Channel DropBlock: An Improved Regularization Method for Fine-Grained Visual Classification

Yifeng Ding[1]
dingyifeng0110@163.com
Shuwei Dong[1]
donggua0812@163.com
Yujun Tong[1]
tongyujun@bupt.edu.cn
Zhanyu Ma[1]
mazhanyu@bupt.edu.cn
Bo Xiao[1]
xiaobo@bupt.edu.cn
Haibin Ling[2]
haibin.ling@gmail.com

[1] School of Artificial Intelligence
Beijing University of Posts and Telecommunications
Beijing, China

[2] The Department of Computer Science
Stony Brook University
New York, USA

Abstract

Classifying the sub-categories of an object from the same super-category (*e.g.*, *bird*) in a fine-grained visual classification (FGVC) task highly relies on mining multiple discriminative features. Existing approaches tackle this problem mainly by introducing attention mechanisms to locate the discriminative parts or feature encoding approaches to extract the highly parameterized features in a weakly-supervised fashion. In this work, we propose a lightweight yet effective regularization method named Channel DropBlock (CDB), in combination with two alternative correlation metrics, to address this problem. The key idea is to randomly mask out a group of correlated channels during training to destruct features from co-adaptations, and thus enhance feature representations. Extensive experiments on three benchmark FGVC datasets show that CDB effectively improves the performance. Code available at https://github.com/PRIS-CV/DropChannelBlock_Pytorch_master.

1 Introduction

Deep convolutional neural networks have achieved great progresses in many computer vision tasks, especially in object recognition with sophisticated model design and abundant data sets. By contrast, Fine-Grained Visual Classification (FGVC) remains very challenging, mainly because of the high intra-class variance and low inter-class variance among fine-grained categories (e.g., *bird species*, *flower types*, *car models*, *etc*.).

The key in FGVC is to learn discriminative feature representations that can distinguish the inter-class differences and align the intra-class variances. Early solutions [1, 2, 14, 17, 30] utilize additional bounding box/part annotations to locate discriminative parts. It later becomes clear that such supervised approaches are hard to generalize because expert human annotations can be cumbersome to obtain and are often error-prone [32]. More recent methods [7, 9, 11, 18, 29, 32, 33] address this issue in a weakly supervised manner. However, these methods usually involve complicated network structure or highly parameterized feature representations, which is hardly applicable to other models and also introduces extra computation overhead in both training and inference stages.

To address the aforementioned concerns, in this paper we propose a novel lightweight yet powerful method, named *Channel Dropblock* (CDB), for weakly supervised FGVC. Since the features in the convolutional layers are correlated, CDB aims at eliminating the effect of co-adaptation among channels. This encourages the network to enhance feature representation and to find more discriminative visual evidence in a FGVC task. To this end, CDB is designed to drop a group of correlated channels in one or several of convolutional layers to motivate the network to distinguish more discriminative parts. Unlike most existing weakly supervised FGVC networks, CDB is a light structure without additional parameters, making it easy to integrate into existing networks.

To guide the channel selection, we propose two novel strategies to measure the channel correlation. One strategy is the *max activation metric*, inspired by MA-CNN [32], which measures channel correlation by computing the distance between peak responses from different channels. We suppose that the smaller the distance is, the closer the correlation between two channels. In this occasion, the channels are clustered into discriminative local regions with diverse activation centers. The other strategy is the *bilinear pooling metric*. It computes the channel correlation matrix in which the cosine similarities between channels are calculated pairwise. In the bilinear pooling metric, the larger the cosine similarity value is, the more similar the two channels are. With this metric, the channels are clustered into specific visual patterns. During training, a channel of the feature map is selected randomly, and then the correlated channels which share similar visual pattern are clustered and masked out based on the correlation matrix.

Compared with existing FGVC networks, the proposed method is more efficient and flexible. This is because it can erase a discriminative visual pattern to prevent the features from co-adaptations by a single forward-backward propagation in a single block, which can be easily applied to all kinds of networks. Moreover, it does not need extra part/object annotation and introduces no computational overhead at inference time.

Our contributions can be summarized as follows:

1) We address the challenges of discriminative feature learning in FGVC tasks by proposing a novel lightweight regularization structure, which drops a group of correlated channels to enhance the network feature representations and hence extract more discriminative patterns.

2) We propose two metrics to measure the pairwise correlation between different feature channels, which can draw insights from feature channels.

