2020 International Conference on Machine Vision and Image Processing (MVIP)
Faculty of Engineering, College of Farabi, University of Tehran, Iran, 19 & 20 Feb. 2020

Class Attention Map Distillation for Efficient Semantic Segmentation

Nader Karimi Bavandpour
Department of Computer Engineering
Sharif University of Technology
Tehran, Iran
nkarimi@ce.sharif.edu

Shohreh Kasaei, *IEEE, Senior Member*
Department of Computer Engineering
Sharif University of Technology
Tehran, Iran
kasaei@sharif.edu

***Abstract*—In this paper, a novel method for capturing the information of a powerful and trained deep convolutional neural network and distilling it into a training smaller network is proposed. This is the first time that a saliency map method is employed to extract useful knowledge from a convolutional neural network for distillation. This method, despite of many others which work on final layers, can successfully extract suitable information for distillation from intermediate layers of a network by making class specific attention maps and then forcing the student network to mimic producing those attentions. This novel knowledge distillation training is implemented using state-of-the-art DeepLab and PSPNet segmentation networks and its effectiveness is shown by experiments on the standard Pascal Voc 2012 dataset.**

***Index Terms*—Semantic Segmentation, Knowledge Distillation, Saliency Maps.**

I. Introduction

Classification is a basic problem in machine learning and many other practical problems involve classification as a part of their solutions. Semantic segmentation is the problem of classifying each pixel of a given image into its related category, hence it needs to deal with spatial information that is required in the output besides the global context information for classifying each pixel. This task has many real-world applications like autonomous driving, robot navigation, and medical image processing.

Traditional segmentation methods use handcrafted features like HoG and SIFT, and then feed those features into classifiers like random forests and SVMs [1]. In the past recent years, Convolutional Neural Networks (CNNs) faced numerous design improvements that lead to outperforming traditional methods in various fields by a large margin, specially in computer vision and natural language processing. Most of the current state-of-the-art methods for semantic segmentation are based on CNNs. Although their performance is satisfactory in terms of accuracy, they often have tens of millions of parameters, high memory usage, and low test speed. This is because deep networks tend to be easier to optimize and shallow networks may not reach their true potential in a typical training setting [1]. One way of dealing with this problem is to use a trained and deep network (teacher) to help the shallow network (student) in the training stage. This is typically performed by using knowledge distillation methods, first introduced in [11], that add additional loss functions between the teacher and student network (beside the standard loss of the task).

Several methods of performing distillation are introduced in semantic segmentation, most of which focus on distilling knowledge from the last convolutional feature maps of each network. In this work, it is realized that saliency map techniques are overlooked for their potential to perform useful encoding of each network's knowledge, specially in their intermediate layers, to be used in knowledge distillation.

In summary, the main contributions of this work are as follows:

- Introducing a new method of attention transfer using class-specific attention maps.
- Presenting the reason that the method is able to successfully distill knowledge from intermediate layers of two networks.
- Showing that the method can be naturally implemented on two state-of-the-art segmentation networks; namely, DeepLab [3] and PSPNet [29].
- Validating the effectiveness of the method on the Pascal Voc [6] dataset.

II. Related Work

Three relevant research areas that are essential to understand the proposed method are briefly reviewed in this section.

Semantic Segmentation: Semantic segmentation is known as a challenging task, which is about to combine global information with detailed local information to predict the structure of an input image in terms of classifying pixels to categories. Semantic segmentation networks are generally larger than classification ones, as they have to extract additional information beside the information needed for classification. The fully convolutional framework, first introduced in [15], added several important improvements to segmentation network design. It can use pretrained weights of classification networks, perform on variable input size, and be trained end-to-end. Because of the vast applications of semantic segmentation, speed versus accuracy trade-off has become another aspect of segmentation network design. Compact and fast networks are important for real-time tasks that use semantic segmentation. Designing such networks is an active area of research; some of the notable methods include [16], [17], [25].

DeepLab (version 3) [3] and PSPNet [29] are two of the most

978-1-7281-6832-6/20/$31.00 ©2020 IEEE

powerful and popular existing segmentation networks. Due to their flexible design, one can choose big and powerful or small and efficient classifier networks as their backbones. They use Atrous convolution and pyramid spatial pooling to capture global context while preserving feature maps' resolution and details. In this work, DeepLab with ResNet101 [9] backbone is adopted as the teacher and PSPNet with ResNet18 backbone is chosen as the student network.

