused by noisier lower-level features. Without bells and whistles, the proposed segmentation network achieves new state-of-the-arts consistently on the six public semantic segmentation datasets,

FIGURE 4.8: Qualitative segmentation examples on COCO-Stuff (1st-4th rows) and PASCAL-Context (5th-7th rows).

COCO-Stuff, SIFT-Flow, CamVid, PASCAL-Person-Part, PASCAL-Context, and Cityscapes.

Chapter 5

Boundary-Aware Feature Propagation

Boundary information could be used for improving segmentation performance and is essential for spatial details in dense prediction. In this chapter, we explore to detect boundaries in scene images and leverage the learned boundary information to boost parsing results. To increase the feature similarity of the same object while keeping the feature discrimination of different objects, we explore to propagate information throughout the image under the control of objects' boundaries. To this end, we first propose to learn the boundary as an additional semantic class to enable the network to be aware of the boundary layout. Then, we propose unidirectional acyclic graphs (UAGs) to model the function of undirected cyclic graphs (UCGs), which structurize the image via building graphic pixel-by-pixel connections, in an efficient and effective way. Furthermore, we propose a boundary-aware feature propagation (BFP) module to harvest and propagate the local features within their regions isolated by the learned boundaries in the UAG-structured image. The proposed BFP is capable of splitting the feature propagation into a set of semantic groups via building strong connections among the same segment region but weak connections between different segment regions. Without bells and whistles, our approach achieves new state-of-the-art segmentation performance on three challenging semantic segmentation datasets, i.e., PASCAL-Context, CamVid, and Cityscapes.

FIGURE 5.1: (Best viewed in color) The boundary-aware feature propagation module builds stronger connections within the same segment and weaker connections between different segments, which helps to enhance the similarity of features belonging to the same segment while keeping discrimination of features belonging to different segments.

5.1 Introduction

Scene segmentation is a challenging and fundamental task that aims to assign semantic categories to every pixels of scene images. The key of scene segmentation refers to parsing and segmenting a scene image into a range of semantic coherent regions. Therefore, it is critical to improve the feature similarity of the same object while keeping the feature discrimination of different objects. To this end, on the one hand, we explore to propagate features throughout the images to share features and harvest context information, which is beneficial for improving the feature similarity. On the other hand, in order to keep the discriminative power of features belonging to different objects, we propose to make use of boundary information to control the information flow during propagation progress. In a word, we propose a boundary-aware feature propagation module to build strong connections within the same segment and weak connections between different segments, as shown in Figure 5.1. This module requires two components: boundary detection and graph construction.

First, boundary detection, which is an implicit task in scene segmentation, is important for meticulous dense prediction. However, in existing segmentation methods,

boundary detection did not attract due attention since boundary pixels only account for a small portion of the entire image and it has little contribution to the performance improvement. In this work, we try to find a way to simultaneously achieve segmentation and boundary detection, and further make use of the learned boundary information to enhance the segmentation performance. With regards to this, we propose to generate the boundary label of semantic objects from the existing object class labels given in the segmentation datasets and define it as an additional class for learning and classification. By doing so, concise boundaries are well learnt and inferred as one additional class because the characteristics of pixels on boundary are different from those of most pixels off boundary. And the parsing of pixels among disputed area (i.e., near the boundary) is enhanced. Moreover, taking boundary as an additional class requires little change on network but makes the network be aware of the boundary layout that could be further used for segmentation improvement.

Second, a graphic model is needed to create the order rules for feature propagation. Convolutional methods [3, 55] are popular in scene segmentation, but they usually consume large computation resources when aggregating features from grand range of receptive fields. Moreover, the convolution kernels could not vary with input resolutions, thus cannot ensure a holistic view of the overall image. Recently, DAG-RNN [75] proposes to use four directed acyclic graphs (DAGs) with different directions to model the function of undirected cyclic graphs (UCGs), which structurize the image by building pixel-by-pixel connections throughout the whole image. However, DAGs require lots of loops to scan the images pixel by pixel. Thus it is very slowly even on low-resolution feature maps, which limits its application on “dilated FCN” [3, 13, 63] and on high resolution datasets like Cityscapes [22]. To address this issue, we propose a more efficient graphic model to achieve faster feature propagation. We find that each DAGs adopted by [75] could be alternatively replaced by two Unidirectional Acyclic Graphs (UAGs), in which the pixels of the same row or column are dealt in parallel with 1D convolution. The proposed UAGs greatly speed up the feature propagation process. Moreover, different from the DAGs that are extremely deep, the proposed UAGs are much shallower and thus alleviate the vanishing gradient problem in backpropagation [182].

Finally, based on the UAG-structured image and the learned boundaries information, we build a boundary-aware feature propagation (BFP) module. In the BFP,

local features of the same segment are shared via unimpeded connections to exchange information that achieves feature assimilation, while features of different segments are exchanged under controlled connections with the guidance of learned boundaries. There are several advantages of our proposed boundary-aware feature propagation (BFP) network. First, since the proposed UAGs deal with pixels of the same row or column in parallel, we achieve the propagation process in a high speed. And the UAGs contain much fewer parameters than convolutional methods. Second, as we express boundary detection as classification of a semantic class, lots of parameters and complex module for boundary detection are saved. Third, with the advice of boundary confidence, the local features are propagated in a more motivated way, enhancing the similarity of features belonging to the same segment while keeping the discrimination of features belonging to different segments.

The main contributions of this chapter can be summarized as follows:

- We show that the boundary can be learned as one of semantic categories, which requires little change on network but obtains essential boundary information.
- We propose some unidirectional acyclic graphs (UAGs) to propagate information among high-resolution images with a high speed.
- We propose a boundary-aware feature propagation module to improve the similarity of local features belonging to the same segment region while keeping the discriminative power of features belonging to different segments.
- We achieve new state-of-the-art performance consistently on PASCAL-Context, CamVid, and Cityscapes.

5.2 Approach

Due to the diverse style and complex layout of scene images, it is necessary to classify every pixel using holistic context information but protect its differentiation from overwhelming by global scene. In this respect, we propose a boundary-aware feature propagation module to arm the local features with a holistic view of contextual awareness but keep the discriminative power of features for different objects. The overall architecture is shown in Figure 5.2. We use the dilated FCN (subsampling by 8) based on ResNet-101 [2] as backbone. The supervisor of loss 2 is the

FIGURE 5.2: An overview of the proposed approach. We use the ResNet-101 (CNN) with the dilated network strategy [3] as backbone and the proposed boundary-aware feature propagation (BFP) module is placed on the top of CNN. The supervisor of loss 2 is the new ground truth of $N+1$ classes with an additional boundary class generated from the original ground truth of N classes.

boundary-aware ground truth ($N+1$ classes) that are generated from the original ground truth (N classes).

5.2.1 Semantic Boundary Detection

Boundary delineation is favourable for meticulous scene parsing. However, because of the various semantic labels and complex layout of objects in segmentation datasets, parsing pixels in the boundary area is always difficult and results in confused prediction. In this work, instead of directly assigning semantic labels to pixels in boundary area, we explore to infer the boundary layout first and improve the segmentation performance with the learned boundary information. Lots of works have contributed to boundary detection [96–100], but most of them focus on edges that sketch the objects. Different from them, we only focus on the boundaries of semantic objects that are predefined in segmentation datasets. We have observed that boundaries have the property of dramatic changes in RGB and feature information. And the boundary label is easy to be generated from the existing ground truth. Consequently, we assume that the boundary could be viewed as an additional semantic category and simultaneously learned with other existing object categories. As shown in Figure 5.2, we obtain a new ground truth (Loss 2, $N+1$ classes) from the original ground truth (Loss 1, N classes) and utilize the new ground truth for supervising the network to learn and infer the boundary layout.

Different from previous boundary detection works that aim to boundary delineation only or deal with segmentation and boundary detection as two detached tasks, our proposed semantic boundary detection is embed with semantic object parsing. Our boundary detection module only targets at boundaries of semantic objects predefined in the training data and generate concise boundary information under interaction with segmentation. These two tasks are combined into one, and they benefit each other. By training them together, the scene segmentation classes help suppress the edges within objects that are not semantic boundary of objects, e.g., edges of eyes. Scene segmentation helps boundary detection to filter out noise and delineate well-directed boundary, which means higher responses for desired boundaries while lower responses for noise boundaries. While boundary detection makes the scene segmentation be aware of the important boundary layout information.

FIGURE 5.3: Each point of the DAGs has three different directions. Thus, the DAGs have to scan the image pixel by pixel and consumes lots of time with many loops. We decompose the four DAGs to six Unidirectional Acyclic Graphs (UAGs). Each of UAGs propagates information toward a single direction, which deals with pixels of each row in parallel and then deals with pixels of each column in parallel. For example, the UAG_S is in the south direction only and the UAG_{S,E} is in the east direction only (based on UAG_S).

5.2.2 Unidirectional Acyclic Graphs

Context is designed to aggregate wide range of surrounding information, thus it desires a holistic view of the overall image without regard to resolution. One popular way is to employ stacked convolution or dilated convolution to enlarge the receptive field [3, 55, 56], but this consumes large computation resources. The work of [75] proposes DAG-RNN to capture long-range context based on directed

FIGURE 5.4: (a) Image; (b) Original ground truth; (c) New generated ground truth: add a boundary class generated from the original ground truth; (d) Boundary confidence map: the probability distribution of the boundary layout; (e) Propagation confidence map: the confidence distribution of propagation.

acyclic graphs (DAGs). As shown in Figure 5.3, pixels are locally connected to form a undirected cyclic graph (UCG) to build propagation channels among the whole image. To overcome the loopy property of UCG, the UCG is decomposed to four DAGs with different directions (southeast, southwest, northeast, northwest). However, since each pixel of the DAGs has three different directed connections, the feature propagation based on DAG-structured images has to scan the image pixel by pixel and requires lots of loops. Thus it is very slowly even on low-resolution feature maps, which limits its application on “dilated FCN” [3, 13, 63] and on high resolution datasets like cityscapes [22] and CamVid [36]. To address this issue, we explore to reduce the number of loops and propagate information in parallel. Herein, we propose some Unidirectional Acyclic Graphs (UAGs) as shown in Figure 5.3, which deal with pixels of each row in parallel and then deals with pixels of each column in parallel. Each DAGs adopted by [75] could be alternatively replaced by two UAGs. For example, the DAG_{SE} is decomposed to UAG_S and $UAG_{S,E}$, where UAG_S is south directed that deals with pixels of the same row in parallel and $UAG_{S,E}$ is east directed (after UAG_S) that deals with pixels of the same column in parallel. As a result, the number of loops for each DAGs is reduced from $H \times W$ to $H + W$, where H and W are the height and width of feature maps. The proposed UAGs greatly speed up the feature propagation process, which is economic and desired in practice, especially for applications that require high resolution and big eyeshot (e.g., self-driving vehicle). Moreover, due to the pixel-by-pixel operation, the recursive layers in DAGs are very deep that could be unrolled to thousand layers. This causes the problem of propagation vanish [182]. The proposed UAGs are much shallower than DAGs and thus alleviate the the problem of propagation vanish.

FIGURE 5.5: As our UAGs are unidirectional and in parallel, we show the propagation process in 1D here for clarity. i_t represents the feature of pixel at position t , h_t is output (hidden state), p_t is the propagation confidence.

5.2.3 Boundary-Aware Feature Propagation

However, unselective propagation would make the features assimilated, which results in smooth representation and weakens the features' discrimination. To classify features in different objects and stuff in scene segmentation, it is beneficial to improve the feature similarity of the same object while keeping the feature discrimination of different objects. Therefore, we introduce the boundary information into feature propagation to control the information flow between different segments. As shown in Figure 5.1, with the learned boundary information, we build strong connections for pixels belonging to the same segment but weak connections for different segments. During propagation, more information is passed via strong connections within the same segment and less information flows crossing different segments. In such a way, pixels get more information from other pixels of the same objects and less information from pixels of other objects. Consequently, features of different objects could keep their discrimination while features of the same object trend to be cognate, which is desired for segmentation. The detailed process of proposed boundary-aware feature propagation is presented below.

