B-CARAFE: Better Content-Aware ReAssembly of FEatures

20220848 Minsu Sun
POSTECH CSE
minsu.sun@postech.ac.kr

Abstract

CARAFE:Content-Aware ReAssembly of FEatures [4] has shown improvement on object detection, instance segmentation, semantic segmentation and image inpainting with contextual information injection. Although CARAFE enables a large receptive field with Kernel Prediction Module, reassembled feature map in manner of content-awareness do actually saturate the region of background beside the the object itself. We propose B-CARAFE, which is improved approach of CARAFE with adopting activation function in feature reassembly step to suppress the saturation of background in feature map retaining the advantage of performance improvement with negligible additional computation.

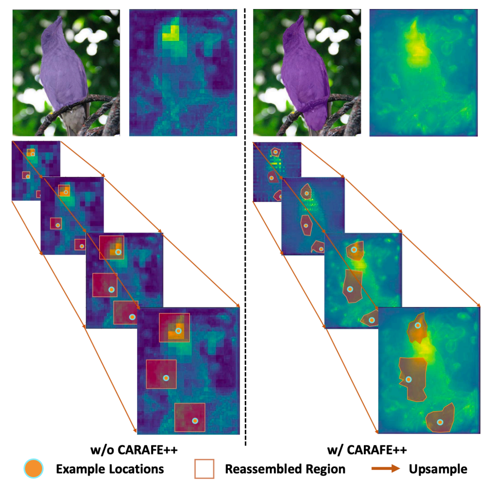

Figure 1. Comparison of Baseline and CARAFE++ [4] Left: Multi-level FPN [6] features from Mask R-CNN baseline (left to dotted line) and Right: Multi-level FPN features from Mask R-CNN with CARAFE++(right to dotted line).

1. Introduction

Content-aware feature reassembly shown by CARAFE [4] has improved feature reassembly step by adopting kernel prediction module and content-aware reassembly module. CARAFE clearly showed consistent and substantial gains with negligible computational overhead. In order to enhance the performance improvement of CARAFE keeping the advantage of negligible computational overhead, aggregating contextual information retaining a large receptive field is critical issue.

CARAFE used kernel prediction module to extract the contextual information from feature map and content-aware reassembly module to aggregate contextual information to the feature map. Re-assembled region by CARAFE showed significant improvement compared to the baseline while the background of object is highly saturated compared to the original baseline, which is unwanted result like what Fig. 1 shows. To suppress the background saturation beside the object in feature map, we adopt activation functions between levels. Experimental result shows that adopting B-CARAFE instead of CARAFE showed improvement.

2. Adopting Activation Function to CARAFE

In order to suppress the background saturation of feature map, it needs to use contextual information for enhancing only objects.

2.1. Original

Original structure of CARAFE [1] does not have any activation function.

2.2. ReLU(Rectified Linear Unit) [2]

$$ReLU(x) = max(0, x)$$

ReLU blocks all signals for negative inputs and releases for positive inputs. The mechanism for ReLU enables CARAFE to use only object concentrated contextual information.

Method	Backbone	Task	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	FPS
Faster R-CNN w/ CARAFE++	ResNet-50	BBox	22.5	39.5	23.1	12.6	24.7	29.6	12.54 fps
Faster R-CNN w/ B-CARAFE(ReLU)	ResNet-50	BBox	21.7	38.9	22.1	13.6	24.3	27.3	13.26 fps
Faster R-CNN w/ B-CARAFE(GELU)	ResNet-50	BBox	23.7	41.4	24.6	13.2	24.9	30.7	13.25 fps

Table 1. Benchmark results on MS COCO 2017

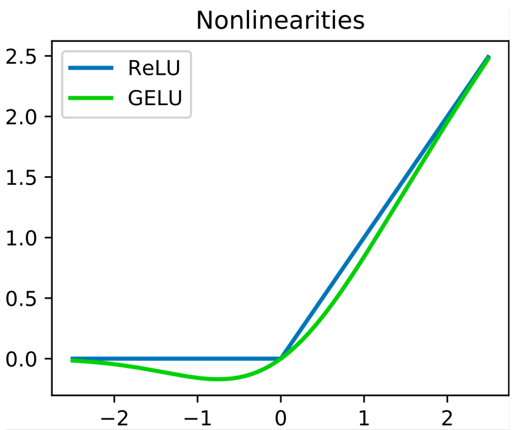

Figure 2. Comparison of ReLU(Rectified Linear Unit) and GELU(Gaussian Error Linear Unit)

2.3. GELU(Gaussian Error Linear Unit) [3]

$$GELU(x) \approx \; 0.5x(1 + \tanh[\sqrt{2/\pi}(x + 0.044715x^3)])$$

GELU, the variation of ReLU, releases signals smoothly compared to the ReLU. The mechanism for GELU enables CARAFE to use object concentrated contextual information but with more global context than ReLU does.

3. Experiments

3.1. Experiment Details

DataSet & Evaluation Metrics(Object Detection) We evaluated B-CARAFE on MS COCO 2017 dataset [7]. Results are evaluated with the standard COCO metric, i.e. AP of IoUs from 0.5 to 0.95.
Implementation Details We evaluated B-CARAFE on Faster R-CNN with the ResNet-50 [5] w/ FPN [6] backbone. Remaining implementations are specified in MMDetction [1].

3.2. Benchmark Results

We first evaluate CARAFE++ in Faster R-CNN in object detection as a baseline.

We first replace CARAFE++ with our B-CARAFE(ReLU). As shown in Tab. 1, B-CARAFE(ReLU) does not improve CARAFE++.

We replace CARAFE++ with out B-CARAFE(GELU) for second experiment. As shown in Tab. 1, B-CARAFE(GELU) improved CARAFE++ by 1.2% (i. e. , from 22.5% to 23.7%) on AP.

4. Conclusion

We have presented Better Content-Aware ReAssembly of FEatures (B-CARAFE), substantially improved version of CARAFE. It boosts the performance gain trading off negligible additional computation by adopting CARAFE. Adopting well-optimized activation to B-CARAFE will significantly boost up the performance

References

[1] https://github.com/open-mmlab/mmdetection. 1, 2

[2] Abien Fred M. Agarap. Deep learning using rectified linear units (relu), 2018. 1

[3] Kevin Gimpel Dan Hendrycks. Gaussian error linear units (gelus), 2016. 2

[4] Rui Xu Ziwei Liu Chen Change Loy Jiaqi Wang, Kai Chen and Dahua Lin. Carafe++: Unified content-aware reassembly of features, 2020. 1

[5] Shaoqing Ren Kaiming He, Xiangyu Zhang and Jian Sun. Deep residual learning for image recognition, 2016. 2

[6] Ross Girshick Kaiming He Bharath Hariharan Tsung-Yi Lin, Piotr Dollar and Serge Belongie. Feature pyramid networks for object detection, 2017. 1, 2

[7] Serge Belongie James Hays Pietro Perona Deva Ramanan Piotr Dolla´r Tsung-Yi Lin, Michael Maire and C Lawrence Zitnick. Mi- crosoft coco, 2014. 2