Saliency Maps: Interpretability is considered important in machine learning models because humans need and want to know how an algorithm is making its decisions. It is one of the big concerns about CNNs and they are sometimes referred to as black boxes. Saliency maps are methods in the direction of knowing more about how CNNs work. They often process a neuron or group of neurons from hidden layers to create an attention map in the input space. Activation maximization [5] is the earliest work to visualize a hidden neuron by optimization in the input space to maximize that neuron. Deconvnet [27] tries to approximate the inverse of a network's layers by introducing deconvnet layers. [7] introduces a strong and model free method. It performs optimization on the input space to delete information from the input image as much as possible while preserving model's decision. This method seems to be the most clear one in terms of interpretability, but it is slower than the other ones. [22] and [23] were the first works that proposed taking gradient of neurons with respect to the input layer and visualizing them in networks with ReLu activation function. This results in high resolution saliency maps, but they are not class discriminative. [30] introduced Class Activation Maps (CAMs), which works on classification networks that perform global average pooling followed by a linear layer after their final convolutional layer. This way, the weights of the linear layer for each output class can be used as weights for summing the channels of the last convolutional feature map. The outcomes would have a low resolution, but they are class discriminative. [20] used the method of [23] to obtain gradients in the desired layer and then used them as weights to to perform the weighted sum on a feature map. This lead to a valuable generalization of CAM that could generate class specific attention maps regardless of the architecture of a network.

Knowledge Distillation: The idea of knowledge distillation first appeared in [11], where the student network uses the teacher's predictions as soft labels (compared to zero and one hard label of ground-truth). Soft labels hold useful information about the structure of a problem and relationships between the categories and provide useful information for training the student. The teacher and student framework is widely used for helping to train compact students. There are also numerous other scenarios where it comes useful. For instance, here is a list of some of the most notable related works:

- [18] trains a deep but thin student using a big shallow teacher.
- [1] uses the teacher to label unlabeled data and uses that additional labeled data to train the student.
- [13] uses distillation to teach an already trained network to classify additional classes without losing performance on the old ones.
- [28] trains an ensemble of students that transfer information to each other during the training phase.
- [8] trains a sequence of identical networks in such a way that each network distills from the previously trained one and this leads to improved performance.
- [21] combines a number of classifiers which are expert at different domains of labels to get one model that is as good as all of those experts at each domain.

Most of the discussed methods are designed for image classification, but [13] applied its method for object detection as well. Other examples of successful work on object detection include [12] and [2]. After classification and detection, one of the first applications of distillation to semantic segmentation was introduced in [19]. They used the prediction of the teacher instead of the ground-truth for training the student. It led to better results because the teacher's output is an easier distribution to learn. Authors of [4] used distillation between feature maps of a freezed and pretrained network on the ImageNet (real data) and a student network which uses synthetic data for training. As such, they reduced the overfitting of the student to synthetic textures. [24] introduced the consistency loss between the student and the teacher to make their segmentation boundary similar. It also takes the L2 norm of the difference between student's output probabilities and the teacher's as another loss. [14] introduced two novel distillation losses in segmentation. Pairwise loss is defined as the mean square distance between elements of affinity matrices of the teacher and student networks (affinity matrix contains inner products between every pair of features which encode pixels). The second loss is called holistic distillation, which uses adversarial learning to make feature maps of student similar to its teacher's, using a discriminator convolutional network. [10] is another relevant work that was developed parallel with [14]. It uses an affinity loss which is almost the same as the pairwise loss in [14], except that they train an auto-encoder for the last convolutional layer of their teacher network before computing its affinity matrix. They also use the direct L2 norm distance between the student's last convolutional features and the teacher's encoded features as an additional loss.