As our UAGs are unidirectional and in parallel, we formulate the propagation process in 1D here for notation clarity. Extension to 2D/3D is straightforward. We denote the feature of pixel at position t as i_t , and the corresponding output (hidden state) is denoted as h_t . The standard propagation process based on our UAG-structured image is formulated as following:

$$h_t = g(U * i_t + W * h_{t-1} + \delta) \quad (5.1)$$

where $*$ is 1D convolution operation, U, W are learnable parameters of 1D convolution and δ is learnable bias. g is element-wise nonlinear activation function (we use ReLU).

For the boundary-aware propagation, we first extract the boundary confidence map from the $(N + 1)$ segmentation confidence maps, as shown in Figure 5.2. We denote the boundary confidence of pixel t as b_t , corresponding to i_t . Based on the boundary confidence map, we generate the propagation confidence map:

$$p_t = 1 - \beta f(\alpha b_t - \gamma) \quad (5.2)$$

where p_t is the propagation confidence that decides how much the information of pixel t to be passed to next region. $\alpha=20$ and $\gamma=4$ are constant chosen by experience, f is sigmoid function to enhance the boundary, β is a learnable parameter. With the propagation confidence, the propagation process can be reformulated as below:

$$h_t = g(U * i_t + W * h_{t-1} p_{t-1} + \delta) \quad (5.3)$$

as shown in Figure 5.5, the propagation is controlled by the boundary and thus models boundary-aware context feature for better parsing of different segments.

For the UAGs that have “two directions”, they are also unidirectional and in parallel. For example, $\text{UAG}_{S,E}$ is formulated below:

$$h_t^j = g(U * i_t^j + W * h_{t-1}^j p_{t-1}^j + \hat{W} * h_{t-1}^{j-1} p_{t-1}^{j-1} + \delta) \quad (5.4)$$

There are two hidden states, h_{t-1}^j and h_{t-1}^{j-1} , input to the current cell, where t and j are the denotations of horizontal and vertical axis, respectively. Finally, the hidden states of the corresponding positions in four UAGs (i.e., $\text{UAG}_{S,E}$, $\text{UAG}_{S,W}$, $\text{UAG}_{N,E}$, $\text{UAG}_{N,W}$) are fused together to generate the final output, as shown in Figure 5.3.

One example of boundary confidence map and propagation confidence map is shown in Figure 5.4. We learn the boundary confidence map under the supervise of new generated ground truth with additional boundary class. To control the progress of feature propagation, propagation confidence map is generated from boundary confidence map. If pixel i_t is in boundary region, then it has a higher boundary probability b_t and hence a smaller propagation probability p_t . Thus, the feature

propagation is suppressed and weak signals are passed to next pixel. Otherwise, it has a strong propagation to spread its features to next pixel.

5.3 Experiments

5.3.1 Implementation Details

Our Network is implemented based on the public platform Pytorch. We use ResNet-101 [2] with the dilated network strategy [3] (subsampling by 8) as our backbone. In detail, the convolution strides for downsampling in last two blocks are reset to 1 and the convolutions of last two blocks are dilated with dilation rates of 2 and 4 receptively. *Pool 5* and layers after it are discarded. The network is trained with mini-batch, batchsize is set to 12 for PASCAL-Context and 8 for Cityscapes and CamVid. Following deeplab-v2 [3], we use the “poly” learning rate policy, where current learning rate Lr_c depends on the base learning rate Lr_b and iterations: $Lr_c = Lr_b \times (1 - \frac{iter}{total_iter})^{0.9}$. Momentum and weight decay are fixed to 0.9 and 0.0001 respectively. We adopt random horizontal flipping and random resizing between 0.5 and 2 to augment the training data.

Most of the scene segmentation datasets do not provide boundary layout, we use the provided segmentation ground truth to generate boundary-aware ground truth, as shown in Figure 5.4 (c). As we adopt the dilated FCN as our backbone, the spatial size is downsampled 8 times in encoding process. Thus to avoid the loss of boundary information in feature maps with smallest spatial size, pixels with distance smaller than 9 pixels (i.e., trimap of 18 pixels) to boundary are defined as boundary pixels and their ground truth labels are set to $N + 1$, where N is the number of classes in datasets. In our experiments, the over-wide boundary (e.g., trimap of 50 pixels) squeezes small objects and weakens the function of boundary in feature propagation. We evaluate our network with mean Intersection-over-Union (**mIoU**). Mathematical definition of mIoU please refer to [45].

Methods	Input Resolution	# Loops	Time (s)
FCN	480×360	none	0.35
DAGs	480×360	10800	17.92
UAGs	480×360	300	0.47
FCN	960×720	none	0.42
DAGs	960×720	43200	56.97
UAGs	960×720	600	0.76

TABLE 5.1: Inference speed comparison of FCN (baseline), DAGs, and UAGs on dilated ResNet-101 with different resolution inputs.

5.3.2 Efficiency Analysis

To evaluate the speed of the proposed UAGs, we report in Table 5.1 the inference time of UAGs and compare it with DAGs [75] on different resolution of input images, based on dilated FCN (subsampling by 8). The number of loops is also recorded. Different from DAGs that have to scan the image pixel by pixel, the proposed UAGs deal with pixels of each row/column in parallel, hence they save a lot of loops. As shown in Table 5.1, the UAGs contain much fewer loops than DAGs and consequently they are much faster than DAGs. Especially with high resolution (e.g., 960×720), DAGs are very slow and time-consuming. The speed of DAGs is highly related with the input resolution that determines the number of loops, thus DAGs are not suitable of high-resolution datasets (e.g., Cityscapes [22]) and FCN with dilated network strategy [3]. Besides the inference time, training of DAGs based on dilated FCN requires one hundred times more GPU hours than our proposed UAGs, which also shows the high efficiency of our approach. To quantitatively compare the segmentation performances of DAGs and proposed UAGs, we evaluate them on VGG-16 [7] with encoder-decoder strategy, exactly in the same way like that in [75]. DAGs and UAGs achieve almost the same results on PASCAL-Context (UAGs 43.0% vs. DAGs 42.6%). This shows that the proposed UAGs realize the same function as DAGs but with a much faster speed.

5.3.3 Ablation Studies

We show the detailed results of ablation studies of the proposed approach in Table 5.2. The proposed UAGs harvest local features and propagate them throughout

the whole image to achieve a holistic context aggregation, which greatly improves the segmentation performance from the baseline (dilated FCN). Then, based on the UAGs, we learn the boundary information and inject it to the propagation process to control the information flow between different regions. With the boundary information, the UAGs build stronger connections of pixels within the same segment and weaker connections between different segments. Thus, features of the same segment become more similar but features of different segments remain discriminative.

We also visualize some examples of the inferred boundary confidence map in Figure 5.6. As shown in Figure 5.6, the inferred boundary map mainly involves boundaries between the semantic segments predefined in datasets instead of other edge information. Therefore, it contains the desired boundary layout of semantic objects and could be used for control of the feature propagation throughout image. The results in Table 5.2 show the effectiveness of the boundary-aware feature propagation (BFP) network. The column “boundary” represents using the learned boundary map to control the propagation process in UAGs. Following [103, 183], we evaluate the performance of BFP near boundaries, as shown in Figure 5.7. We compute the mIoU for the regions within different bands of boundaries.

DT-EdgeNet [103] is the most related to BFP. However, BFP learns the boundary as one of semantic classes and is trained end-to-end, while DT-EdgeNet learns edge with additional EdgeNet and requires a two-step training process for DeepLab and EdgeNet. As shown in Figure 5.6, Our learned boundaries response less to object interior edges than DT-EdgeNet. BFP is proposed to perform feature propagation that is some kind of contextual feature modeling, while DT is used to refine the segmentation scores. We use the DT to filter the segmentation scores of BFP and this brings 0.7% performance gain, which shows that the DT and BFP are complementary.

5.3.4 Comparison with the State-of-the-Art Works

PASCAL-Context [33] provides pixel-wise segmentation annotation for the whole scene image. There are 4998 training images and 5105 testing images in

FIGURE 5.6: Qualitative examples of inferred boundary map.

PASCAL-Context. Following [33], we use the most common 59 classes for evaluation. Testing accuracies of PASCAL-Context are shown in Table 5.3, which shows that the proposed BFP outperforms the state-of-the-art works by a large margin.

FIGURE 5.7: Segmentation performance within band (trimap) around boundaries.

Methods	Backbone	UAGs	Boundary	MS	mIoU
FCN	ResNet-50				41.2
BFP	ResNet-50	✓			49.8
BFP	ResNet-101	✓			50.8
BFP	ResNet-101	✓	✓		52.8
BFP	ResNet-101	✓	✓	✓	53.6

TABLE 5.2: Ablation Study of Boundary-aware Feature Propagation (BFP) Network on PASCAL-Context. Baseline is dilated FCN and MS means multi-scale testing.

CamVid [36] is a road scene image segmentation dataset which provides dense pixel-wise annotations for 11 semantic categories. There are 367 training images, 101 validation images and 233 testing images. The testing results are shown in Table 5.4. It shows again that the proposed BFP outperforms previous state-of-the-arts by a large margin.

Cityscapes [22] is a recent street scene dataset which contains 5000 high-resolution (1024×2048) images with pixel-level fine annotations. There are 2975 training images, 500 validation images and 1525 testing images. 19 classes (e.g., roads, bicycles and cars) are considered for evaluation on the testing sever provided by the organizers. The Category-wise results are shown in Table 5.5. Our BFP is only

Methods	mIoU
O2P [152]	18.1
FCN-8s [10]	39.1
BoxSup [154]	40.5
HO-CRF [156]	41.3
PixelNet [157]	41.4
DAG-RNN [75]	43.7
EFCN [159]	45.0
DeepLab-v2+CRF [3]	45.7
RefineNet [123]	47.3
MSCI [184]	50.3
CCL+GMA [56]	51.6
EncNet [63]	51.7
CGBNet [170]	53.4
BFP (ours)	53.6

TABLE 5.3: Testing accuracies on PASCAL-Context.

Methods	mIoU
DeconvNet [48]	48.9
SegNet [49]	50.2
FCN-8s [45]	52.0
DeepLab [66]	54.7
DilatedNet [55]	65.3
Dilation+FSO [173]	66.1
G-FRNet [175]	68.0
Dense-Decoder [176]	70.9
BFP (ours)	74.1

TABLE 5.4: Testing accuracies on CamVid.

trained on fine annotations, while [169] also uses coarse annotations for training. Some segmentation examples on Cityscapes are shown in Figure 5.8.

5.4 Chapter Summary

In this chapter, we address the challenging issue of scene segmentation. In order to improve the feature similarity of the same segment while keeping the feature discrimination of different segments, we explore to propagate features throughout

FIGURE 5.8: Qualitative segmentation examples on Cityscapes.

the image under the control of inferred boundaries. Towards to this, we first propose to learn the boundary as an additional semantic class to enable the network to be aware of the boundary layout. Then, in order to structurize the image to define the order rules for feature propagation, we propose some unidirectional acyclic graphs (UAGs) to model the function of undirected cyclic graphs (UCGs) in a much more efficient way than DAGs. Based on the proposed UAGs, holistic context is aggregated via harvesting and propagating the local features throughout the whole image efficiently. Finally, we propose a boundary-aware feature propagation (BFP) network to detect and utilize the boundary information for controlling

the feature propagation of the UAG-structured image. The proposed BFP is capable of improving the similarity of local features belonging to the same segment region while keeping the discriminative power of features belonging to different segments. We evaluate the proposed boundary-aware feature propagation network on three challenging semantic segmentation datasets, PASCAL-Context, CamVid, and Cityscapes, which show that the proposed BFP achieves new state-of-the-art segmentation performance consistently.

