PUZZLE-CAM: IMPROVED LOCALIZATION VIA MATCHING PARTIAL AND FULL FEATURES

Sanghyun Jo[*]

GYNetworks
josanghyeokn@gynetworks.com

In-Jae Yu

Korea Advanced Institute of Science and Technology
School of Computing
myhome9830@kaist.ac.kr

ABSTRACT

Weakly-supervised semantic segmentation (WSSS) is introduced to narrow the gap for semantic segmentation performance from pixel-level supervision to image-level supervision. Most advanced approaches are based on class activation maps (CAMs) to generate pseudo-labels to train the segmentation network. The main limitation of WSSS is that the process of generating pseudo-labels from CAMs that use an image classifier is mainly focused on the most discriminative parts of the objects. To address this issue, we propose Puzzle-CAM, a process that minimizes differences between the features from separate patches and the whole image. Our method consists of a puzzle module and two regularization terms to discover the most integrated region in an object. Puzzle-CAM can activate the overall region of an object using image-level supervision without requiring extra parameters. In experiments, Puzzle-CAM outperformed previous state-of-the-art methods using the same labels for supervision on the PASCAL VOC 2012 dataset. Code associated with our experiments is available at https://github.com/OFRIN/PuzzleCAM.

Index Terms— Image segmentation, Deep learning, Neural Networks, Weakly-supervised semantic segmentation

1. INTRODUCTION

Semantic segmentation is a fundamental approach using convolutional neural networks (CNNs) with the aim of correctly predicting the pixel-wise classification of an image. Recently, fully-supervised semantic segmentation (FSSS) has achieved remarkable progress [1, 2, 3]. However, producing large-scale training datasets with precise pixel-level annotations per image is considerably expensive and requires labor-intensive and time-consuming tasks. To solve this issue, many researchers have focused on weakly supervised semantic segmentation (WSSS), which is used to train networks using image-level annotations, scribbles, bounding boxes, and points. Image-level supervision can be more easily conducted than other approaches in a group of weak supervision processes. In this study, we only focused on learning semantic segmentation models using image-level supervision.

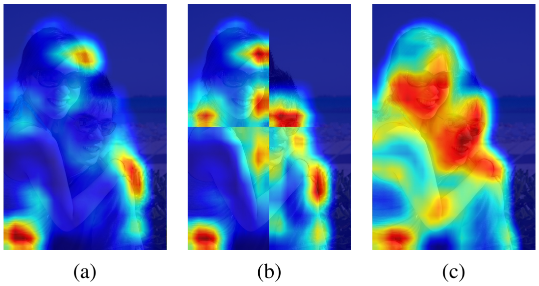


Fig. 1: CAMs generated from tiled and original images: (a) conventional CAMs from the original image, (b) generated CAMs from the tiled images, and (c) predicted CAMs by the proposed Puzzle-CAM.

Most previous methods [4, 5, 6] using WSSS are based on the class activation maps (CAMs) [7] to achieve good performance. However, the generated CAMs are usually focused on small parts of the semantic objects to efficiently classify them, which prevents the segmentation models from learning pixel-level semantic knowledge. Moreover, we can see that the CAMs generated from isolated patches in the tiled image are different from those gained from the original image. As shown in Fig. 1, CAMs of the tiled image comprising tiled patches are significantly inconsistent compared to those of the original image. The differences are factored in by enlarging the supervision gap between FSSS and WSSS by even more.

The above observations gave us the inspiration to address WSSS issues by using an attention-based feature learning method. Thus, we propose Puzzle-CAM for WSSS training to detect integrated regions of objects. Our method applies reconstructing regularization that corresponds to the generated CAMs from the tiled and original images to provide self-supervision. To improve the network prediction consistency further, we introduced a puzzle module that splits the

[*] Thanks to GYNetworks for funding.