3) We conduct extensive experiments on three popular fine-grained benchmark datasets, the results demonstrate that the proposed CDB significantly improves the FGVC performance when applied to baseline networks or integrated into existing methods.

2 Related Work

2.1 Dropout Mechanism

Dropping some units or patches in CNN has been proposed as a regularization method, such as dropout [26], SpatialDropout [27], cutout [6] and DropBlock [12], among them dropout is the inspiration of most other related regulation methods. Based on the way to drop, these methods can be divided into two categories: one contains attention-based methods and the other one drops object information randomly.

Dropout [26] is set to omit each neuron with a probability p during training time, providing a simple yet effective way to avoid overfitting. However, features are correlated spatially that hinders dropout to work effectively when applied to convolutional layers. To solve the problem, SpatialDropout [27], cutout [6] and DropBlock [12] were proposed later. Instead of dropping several inconsecutive units in feature maps, in SpatialDropout [27], entire feature maps are dropped with a probability p randomly, preventing nearby pixels from presenting the same information as dropped neurons. Cutout [6] is applied on input images to erase one part with a random square mask before training. Similarly, the key of DropBlock [12] is set to drop contiguous regions of feature maps, declining the effect of spatial correlated features. ADL [5] is one of the attention based regulation methods, in which drop masks based on attention are applied to feature maps to hide most discriminative parts and activate networks to learn other important features.

Different from dropping channels randomly of SpatialDropout [27] and erasing a patch in input image randomly of Cutout [6], our method drops a group of related channels. In this way, our method reduces the effect of co-adaptation among different channels, and thus can locate more discriminative visual patterns and enhance the features representation in FGVC.

2.2 Fine-grained Classification

The key of fine-grained image classification is to find out the discriminative representations of each class. To this end, a variety of methods have been proposed to increase the accuracy of deep learning network for this task.

Bilinear CNN [18] consists of two extractors, capturing pairwise correlations between the feature channels, to distinguish images from subtle differences, which is an efficient way to obtain more representation of features. RA-CNN [9] is a reinforced attention proposal network to obtain discriminating attention regions and region-based feature representation of multiple scales. MA-CNN [32] was composed by three sub network to realise convolution, channel grouping and part classification respectively, which generates multiple object parts by clustering channels of feature maps into different groups. NTS [29] enables a navigator agent as the region proposal network to detect multiple informative regions under the guidance from a teacher agent. PMG [8] adopts a progressive training strategy that fuses multi-granularity features.

The most relevant work to ours is MA-CNN [32], which also highlights channel correlations and clusters them into groups. However, the setting in MA-CNN restricts itself to the last layer of the feature extractor, which ignores the low-level information in FGVC task. Besides, it adjusts the original network structure by setting a fixed number of individual part classifiers, which poses challenges to implement on other methods or tasks.

Compared with MA-CNN, the advantages of our work are two-folds. First, the proposed CDB is more flexible to be applied to any convolutional feature maps of classification model, and hence can make fully use of both high-level semantics and low-level details. Second, no

adjustment to the original structure and no additional parameter/computation are involved during inference, which makes the method flexible to be integrated into existing networks.

3 CDB: Channel DropBlock

In this section, we present details of Channel DropBlock (CDB). CDB is a dropout-based regularization technique that can be easily applied to convolutional feature maps of a classification model to improve feature representations. We first describe the motivation together with comparison with relevant methods (Section 3.1). We then describe the Channel DropBlock algorithm, which drops correlated channel groups based on channel correlation matrix (Section 3.2 & Section 3.3).

3.1 Motivation

As shown in previous works [24, 31, 32], each channel of the convolutional features corresponds to a visual pattern. However, only parts of patterns contribute to the final prediction due to the co-adaptations between them, which will reduce inference accuracy especially when sub-categories are close and hard to distinguish (*e.g.*, in FGVC tasks). While dropout [26] is effective to destruct co-adaptations in features, it is less effective for convolutional feature channels since such channels are pairwise correlated, and the pattern about the input can still be sent to the next layer if we drop channels individually. This intuition suggests us to mask out a correlated group of channels instead of a single channel to encourage the model to learn more discriminative parts. The main motivation for CDB is to destruct the co-adaptations and induces the model to make full use of more discriminative features. This is achieved by randomly masking out a whole correlated channel group which only contributes to one visual evidence for the final prediction.