Methods	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle	mIoU
CRF-RNN [67]	96.3	73.9	88.2	47.6	41.3	35.2	49.5	59.7	90.6	66.1	93.5	70.4	34.7	90.1	39.2	57.5	55.4	43.9	54.6	62.5
FCN [10]	97.4	78.4	89.2	34.9	44.2	47.4	60.1	65.0	91.4	69.3	93.9	77.1	51.4	92.6	35.3	48.6	46.5	51.6	66.8	65.3
SiCNN [167]	96.3	76.8	88.8	40.0	45.4	50.1	63.3	69.6	90.6	67.1	92.2	77.6	55.9	90.1	39.2	51.3	44.4	54.4	66.1	66.3
DPN [60]	97.5	78.5	89.5	40.4	45.9	51.1	56.8	65.3	91.5	69.4	94.5	77.5	54.2	92.5	44.5	53.4	49.9	52.1	64.8	66.8
LRR [147]	97.7	79.9	90.7	44.4	48.6	58.6	68.2	72.0	92.5	69.3	94.7	81.6	60.0	94.0	43.6	56.8	47.2	54.8	69.7	69.7
DeepLabv2 [3]	97.9	81.3	90.3	48.8	47.4	49.6	57.9	67.3	91.9	69.4	94.2	79.8	59.8	93.7	56.5	67.5	57.5	57.7	68.8	70.4
Piecewise [68]	97.9	81.3	90.3	48.8	47.4	49.6	57.9	67.3	91.9	69.4	94.2	79.8	59.8	93.7	56.5	67.5	57.5	57.7	68.8	71.6
RefineNet [123]	98.2	83.3	91.3	47.8	50.4	56.1	66.9	71.3	92.3	70.3	94.8	80.9	63.3	94.5	64.6	76.1	64.3	62.2	70.0	73.6
GCN [185]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	76.9
DUC [168]	98.5	85.5	92.8	58.6	55.5	65.0	73.5	77.9	93.3	72.0	95.2	84.8	68.5	95.4	70.9	78.8	68.7	65.9	73.8	77.6
DepthSet [179]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	78.2
ResNet-38 [186]	98.5	85.7	93.1	55.5	59.1	67.1	74.8	78.7	93.7	72.6	95.5	86.6	69.2	95.7	64.5	78.8	74.1	69.0	76.7	78.4
PSPNet [13]	98.6	86.2	92.9	50.8	58.8	64.0	75.6	79.0	93.4	72.3	95.4	86.5	71.3	95.9	68.2	79.5	73.8	69.5	77.2	78.4
AAF [187]	98.5	85.6	93.0	53.8	59.0	65.9	75.0	78.4	93.7	72.4	95.6	86.4	70.5	95.9	73.9	82.7	76.9	68.7	76.4	79.1
DFN [87]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	79.3
PSANet [188]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	80.1
DenseASPP [4]	98.7	87.1	93.4	60.7	62.7	65.6	74.6	78.5	93.6	72.5	95.4	86.2	71.9	96.0	78.090.3	80.7	69.7	76.8	80.6	80.6
SVCNet [12]	98.6	86.8	93.2	57.4	62.7	68.1	76.6	80.9	93.6	72.3	95.5	86.7	72.1	96.0	77.3	89.3	85.9	68.3	77.2	81.0
CGNet [170]	98.5	87.1	93.5	54.6	62.8	70.3	77.981.793.9	73.095.887.5	73.095.887.5	73.495.3	74.7	89.7	84.9	70.878.3	81.2					
BFP (ours)	98.7	87.0	93.5	59.8	63.4	68.9	76.8	80.9	93.7	72.8	95.5	87.0	72.1	96.0	77.6	89.0	86.9	69.2	77.6	81.4

TABLE 5.5: Category-wise performance comparison on the Cityscapes test set. Note that the DenseASPP [4] uses stronger backbone DenseNet-161 [5] than Resnet-101 [2] we adopt as backbone.

Chapter 6

PhraseClick: Integration of “What” and “Where” for Interactive Segmentation

Most of scene parsing models cannot correctly segment the unknown/unseen objects, which have never appeared in the training data. However, when comes to real applications, different sets of semantic categories are required for different scenarios, for example, “pillow” and “curtain” for indoor scene analysis, “traffic light” and “traffic sign” for road scene parsing. It is impractical to annotate a benchmark that contains all the possible object categories in the world. Therefore, the ability to segment any unknown/unseen objects is required. Besides, it is well known that large-scale benchmarks are desired for deep-learning-based scene parsing approaches. However, it is time-consuming and resource-expensive to annotate datasets for scene parsing. Therefore, auxiliary annotation tools are demanded to speed up the annotation process. This motivates us to build interactive image segmentation model, which aims to segment the image according to user interactive inputs. The interactive segmentation model responses to user’s interactions instead of pre-defined class labels, thus we can select any objects with such model. Interactive segmentation can be used as a dataset annotation tool, which greatly speed up the pixel-level annotation process. What’s more, with interactive segmentation, user can select any target-of-interest pieces by providing interactions.

Existing interactive object segmentation methods mainly take spatial interactions such as bounding boxes or clicks as input. However, these interactions do not contain information about conspicuous attributes of the target-of-interest and thus cannot quickly specify what the selected object exactly is, especially when there are diverse scales of candidate objects or the target-of-interest contains multiple objects. Therefore, excessive user interactions are often required to reach desirable results. On the other hand, in existing approaches attribute information of objects is often not well utilized in interactive segmentation. We propose to employ phrase expressions as another interaction input to infer the attributes of target object. In this way, we can 1) leverage spatial clicks to locate the target object and 2) utilize semantic phrases to qualify the attributes of the target object. Specifically, the phrase expressions focus on “what” the target object is and the spatial clicks are in charge of “where” the target object is, which together help to accurately segment the target-of-interest with smaller number of interactions. Moreover, the proposed approach is flexible in terms of interaction modes and can efficiently handle complex scenarios by leveraging the strengths of each type of input. Our multi-modal phrase+clicks approach achieves new state-of-the-art performance on interactive object segmentation.

6.1 Introduction

Interactive object segmentation (or interactive object selection) aims to accurately segment the image into foreground and background given a minimal amount of user interactive inputs. These inputs usually come in the form of user clicks/strokes [27–32] or bounding boxes [23–26]. This form of input gives hard constraints regarding the location of the object of interest. With the recent advances in deep learning such methods can now often select familiar objects with a small amount of input. Alternately, systems have been proposed that instead use language-based input to drive the selection [110–112]. Natural language phrases can be used by a neural network to infer high-level attribute information about what the object of interest looks like that can then be used to select the objects.

While great strides have been made in interactive selection, each of these interaction approaches may still fall short and require additional and excessive user interaction. For example, click-based methods are required to infer the target object given only

FIGURE 6.1: (Best viewed in color) Both the click and phrase inputs have their advantages and disadvantages. Here we show ambiguities that exist when using only click or phrase interactions. We propose to utilize both interaction types to make selection flexible and robust to handle complex scenarios.

spatial constraints and usually are trained to select entire objects. However, the region of user interest may instead be an object part or a combination of multiple objects. For example, the first row in Figure 6.1, the click on the boy may indicate the trousers, the boy, or the whole foreground (boy and chair). This leads to ambiguities that must be overcome with additional user inputs, which directly runs counter to the goal of minimizing user interaction. Click-based methods also generally assume accurate input, but with mobile devices it can be difficult for users to accurately click on objects, especially given that the user’s finger is occluding the object of interest. It still remains a significant challenge to accurately segment a target-of-interest with one or few rough clicks.

On the other hand, language-driven segmentation methods learn the overall appearance of objects and must infer their location. A language phrase naturally and easily overcomes ambiguities such as whether the target is an object, object part, or collection of objects. A phrase can also provide rough spatial information

FIGURE 6.2: The mistakes (circled in red) may be difficult to describe precisely to a phrase-based selection method but are easy to click on for a click-based selection method.

(the woman on the left). However, in many cases an object name and rough location is not sufficient to produce a desired result. For example, in images where there are multiple objects with similar appearance (an image of dozens of cows, see Figure 6.1) it can be very difficult to articulate using language a single target instance. In such cases, directly clicking on the object is much easier for a user. It can also be difficult to articulate some required corrections that don’t correspond to an entire semantically-meaningful region, e.g. the mistakes in Figure 6.2.

The strengths and weaknesses of click-based and phrase-based inputs are complementary with clicks giving hard spatial constraints and phrases giving high-level attribute information. An effective combination of these inputs may reduce the amount of user interactions needed for accurately selecting objects of interest.

Given this observation, we propose to combine click-based and phrase-based interactions within a single neural network to allow more effective user interaction. While we use a convolutional neural network (CNN) to process the input image and clicks, our novel attribute guided feature attention module effectively integrates the phrase into the CNN. Our network uses the click information to locate the objects and leverages the phrase expression to infer the attributes of objects, i.e. the spatial clicks are in charge of “where” the selected object is and the language expressions focus on “what” the selected object is.

The main contributions of this chapter can be summarized as follows:

- To the best of our knowledge, this is the first work to leverage both clicks and phrases for interactive object segmentation. The proposed approach is more flexible in terms of interaction and can better handle complex scenarios via utilizing advantages of these two interactions.
- We propose an attribute guided feature attention module to extract discriminative semantic attribute information from phrase expressions and integrate the attribute and location information into the network.
- Extensive experimental results have demonstrated that phrase expressions are indeed effective at boosting the performance of interactive segmentation, especially in some complex scenarios and with few clicks.

6.2 Approach

6.2.1 Network Architecture

The overall architecture of our approach is shown in Figure 6.3. We use the ResNet-101 [2] based DeepLabv3+ [128] as a backbone architecture. The proposed approach takes both clicks and a phrase as interaction input. The clicks are transformed to distance maps and concatenated with original image to form a 5-channel input for the CNN as in [27]. The phrase is processed by our novel attribute guided feature attention module. First the phrase is encoded by a word-to-vector model and then a bi-directional LSTM extracts the holistic context of phrase. The context information from the phrase and image is then used for attribute prediction to generate attribute attention weights. We utilize the attribute attention to guide the segmentation process for better results. During testing, we employ graph cut [32] as a post-processing tool to refine the final segmentation mask.

6.2.2 Click Interaction

Click-based input is commonly used to provide location information and give spatial constraints about a target of interest. We also use clicks to provide spatial

FIGURE 6.3: The proposed approach takes both clicks and a phrase as interaction input. Clicks are employed to localize the target-of-interest and gradually refine the details, and the phrase is used to infer semantic attributes for quickly discriminating ambiguous candidate objects. It is *not* necessary to enter both types of interactions at the same time. We use the ResNet-101 based DeepLabv3+ as a segmentation architecture for fine tuning. The *Context Encoder* is ASPP+Pooling and the *Decoder* uses skip connections to extract low-level features, as does DeepLabv3+. Graph Cut is employed as a post-processing method during testing.

information about the object which is then supplemented by attribute information from phrases.

Click Input Transformation

Click information is usually introduced at the beginning of a network. As in [27], the user provides a set of sequential clicks to segment the target region. The interactions contain positive clicks $\mathcal{C}^+$ on the target region and negative clicks $\mathcal{C}^-$ on “background”. We employ a Euclidean distance transformation to transform the positive clicks $\mathcal{C}^+$ and negative clicks $\mathcal{C}^-$ to two Euclidean distance maps $\mathcal{D}^+$ and $\mathcal{D}^-$. We concatenate the 3 channels of the input image $\mathcal{I}$ and the two distance maps ($\mathcal{D}^+, \mathcal{D}^-$) to form the 5 channel input:

$$\mathcal{X} = \mathcal{I} \oplus \mathcal{D}^+ \oplus \mathcal{D}^- \quad (6.1)$$

where $\oplus$ denotes concatenation and $\mathcal{X}$ is the 5-channel input to the network. To convert the original CNN [2] to our interactive segmentation network, we modify its first convolution layer to take the 5-channel input $\mathcal{X}$.

Click Simulation Protocol

We follow the simulation strategies proposed in [27] to form a set of synthetic positive and negative clicks. Furthermore, to encourage the model to learn to generate correct prediction from ambiguous clicks, we follow [189] and introduce another sampling strategy that samples clicks on the overlapped foreground and overlapped background of different object masks, for example, the positive and negative clicks in Figure 6.3.

6.2.3 Attribute Feature Attention Module

While clicks as a kind of spatial interaction can indicate the position of target object, such spatial interactions can not conspicuously express the semantic attributes of a target object. Semantic attributes are commonly used to discriminate instance objects of the same class in image captioning [190, 191] and referring expression comprehension [111, 129–132]. To let the network quickly be aware of both the

location and attributes of the target object, besides the spatial interactions, we explore introducing semantic-level interactions for attribute inference.

To introduce semantic-level interactions into the network, we propose an attribute guided feature attention module. CNN features contain rich information like attributes, which are preserved in different feature channels [134]. Given different visual patterns, the largest responses in CNNs are captured in different feature channels. Since phrases are good at describing semantic attributes of objects, we use the input phrase to compute channel-wise attribute attention. We then leverage this attribute attention to emphasize some specific channels and suppress others in order to help the network to determine the object of interest.