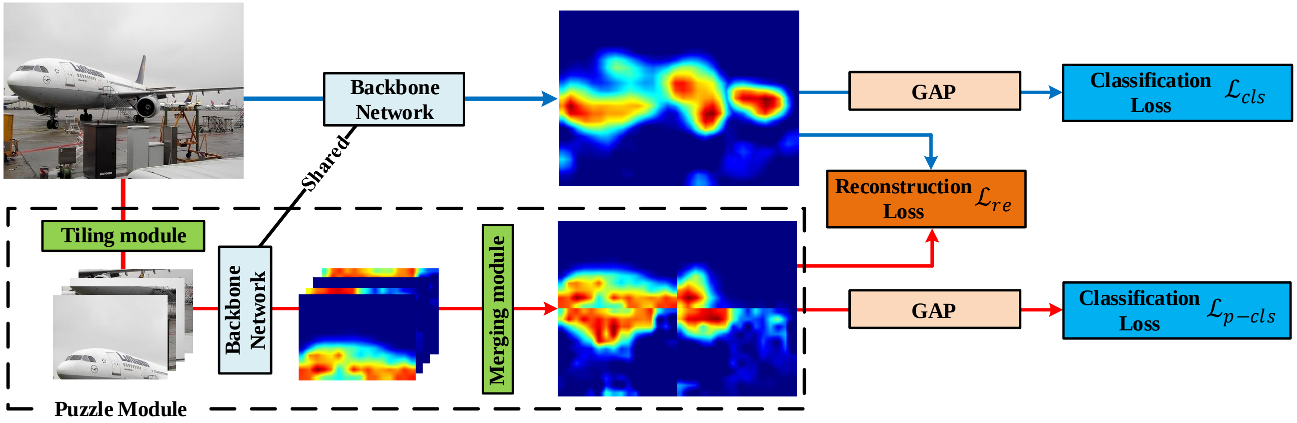

Backbone Network
Shared
GAP
Classification Loss $\mathcal{L}_{cls}$
Reconstruction Loss $\mathcal{L}_{re}$
Tiling module
Backbone Network
Merging module
GAP
Classification Loss $\mathcal{L}_{p-cls}$
Puzzle Module

Fig. 2: The overall architecture of the proposed Puzzle-CAM showing the integration of reconstructing regularization and the puzzle module.

image and merges CAMs generated from the tiled image. Puzzle-CAM consists of a Siamese neural network with reconstructing regularization loss that reduces the differences between the original and merged CAMs. Our experiments yielded both quantitative and qualitative results that demonstrate the superiority of our approach.

Our main contributions are as follows:

- We propose Puzzle-CAM that incorporates reconstructing regularization with a puzzle module, to effectively enhance the quality of CAMs without adding layers.
- Puzzle-CAM outperformed existing state-of-the-art methods with the same level of supervision on the PASCAL VOC 2012 dataset.

2. RELATED WORK

2.1. Attention Mechanisms Using CNNs

These provides fine-grained information on the features learned in CNNs. Simonyan *et al.* [8] used the error back-propagation strategy to visualize semantic regions whereas the combined attention model used the global average pooling (GAP) layer in the CNNs to generate the CAMs [7] more efficiently. Last, a final classifier is used to generate attention maps. To the best of our knowledge, which attention mechanism is chosen does not have a great effect on achieving high performance with WSSS, and so we based Puzzle-CAM on the combined attention model because it is more manageable than the other attention mechanism.

2.2. Weakly Supervised Semantic Segmentation

Unlike FSSS, which requires pixel-wise labels for an image, WSSS employs lower level labeling, such as bounding boxes [9], scribbles [10], and image-level classification labels [4, 6]. Recently, the performance of WSSS has been significantly boosted by incorporating the CAMs. Most previous WSSS methods refine the CAMs generated by the image classifier to approximate the segmentation mask [4, 11, 12, 13, 6]. AffinityNet [4] trains an additional network to learn similarities between the pixels, which often generates a transition matrix and multiplies with CAM to adjust its activation coverage. IRNet [11] generates a transition matrix from the boundary activation map and extends the method to achieve weakly supervised instance segmentation and WSSS. SEAM [5] aims to refine CAMs using a pixel correlation module that captures context appearance information for each pixel and alters the original CAMs by using learned affinity attention maps.

3. METHODOLOGY

3.1. Motivation

Most WSSS approaches are based on CAMs to obtain a segmentation mask using image-level supervision. Usually, the CAMs are focused on the discriminative region of the object. The well-known reason is that a normal image classifier only uses the classification loss, which induces a partial region that activated during training. The CAMs generated from a tiled image causes over-activation since the image does not have global-context information. To address this issue, we propose Puzzle-CAM that improves the network for consistent prediction to match partial and full features. Puzzle-CAM contains designed loss functions to match the CAMs generated from a tiled image with the original image (Fig. 2).

