See More for Scene: Enlarging Effective Receptive Field for Scene Classification via Pairwise Consistency

Anonymous Author(s)
Affiliation
Address
email

Abstract

Scene classification is a valuable classification subtask and has its own characteristics which still needs more in-depth studies. Basically, scene characteristics are distributed over the whole image, which causes the need of "seeing" comprehensive and informative regions. Previous works mainly focus on region discovery and aggregation, while rarely involves the inherent properties of CNN along with its potential ability to satisfy the requirements of scene classification. In this paper, we propose to understand scene image and the CNN classification model in terms of effective receptive field (ERF). From this new perspective, we find that large ERF is yielded for the need of exploring more regions. Meanwhile, the analysis about non-linear layer, ReLU, helps us to build the positive correlation between ERF size and the number of activated positions in feature maps. Pursuing enlarged ERF for improving scene classification, we propose a new learning scheme and a tailored loss to achieve it. The proposed learning target is to make the outputs of an input image and its modified version to be consistent. The tailored loss is implemented as pairwise consistency of prediction to leverage category-relevance information. Experiments on Places365 show the significant improvements of our method with various CNNs. Our method shows an inferior result on object dataset, ImageNet, which indicates that it really captures the unique characteristics of scenes.

1 Introduction

Image Classification is one of the fundamental tasks of computer vision, and attracts a lot of attention as a popular task for facilitating the improvement of convolution neural network (CNN) [14, 23, 26, 10, 12]. As a valuable classification subtask, scene classification has its own characteristics which still needs further study. In comparison with objects, scenes are more complex and have many differences. One of main differences is that the characteristics of scenes are basically distributed over the whole image, while those of objects are confined within a clear boundary. This phenomenon inspires a core idea of scene classification, "seeing" comprehensive and informative regions in the image.

Most scene classification methods aim to extract unspecific [7, 25, 5] or specific regions [29, 35, 1], then aggregate them via statistical models [7, 5, 29] or relation modeling approaches [25, 1]. The main challenge is that the extraction and aggregation are partially independent from the main CNN backbone. It induces some issues as follows. 1) some sort of incompatibility caused by the separated ways to extract and represent regions. 2) inevitable computational consumption from extra region extraction and aggregation processes. It is less studied to investigate the inherent properties of CNN and make it adaptive to the needs of scene classification in a more organic and effective way.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

In this paper, we explore a new way to understand scene images and the CNN classification models in terms of effective receptive field (ERF) [19]. ERF is defined as the region contains the input pixels within a receptive field that have non-negligible impact on the output. From the perspective of ERF, we obtain some observations that reveal the differences in the characteristics of CNNs between scene and object classification, which also correlates well with the properties of these two kinds of images that described above. One of the interesting findings is that the need of exploring more informative regions in scene classification could drive the development of large effective area in a receptive field of the corresponding CNN model.

This finding leads us to study further the benefit of large ERF for improving scene classification. As we measure the effectiveness (impact on the output) of pixels within a receptive field by the deviation, it is theoretically possible to investigate the distribution of impact based on the networks. An earlier work has proved the distribution of impact under the setting of many stacked linear layers (convolution) distributes as a Gaussian. However, it is still very hard to figure out the complex relation between ERF and the CNNs, due to the mixed combination of linear and non-linear layers. Here, we focus on the non-linear layer (ReLU), and find the positive correlation between large ERF and more activation regions in the convolution feature maps.

Based on this correlation, we propose a new learning scheme and a tailored loss for scene classification. Our learning target is to make the outputs of an input image and its modified version to be consistent so as to drive CNN to activate more in the feature maps along with the enlarged ERF. It is also motivated by the observation that a scene image can retain its semantic meaning even if some regions in it are covered. Specifically, the modified image is generated by partially erasing input image with the guidance of model preference. The tailored loss is implemented as pairwise consistency of predictions, which can leverage the category-relevance information.

Experiments on one of the largest scene datasets, Places365 [38], demonstrate the effectiveness of the proposed method with various CNNs. Moreover, with the newly proposed metric, it is concluded that the proposed method indeed implements better performance by enlarging ERF, which verifies our findings. More importantly, we experimentally find that the proposed method really captures the unique characteristics of the scene image, while is not applicable to object classification.

2 Related work

Scene classification. In the era of CNN, many works [7, 4, 6, 16, 1, 33] explore various region extraction and encoding methods based on CNN representations to improve scene classification. They generally need lots of extra overhead compared to a single CNN classifier and is not easy to apply on large scale datasets. A few researches focus on specific deep models designed for scene classification, including Dictionary Learning CNN [17], spatial unstructured layer [8], and contextual relation learning [40]. In contrast, we aim to investigate the inherent properties of scene CNN model, and propose an advanced training method to enhance it without any change of architecture.