We initially developed CDB as an attention-based approach that specifically removes the most important channel groups from the input feature. This attempt is similar to the idea in ADL [5], in that we develop an importance branch and a dropout branch, which are selected stochastically and work adversarially to highlight important channels and remove maximally activated group. The attention-guided CDB achieves limited improvements compared with random selection, because the random one may yield more occlusion combinations, and are more likely to destruct co-adaptions between channels. We focus on Channel DropBlock with random selection for all of our experiments.

Compared with MA-CNN [32] that clusters channels on the final feature map and settles individual classifier for each cluster, the proposed CDB is designed as a regularization block which is more flexible to be applied on any convolutional feature maps of classification model.

Compared with SpatialDropout [27], CDB emphasizes that channels are correlated with each other, visual evidence can still be sent to the next layer with individually dropout.

Compared with DropBlock [12] that drops correlated units spatially, the proposed CDB calculates correlations channel-wise and can captures more precise visual evidences with two unique correlation metrics we provide.

3.2 Channel DropBlock Algorithm

Algorithm 1 and Figure 1 show the main procedure of the Channel DropBlock. Specifically, the input of CDB is a convolutional feature map $\mathbf{F} \in \mathbb{R}^{C \times H \times W}$, where C is the number of

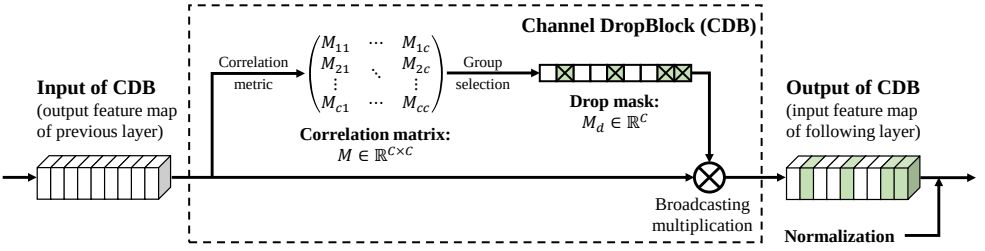

Figure 1: Illustration of the CDB block. The channel correlation matrix is generated based on different metrics. Then one channel and its corresponding visual group is randomly dropped by applying the drop mask to the input feature map.

Algorithm 1 Training of Channel DropBlock.

Input: Input feature map $\mathbf{F}$; Drop rate γ;

1: Calculate correlation matrix M
2: Randomly select a channel in M with equal probabilities and generate drop mask M_d with top γ most correlated channels setting as zero
3: Apply the mask: $\mathbf{F} = \mathbf{F} \times M_d$
4: Normalize the features: $\mathbf{F} = \frac{1}{1-\gamma}\mathbf{F}$
5: **return** $\mathbf{F}$;

channels, H and W denote to the height and width of $\mathbf{F}$, respectively. We obtain the correlation matrix $M \in \mathbb{R}^{C\times C}$ by calculating pairwise similarities between each feature channel (described in Section 3.3). To obtain the drop mask, CDB first randomly selects one line in M, and produces the drop mask $M_d \in \mathbb{R}^{C}$ by setting top γ most correlated elements as 0 and other elements as 1. The drop mask is then applied to the input feature map by broadcasting multiplication. In this way, features in a contiguous group are dropped together, which hides one certain discriminative pattern and encourages the model to learn other discriminative information that can also contribute to the final prediction. Similar to dropout, the proposed CDB only works in the training stage with normalization, no additional parameters and calculation costs are involved in the inference time.

CDB has two main hyperparameters: *insert_pos* and γ. The parameter *insert_pos* indicates where the CDB is applied, and γ controls the number of channels in the dropped group.

Influence of *insert_pos*. As the structure of CNN getting deeper, the neurons in high layers are strongly respond to entire images and rich in semantics, but inevitably lose detailed information from small discriminative regions. With different setting of *insert_pos*, the information of the input feature map differs. In our experiments, we settle an ablation study (described in Table 2) applying the proposed CDB block on varying layers in CNN.

Setting the value of γ. Another hyperparameter involves when we aggregate correlated channels into groups. Here we define γ as the percentage of channels in a dropped group when conducting CDB. In practice, different correlation metrics will result in different cluster numbers and the number of channels in each cluster, so the setting of γ is distinct from the correlation metrics we choose.