3.2. The Employed CAM Method

We first introduce the CAM method for producing the initial attention map. Given feature extractor F and classifier θ, we can generate CAMs A that are the collection of CAMs for all of the classes. After training the classifier by image-level supervision, we apply the weights of the c-channel classifier as θ^c on feature map $f = F(I)$ from input image I to obtain the CAM of class c as follows:

$$A_c = \theta_c^\top f. \tag{1}$$

The generated CAM is normalized by using the maximum value of A_c. Finally, we obtain the CAMs for all of the classes (A) by concatenating A_c.

3.3. The Puzzle Module

When matching partial and full features, the key is to narrow the gap between FSSS and WSSS. The puzzle module consists of tiling and merging modules. From input image I of size $W \times H$, the tiling module generates non-overlapping a tiled patches $\{I^{1,1}, I^{1,2}, I^{2,1}, I^{2,2}\}$ of size $W/2 \times H/2$. Next, we generate $A^{i,j}$ CAMs for each $I^{i,j}$. Finally, the merging module attaches all $A^{i,j}$ into a single CAMs A^{re} that has the same shape as A^s which is the CAMs of I.

3.4. The Loss Design for Puzzle-CAM

We employed a GAP layer at the end of the network to incorporate prediction vector $\hat{Y} = \sigma(G(A_c))$ for image classification and to adopt multi-label soft margin loss for the classification task. For notational convenience, we define Y_t as

$$\hat{Y}_t = \begin{cases} \hat{Y}, & \text{if } Y = 1 \\ 1 - \hat{Y}, & \text{otherwise} \end{cases} \tag{2}$$

$$\ell_{cls}(\hat{Y}, Y) = -log(Y_t). \tag{3}$$

The CAMs of the original (A^s) and tiled (A^{re}) images are converted using the GAP layer with prediction vectors $\hat{Y}^s = G(A^s)$ and $\hat{Y}^{re} = G(A^{re})$, respectively. The classification losses for the original and reconstructed images are respectively calculated as

$$\mathcal{L}_{cls} = \ell_{cls}(\hat{Y}^s, Y), \tag{4}$$

$$\mathcal{L}_{p-cls} = \ell_{cls}(\hat{Y}^{re}, Y). \tag{5}$$

These two classification losses are used to improve the performance of the image classification. To reinforce the CAMs from the original image, we added reconstructing regularization to correspond with the original and reconstructed CAMs. The reconstruction loss for the original CAM can be easily defined as

$$\mathcal{L}_{re} = \left\| A^s - A^{re} \right\|_1. \tag{6}$$

Table 1: Ablation study of the Puzzle-CAM loss functions using ResNet-50 as the backbone.

L_{cls}	L_{p-cls}	L_{re}	mIoU (%)
✓			47.82
✓	✓		47.70
✓		✓	49.21
✓	✓	✓	**51.53**

In summary, the final loss of Puzzle-CAM can be written as

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{p-cls} + \alpha\mathcal{L}_{re}. \tag{7}$$

where α is the balance of the weights for the different losses. The classification losses, $\mathcal{L}_{cls}$ and $\mathcal{L}_{p-cls}$, are used to roughly estimate the region of the object. The reconstruction loss, $\mathcal{L}_{re}$, is used to narrow the gaps between the pixel- and image-level supervision processes. We report details of the network training settings and probe into the effectiveness of the proposed module in the experiments section.