Erasing methods in CNN. Erasing regions or pixels in images or feature maps is an effective method in weakly supervised object localization [24, 28, 34, 2] / segmentation [15], due to the observation that the models often only concentrate on small and most discriminative regions of object when supervised by only image-level labels. Our idea is inspired by this but study further the properties of CNN from the perspective of effective receptive field, which has not been explored. In addition, we explore the erasing method with a tailored loss and extending them into classification task with superior performances, which enriches the understanding of the intrinsic correlations between the erasing method and image classification.

3 Effective receptive field

The receptive field is a local region of the input that can support the output response. For example, the receptive field of each output response after a convolution layer is a rectangular area determined by the convolution kernel. The receptive field size of the final outputs will become larger when

stacking more convolution layers or down-sampling layers. In many tasks [21, 30, 9, 22, 39], it has been verified that a big receptive field is a crucial factor for the model performance, and various technologies (like dilated convolution [30], and deformable convolution [3]) have been proposed to meet this need.

From a more detailed perspective, pixels in a receptive field have different impact on a neural output, as the paths of input pixels connected to the output vary over their positions. Pixels near the center of a receptive field have larger impact on the corresponding output, compared to pixels far away. In contrast to many researches on the receptive field size, there are a few works to investigate the distribution of impact within a receptive field, and how its change affects the CNN models.

Here, we aim to study the effective area of a receptive field, also termed **Effective Receptive Field** (ERF), which consists of the important pixels that have strong impact on the target output. In an earlier work, Luo et al. [19] theoretically prove that in some cases the distribution of impact within a receptive field distributes as a Gaussian. For that reason, they directly treat effective receptive field as a measurable shape, which size is represented by one or two standard deviation. In contrast, we present a more general way to measure ERF, and analyze the difference among ERFs derived from CNN models with the same architecture but trained on various datasets, which expands the existing researches.

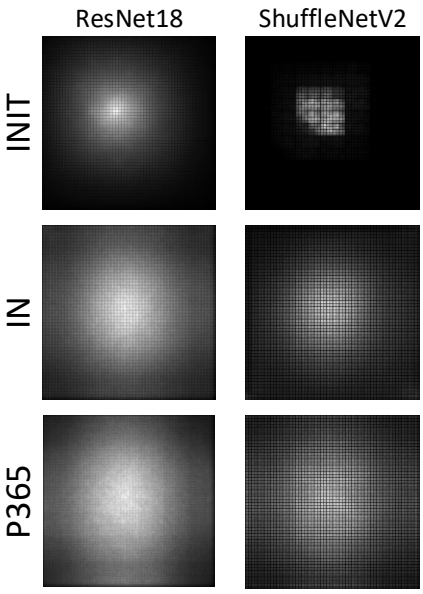

Figure 1: The impact maps from models without training "INIT", trained on ImageNet "IN", and Places365 "P365".

Following the setting in [19], we measure the impact by the partial derivative $\partial y^l_{m,n}/\partial \mathbf{x}_{i,j}$, where $y_{m,n}$ means the mean value of feature vector indexed by (m, n) in the output feature maps Y of layer l, and $\mathbf{x}^l_{ij}$ denotes the pixel vector at position (i, j) of input image X. For generating the value of pixel $x_{i,j}$ in the impact map of input, we average the absolute values of gradient vector $\mathbf{x}^l_{i,j}$. All computation can be easily implemented with current deep learning platform.

In practice, we choose the final convolution layer from last stage as layer l. When aggregating the features over the channel dimension, due to the weight sharing property in the convolution, all pixels in the aggregated feature map will have the same theoretical distribution of impact. So, we only consider the center point in the output feature maps as (m, n) for simplifying analysis. In order to eliminate the effect of input image, the impact map is presented in terms of expectations over input distribution, by averaging the results of one thousand randomly picked images. Figure 1 shows some examples of the impact map.

Modern CNN architectures contain many convolution layers and non-linear layers stacked in an alternating way, which violates the basic assumptions that support the Gaussian distribution conclusion. Since there is no strict theoretical distribution, it is inappropriate to measure ERF with some shape priors (like rectangular or ellipse). We propose to compute a histogram H on the impact map $\mathcal{I}$,

$$H_s = \sum_{i=0}^{HW} \mathbf{I}[\mathcal{I}_i \in r_s],\ s \in \{0, \ldots, S\} \tag{1}$$

where r_s is the s-th bin, S is the number of bins, and $\mathbf{I}[\cdot]$ is the indicator function. Before computing histogram, $\mathcal{I}$ is normalized into range $[0, 1]$ by performing min-max normalization, like $\mathcal{I} = \frac{\mathcal{I} - \min(\mathcal{I})}{\max(\mathcal{I}) - \min(\mathcal{I})}$.