III. Proposed Method

In this section, the proposed novel distillation method between two segmentation CNNs is explained. As it is mentioned in the previous section, [18] introduced a method to distill the knowledge between two feature maps using a distance which is applied directly to all of their elements. [26] showed that using an attention map which is created by taking summation of feature map channels in a specific layer with uniform weights can improve the performance boost with respect to the method of [18]. The method of [26] is called Global Attention Map (GAM) distillation in this paper, because each feature map is mapped to a single attention matrix. In this section, the idea of transferring attention maps is investigated further by creating

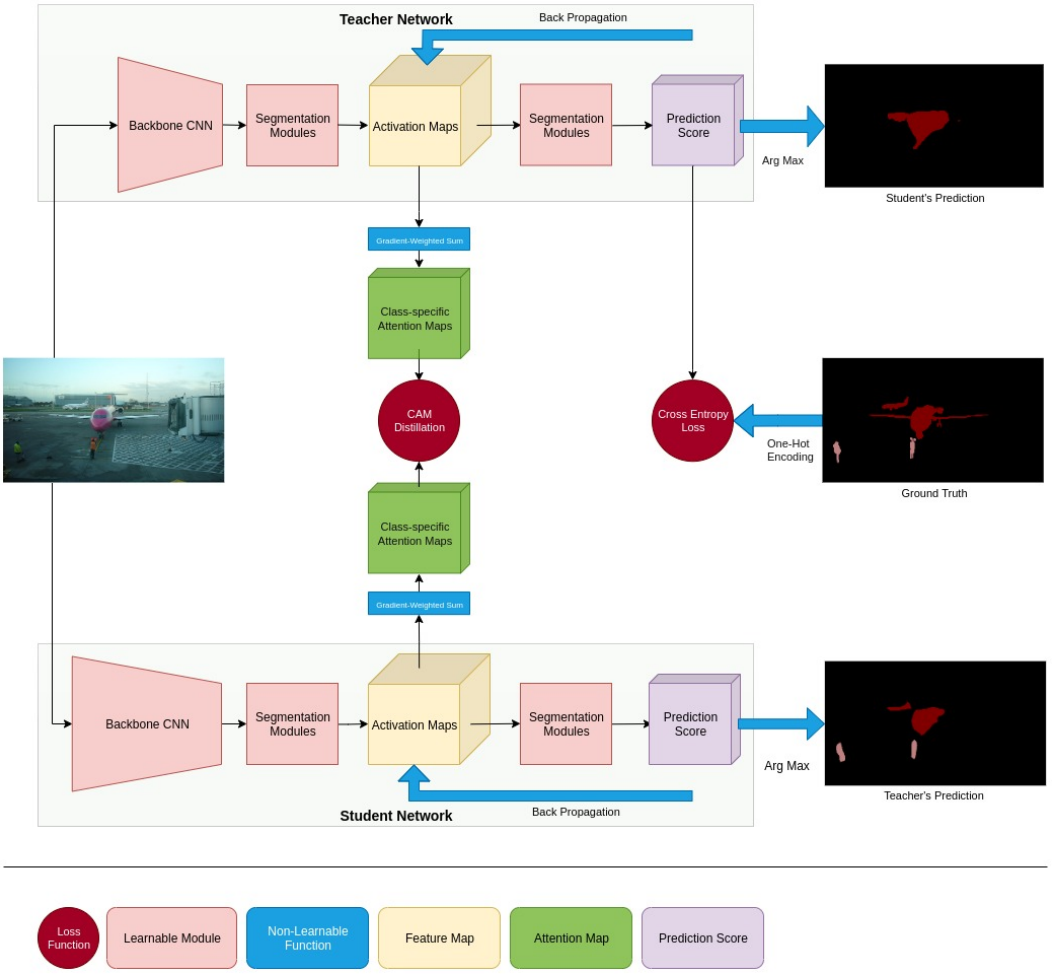

Fig. 1. CAM distillation and classic segmentation loss for training a lightweight student network with small backbone by using a strong teacher network having a bigger backbone.

class specific attention maps from each feature map, and using them for a novel distillation scenario called Class Attention Map (CAM) distillation, because it creates several attention maps from each feature map, each of which corresponds to a specific class that is predictable by the model. The high level architecture of using the CAM distillation and standard cross entropy loss for semantic segmentation is depicted in Fig. 1. In the remainder of this section, first a mathematical notation is presented and then the proposed method to create CAMs and a loss function to distill them between student and teacher networks is formally introduced.