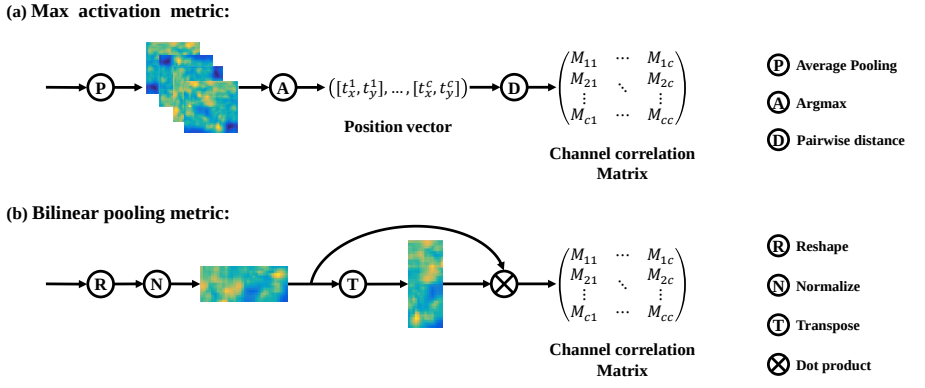

Figure 2: An illustration of channel correlation metrics: (a) max activation that groups channels into discriminative local region, and (b) bilinear pooling metric that groups channels based on visual pattern.

3.3 Channel Correlation

Ideally, a correlation metric should be symmetric and can cluster feature channels into different visual pattern groups. In this paper we examine two candidate metrics to evaluate the correlation between channels.

Max activation metric. In order to distribute feature channels into groups, an intuitive idea is to divide them into different focused local regions. Inspired by MA-CNN [32], we treat channels with close maximal activation position as a pattern group. We conduct 3×3 average pooling to smooth feature maps and use the $argmax(\cdot)$ operation to get the coordinates of the peak response in each feature channel, which turns the input feature map $\mathbf{F}$ into a position matrix $P \in \mathbb{R}^{C \times 2}$ given by:

$$P = [(t_x^1, t_y^1), (t_x^2, t_y^2), \cdots, (t_x^C, t_y^C)], \tag{1}$$

where t_x^i, t_y^i are the coordinates of the peak response of the i^{th} channel. We then compute pairwise Euclidean distance between each activation position and obtain the correlation matrix M:

$$M(i,j) = \|(t_x^i, t_y^i) - (t_x^j, t_y^j)\|^2. \tag{2}$$

In this metric, feature channels are grouped into discriminative local regions. Further more, it is a parameter-free metric where no learnable parameters are involved. Figure 2 (a) shows the procedure of the max activation metric.

Bilinear pooling metric. We also examine a correlation metric based on the bilinear pooling operator [18], which calculates normalized cosine distance to measure channel similarities. In this approach, the input feature map $\mathbf{F}$ is reshaped into a matrix with a shape of $C \times HW$, which is denoted as $\mathbf{X} \in \mathbb{R}^{C \times HW}$. Then the reshaped matrix is fed through a normalization function followed by a bilinear pooling operator to get spatial relationship among channels:

$$M \leftarrow \mathcal{N}(\mathbf{X})\mathcal{N}(\mathbf{X}^\mathbf{T}), \tag{3}$$

where $\mathcal{N}(\cdot)$ indicates the L2 normalization function over the second dimension of the matrix. $\mathbf{X}\mathbf{X}^\mathbf{T}$ is the homogeneous bilinear feature. Compared with the max activation metric, each

channel group in this metric indicates one specific visual pattern. Similarly, no trainable parameters are involved in both the training phase and the inference phase. Figure 2 (b) shows the procedure of the bilinear pooling metric.

4 Experimental Results and Discussions

Datasets. We evaluate the performance of the proposed method on three fine-grained benchmark datasets: CUB-200-2011 (CUB) [28], Stanford-Cars (CAR) [15], and FGVC-Aircraft (AIR) [20], together with two classical image classification benchmark datasets: CIFAR-10 (C10) [16] and CIFAR-100 (C100) [16].

Implementation details. We use VGG19 [25] and ResNet50 [13] as backbone networks, and replace the origin classifier with an average pooling layer followed by a fully connected layer. We apply CDB with γ set to 20% for the max activation metric and 5% for the bilinear pooling metric. We use open-source PyTorch [21] as our code-base, and train all the models on a single GTX 1080Ti GPU. Optimization is performed using Stochastic Gradient Descent with momentum of 0.9 and weight decay of 5e-4.