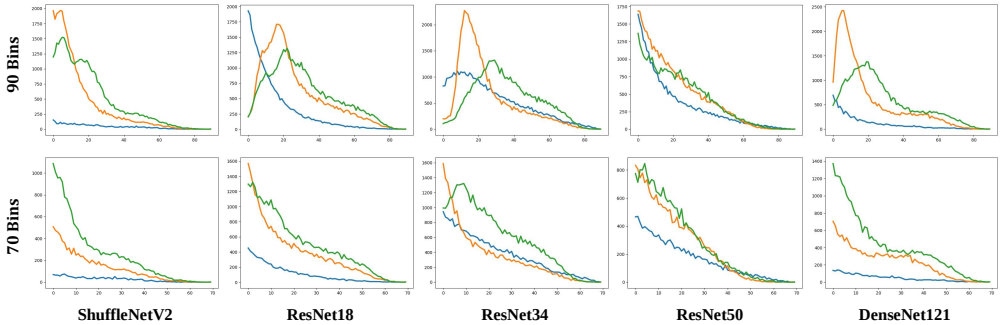

Figure 2: The histograms of the impact maps. Bigger integration area under the lines means larger effective receptive field. “Green”, “Orange”, and “Blue” denote the results from models trained on Places, ImageNet, and only initialized, respectively.

Using histogram to measure ERF has two main advantages. One is that there is no shape prior as we directly show the distribution of impact. The other is that it may reduce the effects of random noise because calculating depends on a certain range instead of a fixed threshold [19]. As shown in Figure 2, we investigate five kinds of models (ShuffleNetV2_1.5 [20], ResNet18 / 34 / 50 [10], DenseNet121 [12]) trained on two datasets (ImageNet [13] and Places365 [38]). The number of bins is set to 100. We show two kinds of histogram, “90 Bins” and “70 Bins” (remove last 10% and 30% bins with the lowest values, respectively). The effective receptive field size can be calculated as the summation of the numbers within a certain set of bins.

Some observations are listed as follows: 1) ERF is quite small at the initialization, but increases a lot at the end of training; 2) generally, ERF of the model trained on scene images (Places365) is larger than that on object images (ImageNet); 3) when the output feature dimension is large (like 2048 in ResNet50), the difference in ERF between these two datasets becomes small. These observations inspire us to explore the correlation between large ERF and scene classification, and consider the possibility of enlarging the ERF to improve the performance.

4 Methods

In this section, we will present a deeper analysis of ERF and connect it to the convolution feature maps. Then inspired by the properties of scenes, we propose an advanced training scheme with a tailored loss for improving scene classification in terms of enlarged ERF.

4.1 Connecting ERF and the convolution feature maps

Based the definition, it is intuitive to increase the effective receptive field by stacking more convolution layers. However, it will induce more parameters and may be difficult to optimize [31] or cause overfitting [10]. Here, our goal is to increase ERF with keeping the architectures unchanged. Before introducing our method, we need to delve a little deeper into ERF and bridge the gap between ERF and the convolution feature maps.

With measuring impact by the derivative, we note that the effective receptive field depends on the network and the input. If we analyze it over the expectation of input distribution, the focus point will be narrow down to the network (including the architecture and weights). Roughly, there are linear layers (convolution layer, batch normalization layer) and non-linear layers (ReLU, tanh, sigmoid) in CNN[1], which makes it very hard to figure out what changes will be better for large ERF.

[1]Theoretically, the output from batch normalization layer will be connected to all inputs, which means its receptive field is the whole spatial locations. However, we empirically find that the receptive field is independent to the batch normalization layer by using the derivative to measure, which is consistent with the results in [19].

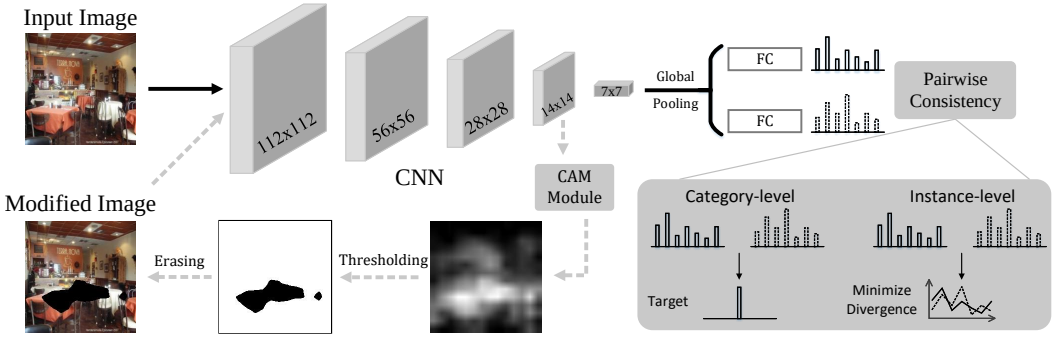

Figure 3: The overview of our proposed method.