Suppose $\mathbf{A} \in \mathbf{R}^{d \times w \times h}$ is an intermediate feature map from a segmentation network with spatial dimensions $h \times w$ and number of channels d. The notation $\mathbf{A}^k(\mathbf{x})$ is used to show the element at depth k and spatial dimensions are indexed by the vector $\mathbf{x}$, and different feature maps are indexed as $\mathbf{A_i}$. The element-wise power operator on matrix $\mathbf{A}$ is denoted by $|\mathbf{A}|^p$. The GAM attention matrix for the feature map at layer i is defined as [26]

$$\mathbf{G}_i = \sum_k |\mathbf{A_i}^k|^p \tag{1}$$

where $p = 2$ in this paper's experiments. If the $\mathbf{G}_i$ computed from the student network is denoted by $\mathbf{S}_i$, and $\mathbf{G}_i$ computed from the teacher network by $\mathbf{T}_i$, then the GAT distillation loss function can be written as [26]

$$\ell_{\text{GAT}_i} = \| \frac{\mathbf{T}_i}{\|\mathbf{T}_i\|_2} - \frac{\mathbf{S}_i}{\|\mathbf{S}_i\|_2} \|_2 \tag{2}$$

which is then used in combination with segmentation loss with a weighted sum

$$\ell_{\text{total}} = \ell_{\text{seg}} + \sum_i \lambda_i \ell_{\text{GAT}_i}. \tag{3}$$

Here, the segmentation loss ℓ_{seg} is the widely used cross entropy function between the student network's normalized predictions and ground-truth labels. The loss in (2) was originally defined for image classification, but it can be readily used in semantic segmentation, as well. Inspired by the work of [20], class specific attentions are used in this paper to introduce a novel loss function for knowledge distillation. Suppose $\mathbf{L} \in \mathbf{R}^{c \times w' \times h'}$ denotes the unnormalized output scores of a segmentation network (Logits) and c is the number

of classes that the network is able to predict. For computing class specific attentions using feature map $\mathbf{A}$, a matrix of gradients $\mathbf{W}$ is created as defined below

$$\mathbf{W}_c^k(\mathbf{x}) = \sum_{\mathbf{y}} \frac{\partial \mathbf{L}^c(\mathbf{y})}{\partial \mathbf{A}^k(\mathbf{x})}. \tag{4}$$

Although the spatial dimensions of $\mathbf{W}$ can be eliminated using a summation to get a one-dimensional weight vector, experiments show that spatial information in $\mathbf{W}$ can result in a more successful knowledge distillation training. Using the weight matrix $\mathbf{W}$, the CAM attention matrix is defined by

$$\mathcal{E}^c(\mathbf{x}) = \max\left(\sum_k \mathbf{W}_c^k(\mathbf{x}) \times \mathbf{A}^k(\mathbf{x}), 0\right). \tag{5}$$

As this equation is similar to the one used for creating saliency maps, it removes negative values to make the result more meaningful for human's understanding. Finally, the novel CAM loss function to distill knowledge between two networks is defined by

$$\ell_{\text{CAM}} = \left\| \frac{|\mathcal{E}_T|^2}{\||\mathcal{E}_T|^2\|_2} - \frac{|\mathcal{E}_S|^2}{\||\mathcal{E}_S|^2\|_2} \right\|_2^2 \tag{6}$$

in which $\mathcal{E}_T$ and $\mathcal{E}_S$ denote CAMs for the teacher and student networks, respectively. The overall loss function is a weighted sum with ℓ_{seg}, defined by