4. EXPERIMENTAL RESULTS

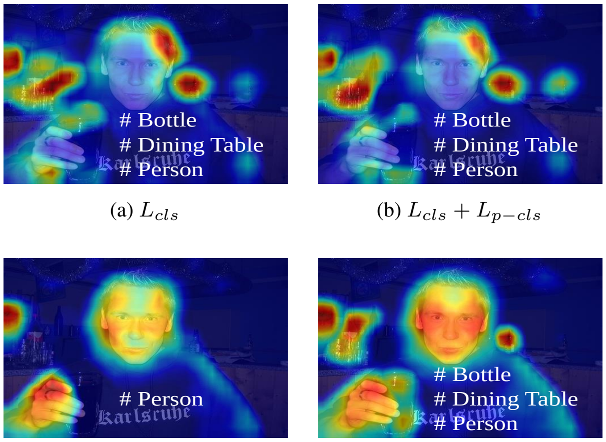
4.1. Implementation Details

We evaluated our method using the PASCAL VOC 2012 dataset [14]. The dataset was separated into 1,464 images for training, 1,449 for validation, and 1,456 for testing. Following the experimental protocol used in previous methods [4, 5, 6], we took additional annotations from the Semantic Boundary Dataset [15] to build an augmented training set with 10,582 images. The images were randomly re-scaled in the range of [320, 640] and then cropped to 512 × 512 as the network inputs. For all experiments, we set $\alpha = 4$ as the maximum and linearly ramped up α to its maximum value by half epochs. During inference, we utilized classifiers without the puzzle module. Thus, we adopted multi-scale and horizontal flip to generate pseudo-segmentation labels. We used four TITAN-RTX GPUs for the model to train the dataset.

4.2. Ablation Studies

We conducted ablation studies on the main components of Puzzle-CAM by applying the mean Intersection-Over-Union (mIoU) metric (Table 1), for which the baseline was $mIoU = 47.82\%$. With the proposed reconstructing regularization ($\mathcal{L}_{re}$) of the tiled patches, the baseline was boosted to $mIoU = 49.21\%$, while the proposed classification loss from the tiled patches ($\mathcal{L}_{p-cls}$) was similar to the baseline. Both $\mathcal{L}_{re}$ and $\mathcal{L}_{p-cls}$ consistently improved the baseline by a 3.71%.

We visualized the CAMs by using combinations of loss functions for each of them (see Eq. 3). When the classification losses (L_{p-cls}) only are employed, the result will show no marginal differences. Meanwhile, when the reconstruction loss (L_{re}) only is employed, the result will show improved



(a) L_{cls} (b) $L_{cls} + L_{p-cls}$

(c) $L_{cls} + L_{re}$ (d) $L_{cls} + L_{p-cls} + L_{re}$

Fig. 3: Visualization of the prediction tags and CAMs by using combinations of loss functions. In (d), the final CAMs not only suppressed over-activation but also expanded the CAMs into complete object activation coverage .

Table 2: Quality of the pseudo semantic segmentation labels in mIoU, evaluated on the PASCAL VOC 2012 training set [14]. RW, random walk with AffinityNet [4]; dCRF, dense conditional random field [16].

Method	Backbone	CAM (%)	CAM +RW (%)	CAM+RW +dCRF (%)
AffinityNet [4]	ResNet-50	47.82	58.10	59.70
Puzzle-CAM	ResNet-50	51.53	64.16	64.70
Puzzle-CAM	ResNeSt-50	57.59	69.48	69.91
Puzzle-CAM	ResNeSt-101	61.85	71.92	72.46
Puzzle-CAM	ResNeSt-269	**62.45**	**74.14**	**74.67**

localization ability for some classes compared to the original but the method will fail to predict several classes. When both sets of losses are combined, the result will show improved localization without suffering classification losses.

4.3. Comparison with Existing State-of-the-Art Methods

To further improve the accuracy of pseudo pixel-level annotations, we followed the approach in [4] to train AffinityNet based on Puzzle-CAM. We adopted ResNeSt architecture that universally improves the learned feature representations to boost performance across image classification, object detection, instance segmentation and semantic segmentation. In Table 2, we report the performances with the original CAMs used by the baseline AffinityNet [4] and Puzzle-CAM.

The final synthesized pseudo-labels achieved 74.67% mIoU on the PASCAL VOC 2012 *train* set. Puzzle-CAM was then used to train the segmentation model DeepLabv3+ [1] with the ResNeSt-269 [18] backbone using the pseudo-labels in full supervision mode to achieve the final segmentation results. Table 3 reports a comparison of the mIoU values for the proposed method and the previous approaches. Compared to the baseline methods, Puzzle-CAM had remarkably improved performances on both the *val* and *test* sets with the same settings for training. Fig. 4 shows some qualitative results on the *val* set that illustrate that the proposed method worked well on both large and small objects.

Table 3: Comparison of Puzzle-CAM and existing state-of-the-art methods on the PASCAL VOC 2012 *val* and *test* datasets. $\mathcal{I}$, image-level labels; $\mathcal{S}$, external saliency models.