To simply analysis, we only focus on the non-linear layer, and analyze the gradient function of ReLU as it is the widest used non-linear operation. The formula of ReLU is $\sigma(x_c) = \max(0, x_c)$, where c means the index of channels. It is easy to deduce the gradient function as follows,

$$\sigma^{'}(x_c) = \begin{cases} 1, & x_c > 0 \\ 0, & x_c <= 0 \end{cases}. \tag{2}$$

It can be concluded that it is necessary to keep the output of ReLU layer positive for making effective connection. From the perspective of expectation, we can obtain $\mathbf{E}\left[\sum_c \sigma^{'}(x_c)\right] = \sum_c P(x_c > 0)$, with the assumption that x_c has independent distribution along the dimension C. This inspires that if we expect larger impact (expectation of summation of gradients over channel dimensions) of input pixels on the output, it is a solution to activate more positions in the output feature maps (keep them positive).

This connection is not easy to build on a single image, due to the complex effects from other layers and their weights. But we may have a relatively clear view when verifying it with a large image set. It has been empirically found [1] that the sizes of activated regions (consisting of spatial positions with positive values in feature maps) from models trained on scene images are generally larger than that on object images, which is consistent with our observation of ERF.

4.2 Advanced training via pairwise consistency

Some earlier works [28, 34, 2] have effectively demonstrated that erasing certain regions in the image or feature maps can force CNN to cover the full extent of object in the context of object classification along with the classification performance degradation. We conclude that erasing regions can be a useful way to help the CNN models focus on more information of the image, which are reflected by more activated positions in the feature maps.

Based on this, we propose a new learning scheme and design a tailored loss for scene classification. Our design also correlates well with an observation about scene images, which is that if some regions of an image are hiding, we still have a high probability to classify this image as the same category or even the same image. In this way, we decide to make the outputs of an input image and its modified version to be consistent. The main components of this learning process are the modifying way and the consistency function.

Modifying image. Here, we will discuss which regions need to be erased. Normally, It is intuitive to randomly choose regions, which is independent to the model and may not fit well. What is expected is the kind of regions that could reflect the characteristics of images or models and are relevant to the demands of classification.

Fortunately, it is widely recognized [32, 27, 36, 2] that the convolution feature maps can indicate the model preferences in terms of spatial positions. A simple way [32, 2] to leverage this property is to average the feature maps over channel dimensions, which will present the model-relevant regions as

a kind of learned knowledge. Furthermore, class activation map (CAM) [37] is proposed to direct the attention to class-specific regions, which reveals the contents that provide evidences for making the corresponding prediction. In our design, class-specific regions (produced by CAM) can present more targeted information, and perform better in deriving region exploration as desired.

The generation of CAM depends on a specific module, termed CAM module, that sequentially contains a convolution layer (1×1 in our setting), a global average pooling layer, and a fully connected layer (as a classifier). Then, the output M of CAM is computed as follows,

$$M = \mathcal{F}(A, W^t) = \sum_{c=1}^{C} W_c^t A_c, \tag{3}$$

where $A \in \mathbb{R}^{C \times H \times W}$ is the output from the convolution layer, and W^t is the weights in the classifier to generate the prediction of class t. Most modern CNN architectures can meet the demands of CAM generation. So it is quite straightforward to use the original linear classifier of CNN to produce CAM.

However, we attach a CAM module to the input of last down-sampling layer (normally, spatial dimension is 14×14) for two reasons. One is that larger resolution feature maps may contain more and better detail information. The other is that decoupling the classifier and the generation of CAM would reduce the performance loss caused by erasing more relevant regions derived from the original classifier. In addition, we use the prediction (category with the maximum output from the original classifier) as class t, not the ground truth, because the focus point is the model's characteristics, not the properties of the input images.

Guided by the output M, we can use a threshold γ to convert it to a binary mask M_B, which indicates the regions that need to be erased. Then, a modified image I^* is yielded by multiplying the input image I and the binary mask M_B. The formulation of this process is shown as follows,

$$I^* = M_B \odot I, \; M_B = T(M, \gamma), \tag{4}$$

where $\odot$ means element-wise multiplication, T denotes the thresholding function.