$$\ell_{\text{total}} = \ell_{\text{seg}} + \lambda \ell_{\text{CAM}}. \tag{7}$$

Note that any possible difference of spatial dimensions between attention maps of teacher and student networks is compensated by a simple operation of bilinear upsampling in both GAM and CAM distillation methods. Computing the CAM loss function requires partial back propagation to the chosen feature map for each class in both teacher and student networks, which increases the training time relative to the GAM distillation method. On the other hand, the information in the CAM matrices seem to be more architecture-independent than the GAM matrix, as they are created using a method that is designed to create saliency maps for humans. CAMs are better to be created from feature maps from intermediate layers of a network. The reason is that despite image classification networks, there are no pooling or strided convolution operations in the last layer of modern semantic segmentation networks, and this causes the receptive field to be limited there. For example Deeplab and PSPNet and many other modern segmentation networks' last operation is a 1×1 or 3×3 convolution. It is easy to see that as the convolution is a linear operation, in the case of a 1×1 convolution being the network's last layer, the gradient matrix $\mathbf{W}$ will be the weights of the convolution and (5) will simply produce the prediction score of the corresponding class scaled by the number of elements in the summation. It is worth noting that even two or more convolutional functions with limited receptive fields and nonlinearities among them may not solve this problem completely. This is because (5) will still produce a near exact first order approximation of prediction layer from them and will add only little information to what that already exist in the output score matrix. While many of other distillation methods (such as [10], [14] and even [26] as experiments of this research show) perform better on a network's last layers, the CAM distillation method can perform better on intermediate layers by design.

IV. Experimental Results

The standard Pascal Voc 2012 dataset is used to validate the proposed method. It contains 1,464 labeled images for training, 1,449 for validation, and 1,456 for test. This dataset is widely used for the semantic segmentation task and measuring the mean Intersection over Union (mIoU) metric over the validation set is usually adopted for reporting the results. There are 21 classes present in this dataset, including background class, which must be included in computing the mIoU.

The teacher networks is the Deeplab version 3 with ResNet101 backbone which has 58,630,997 trainable parameters and the student network is the PSPNet with ResNet18 backbone with 12,129,170 trainable parameters. All of the weights defined as λ in loss functions (7) and (3) are set to 1. This value is the result of trying values 10, 1, and 0.1 and choosing the best one. All of the models are trained with a similar configuration of batch size of 6, total epochs of 130, and a starting learning rate of 0.01. Each training image is preprocessed by the operations of random scaling to 0.5 to 2 times of their original size, horizontal random flip, and finally a random crop of 380×380. For validation, each image is resized so that the smaller side is 480 pixels and then a center crop of size 480×480 is taken from it. In the experiments of this work, no augmentation is added to the standard Pascal Voc dataset (as some of other papers). The teacher and student networks use the ImageNet pretrained weights in their backbones and their segmentation parts are randomly initialized.

As mentioned earlier, the CAM distillation method performs better when applied to intermediate layers instead of the last ones. Fortunately, the designs of both PSPNet and Deeplab naturally satisfy having the best possible characteristics to be used in the CAM based distillation. It is because they both place their segmentation functions after the backbone *in a parallel layer*, followed by a convolution. This means that choosing the feature maps before the parallel layers has two great benefits: i) Because the segmentation modules are parallel, it remains close to the last layer, and the partial backpropagations needed for computing CAMs will be fast. ii) Because of the sudden increase of receptive fields which is the effect of high rate dilated convolutions and global average pooling, the CAM matrix will contain rich information that is not present in the score map matrix. The experiments in this section are performed on two different layers, and in the names of the methods the middle layer refers to the layer before parallel segmentation modules, and the end layer refers to the last convolutional layer of a segmentation network.

Table I lists the results for different training scenarios. Three different random seeds are used for each experiment to make the numbers more meaningful. As the table shows, the GAT

TABLE I
AVERAGE AND STANDARD DEVIATION OF MIOU METRIC OF 3 RUNS WITH DIFFERENT RANDOM SEEDS FOR DIFFERENT TRAINING METHODS ON THE VALIDATION SET OF PASCAL VOC 2012.

Network	Avg. of mIoU	Std. of mIoU
Teacher	76.54	NA
No Distillation	66.59	0.29
GAT-Middle-End	66.84	0.34
GAT-End	67.11	0.28
Proposed CAM-End	67.06	0.36
Proposed CAM-Middle	67.25	0.29

TABLE II
AVERAGE MEMORY CONSUMPTION AND SPEED FOR TRAINING EACH BATCH OF 6 IMAGES FOR DIFFERENT TRAINING METHODS.