Method	Backbone	Supervision	val	test
AffinityNet [4]	Wide-ResNet-38	$\mathcal{I}$	61.7	63.7
DSRG [12]	ResNet-101	$\mathcal{I}+\mathcal{S}$	61.4	63.2
SeeNet [13]	ResNet-101	$\mathcal{I}+\mathcal{S}$	63.1	62.8
IRNet [4]	ResNet-50	$\mathcal{I}$	63.5	64.8
FickleNet [6]	ResNet-101	$\mathcal{I}+\mathcal{S}$	64.9	65.3
ICD [17]	ResNet-101	$\mathcal{I}$	64.1	64.3
SEAM [5]	Wide-ResNet-38	$\mathcal{I}$	64.5	65.7
Ours (Puzzle-CAM)	ResNeSt-101	$\mathcal{I}$	66.9	67.7
Ours (Puzzle-CAM)	ResNeSt-269	$\mathcal{I}$	**71.9**	**72.2**

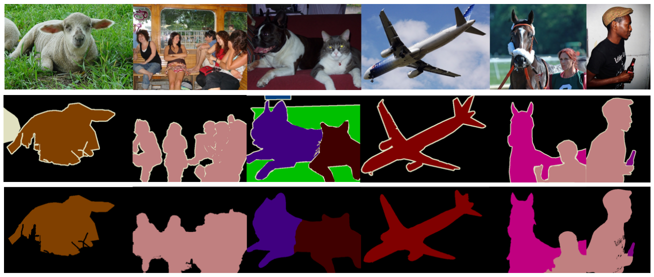


Fig. 4: Qualitative segmentation results on the PASCAL VOC 2012 *val* set. Top: original images. Middle: ground truth. Bottom: prediction of the segmentation model trained using the pseudo-labels from Puzzle-CAM.

5. CONCLUSIONS

In this paper, we proposed the Puzzle-CAM algorithm to narrow the supervision gap between FSSS and WSSS using image-level labels. To improve the network for generating consistent CAMs, we designed a puzzle module and adopted reconstructing regularization to match partial and full features. Not only did Puzzle-CAM consistently generate features from local tiled patches but it also fitted the shape of the ground truth masks better. The segmentation network trained by our synthesized pixel-level pseudo-labels achieved state-of-the-art performance on the PASCAL VOC 2012 dataset, which proves the effectiveness of our approach. We believe that the concepts of Puzzle-CAM as a training module can be generalized and will benefit other weakly- and semi-supervised tasks, such as semantic and instance segmentation.

6. REFERENCES

[1] 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.

[2] Jonathan Long, Evan Shelhamer, and Trevor Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille, “Semantic image segmentation with deep convolutional nets and fully connected crfs,” *arXiv preprint arXiv:1412.7062*, 2014.

[4] Jiwoon Ahn and Suha Kwak, “Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4981–4990.

[5] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen, “Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[6] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon, “Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 5267–5276.

[7] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba, “Learning deep features for discriminative localization,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929.

[8] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman, “Deep inside convolutional networks: Visualising image classification models and saliency maps,” *arXiv preprint arXiv:1312.6034*, 2013.

[9] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele, “Simple does it: Weakly supervised instance and semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 876–885.

[10] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun, “Scribblesup: Scribble-supervised convolutional networks for semantic segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3159–3167.

[11] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak, “Weakly supervised learning of instance segmentation with inter-pixel relations,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2209–2218.

[12] Zilong Huang, Xinggang Wang, Jiasi Wang, Wenyu Liu, and Jingdong Wang, “Weakly-supervised semantic segmentation network with deep seeded region growing,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7014–7023.

[13] Qibin Hou, PengTao Jiang, Yunchao Wei, and Ming-Ming Cheng, “Self-erasing network for integral object attention,” in *Advances in Neural Information Processing Systems*, 2018, pp. 549–559.

[14] 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, pp. 303–338, 2010.

[15] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik, “Semantic contours from inverse detectors,” in *2011 International Conference on Computer Vision*. IEEE, 2011, pp. 991–998.

[16] Philipp Krähenbühl and Vladlen Koltun, “Efficient inference in fully connected crfs with gaussian edge potentials,” *Advances in neural information processing systems*, vol. 24, pp. 109–117, 2011.

[17] Junsong Fan, Zhaoxiang Zhang, Chunfeng Song, and Tieniu Tan, “Learning integral objects with intra-class discriminator for weakly-supervised semantic segmentation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4283–4292.

[18] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al., “Resnest: Split-attention networks,” *arXiv preprint arXiv:2004.08955*, 2020.