Pairwise consistency. Given the input image I and its modified version I^*, we propose a tailored loss function to measure the similarity of their outputs so as to achieve pairwise consistency. Specifically, two consistency forms can be considered, category-level and instance-level. In category-level consistency, we assume that the pair of images belongs to the same category. In instance-level consistency, a stricter constraint is created by enforcing the same prediction of these two images.

In practice, we keep the main component of CNN shared, and only create specific linear classifier for two images, separately. In inference, the classifier of the modified image is discarded and only the original image is used as input to make a prediction. For category-level consistency, two cross-entropy loss with the same ground truth are used. For instance-level consistency, we use Kullback-Leibler divergence as the metric to measure similarity between the output predictions, and also keep supervising the main branch with the ground truth. Our final loss combines these two consistency forms, and are formulated as

$$L = \sum_{k=1}^{K} -g_k \log(p_k) + \alpha \sum_{k=1}^{K} -g_k \log(q_k) + \beta \sum_{k=1}^{K} -q_k \log(\frac{q_k}{p_k}), \tag{5}$$

where $p, \; q$ denote the predictions of the original and modified images respectively, g is the ground truth (represented as one-hot vector). we empirically set $\alpha = 1$ and $\beta = 2$ in all experiments.

Discussion. Our method only influences the training stage, and shows no change in inference. Although the proposed method needs double memory usage and computational time during training, we believe it is tolerable as an one-off process.

Table 1: Top-1 and Top-5 accuracy (%) on Places365-small and Places365 dataset. "RF" is short for receptive field, which is computed according to the CNN architectures. "Param." denotes the number of parameters.

Dataset	Model	Param.	RF	Baseline		Our Method	
				Top-1	Top-5	Top-1	Top-5
Places365 -small	ShuffleNetV2	2.85M	527	47.67	78.68	48.91	79.76
	ResNet18	11.36M	435	48.21	79.22	49.70	80.62
	ResNet34	21.47M	899	48.52	79.58	50.72	81.31
	ResNet50	24.26M	427	49.66	80.67	50.92	81.70
	DenseNet121	7.33M	2071	49.48	80.87	51.55	82.20
Places365	ShuffleNetV2	2.85M	527	54.01	84.05	54.95	84.56
	ResNet18	11.36M	435	54.36	84.50	55.12	84.94
	ResNet34	21.47M	899	54.84	84.72	56.23	85.79
	ResNet50	24.26M	427	55.53	85.85	56.61	86.15
	DenseNet121	7.33M	2071	55.40	85.50	56.84	86.44

5 Experiments

In this section, we empirically evaluate the effectiveness of the proposed method and conduct some ablation studies for the deeper analyses.

Datasets. Places365 [38] is one of largest scene-centric datasets with over 10 million labeled images of scenes, which consists of 365 scene categories and two versions of training sets, Places365-challenge and Places365-standard. We choose Places365-standard as the training set in following experiments, which provides 1.8 million images. The validation set has 36,500 images, where 100 images per category. We report Top-1 and Top-5 accuracy (single crop) on the validation set. Meanwhile, for fast evaluation, we also randomly choose 1000 images per category from Places365-standard set, and construct a new training set, termed **Places365-small**.

Implementation details. We evaluate the proposed method on several CNN architectures (implemented based on torchvision models[2]), including ResNet [10], ShuffleNetV2 [20], and DenseNet [12]. We use simple data augmentation with random cropping (from 256×256 to 224×224) and random horizontal flipping during training. In inference, we apply center-cropping in resized 256×256 images. All models are trained from scratch by synchronous SGD with momentum 0.9 and a mini-batch size of 256, starting from learning rate 0.1 and decreasing it by cosine annealing strategy [18]. All experiments are conducted with the PyTorch framework on 4 Titan RTX devices. The weight decay is $2e^{-4}$ for ResNet and DenseNet on Places365, $1e^{-4}$ for ShuffleNetV2 on Places365, and $5e^{-4}$ for all models on Places365-small. We conduct experiments on Places365 with 40 epochs and Places365-small with 30 epochs. We simply evaluate the hyper-parameter γ in range $\{1.0, 3.0, 5.0\}$, and find $\gamma = 3.0$ is a better choice. Although, we believe more detailed evaluation of γ may lead to better performances.

5.1 Main results

Comparison with the baseline. The main results are shown in Table 1. We compare the baseline and our method with five CNNs (ShuffleNetV2, ResNet18 / 34 / 50, and DenseNet121). It can be observed that the proposed method significantly improves the performances. For example, when applied on ResNet34, the proposed method achieves the largest improvements of 2.20%, 1.39% on Places365-small and Places365 datasets, respectively. Note that these improvements are very significant, because there are less performance differences (e.g. less than 1.5% for ResNet18 and ResNet50) between various models with the standard training scheme.