To learn the attributes from phrase expression, we first use a word-to-vector model to embed each word $w_t \in \{w_1, \dots, w_T\}$:

$$v_t = \text{word2vec}(w_t) \quad (6.2)$$

Then we employ the bi-directional LSTM (bi-LSTM) to encode the holistic context of phrase expression:

$$\begin{aligned} \vec{h}_t &= \overrightarrow{LSTM}(v_t, \vec{h}_{t-1}) \\ \overleftarrow{h}_t &= \overleftarrow{LSTM}(v_t, \overleftarrow{h}_{t+1}) \end{aligned} \quad (6.3)$$

Besides the context from phrase expression, we also extract channel-wise feature vector from the context feature of the CNN, $f = \text{pooling}(F)$, where *pooling* is a global average pooling operation, F with spatial size of $H \times W$ is the context feature of CNN, f with spatial size of 1×1 is visual attribute vector that preserves visual attributes in channels. Then we concatenate the visual attribute vector and bi-directional hidden vectors to generate final attribute context vector representation:

$$\mathcal{C} = \vec{h} \oplus \overleftarrow{h} \oplus f \quad (6.4)$$

where $\vec{h} = (\vec{h}_1, \dots, \vec{h}_t, \dots, \vec{h}_T)$ and $\overleftarrow{h} = (\overleftarrow{h}_1, \dots, \overleftarrow{h}_t, \dots, \overleftarrow{h}_T)$, and $\mathcal{C}$ is attribute context vector. Next we adapt the attribute context vector to make it have the same number of channels with features in CNN, and normalize the values to range of $[-1, 1]$ by:

$$\mathcal{A} = \tanh(\text{Adapt}(\mathcal{C}, \Theta)) \quad (6.5)$$

FIGURE 6.4: **Attribute Attention:** It emphasizes feature channels that have larger response in semantic attribute learning, which is based on the phrase interaction, click interaction and visual patterns.

where $\tanh(x) = (e^x - e^{-x}) / (e^x + e^{-x})$, *Adapt* is **Conv+BN** and Θ is its learnable parameters. Now we get the attribute attention weight $\mathcal{A}$, in which different attribute channels have different attention weights in the range of $[-1, 1]$. Finally, as shown in Figure 6.4, we apply the attribute attention weights on feature maps in CNN:

$$\hat{F} = \beta \mathcal{A} * F + F \quad (6.6)$$

where β is a learnable weight and $*$ denotes channel-wise multiplication. To have our approach more flexible in terms of interaction methods, the phrase interaction is not compulsory. When there is no phrase interaction input, the attribute attention weight $\mathcal{A}$ is set to 0.

With the attribute attention, the feature channels that represent the same attribute information as the phrase expression are emphasized while others are suppressed. Through this way, the network can quickly determine candidate objects with the attributes inferred from phrase interaction. More important, the attribute attention module helps to bridge the clicks and phrases information together. The click information provides the location of target object and further refines the initial prediction, while the phrase expression provides high-level semantic attribute to help the network disambiguate confused potential objects of interest. These two interaction inputs work together to reduce the interaction times for improved performance.

6.2.4 Training Loss

We use the binary cross-entropy loss for the segmentation branch training. Given the segmentation ground truth t and the segmentation branch output s , the

segmentation loss is:

$$\mathcal{L}_m = -t \log(\sigma(s)) - (1 - t) \log(1 - \sigma(s)) \quad (6.7)$$

where $\sigma(x) = 1/(1 + e^{-x})$ is sigmoid function.

Additionally, in order to help the network optimize the attribute learning process and catch meaningful visual and phrasal attributes, we introduce an attribute loss for the attribute learning module. We first generate the attribute label $\{a_i\}_{i=1}^N$ following [129], $a_i \in \{0, 1\}$ indicates whether the i -th attribute word exists in the input phrase. Denoting the output of attribute learning module as p_i , the objective function for attribute module is defined as:

$$\mathcal{L}_a = \sum_{i=1}^N w_i (a_i \log(\sigma(p_i)) + (1 - a_i) \log(1 - \sigma(p_i))) \quad (6.8)$$

where w_i is a weighting factor to address the unbalance of different attributes, and a_i indicates whether the i -th attribute word exists in the input phrase. The final loss for the proposed network is:

$$\mathcal{L} = \omega_m \mathcal{L}_m + \omega_a \mathcal{L}_a \quad (6.9)$$

ω_m and ω_a are used to balance the contributions of the segmentation loss and attribute loss.

6.3 Experiments

6.3.1 Implementation Details

The network and all the experiments are implemented based on the public Pytorch platform. We use ResNet-101 [2] based DeepLabv3+ [128] as the backbone of the segmentation branch. The first convolution layer is modified to $5 \times 64 \times 7 \times 7$ to deal with the 5-channel input. The proposed attribute attention module is placed between the encoder and decoder of Deeplabv3+, i.e., after ASPP. The parameters of newly added layers are randomly initialized from a Gaussian distribution with standard variance of 10^{-2} . Batch Normalization [151] is used after newly added

FIGURE 6.5: Examples of our phrase annotations for Grabcut and Berkeley.

convolution layers to accelerate training process. The network is trained by SGD with batch size set to 10. For batch processing, we resize the inputs to 512×512 pixels during training. We adopt random horizontal flipping to augment the training data. Our network is trained end-to-end with SGD. The learning rate is set to 1×10^{-8} and the parameters of the new layers are trained with a higher learning rate 1×10^{-7} . Momentum and weight decay are fixed to 0.99 and 1×10^{-4} respectively. We empirically set ω_m and ω_a to 1 and 10. The network is evaluated with Intersection-over-Union (**IoU**) [45].

To jointly train the attribute learning branch and segmentation branch, we train our network on RefCOCO [129] first, which contains referring phrase expressions and segmentation masks for every instance objects in each of 19994 images. Then, for interactive object segmentation datasets like PASCAL VOC, we take the name of categories as the phrase input, like person and car, to fine-tune the attribute learning branch. We annotate phrase expressions for the Grabcut and Berkeley datasets (see Figure 6.5).

6.3.2 Ablation Studies

We conduct the ablation studies on RefCOCO [129], as shown in Table 6.1. First, to verify the effectiveness of the proposed attribute attention module, we discard the click interaction and only take the phrase as input. With *Phrase-only* input, our network can achieve an acceptable result of 50.98%. We also display some visualized

Interaction Inputs	Phrase	Click	Graph Cut	IoU
<i>Phrase-only</i>	✓			50.98
<i>Click-only</i>		✓		77.93
<i>PhraseClick</i>	✓	✓		83.94
<i>PhraseClick</i>	✓	✓	✓	85.02

TABLE 6.1: Ablation study of the proposed approach on testA of RefeCOCO.

segmentation score maps in Figure 6.6. Both the quantitative and qualitative results demonstrate that the attribute attention module does help the network to select the target-of-interest according to referential phrase expression.

We also conduct a *Click-only* experiment in which the $\vec{h}$ and $\overleftarrow{h}$ in Eq. (6.4) are set to 0. This achieves 77.93% IoU by 3 clicks, which is much higher than phrase only. It shows the advantage of click interaction for interactive object segmentation. Next, we verify whether the phrase interaction can speed up the click-based interactive segmentation process and improve the segmentation performance. To this end, we first compare the segmentation performance of *Click-only* v.s. *PhraseClick*. As shown in Table 6.1, compared with *Click-only* without phrase, *PhraseClick* achieves 6.01% better performance on IoU. This demonstrates that the phrase expression and attribute attention module visibly enhances the performance of click-based interactive segmentation. Note, in Table 6.1, the interaction inputs for *PhraseClick* are one phrase and two clicks (one positive and one negative). For fair comparison, there is one more positive click for *Click-only* than *PhraseClick* in Table 6.1, i.e. the number of interaction inputs for *Click-only* and *PhraseClick* is the same. Next, we show the segmentation performance with different number of interaction times in Fig. 6.7. We start with an initial positive click, then gradually add a phrase/click as the next interaction according to the current segmentation prediction. As shown in Fig. 6.7, to reach a specific IoU performance, the network taking the *PhraseClick* as interaction inputs is faster than the network that only takes *Click* input. This shows the phrase expression helps to speed up the interaction process and improve the segmentation performance. During the inference, we employ the graph cut [32] as post-processing method to refine the segmentation mask, which improve the performance by 1.08% IoU. The performance gain is brought by the refinement in boundary regions.

Note that in practical applications, the amount of time to input a phrase depends

FIGURE 6.6: We display some segmentation score maps (soft value from 0 to 1) of referential object segmentation. Here the network only takes the phrase as interaction input.

greatly on implementation and the input device (mouse, trackpad, or finger). Given a strong voice-to-text algorithm (such as those in Siri and Alexa), the verbal command can be issued during the time the user moves the mouse to the position to add a click. Thus, in an ideal implementation, a phrase can be uttered at least

FIGURE 6.7: Segmentation accuracy with different number of interaction times.

as quickly as click, perhaps much faster depending on the user and device. In our specific case, we do not have a sophisticated voice-to-text system, so we allow the user to type in the desired phrase. This will be slower than using voice and likely slower than clicks. But the scope of this work is investigating the computer vision aspects of this and not building an ideal UI system across multiple platforms (desktop with mouse, laptop with trackpad, phone with finger) to fully evaluate the optimal interactivity of the method.

6.3.3 Results on Benchmarks

Interactive Object Segmentation

We first compare the proposed approach with previous state-of-the-art interactive segmentation methods. Interactive object segmentation enables users to add interactive information until the desired selection is reached.

We conduct the interactive segmentation experiments on three public benchmarks with instance object annotations: PASCAL VOC [135], Grabcut [23], Berkeley [194]. PASCAL VOC is benchmark for semantic/instance segmentation, Grabcut and Berkeley are benchmarks for interactive segmentation. PASCAL VOC contains 20 object categories with 1464 images for training and 1449 images for validation. Grabcut has 50 images that have obvious foreground object and background stuff distributions. Thus, it is easy to reach a good performance. Berkeley has 100 images for interactive segmentation.

Methods	PASCAL VOC (85% IoU)	GrabCut (90% IoU)	Berkeley (90% IoU)
GraphCut [32]	15.06	11.10	14.33
Geodesic Matting [192]	14.75	12.44	15.96
Random Walker [114]	11.37	12.30	14.02
Geodesic Convexity [193]	11.73	8.38	12.57
iFCN [27]	6.88	6.04	8.65
RIS-Net [28]	5.12	5.00	6.03
ITIS [120]	3.80	5.60	-
LDN [122]	-	4.79	-
DEXTR [29]	4.00	4.00	4.00
FCTSFN [109]	4.58	3.76	6.49
<i>PhraseClick</i> (Ours)	3.12	2.06	3.26

TABLE 6.2: Comparison with previous state-of-the-art interactive object segmentation methods. We demonstrate the number of interactions that each method needed to reach a certain segmentation performance.

To evaluate the performance of interactive object segmentation, the standard process is: 1) starting with an initial positive click at the center of the target-of-interest, the network generates an initial segmentation mask for the target object; 2) then a succeeding positive/negative click is added at the center of the largest wrongly predicted piece to add/remove this piece; 3) the 2nd step is repeated again and again until the desirable accuracy is reached or the maximum number (set to 20 as [27]) of clicks is reached. Different from existing interactive segmentation methods that only use clicks as input, the proposed approach accepts both the clicks and phrases as interaction input. Therefore, to evaluate our approach and compare with existing state-of-the-art works, we input the phrase for the target-of-interest after the initial click in the first step, which together are counted as two interaction times. Then the subsequent interactions only use clicks. Table 6.2 shows the number of interactions that each approach requires to reach a certain IoU. We achieve new state-of-the-art performance on the three benchmarks, which shows the advantage of the proposed approach.