For the FGVC datasets, the input images are resized to 448×448. All of the models are pretrained on ImageNet and fine tuned for another 100 epochs with batches of 16 images. The initial learning rate is set to 0.001 and drops to 0 using cosine anneal schedule [19]. For the C10/C100 datasets, the input images are first zero-padded with 4 pixels on each side, then randomly cropped into 32×32. Models are trained from scratch for 200 epochs with each batch containing 128 images. The initial learning rate is set to 0.1 and drops to 0 using cosine anneal schedule.

4.1 Fine-grained Image Classification

The comparisons of our method with other state-of-the-art methods on three benchmark FGVC datasets are presented in Table 1. The first block lists recent works for weakly supervised FGVC.

Apply to baseline networks. The second block in Table 1 summarizes the performance of CDB on two baseline networks and three benchmark FGVC datasets. For the CUB dataset we follow the enhanced network setting and augmentation strategy in PMG [8], in which the classifier is combined with two fully connected layers, the input images are resize to a size of 550×550, and randomly cropped to 448×448 in the training time, centrally cropped to 448×448 in the inference time.

We find that CDB with two correlation metrics outperforms the baselines on all FGVC datasets by a clear margin. Since FGVC focuses on differentiating sub-categories that share close similarities, it requires much more fine-grained visual evidence for prediction. The CDB block drops one entire visual group each iteration that induces model to distil other discriminative parts, thus being extremely suitable for this task.

Apply to SOTA methods. The third block in Table 1 shows the results applying CDB on existing FGVC methods. We choose BCNN and PMG as benchmark models, the former one is a classical feature encoding-based approach which encode higher order information on features, the latter one is a SOTA approach which adopt a progressive training strategy that fuses multi-granularity features. CDB further improves the accuracies for a relative margin of 0.5%, 0.6% ,1.4% on BCNN, and 0.3%, 0.3%, 0.4% on PMG.

Method	Backbone	CUB (%)	CAR (%)	AIR (%)
FT VGGNet [25]	VGG19	85.8	84.6	85.8
FT ResNet [13]	ResNet50	86.6	92.3	90.8
BCNN [18]	VGG16	84.1	91.3	89.0
MA-CNN [32]	VGG19	86.5	92.8	89.9
MC-Loss [4]	ResNet50	87.3	93.7	92.6
CIN [10]	ResNet50	87.5	94.1	92.8
API-Net [33]	ResNet50	87.7	94.8	93.0
DCL [2]	ResNet50	87.8	94.5	93.0
PMG [8]	ResNet50	89.6	<u>95.1</u>	93.4
CDB-MA (VGGNet)	VGG19	86.2	87.0	87.8
CDB-BP (VGGNet)	VGG19	86.1	86.6	85.9
CDB-MA (ResNet)	ResNet50	86.9	93.5	91.9
CDB-BP (ResNet)	ResNet50	87.2	93.2	91.5
CDB-MA (BCNN)	VGG16	84.6	91.9	90.4
CDB-BP (BCNN)	VGG16	84.5	91.7	90.0
CDB-MA (PMG)	ResNet50	**89.9**	**95.4**	<u>93.7</u>
CDB-BP (PMG)	ResNet50	<u>89.7</u>	**95.4**	**93.8**

Table 1: Comparison results on CUB-200-2011, Stanford Cars, and FGVC-Aircraft datasets. CDB is applied on features from *conv*2 and *conv*3, with γ set to 20% for the max activation metric and 5% for the bilinear pooling metric. CDB-MA and CDB-BP indicate CDB with the max activation metric and the bilinear pooling metric, respectively. The best and second-best results are marked respectively in bold and underlined fonts.

4.2 Ablation study

We conduct ablation studies to analyze the influence of insert position, and compare CDB with other regularization techniques. We also settle experiments analyzing the performance of CDB on traditional image classification tasks. The following experiments are all conducted on the Stanford-Cars dataset with ResNet50 as backbone if not particularly mentioned.

Where to apply CDB. In order to judge the influence of *insert_pos*, we apply CDB on variant layers in CNN. Specifically, we define v1 to v5 indicating the last layer of *conv*1 to *conv*5 in ResNet50, and *pool*1 to *pool*5 in VGG19, indicating feature positions with multi-level information from low-level details to high-level semantics, and apply CDB with different *insert_pos* alone or their combinations. The experimental results in Table 2 suggest that with the *insert_pos* on both v2&v3 gives the best results. This is because the middle features in v2 and v3 contains both high-level semantics and low-level details, which contributes to the FGVC task.