Interestingly, we also find the positive correlation between the receptive field and the improvement of the proposed method. For example, ResNet34 and DenseNet121 have larger receptive field

[2]https://github.com/pytorch/vision

Table 2: Comparison with other erasing methods, HaS [24], ACoL [34], and ADL [2].

Methods	HaS	ADL	ACoL	Ours
ResNet18	49.25	48.70	49.03	49.70
ResNet34	49.91	48.68	49.55	50.72

Table 3: The position of CAM module.

Model	ResNet18	ResNet34	DenseNet
stage4	49.62±0.22	49.77±0.08	51.29±0.08
stage3	49.70±0.10	50.72±0.05	51.55±0.17

Table 4: Comparison with object classification. All experiments are based on ResNet18, and trained with the same settings. Top-1 accuracy (single crop) is reported.

Dataset	Places365-small	Places365	ImageNet
Baseline	47.80	53.77	69.76
Ours	48.68	54.12	69.57

sizes (899 and 2071) along with bigger relative improvements (4.53% and 4.18%), compared to ResNet50 with smaller receptive field size (427) and relative improvement (2.54%). It suggests that the proposed method benefits much from the large receptive field, as it facilitates the ability of our method to explore more effective connections between the output features and the input image. Another interesting finding is that the performance ranking of ResNet50 and DenseNet121 is different between baseline and our method. Considering the large difference in the receptive field size, we conclude that the proposed method is able to leverage the potential value of big receptive field, which may be limited in the standard training scheme.

Comparison with other erasing methods. We compare the proposed method with other related erasing methods, HaS [24], ACoL [34], and ADL [2]. All three methods are proposed for weakly supervised object localization task, which concentrate on the characteristics of object images, and suffer degraded classification performances (excluding HaS). While ACoL and ADL achieve the comparable or worse classification performances in WSOL, they clearly outperform the baselines in the context of scene classification, as shown in Table 2. Naturally, as a data augmentation method, HaS still show a superior performance when applied on scene images. However, all these methods yield inferior results to our method. This can be attributed to the tailored pairwise consistency loss and the effective region generation method.

Comparison with object classification. We also evaluate the proposed method on ImageNet [13], the most popular object dataset. All comparison experiments in Table 4 use the baseline training procedure in [11] with a total batch size of 256, and 90 epochs. It can be observed that our method has a negative effect on object classification and show inferior performance compared to the baseline. Note that the complex data augmentation strategy is tuned well for object classification, which causes the worse results in scene classification. Despite this bad impact, the proposed method still obtain better performances. This comparison clearly indicates that our method is specifically designed for scene images and leverage their unique attributes.

5.2 Ablation studies

The position of CAM module. We evaluate two positions for inserting the CAM module, stage3 (convolution layers operated on 14×14 feature maps) and stage4 (convolution layers operated on 7×7 feature maps). Although we assume that the erased image will have the same category distribution as the original input, it may be hard to hold when the erased regions cover lots of discriminative parts. As a result, we insert the CAM module at stage3, which is a little far away from the final classifier and produces the less discriminative regions. As shown in Table 3, it can be seen that this strategy works well on the models with large receptive field, like ResNet34 and DenseNet121, and retain the similar performances on the models with small receptive field, such as ResNet18. Limited by the computation resource, we only report the error bar in Table 3 with 3 repeated runs. The results show that the random deviations are relatively small and do not interfere with our conclusions.

Table 5: Consistent or not? "AM" means Attention Mining loss [15], which aims to minimize the prediction value of the modified image at the ground truth.

Loss	AM	CPC
ResNet18	48.85	49.12
ResNet34	48.89	49.92

Table 6: The effect of two kinds of pairwise consistency. "CPC", "IPC" are short for the category-level and instance-level pairwise consistency, respectively.

CPC	IPC	ResNet		
		18	34	50
✓	-	49.12	49.92	50.43
-	✓	49.65	50.36	50.85
✓	✓	49.70	50.72	50.92
Baselines		48.21	48.52	49.66

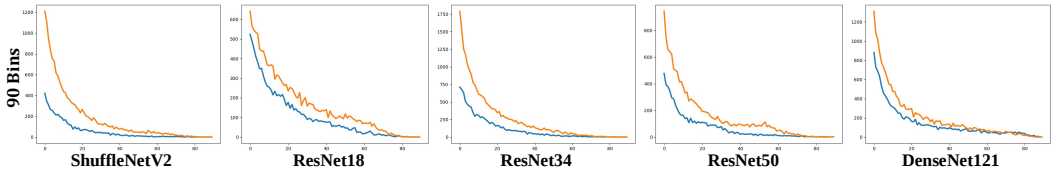

Figure 4: Comparison with the baseline. ERF is computed at the center point of the convolution maps from stage3. "Orange" and "Blue" lines correspond to our method and the baseline respectively.