Furthermore, in comparison with methods that accept a single type of interaction, the proposed approach is more flexible in terms of interaction and can better handle complex scenarios via utilizing advantages of both clicks and phrases. Although we fix the order of phrase input to be the second interaction after the initial positive click in Table 6.2, this is just for the convenience of evaluation. In practice, user can input the phrase at any time and randomly choose phrases and clicks interactions in any order, e.g. phrase first and then refine with clicks. We show some interaction

examples in Figure 6.8. In the first two rows in Figure 6.8, we start with a phrase to get the initial segmentation prediction, then refine the initial prediction with clicks. The first row demonstrates the advantages of clicks at locating “where” the object is when there are multiple objects with the similar appearance. The second row shows the abilities of clicks to correct mistakes and refine details. In the 3rd-4th rows in Figure 6.8, phrases are employed to disambiguate the confused results of clicks, which shows the advantages of phrases at describing “what” the object of interest is. The first four rows show how the integration of clicks and phrases help to improve the performance of interactive segmentation and better handle complex scenarios. Also, our network can work in a way same as previous interactive segmentation methods that only take clicks as interaction inputs, as shown in the last two rows in Figure 6.8.

For the unknown/unseen objects that haven’t been seen in training data, e.g., the UFO, the user can enter “the circle in the middle” to segment it. In other word, user can select objects by giving a natural description of its characteristics. What’s more, the proposed method can always fall back on clicks if the phrases were not understood by system. The pieces selected by clicks can be any objects, no matter known or unknown. With PhraseClick, user can select any target-of-interest by providing interactions.

Referential Object Segmentation

Although the proposed network is designed for interactive object segmentation, it can also be used for referential object segmentation, which generates a segmentation mask for a referential object according to a referring expression. Most existing referential object segmentation methods have multiple stages. For example, MAttNet [111], based on Mask R-CNN [124], scores and ranks the similarity between phrase embedding and instance objects generated by Mask R-CNN, and then outputs the mask of the best matched object. Therefore, there are 3 stages in MAttNet: region proposal, mask generation, and matching of phrase and instance objects. Different from the multi-stage methods, our proposed approach can be viewed as one-stage grounding method that completely discards the region proposal stage and similarity matching stage. It directly generates a segmentation mask according to the referring expression. More importantly, existing referential object segmentation methods only compute an initial segmentation and cannot further

Methods	RMI [110]	RNN [195]	MAttNet [111]	<i>Phrase</i>	<i>PhraseClick</i>
IoU	45.18	55.33	56.51	50.42	80.02

TABLE 6.3: Comparison with state-of-the-art referring segmentation methods.

correct the mistakes, thus they cannot meet the requirement of practical application for interactive segmentation. The proposed approach can gradually refine segmentation result until it meets the user’s requirement.

We compare with referential object segmentation methods on the validation set of RefCOCO [129]. RefCOCO consists of 142209 referring expressions for 50000 objects in 19994 images from MS COCO [139]. As shown in Table 6.3, although our *Phrase* is proposed as part of a network design that includes clicking, it achieves competitive results with phrase-only methods. With only one click (*PhraseClick*), the proposed approach significantly outperforms the state-of-the-arts by a very large margin, illustrating the utility of adding clicks to phrase-based selection.

6.4 Chapter Summary

In this chapter, we propose an interactive segmentation network that can take either clicks or phrases or both as interaction input, which utilizes the complementary merits of these two interactions. Existing interactive object segmentation methods mainly take spatial interaction as input such as bounding boxes or clicks. As these methods receive only spatial constraints, they cannot provide high-level attribute information to the network and thus may require additional user input to overcome this drawback in challenging cases. To speed up the interaction process, we propose to make use of attribute information to quickly determine what the target-of-interest is with fewer clicks. We introduce phrase expression as another interaction input to infer semantic attributes and propose an attribute attention module to integrate such attributes information into the network. Specifically, the language phrases focuses on “what” the object is and the spatial clicks are in charge of “where” the selected object is. Extensive experimental results have shown that the proposed approach is flexible in terms of interaction and can handle complex scenarios well.

FIGURE 6.8: Different kinds of interaction processes. The proposed approach is more flexible in terms of interaction ways and can better handle complex scenarios via utilizing advantages of these two interactions.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we have explored how to enhance scene parsing performance from different aspects. We start from introducing four challenges in scene parsing, and proposed four segmentation models to address these challenges. We figure out that context information is essential for scene parsing. Meanwhile, multi-scale and multi-level feature aggregation benefits scene parsing a lot. And boundary detection and delineation helps to provide spatial details for dense prediction. Finally, since it is time-consuming and resource-expensive to develop scene parsing benchmarks and it is impractical to annotate all the possible object categories in the world, interactive segmentation tools are desired.

Motivated by these challenges, we propose four deep-learning-based segmentation models. The first segmentation model, CGBNet, employs context encoding and multi-path decoding to enhance the segmentation performance from different aspects. The second segmentation model further introduce shape- and scale-variant context for enhancing high-level feature representation. The third model explores to detect the boundaries of objects and utilize the learned boundaries to increase the feature similarity of the same object while keeping the feature discrimination of different objects. In the mean time, in the fourth model, we build an interactive segmentation network which aims to segment the image according to users' interactions.

In align with the challenges and motivations discussed in this thesis, we elaborate and conclude the contributions of the proposed four segmentation models as follows:

- 1) **Context encoding.** We propose two context encoding modules to customize high-level information for every pixel. Firstly, in Chapter 3, we propose a novel context-contrasted local feature that is tailored for scene segmentation and propose a context-contrasted local (CCL) model to aggregate multi-scale context-contrasted local features. The proposed CCL brings 5.8% IoU performance gain on Pascal-Context [33]. Secondly, in Chapter 4, we propose a novel paired convolution to infer the semantic correlation of two pixels and based on that to generate a semantic-correlated region for each pixel. Based on the semantic-correlated region, we propose a shape-variant context aggregated within the semantic-correlated region to model the diverse shapes and scales of contexts, which greatly enhances the modeling ability of network. The proposed shape-variant context(SVC) brings significant 9.7% and 7.0% IoU performance gain on Pascal-Context [33] and COCO-Stuff [34] respectively, which demonstrates the significance of SVC. Besides, the feature prorogation module in Chapter 5 also encoding and propagate high-level context clues to each pixels, which brings 8.6% IoU performance gain on Pascal-Context [33]. All the three works demonstrates the effectiveness of context encoding for scene parsing, which is also in line with our motivation.
- 2) **Multi-scale and multi-level feature aggregation.** In Chapter 3, We propose a gated sum scheme to adaptively aggregate appropriate-scale features at each spatial location, which is an efficient and effective way to utilize multi-scale and multi-level features for enhancing parsing performance. The gated sum scheme significantly boosts the segmentation performance by 3.3% IoU on Pascal-Context [33]. The boundary delineation refinement module leverages the very-low-level features to refine the boundary details and also enhance the coalescence of multi-level tasks (e.g. high-level recognition and low-level boundary delineation) in a single network. And in Chapter 4, the shape-variant context (SVC) module is scale- and shape-variant, it helps to customize appropriate-scale context information for different pixels and thus implicitly leverage multi-scale information. We propose a boundary delineation model to refine the prediction of boundary, which encourages

the location-informative very-low-level features near the boundaries and suppressed them far from the boundaries.

- 3) **Boundary delineation.** To generate more spatial details in dense prediction, we propose a boundary delineation module in CGBNet to refine the prediction of boundary, which encourages the location-informative very-low-level features near the boundaries and suppressed them far from the boundaries. We show that the boundary can be learned as one of semantic categories, which requires little change on network but obtains essential boundary information. Meanwhile, in the third model BFP, we show that the boundary can be learned as one of semantic categories. The two tasks (scene parsing and boundary detection) are combined into one, and they benefit each other. By training them together, the scene parsing classes help suppress the edges within objects that are not semantic boundary of objects, e.g., edges of eyes. Scene segmentation helps the boundary detection to filter out noise and delineate well-directed boundary, while boundary detection makes the scene parsing be aware of the important boundary layout information and provide more spatial details.
- 4) **Unseen/Unknown objects' segmentation.** To segment any pieces including unseen/unknown objects that never appeared in training data, in Chapter 6, we propose a novel interactive segmentation model (PhraseClick) that employ both phrase expressions and clicks. The proposed interactive segmentation model can be used to segment any objects including unseen or unknown objects, it greatly speeds up the pixel-wise annotation progress. To the best of our knowledge, this is the first work to leverage both clicks and phrases for interactive object segmentation. The proposed approach is more flexible in terms of interaction and can better handle complex scenarios via utilizing advantages of these two interactions. We propose an attribute guided feature attention module to extract discriminative semantic attribute information from phrase expressions and integrate the attribute and location information into the network. Extensive experimental results have demonstrated that phrase expressions are indeed effective at boosting the performance of interactive segmentation, especially in some complex scenarios and with few clicks.

7.2 Future works

With the powerful deep neural networks as backbone in segmentation models, significant progress has been achieved for scene parsing in thesis. In the future work, we will continue exploring to develop stronger segmentation models. Beyond this, we will pay attention to following unfolding works:

- **Unsupervised or weakly supervised segmentation.** Current state-of-the-arts segmentation networks still require the fully annotated ground truth, which is rare, expensive and difficult to obtain in real world [33]. One of the way is employ interactive segmentation to speed up the pixel-wise annotation progress, like PhraseClick in this thesis. Another way is to exploit unsupervised or weakly supervised methods to train segmentation networks. In such case, we can make use of the images in the wild and greatly enlarge the training data. In the future, we will aim to learn robust semantic segmentation networks in unsupervised or weakly supervised condition. One of the possible approaches is to exploit the image-level label and combine with multi-label classification to weakly supervise the training of segmentation networks.
- **Real-time segmentation.** Current segmentation networks have achieved very good accuracy but consume too much resources, such as GPU memory and time. It is unrealistic to apply these segmentation networks in practice. To achieve effective and efficient segmentation network, the speed and the amount of calculation must be taken into account. Thus, real-time scene parsing networks will be one of the next stations. There are two possible ways to achieve this goal. One possible method is teacher-student network, in which a shallow but fast student network is supervised by a deeper and larger teacher network pre-trained on segmentation datasets. The deep neural networks are more like black boxes, there may be lots of redundancy in the deeper teacher network. The goal of teacher-student network is to remove these redundance and minimize the difference between teacher part and student part. After training, the larger teacher network will be abandoned and the fast student network is used for real-time scene segmentation. Another way is to carefully design a suitable network architecture which is fast and does not sacrifice to much accuracy.

- **Panoptic segmentation.** Panoptic segmentation [196, 197] is a new challenging task that combine the typical tasks of scene parsing and instance segmentation. It requires to assign a semantic label to each pixel but also detect each countable object instance. In panoptic segmentation, the pixels either belong to countable “things” (e.g. car, person) or uncountable “stuff” (e.g. sky, ground). Compared to scene parsing, panoptic segmentation further requires to assign the countable objects instance IDs. Compared to instance segmentation, panoptic segmentation further requires to classify the pixels belonging to uncountable “stuff”. Panoptic segmentation can provide more comprehensive information than semantic segmentation and instance segmentation. For example, with panoptic segmentation, we are able to understand how many instance objects for each class and how these instance objects relate with each other and the background stuff. Therefore, panoptic segmentation has more promising applications and demands research attention. We will explore to develop robust panoptic segmentation models in the future.

Author's Publications

Author's publications

- **Henghui Ding**, Scott Cohen, Brian Price, and Xudong Jiang. *PhraseClick: Toward Achieving Flexible Interactive Segmentation by Phrase and Click*. In European Conference on Computer Vision (**ECCV**), Springer, 2020. (**Spot-light**, Acceptance Rate 5.3%)
- **Henghui Ding**, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. *Semantic Correlation Promoted Shape-Variant Context for Segmentation*. In Proceedings of the Conference on Computer Vision and Pattern Recognition (**CVPR**), IEEE, 2019. (**Oral**, Acceptance Rate **5.6%**)
- **Henghui Ding**, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang. *Boundary-Aware Feature Propagation for Scene Segmentation*. In Proceedings of the International Conference on Computer Vision (**ICCV**), IEEE, 2019.
- **Henghui Ding**, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. *Context Contrasted Feature and Gated Multi-scale Aggregation for Scene Segmentation*. In Proceedings of the Conference on Computer Vision and Pattern Recognition (**CVPR**), IEEE, 2018. (**Oral**, Acceptance Rate **2.1%**)
- **Henghui Ding**, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. *Semantic Segmentation with Context Encoding and Multi-Path Decoding*. In IEEE Transactions on Image Processing (**TIP**), 2020.
- **Henghui Ding***, Bing Shuai*, Ting Liu, Gang Wang, and Xudong Jiang. *Towards Achieving Robust High-level and Low-level Scene Parsing*. In IEEE Transactions on Image Processing (**TIP**), 2019. (*Equal contribution).