Network	Time (millisecond)	Memory (Megabyte)
No Distillation	170	1743
GAM-Middle-End	450	4069
GAM-End	440	4041
Proposed CAM-End	500	4031
Proposed CAM-Middle	28000	6733

distillation method performs better when applied on the last layer. On the contrary, the CAM distillation method performs the best when applied on the middle feature maps. The CAM-Middle distillation also gives the best performance boost as it can be seen in the table.

Table II lists the memory and time requirements of each method. The GAM-Middle method needs reasonably more memory and much more computations with respect to other methods. This is because partial backpropagation have to be performed 21 times (equal to the number of classes presented in the Pascal Voc) in both teacher and student networks to compute CAM matrices. Note that the GAM distillation method has a particularly simple implementation and almost all other semantic segmentation distillation loss functions are more complex. In addition, nearly none of other works report the time needed in the training stage and therefore there is no reference to compare the proposed methods' complexity with. It is worth noting that this complexity and memory burden is only in the training stage, and the proposed method adds no extra complexity or memory requirement to the student network in its test phase.

Fig. 2 shows the GAM and CAM attention maps for a typical image. While GAM provides a single non-discriminative map, each CAM provides a useful class specific attention information. It is also worth noting that experiments are performed to see if distilling the CAM matrix for only one randomly presented class, or only the classes presented in each image performs good. But, the performance boost drop meaningfully for both scenarios.

Fig. 3 shows all of CAMs for a single image. It seems to be a possibility in this images that the networks look for relevant cues for classes that are not presented in the image. This is why the distillation of CAMs for absent classes have a positive

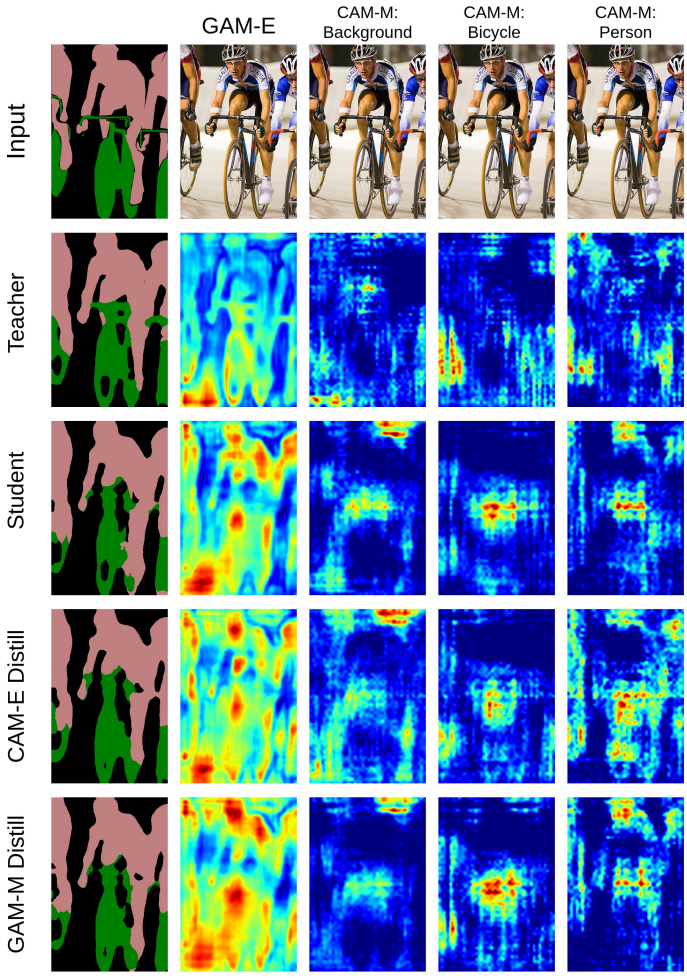

Fig. 2. Visualization of GAM and CAM matrices for an image from Pascal Voc's validation set. The post-fixes -M and -E mean that method is applied on the middle layer and end layer feature maps, respectively.

effect on the performance boost.