Pairwise consistency is the key factor of the propose method, and motivated by the specific properties of scenes. Firstly, we want to discuss the value of the consistency idea. Li et al. [15] proposed a different loss (attention mining loss) to minimize the prediction value of the modified image at the ground truth. Table 5 shows the results from the attention mining loss and the category-level pairwise consistency loss. We can see that our proposed consistency loss outperforms the attention mining loss, especially on ResNet34. It infers that the consistency target provides a stronger driving force for the model, and is a better way to explore more region information for scenes.

Beyond the category-level pairwise consistency (CPC), we also introduce the instance-level pairwise consistency (IPC) as a more powerful tool. As shown in Table 6, It can be observed that IPC significantly outperforms CPC on various CNNs. It is not surprising because IPC provides a clearer and more precise guidance by investigating the category-relevance information from the prediction distribution. Finally, we combine these two losses to produce a slightly better and robust model.

5.3 Analysis of ERF

Figure 4 shows the histograms of the impact maps from our proposed method (orange line) and the baseline (blue line). It is clearly shown that the proposed method could enlarge the effective area of receptive field as desired, compared to the baseline. Here, we show the results of the center point of the convolution maps from stage3. More details of comparison results are shown in Appendix.

6 Conclusion

In this work, we propose to investigate the scene CNN classification model in terms of effective receptive field (ERF). A new metric, histogram, is introduced to measure ERF, and help to analyze the difference in ERF between various classification models. An interesting finding is that ERF in scene CNN generally has a bigger area than that in object CNN. Based on this, we propose a new learning scheme and a tailored loss to force CNN to activate more positions in the feature maps so as to achieve enlarged ERF for improving scene classification. Extensive experiments on large scale scene dataset, Places365, demonstrate the effectiveness of the proposed method. Meanwhile, the worse performance on object dataset, ImageNet, indicates that our method is specifically designed for scenes and captures their unique attributes. Some comparison analyses also verify the ability of our method to enlarge ERF.

References

[1] Gongwei Chen, Xinhang Song, Haitao Zeng, and Shuqiang Jiang. Scene Recognition with Prototype-Agnostic Scene Layout. *IEEE Transactions on Image Processing*, 29:5877–5888, 2020. 1, 2, 4.1

[2] Junsuk Choe and Hyunjung Shim. Attention-based dropout layer for weakly supervised object localization. In *CVPR*, pages 2214–2223, 2019. 2, 4.2, 2, 5.1

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable Convolutional Networks. In *ICCV*, pages 764–773, 2017. 3

[4] Mandar Dixit, Si Chen, Dashan Gao, Nikhil Rasiwasia, and Nuno Vasconcelos. Scene classification with semantic Fisher vectors. In *CVPR*, pages 2974–2983. IEEE, jun 2015. 2

[5] Mandar Dixit, Yunsheng Li, and Nuno Vasconcelos. Semantic Fisher Scores for Task Transfer: Using Objects to Classify Scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(8):1–1, 2019. 1

[6] Mandar D. Dixit and Nuno Vasconcelos. Object based Scene Representations using Fisher Scores of Local Subspace Projections. In *NeurIPS*, pages 2811–2819, 2016. 2

[7] Yunchao Gong, Liwei Wang, Ruiqi Guo, and Svetlana Lazebnik. Multi-scale Orderless Pooling of Deep Convolutional Activation Features. In *ECCV*, pages 1–17, 2014. 1, 2

[8] Munawar Hayat, Salman H. Khan, Mohammed Bennamoun, and Senjian An. A Spatial Layout and Scale Invariant Feature Representation for Indoor Scene Classification. *IEEE Transactions on Image Processing*, 25(10):4829–4841, 2016. 2

[9] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. In *ICCV*, pages 2980–2988. IEEE, oct 2017. 3

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, pages 770–778. IEEE, jun 2016. 1, 3, 4.1, 5

[11] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of Tricks for Image Classification with Convolutional Neural Networks. In *CVPR*, pages 558–567, 2019. 5.1

[12] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely Connected Convolutional Networks. In *CVPR*, pages 2261–2269. IEEE, jul 2017. 1, 3, 5

[13] Jia Deng, Wei Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 3, 5.1

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *NeurIPS*, pages 1–9, 2012. 1