Author's other publications

- Jun Liu, **Henghui Ding**, Amir Shahroudy, Ling-Yu Duan, Xudong Jiang, Gang Wang, and Alex C. Kot. *Feature Boosting Network For 3D Pose Estimation*. In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2019.
- Xiaohong Wang, Xudong Jiang, **Henghui Ding**, and Jun Liu. *Bi-directional Dermoscopic Feature Learning and Multi-scale Consistent Decision Fusion for Skin Lesion Segmentation*. In IEEE Transactions on Image Processing (TIP), Accept.
- Xiaohong Wang, **Henghui Ding**, and Xudong Jiang. *Dermoscopic Image Segmentation through the Enhanced High-level Parsing and Class Weighted Loss*. In Proceedings of the International Conference on Image Processing (ICIP), IEEE, 2019.
- Jianhan Mei, Ziming Wu, Xiang Chen, Yu Qiao, **Henghui Ding**, and Xudong Jiang. *DeepDeblur: Text Image Recovery From Blur to Sharp*. In Multimedia Tools and Applications (MTAP), Springer, 2019.
- Jianhan Mei, **Henghui Ding**, and Xudong Jiang. *Object 6D pose estimation with non-local attention*. In Proceedings of the Twelfth International Conference on Digital Image Processing (ICDIP), 2020.

References

- [1] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, “Going deeper with convolutions,” in *The IEEE conference on computer vision and pattern recognition*, 2015. [xix](#), [1](#), [9](#), [10](#), [11](#), [24](#), [54](#)
- [2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. [xix](#), [xxiv](#), [1](#), [3](#), [5](#), [9](#), [10](#), [24](#), [26](#), [29](#), [54](#), [64](#), [76](#), [83](#), [91](#), [97](#), [99](#), [102](#)
- [3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” *arXiv:1606.00915*, 2016. [xxi](#), [1](#), [3](#), [9](#), [12](#), [13](#), [14](#), [15](#), [19](#), [24](#), [25](#), [26](#), [29](#), [31](#), [39](#), [40](#), [46](#), [47](#), [50](#), [54](#), [55](#), [61](#), [62](#), [64](#), [67](#), [69](#), [70](#), [75](#), [77](#), [79](#), [80](#), [83](#), [84](#), [88](#), [91](#)
- [4] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang, “Denseaspp for semantic segmentation in street scenes,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. [xxiv](#), [13](#), [50](#), [55](#), [61](#), [70](#), [91](#)
- [5] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger, “Densely connected convolutional networks,” in *CVPR*, 2017. [xxiv](#), [1](#), [9](#), [24](#), [54](#), [91](#)
- [6] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al., “Recent advances in convolutional neural networks,” *Pattern Recognition*, 2017. [1](#), [24](#)

-
- [7] Karen Simonyan and Andrew Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv:1409.1556*, 2014. [1](#), [3](#), [9](#), [10](#), [24](#), [29](#), [54](#), [84](#)
 - [8] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015. [1](#), [9](#), [19](#), [24](#)
 - [9] Ross Girshick, “Fast r-cnn,” in *The IEEE international conference on computer vision*, 2015. [1](#), [9](#), [24](#)
 - [10] Evan Shelhamer, Jonathon Long, and Trevor Darrell, “Fully convolutional networks for semantic segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, 2016. [1](#), [9](#), [14](#), [24](#), [26](#), [32](#), [39](#), [42](#), [46](#), [47](#), [48](#), [49](#), [50](#), [68](#), [69](#), [88](#), [91](#)
 - [11] Henghui Ding, Scott Cohen, Brian Price, and Xudong Jiang, “Phraseclick: Toward achieving flexible interactive segmentation by phrase and click,” in *European conference on computer vision*. Springer, 2020.
 - [12] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang, “Semantic correlation promoted shape-variant context for segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019, pp. 8885–8894. [9](#), [24](#), [91](#)
 - [13] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, “Pyramid scene parsing network,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017. [1](#), [3](#), [9](#), [12](#), [13](#), [15](#), [19](#), [24](#), [26](#), [49](#), [50](#), [54](#), [55](#), [61](#), [62](#), [70](#), [75](#), [80](#), [91](#)
 - [14] Matthew D Zeiler and Rob Fergus, “Visualizing and understanding convolutional networks,” in *European conference on computer vision*, 2014. [1](#), [24](#)
 - [15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al., “Imagenet large scale visual recognition challenge,” *International Journal of Computer Vision*, vol. 115, no. 3, 2015. [2](#), [5](#), [24](#), [38](#), [54](#), [64](#)

- [16] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo, “Multi-label image recognition with graph convolutional networks,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. [2](#)
- [17] Christian Zimmermann and Thomas Brox, “Learning to estimate 3d hand pose from single rgb images,” in *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. [2](#)
- [18] Wei Xiong, Jiahui Yu, Zhe Lin, Jimei Yang, Xin Lu, Connelly Barnes, and Jiebo Luo, “Foreground-aware image inpainting,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. [2](#)
- [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015. [3](#)
- [20] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in *CVPR*, 2006. [3](#)
- [21] Wei Liu, Andrew Rabinovich, and Alexander C Berg, “Parsenet: Looking wider to see better,” *arXiv:1506.04579*, 2015. [3](#), [46](#)
- [22] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele, “The cityscapes dataset for semantic urban scene understanding,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223. [4](#), [8](#), [22](#), [48](#), [69](#), [75](#), [80](#), [84](#), [87](#)
- [23] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake, “Grabcut: Interactive foreground extraction using iterated graph cuts,” in *ACM transactions on graphics (TOG)*. ACM, 2004, vol. 23, pp. 309–314. [6](#), [17](#), [18](#), [94](#), [106](#)
- [24] Ning Xu, Brian Price, Scott Cohen, Jimei Yang, and Thomas Huang, “Deep grabcut for object selection,” in *BMVC*, 2017. [18](#)
- [25] Suyog Dutt Jain and Kristen Grauman, “Predicting sufficient annotation strength for interactive foreground segmentation,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013.

-
- [26] Victor S Lempitsky, Pushmeet Kohli, Carsten Rother, and Toby Sharp, “Image segmentation with a bounding box prior,” in *ICCV*, 2009, vol. 76. 6, 17, 18, 94
- [27] Ning Xu, Brian Price, Scott Cohen, Jimei Yang, and Thomas S Huang, “Deep interactive object selection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 373–381. 6, 17, 18, 19, 20, 94, 97, 99, 107
- [28] JunHao Liew, Yunchao Wei, Wei Xiong, Sim-Heng Ong, and Jiashi Feng, “Regional interactive image segmentation networks,” in *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 2746–2754. 17, 18, 20, 107
- [29] Kevis-Kokitsi Maninis, Sergi Caelles, Jordi Pont-Tuset, and Luc Van Gool, “Deep extreme cut: From extreme points to object segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 616–625. 18, 107
- [30] Hoang Le, Long Mai, Brian Price, Scott Cohen, Hailin Jin, and Feng Liu, “Interactive boundary prediction for object selection,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 18–33. 17, 18, 20
- [31] Jianbo Shi and Jitendra Malik, “Normalized cuts and image segmentation,” *Departmental Papers (CIS)*, p. 107, 2000. 17
- [32] Yuri Y Boykov and M-P Jolly, “Interactive graph cuts for optimal boundary & region segmentation of objects in nd images,” in *IEEE international conference on computer vision*. IEEE, 2001, vol. 1, pp. 105–112. 6, 17, 94, 97, 104, 107
- [33] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 7, 8, 20, 46, 69, 85, 86, 112, 114

- [34] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari, “Coco-stuff: Thing and stuff classes in context,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 7, 67, 112
- [35] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari, “Coco-stuff: Thing and stuff classes in context,” *arXiv:1612.03716*, 2016. 8, 21, 47, 49
- [36] Gabriel J Brostow, Jamie Shotton, Julien Fauqueur, and Roberto Cipolla, “Segmentation and recognition using structure from motion point clouds,” in *ECCV*, 2008, pp. 44–57. 8, 22, 68, 80, 87
- [37] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao, “Sun rgb-d: A rgb-d scene understanding benchmark suite,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 8, 21, 46
- [38] Ce Liu, Jenny Yuen, and Antonio Torralba, “Nonparametric scene parsing: Label transfer via dense scene alignment,” in *CVPR*. IEEE, 2009. 8, 21, 47, 67
- [39] Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille, “Detect what you can: Detecting and representing objects using holistic models and body parts,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 8, 21, 68
- [40] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba, “Semantic understanding of scenes through the ade20k dataset,” *arXiv preprint arXiv:1608.05442*, 2016. 8
- [41] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, “You only look once: Unified, real-time object detection,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 9, 24
- [42] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *CVPR*, 2014.
- [43] Chunze Lin, Jiwen Lu, Gang Wang, and Jie Zhou, “Graininess-aware deep feature learning for pedestrian detection,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 732–747. 9, 24

-
- [44] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012. [10](#), [11](#), [54](#)
- [45] Jonathan Long, Evan Shelhamer, and Trevor Darrell, “Fully convolutional networks for semantic segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. [11](#), [15](#), [18](#), [19](#), [56](#), [62](#), [64](#), [67](#), [68](#), [83](#), [88](#), [103](#)
- [46] Jianhan Mei, Henghui Ding, and Xudong Jiang, “Object 6d pose estimation with non-local attention,” in *Twelfth International Conference on Digital Image Processing (ICDIP 2020)*. International Society for Optics and Photonics, 2020, vol. 11519, p. 115191H. [11](#)
- [47] Jun Liu, Henghui Ding, Amir Shahroudy, Ling-Yu Duan, Xudong Jiang, Gang Wang, and Alex Kot Chichung, “Feature boosting network for 3d pose estimation,” *IEEE transactions on pattern analysis and machine intelligence*, 2019. [11](#), [33](#)
- [48] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han, “Learning deconvolution network for semantic segmentation,” in *IEEE International Conference on Computer Vision*, 2015. [12](#), [35](#), [47](#), [68](#), [88](#)
- [49] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2017. [12](#), [68](#), [88](#)
- [50] Xiaoxiao Li, Ziwei Liu, Ping Luo, Chen Change Loy, and Xiaoou Tang, “Not all pixels are equal: Difficulty-aware semantic segmentation via deep layer cascade,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3193–3202. [12](#)
- [51] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia, “Icnet for real-time semantic segmentation on high-resolution images,” in *The European Conference on Computer Vision (ECCV)*, September 2018. [12](#)
- [52] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello, “Enet: A deep neural network architecture for real-time semantic segmentation,” *arXiv:1606.02147*, 2016.