V. CONCLUSION AND FUTURE WORK

One of the main methods of producing saliency maps was used to distill knowledge between two networks with different architectures. Experiments showed that it can successfully boost the student network's performance. Higher levels of deep networks contain more abstract information. In an extreme example, the normalized prediction layer is trained to have a pure information about the structure of the problem and forget as much as possible about the details of instances of the objects. Even two identical network architectures might find two different local optimums in their training stages, and the chance of having distant representations for each input decrease as the depth of layer of representation increase. This fact has attracted researchers to invent methods that can distill information from deeper and near last feature maps of two networks. The proposed method solved this problem by taking the intermediate feature maps and transforming it to meaningful representations that wash out restrictive details

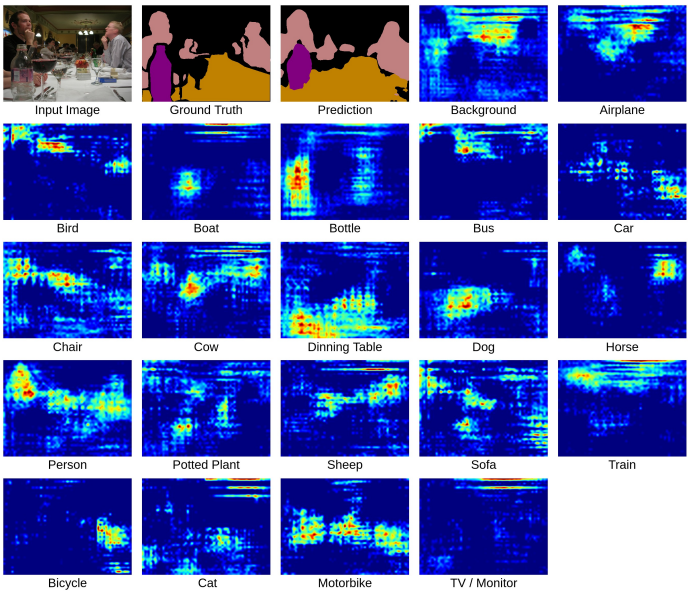

Fig. 3. Visualization of all CAM matrices for an image from Pascal Voc's validation set.

for distillation and hold useful information that can guide the student in the optimization space.

In future, the community may want to pay more attention to use saliency maps to invent more novel distillation functions. A lot of these methods try to map a network's representation space back to the input space or a space that humans understand. This space may have good potentials for knowledge distillation.

References

[1] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in neural information processing systems*, pages 2654–2662, 2014.

[2] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *Advances in Neural Information Processing Systems*, pages 742–751, 2017.

[3] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[4] Yuhua Chen, Wen Li, and Luc Van Gool. Road: Reality oriented adaptation for semantic segmentation of urban scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7892–7901, 2018.

[5] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

[6] 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*, 88(2):303–338, 2010.

[7] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3429–3437, 2017.

[8] Tommaso Furlanello, Zachary C Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. *arXiv preprint arXiv:1805.04770*, 2018.

[9] 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*, pages 770–778, 2016.

[10] Tong He, Chunhua Shen, Zhi Tian, Dong Gong, Changming Sun, and Youliang Yan. Knowledge adaptation for efficient semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 578–587, 2019.

[11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[12] Judy Hoffman, Saurabh Gupta, and Trevor Darrell. Learning with side information through modality hallucination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 826–834, 2016.

[13] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

[14] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2604–2613, 2019.

[15] 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*, pages 3431–3440, 2015.

[16] Sachin Mehta, Mohammad Rastegari, Anat Caspi, Linda Shapiro, and Hannaneh Hajishirzi. Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 552–568, 2018.

[17] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016.

[18] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[19] German Ros, Simon Stent, Pablo F Alcantarilla, and Tomoki Watanabe. Training constrained deconvolutional networks for road scene semantic segmentation. *arXiv preprint arXiv:1604.01545*, 2016.

[20] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[21] Chengchao Shen, Xinchao Wang, Jie Song, Li Sun, and Mingli Song. Amalgamating knowledge towards comprehensive classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3068–3075, 2019.

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

[23] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[24] Jiafeng Xie, Bing Shuai, Jian-Fang Hu, Jingyang Lin, and Wei-Shi Zheng. Improving fast segmentation with teacher-student learning. *arXiv preprint arXiv:1810.08476*, 2018.

[25] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 325–341, 2018.

[26] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

[27] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[28] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4320–4328, 2018.

[29] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[30] 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*, pages 2921–2929, 2016.