[15] Kunpeng Li, Ziyan Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Tell Me Where to Look: Guided Attention Inference Network. In *CVPR*, 2018. 2, 5, 5.2

[16] Yao Li, Lingqiao Liu, Chunhua Shen, and Anton van den Hengel. Mining Mid-level Visual Patterns with Deep CNN Activations. *International Journal of Computer Vision*, 121(3):344–364, 2017. 2

[17] Yang Liu, Qingchao Chen, Wei Chen, and Ian Wassell. Dictionary Learning Inspired Deep Network for Scene Recognition. In *AAAI*, pages 7178–7185, 2018. 2

[18] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *ICLR*, pages 1–16, 2017. 5

[19] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *NeurIPS*, pages 4905–4913, 2016. 1, 3, 3, 1

[20] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. In *ECCV*, pages 122–138, 2018. 3, 5

[21] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel matters - Improve semantic segmentation by global convolutional network. In *CVPR*, pages 1743–1751, 2017. 3

[22] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jonathon Shlens. Stand-Alone Self-Attention in Vision Models. In *NeurIPS*, 2019. 3

[23] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*, pages 1–14, 2015. 1

[24] Krishna Kumar Singh and Yong Jae Lee. Hide-and-Seek: Forcing a Network to be Meticulous for Weakly-Supervised Object and Action Localization. In *ICCV*, pages 3544–3553, 2017. 2, 2, 5.1

[25] Xinhang Song, Shuqiang Jiang, and Luis Herranz. Multi-Scale Multi-Feature Context Modeling for Scene Recognition in the Semantic Manifold. *IEEE Transactions on Image Processing*, 26(6):2721–2735, jun 2017. 1

[26] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015. 1

[27] Xiu-Shen Wei, Jian-Hao Luo, Jianxin Wu, and Zhi-Hua Zhou. Selective Convolutional Descriptor Aggregation for Fine-Grained Image Retrieval. *IEEE Transactions on Image Processing*, 26(6):2868–2881, jun 2017. 4.2

[28] Yunchao Wei, Jiashi Feng, Xiaodan Liang, and Ming-ming Cheng. Object Region Mining with Adversarial Erasing : A Simple Classification to Object Region Mining with Adversarial. In *CVPR*, pages 1568–1576, 2017. 2, 4.2

[29] Ruobing Wu, Baoyuan Wang, and Wenping Wang. Harvesting Discriminative Meta Objects with Deep CNN Features for Scene Classification. In *ICCV*, pages 1287–1295, 2015. 1

[30] Fisher Yu and Vladlen Koltun. Multi-Scale Context Aggregation by Dilated Convolutions. In *ICLR*, 2016. 3

[31] Sergey Zagoruyko and Nikos Komodakis. Wide Residual Networks. In *BMVC*, 2016. 4.1

[32] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, pages 1–13, 2017. 4.2

[33] Haitao Zeng, Xinhang Song, Gongwei Chen, and Shuqiang Jiang. Learning Scene Attribute for Scene Recognition. *IEEE Transactions on Multimedia*, pages 1–1, 2019. 2

[34] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas Huang. Adversarial Complementary Learning for Weakly Supervised Object Localization. In *CVPR*, pages 1325–1334, 2018. 2, 4.2, 2, 5.1

[35] Zhengyu Zhao and Martha Larson. From Volcano to Toyshop: Adaptive Discriminative Region Discovery for Scene Recognition. In *ACM MM*, 2018. 1

[36] Heliang Zheng, Jianlong Fu, Tao Mei, and Jiebo Luo. Learning Multi-attention Convolutional Neural Network for Fine-Grained Image Recognition. In *ICCV*, pages 5219–5227. IEEE, oct 2017. 4.2

[37] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning Deep Features for Discriminative Localization. In *CVPR*, pages 2921–2929, 2016. 4.2

[38] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 Million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2018. 1, 3, 5

[39] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable ConvNets v2: More Deformable, Better Results. In *CVPR*, 2019. 3

[40] Zhen Zuo, Bing Shuai, Gang Wang, Xiao Liu, Xingxing Wang, Bing Wang, and Yushi Chen. Learning Contextual Dependence With Convolutional Hierarchical Recurrent Neural Networks. *IEEE Transactions on Image Processing*, 25(7):2983–2996, jul 2016. 2

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 4.2.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Appendix
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] The data is publicly available, and code is easy to implement by referring to above description and open resources.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 5.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We have no enough resource to complete error bars for all experiments. But we did some test and found the random error can be neglected in our comparisons, as shown in 5.2.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section 5.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No] It is not stated in the corresponding website and paper.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] The assets are publicly available.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] The related information can be found in the original paper that produced the dataset.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]