- [53] Eduardo Romera, José M Alvarez, Luis M Bergasa, and Roberto Arroyo, “Erfnet: Efficient residual factorized convnet for real-time semantic segmentation,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 263–272, 2018. [12](#)
- [54] Aurelien Lucchi, Yunpeng Li, Xavier Boix, Kevin Smith, and Pascal Fua, “Are spatial and global constraints really necessary for segmentation?,” in *IEEE International Conference on Computer Vision*, 2011. [12](#)
- [55] Fisher Yu and Vladlen Koltun, “Multi-scale context aggregation by dilated convolutions,” *arXiv:1511.07122*, 2015. [12](#), [13](#), [25](#), [29](#), [49](#), [50](#), [55](#), [61](#), [68](#), [75](#), [79](#), [88](#)
- [56] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang, “Context contrasted feature and gated multi-scale aggregation for scene segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. [13](#), [29](#), [32](#), [55](#), [56](#), [61](#), [62](#), [67](#), [69](#), [79](#), [88](#)
- [57] Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang, “Boundary-aware feature propagation for scene segmentation,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 6819–6829. [12](#)
- [58] Francesco Visin, Marco Ciccone, Adriana Romero, Kyle Kastner, Kyunghyun Cho, Yoshua Bengio, Matteo Matteucci, and Aaron Courville, “Reseg: A recurrent neural network-based model for semantic segmentation,” in *CVPRW*, 2016. [12](#)
- [59] Wonmin Byeon, Thomas M. Breuel, Federico Raue, and Marcus Liwicki, “Scene labeling with lstm recurrent neural networks,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2015. [12](#), [15](#), [29](#)
- [60] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen-Change Loy, and Xiaoou Tang, “Semantic image segmentation via deep parsing network,” in *IEEE International Conference on Computer Vision*, 2015. [12](#), [15](#), [25](#), [50](#), [91](#)
- [61] Mohammadreza Mostajabi, Payman Yadollahpour, and Gregory Shakhnarovich, “Feedforward semantic segmentation with zoom-out features,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. [12](#), [14](#), [26](#), [32](#)

-
- [62] Abhishek Sharma, Oncel Tuzel, and Ming-Yu Liu, “Recursive context propagation network for semantic scene labeling,” in *Advances in Neural Information Processing Systems*, 2014. [12](#), [48](#), [68](#)
- [63] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Amrith Tyagi, and Amit Agrawal, “Context encoding for semantic segmentation,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018. [12](#), [19](#), [46](#), [49](#), [55](#), [69](#), [75](#), [80](#), [88](#)
- [64] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu, “Dual attention network for scene segmentation,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3146–3154. [12](#), [19](#)
- [65] Philipp Krähenbühl and Vladlen Koltun, “Efficient inference in fully connected crfs with gaussian edge potentials,” in *Advances in neural information processing systems*, 2011. [12](#), [14](#), [15](#), [32](#), [40](#)
- [66] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille, “Semantic image segmentation with deep convolutional nets and fully connected crfs,” in *ICLR*, 2015. [12](#), [14](#), [26](#), [32](#), [49](#), [67](#), [68](#), [69](#), [88](#)
- [67] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr, “Conditional random fields as recurrent neural networks,” in *The IEEE International Conference on Computer Vision*, 2015. [12](#), [19](#), [46](#), [50](#), [69](#), [91](#)
- [68] G. Lin, C. Shen, A. van den Hengel, and I. Reid, “Efficient piecewise training of deep structured models for semantic segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. [12](#), [14](#), [26](#), [46](#), [47](#), [48](#), [50](#), [68](#), [69](#), [91](#)
- [69] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun, “Large kernel matters – improve semantic segmentation by global convolutional network,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017. [13](#), [14](#), [26](#), [61](#), [70](#)

- [70] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei, “Deformable convolutional networks,” in *The IEEE International Conference on Computer Vision*, 2017. [13](#), [61](#)
- [71] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun, “Learning hierarchical features for scene labeling,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, 2013. [14](#), [26](#), [48](#), [68](#)
- [72] Shu Liu, Xiaojuan Qi, Jianping Shi, Hong Zhang, and Jiaya Jia, “Multi-scale patch aggregation (mpa) for simultaneous detection and segmentation,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2016. [14](#)
- [73] Pedro Pinheiro and Ronan Collobert, “Recurrent convolutional neural networks for scene labeling,” in *International Conference on Machine Learning*, 2014. [14](#), [26](#), [48](#), [68](#)
- [74] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik, “Hypercolumns for object segmentation and fine-grained localization,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2015. [14](#), [26](#), [32](#), [56](#), [62](#)
- [75] Bing Shuai, Zhen Zuo, Bing Wang, and Gang Wang, “Scene segmentation with dag-recurrent neural networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. [14](#), [15](#), [25](#), [26](#), [29](#), [30](#), [31](#), [32](#), [40](#), [46](#), [48](#), [49](#), [67](#), [68](#), [75](#), [79](#), [80](#), [84](#), [88](#)
- [76] Jonathan T Barron and Ben Poole, “The fast bilateral solver,” in *European Conference on Computer Vision*, 2016. [14](#)
- [77] Varun Jampani, Martin Kiefel, and Peter V Gehler, “Learning sparse high dimensional filters: Image filtering, dense crfs and bilateral neural networks,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2016. [14](#)
- [78] Stephen Gould, Richard Fulton, and Daphne Koller, “Decomposing a scene into geometric and semantically consistent regions,” in *International Conference on Computer Vision*. IEEE, 2009, pp. 1–8. [15](#)

-
- [79] Ce Liu, Jenny Yuen, and Antonio Torralba, “Sift flow: Dense correspondence across scenes and its applications,” *IEEE transactions on pattern analysis and machine intelligence*, 2011. [47](#), [48](#), [68](#)
- [80] Jimei Yang, Brian Price, Scott Cohen, and Ming-Hsuan Yang, “Context driven scene parsing with attention to rare classes,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2014. [48](#), [68](#)
- [81] Joseph Tighe and Svetlana Lazebnik, “Finding things: Image parsing with regions and per-exemplar detectors,” in *The IEEE conference on computer vision and pattern recognition*, 2013. [15](#), [48](#), [68](#)
- [82] M Pawan Kumar and Daphne Koller, “Efficiently selecting regions for scene understanding,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 3217–3224. [15](#)
- [83] Diane Larlus and Frédéric Jurie, “Combining appearance models and markov random fields for category level object segmentation,” in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–7. [15](#)
- [84] Chris Russell, Pushmeet Kohli, Philip HS Torr, et al., “Associative hierarchical crfs for object class image segmentation,” in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 739–746. [15](#)
- [85] Xiaodan Liang, Xiaohui Shen, Donglai Xiang, Jiashi Feng, Liang Lin, and Shuicheng Yan, “Semantic object parsing with local-global long short-term memory,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3185–3193. [15](#), [69](#)
- [86] Qichuan Geng, Xinyu Huang, Zhong Zhou, and Ruigang Yang, “A network structure to explicitly reduce confusion errors in semantic segmentation,” *arXiv:1808.00313*, 2018. [15](#)
- [87] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang, “Learning a discriminative feature network for semantic segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. [15](#), [50](#), [55](#), [70](#), [91](#)

REFERENCES

- [88] James W Davis, Christopher Menart, Muhammad Akbar, and Roman Ilin, “A classification refinement strategy for semantic segmentation,” *arXiv:1801.07674*, 2018. [16](#), [62](#)
- [89] Yu-Hui Huang, Xu Jia, Stamatios Georgoulis, Tinne Tuytelaars, and Luc Van Gool, “Error correction for dense semantic image labeling,” *arXiv:1712.03812*, 2017. [16](#)
- [90] Pablo Arbelaez, Michael Maire, Charles Fowlkes, and Jitendra Malik, “Contour detection and hierarchical image segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 5, pp. 898–916, 2011. [16](#)
- [91] Piotr Dollár and C Lawrence Zitnick, “Structured forests for fast edge detection,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1841–1848.
- [92] Scott Konishi, Alan L. Yuille, James M. Coughlan, and Song Chun Zhu, “Statistical edge detection: Learning and evaluating edge cues,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 1, pp. 57–74, 2003.
- [93] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson, “Crisp boundary detection using pointwise mutual information,” in *European Conference on Computer Vision*. Springer, 2014, pp. 799–814. [16](#)
- [94] Joseph J Lim, C Lawrence Zitnick, and Piotr Dollár, “Sketch tokens: A learned mid-level representation for contour and object detection,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3158–3165. [16](#)
- [95] Piotr Dollár and C Lawrence Zitnick, “Fast edge detection using structured forests,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 8, pp. 1558–1570, 2015. [16](#)
- [96] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani, “Deepedge: A multi-scale bifurcated deep network for top-down contour detection,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4380–4389. [16](#), [17](#), [78](#)

-
- [97] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani, “High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 504–512. [16](#), [17](#)
 - [98] Jyh-Jing Hwang and Tyng-Luh Liu, “Pixel-wise deep learning for contour detection,” *arXiv:1504.01989*, 2015.
 - [99] Wei Shen, Xinggang Wang, Yan Wang, Xiang Bai, and Zhijiang Zhang, “Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3982–3991.
 - [100] Saining Xie and Zhuowen Tu, “Holistically-nested edge detection,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1395–1403. [16](#), [17](#), [78](#)
 - [101] Wei Shen, Bin Wang, Yuan Jiang, Yan Wang, and Alan Yuille, “Multi-stage multi-recursive-input fully convolutional networks for neuronal boundary detection,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2391–2400. [16](#)
 - [102] Zeeshan Hayder, Xuming He, and Mathieu Salzmann, “Boundary-aware instance segmentation,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5696–5704. [16](#), [17](#)
 - [103] Liang-Chieh Chen, Jonathan T Barron, George Papandreou, Kevin Murphy, and Alan L Yuille, “Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4545–4554. [16](#), [17](#), [85](#)
 - [104] Iasonas Kokkinos, “Pushing the boundaries of boundary detection using deep learning,” *arXiv:1511.07386*, 2015. [16](#)
 - [105] Eric N Mortensen and William A Barrett, “Intelligent scissors for image composition,” in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. ACM, 1995. [17](#)

- [106] Michael Kass, Andrew Witkin, and Demetri Terzopoulos, “Snakes: Active contour models,” *International journal of computer vision*, vol. 1, no. 4, 1988. [17](#)
- [107] Yin Li, Jian Sun, Chi-Keung Tang, and Heung-Yeung Shum, “Lazy snapping,” *ACM Transactions on Graphics (ToG)*, 2004. [17](#)
- [108] Vladimir Vezhnevets and Vadim Konouchine, “Growcut: Interactive multi-label nd image segmentation by cellular automata,” in *proc. of Graphicon*. Citeseer, 2005, vol. 1, pp. 150–156. [17](#)
- [109] Yang Hu, Andrea Soltoggio, Russell Lock, and Steve Carter, “A fully convolutional two-stream fusion network for interactive image segmentation,” *Neural Networks*, vol. 109, 2019. [17](#), [107](#)
- [110] Chenxi Liu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan Yuille, “Recurrent multimodal interaction for referring image segmentation,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1271–1280. [17](#), [19](#), [94](#), [109](#)
- [111] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg, “Mattnet: Modular attention network for referring expression comprehension,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1307–1315. [19](#), [20](#), [99](#), [108](#), [109](#)
- [112] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell, “Segmentation from natural language expressions,” in *European Conference on Computer Vision*. Springer, 2016, pp. 108–124. [17](#), [19](#), [94](#)
- [113] Eirikur Agustsson, Jasper RR Uijlings, and Vittorio Ferrari, “Interactive full image segmentation by considering all regions jointly,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11622–11631. [17](#), [18](#), [20](#)
- [114] Leo Grady, “Random walks for image segmentation,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, , no. 11, pp. 1768–1783, 2006. [17](#), [107](#)
- [115] Xue Bai and Guillermo Sapiro, “A geodesic framework for fast interactive image and video segmentation and matting,” in *2007 IEEE 11th International Conference on Computer Vision*. IEEE, 2007, pp. 1–8. [17](#)

-
- [116] Antonio Criminisi, Toby Sharp, and Andrew Blake, “Geos: Geodesic image segmentation,” in *European Conference on Computer Vision*. Springer, 2008, pp. 99–112.
 - [117] Brian L Price, Bryan Morse, and Scott Cohen, “Geodesic graph cut for interactive image segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 3161–3168. [17](#)
 - [118] David Acuna, Huan Ling, Amlan Kar, and Sanja Fidler, “Efficient interactive annotation of segmentation datasets with polygon-rnn++,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 859–868. [18](#)
 - [119] Lluís Castrejon, Kaustav Kundu, Raquel Urtasun, and Sanja Fidler, “Annotating object instances with a polygon-rnn,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5230–5238. [18](#)
 - [120] Sabarinath Mahadevan, Paul Voigtlaender, and Bastian Leibe, “Iteratively trained interactive segmentation,” *BMVC*, 2018. [18](#), [107](#)
 - [121] Dim P Papadopoulos, Jasper RR Uijlings, Frank Keller, and Vittorio Ferrari, “Extreme clicking for efficient object annotation,” in *IEEE International Conference on Computer Vision*, 2017, pp. 4930–4939. [18](#)
 - [122] Zhuwen Li, Qifeng Chen, and Vladlen Koltun, “Interactive image segmentation with latent diversity,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 577–585. [18](#), [107](#)
 - [123] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid, “Refinenet: Multi-path refinement networks for high-resolution semantic segmentation,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017. [19](#), [26](#), [46](#), [47](#), [49](#), [50](#), [54](#), [62](#), [69](#), [70](#), [88](#), [91](#)
 - [124] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, “Mask r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969. [19](#), [108](#)
 - [125] Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick, “Learning to segment every thing,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4233–4241.

- [126] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia, “Path aggregation network for instance segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8759–8768. [19](#)
- [127] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam, “Masklab: Instance segmentation by refining object detection with semantic and direction features,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4013–4022. [19](#)
- [128] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818. [19](#), [97](#), [102](#)
- [129] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg, “Referitgame: Referring to objects in photographs of natural scenes,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 787–798. [19](#), [20](#), [99](#), [102](#), [103](#), [109](#)
- [130] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg, “Modeling context in referring expressions,” in *European Conference on Computer Vision*. Springer, 2016, pp. 69–85.
- [131] Xihui Liu, Zihao Wang, Jing Shao, Xiaogang Wang, and Hongsheng Li, “Improving referring expression grounding with cross-modal attention-guided erasing,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
- [132] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel, “Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. [19](#), [20](#), [99](#)
- [133] Vasili Ramanishka, Abir Das, Jianming Zhang, and Kate Saenko, “Top-down visual saliency guided by captions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7206–7215. [20](#)

-
- [134] Xiao Chu, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang, “Structured feature learning for pose estimation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4715–4723. [20](#), [100](#)
- [135] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman, “The pascal visual object classes (voc) challenge,” *International journal of computer vision*, vol. 88, no. 2, 2010. [20](#), [106](#)
- [136] Jianxiong Xiao, Andrew Owens, and Antonio Torralba, “Sun3d: A database of big spaces reconstructed using sfm and object labels,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013. [21](#), [46](#)
- [137] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus, “Indoor segmentation and support inference from rgb-d images,” in *European Conference on Computer Vision*, 2012. [21](#), [46](#)
- [138] Allison Janoch, Sergey Karayev, Yangqing Jia, Jonathan T Barron, Mario Fritz, Kate Saenko, and Trevor Darrell, “A category-level 3d object dataset: Putting the kinect to work,” in *Consumer Depth Cameras for Computer Vision*. 2013. [21](#), [46](#)
- [139] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick, “Microsoft coco: Common objects in context,” in *ECCV*. 2014. [21](#), [47](#), [109](#)
- [140] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba, “Scene parsing through ade20k dataset,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017. [22](#), [48](#), [49](#)
- [141] Jianhan Mei, Ziming Wu, Xiang Chen, Yu Qiao, Henghui Ding, and Xudong Jiang, “Deepdeblur: text image recovery from blur to sharp,” *Multimedia Tools Appl.*, vol. 78, no. 13, pp. 18869–18885, 2019. [24](#)
- [142] Xiaohong Wang, Henghui Ding, and Xudong Jiang, “Dermoscopic image segmentation through the enhanced high-level parsing and class weighted loss,” in *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019, pp. 245–249. [24](#)
- [143] Xiaohong Wang, Xudong Jiang, Henghui Ding, and Jun Liu, “Bi-directional dermoscopic feature learning and multi-scale consistent decision fusion for

- skin lesion segmentation,” *IEEE Transactions on Image Processing*, vol. 29, pp. 3039–3051, 2019.
- [144] Bing Shuai, Ting Liu, and Gang Wang, “Improving fully convolution network for semantic segmentation,” *arXiv:1611.08986*, 2016. [24](#)
- [145] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille, “Attention to scale: Scale-aware semantic image segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. [26](#), [69](#)
- [146] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015. [26](#)
- [147] Golnaz Ghiasi and Charless C Fowlkes, “Laplacian pyramid reconstruction and refinement for semantic segmentation,” in *European Conference on Computer Vision*, 2016. [26](#), [50](#), [62](#), [91](#)
- [148] Rosemarie Garland-Thomson, *Staring: How we look*, Oxford University Press, 2009. [31](#)
- [149] Alex Graves, “Offline arabic handwriting recognition with multidimensional recurrent neural networks,” in *Guide to OCR for Arabic scripts*. 2012. [33](#)
- [150] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton, “Speech recognition with deep recurrent neural networks,” in *ICASSP*. IEEE, 2013. [33](#)
- [151] Sergey Ioffe and Christian Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *ICML*, 2015. [39](#), [64](#), [102](#)
- [152] Joao Carreira, Rui Caseiro, Jorge Batista, and Cristian Sminchisescu, “Semantic segmentation with second-order pooling,” *ECCV*, 2012. [46](#), [88](#)
- [153] Jifeng Dai, Kaiming He, and Jian Sun, “Convolutional feature masking for joint object and stuff segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. [46](#)
- [154] Jifeng Dai, Kaiming He, and Jian Sun, “Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation,” in *The IEEE International Conference on Computer Vision*, 2015. [46](#), [69](#), [88](#)

-
- [155] Saining Xie, Xun Huang, and Zhuowen Tu, “Convolutional pseudo-prior for structured labeling,” *arXiv:1511.07409*, 2015. [46](#)
- [156] Anurag Arnab, Sadeep Jayasumana, Shuai Zheng, and Philip HS Torr, “Higher order conditional random fields in deep neural networks,” in *European Conference on Computer Vision*, 2016. [46](#), [69](#), [88](#)
- [157] Aayush Bansal, Xinlei Chen, Bryan Russell, Abhinav Gupta, and Deva Ramanan, “Pixelnet: Towards a general pixel-level architecture,” *arXiv:1609.06694*, 2016. [46](#), [88](#)
- [158] Zifeng Wu, Chunhua Shen, and Anton van den Hengel, “Bridging category-level and instance-level semantic image segmentation,” *arXiv:1605.06885*, 2016. [46](#), [69](#)
- [159] Bing Shuai, Henghui Ding, Ting Liu, Gang Wang, and Xudong Jiang, “Toward achieving robust low-level and high-level scene parsing,” *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1378–1390, 2019. [46](#), [69](#), [88](#)
- [160] Hexiang Hu, Zhiwei Deng, Guang-Tong Zhou, Fei Sha, and Greg Mori, “Labelbank: Revisiting global perspectives for semantic segmentation,” *arXiv:1703.09891*, 2017. [46](#), [67](#)
- [161] Wei-Chih Hung, Yi-Hsuan Tsai, Xiaohui Shen, Zhe L Lin, Kalyan Sunkavalli, Xin Lu, and Ming-Hsuan Yang, “Scene parsing with global context embedding,” in *ICCV*, 2017. [46](#), [55](#), [69](#)
- [162] Piotr Bilinski and Victor Prisacariu, “Dense decoder shortcut connections for single-pass semantic segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6596–6605. [46](#)
- [163] Xiaofeng Ren, Liefeng Bo, and Dieter Fox, “Rgb-(d) scene labeling: Features and algorithms,” in *CVPR*. IEEE, 2012. [47](#)
- [164] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla, “Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding,” *arXiv:1511.02680*, 2015. [47](#)

- [165] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” *arXiv:1511.00561*, 2015. [47](#), [49](#)
- [166] Xiaodan Liang, Hongfei Zhou, and Eric Xing, “Dynamic-structured semantic propagation network,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 752–761. [49](#), [67](#), [70](#)
- [167] Ivan Krešo, Denis Čaušević, Josip Krapac, and Siniša Šegvić, “Convolutional scale invariance for semantic segmentation,” in *German Conference on Pattern Recognition*. Springer, 2016, pp. 64–75. [50](#), [91](#)
- [168] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell, “Understanding convolution for semantic segmentation,” in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 1451–1460. [50](#), [91](#)
- [169] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” *arXiv:1802.02611*, 2018. [54](#), [56](#), [62](#), [88](#)
- [170] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang, “Semantic segmentation with context encoding and multi-path decoding,” *IEEE Transactions on Image Processing*, 2020. [54](#), [88](#), [91](#)
- [171] Zhicheng Yan, Hao Zhang, Yangqing Jia, Thomas Breuel, and Yizhou Yu, “Combining the best of convolutional layers and recurrent layers: A hybrid network for semantic segmentation,” *arXiv:1603.04871*, 2016. [55](#)
- [172] Francois Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017. [64](#)
- [173] Abhijit Kundu, Vibhav Vineet, and Vladlen Koltun, “Feature space optimization for semantic video segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. [68](#), [88](#)
- [174] Simon Jégou, Michal Drozdal, David Vazquez, Adriana Romero, and Yoshua Bengio, “The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation,” in *CVPRW*. IEEE, 2017. [68](#)

-
- [175] Md Amirul Islam, Mrigank Rochan, Neil DB Bruce, and Yang Wang, “Gated feedback refinement network for dense image labeling,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. [68](#), [88](#)
- [176] Piotr Bilinski and Victor Prisacariu, “Dense decoder shortcut connections for single-pass semantic segmentation,” in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018. [68](#), [69](#), [88](#)
- [177] Fangting Xia, Peng Wang, Liang-Chieh Chen, and Alan L Yuille, “Zoom better to see clearer: Human and object parsing with hierarchical auto-zoom net,” in *ECCV*, 2016. [69](#)
- [178] Di Lin, Yuanfeng Ji, Dani Lischinski, Daniel Cohen-Or, and Hui Huang, “Multi-scale context intertwining for semantic segmentation,” in *ECCV*, 2018. [69](#)
- [179] Shu Kong and Charless C Fowlkes, “Recurrent scene parsing with perspective understanding in the loop,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 956–965. [70](#), [91](#)
- [180] Tsung-Wei Ke, Jyh-Jing Hwang, Ziwei Liu, and Stella X Yu, “Adaptive affinity fields for semantic segmentation,” in *ECCV*, 2018. [70](#)
- [181] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia, “Psanet: Point-wise spatial attention network for scene parsing,” in *ECCV*, 2018. [70](#)
- [182] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio, “On the difficulty of training recurrent neural networks,” in *International Conference on Machine Learning*, 2013, pp. 1310–1318. [75](#), [80](#)
- [183] Pushmeet Kohli, Philip HS Torr, et al., “Robust higher order potentials for enforcing label consistency,” *International Journal of Computer Vision*, vol. 82, no. 3, pp. 302–324, 2009. [85](#)
- [184] Di Lin, Yuanfeng Ji, Dani Lischinski, Daniel Cohen-Or, and Hui Huang, “Multi-scale context intertwining for semantic segmentation,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. [88](#)

- [185] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun, “Large kernel matters—improve semantic segmentation by global convolutional network,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4353–4361. [91](#)
- [186] Zifeng Wu, Chunhua Shen, and Anton Van Den Hengel, “Wider or deeper: Revisiting the resnet model for visual recognition,” *Pattern Recognition*, 2019. [91](#)
- [187] Tsung-Wei Ke, Jyh-Jing Hwang, Ziwei Liu, and Stella X. Yu, “Adaptive affinity fields for semantic segmentation,” in *ECCV*, 2018. [91](#)
- [188] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia, “PSANet: Point-wise spatial attention network for scene parsing,” in *ECCV*, 2018. [91](#)
- [189] Jun Hao Liew, Scott Cohen, Brian Price, Long Mai, Sim-Heng Ong, and Jiashi Feng, “Multiseg: Semantically meaningful, scale-diverse segmentations from minimal user input,” in *The IEEE International Conference on Computer Vision*, 2019. [99](#)
- [190] Ting Yao, Yingwei Pan, Yehao Li, Zhaofan Qiu, and Tao Mei, “Boosting image captioning with attributes,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4894–4902. [99](#)
- [191] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo, “Image captioning with semantic attention,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4651–4659. [99](#)
- [192] Xue Bai and Guillermo Sapiro, “Geodesic matting: A framework for fast interactive image and video segmentation and matting,” *International journal of computer vision*, vol. 82, no. 2, pp. 113–132, 2009. [107](#)
- [193] Varun Gulshan, Carsten Rother, Antonio Criminisi, Andrew Blake, and Andrew Zisserman, “Geodesic star convexity for interactive image segmentation,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 3129–3136. [107](#)
- [194] Kevin McGuinness and Noel E Oconnor, “A comparative evaluation of interactive segmentation algorithms,” *Pattern Recognition*, vol. 43, no. 2, pp. 434–444, 2010. [106](#)

- [195] Ruiyu Li, Kaican Li, Yi-Chun Kuo, Michelle Shu, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia, “Referring image segmentation via recurrent refinement networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5745–5753. [109](#)
- [196] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár, “Panoptic segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9404–9413. [115](#)
- [197] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun, “Upsnet: A unified panoptic segmentation network,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8818–8826. [115](#)