Metric	v1 (%)	v2 (%)	v3 (%)	v4 (%)	v5 (%)	v2&v3 (%)
MA	92.9	93.2	93.2	92.6	93.0	**93.5**
BP	93.1	92.7	93.0	92.8	92.9	**93.2**

Table 2: Ablation study on insert position.

Comparison with other regularization techniques. We compare the proposed CDB with different dropout based regularization techniques including dropout [26], cutout [6], SpatialDropout [27], and DropBlock [12]. We conduct experiments on the FGVC datasets with ResNet50 as backbone, and train the model with different settings and report the best results. As shown in Table 3, CDB achieves the best accuracy of 87.2%, 93.5%, and 91.9% on three FGVC benchmark datasets, respectively, which confirms its significance.

Method	insert_pos	CUB (%)	CAR (%)	AIR (%)
Baseline	–	86.6	92.3	90.9
Dropout (p=0.2) [26]	v2	86.9	92.7	91.7
SpatialDropout (p=0.1) [27]	v2	86.9	92.8	91.5
Cutout [6]	–	87.1	92.7	90.9
DropBlock (p=0.25) [12]	v2&v3	86.8	93.1	90.9
CDB-MA	v2&v3	86.9	**93.5**	**91.9**
CDB-BP	v2&v3	**87.2**	93.2	91.5

Table 3: Comparison results with different regularization techniques. p indicates the drop probability of each method.

Comparison between different correlation metrics. It can be found that both the proposed max activation metric and the bilinear pooling metric outperform the baseline settings. However, it is not consistent on different datasets. To give more quantitative insight into this trend, we take all the experimental results of each metric as trails and use the non-parametric Wilcoxon Signed-Rank Test [23] to quantify their differences. The returned p-values equals 0.025 demonstrates a clear distribution difference between them. To this end, we give both the two metrics as candidates to better deal with varying real-world cases.

Traditional Image Classification. We conduct identical experiments on CIFAR-10 and CIFAR-100 dataset to evaluate the performance of CDB on traditional image classification task. CIFAR-10 has 10 distinct classes, such as cat, dog, car, and boat. CIFAR-100 contains 100 classes, but requires much more fine-grained recognition compared to CIFAR-10 as some classes are very visually similar. For example, it contains five different classes of trees: maple, oak, palm, pine, and willow.

Method	C10 (%)	C100 (%)
ResNet50	95.1	78.1
ResNet50 + CDB-MA	95.4	**79.4**
ResNet50 + CDB-BP	**95.5**	78.5
VGG19	93.7	71.8
VGG19 + CDB-MA	93.7	72.5
VGG19 + CDB-BP	**93.9**	**73.4**

Table 4: Comparison results on CIFAR-10 and CIFAR-100 datasets.

Table 4 summarize the performance on traditional image classification. It can be observed that the improvement in C10 is limited while in C100 significant. As C100 requires much more fine-grained recognition compared to C10, it demonstrate that CDB is more effective in fine-grained tasks.

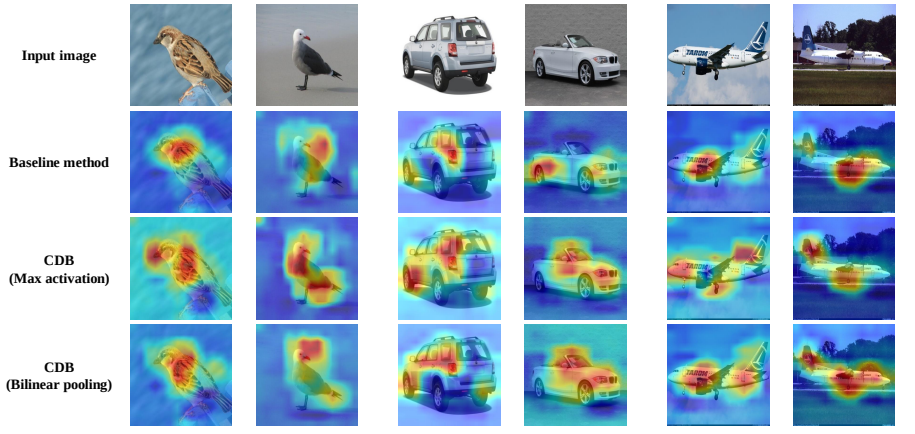

Figure 3: Visualization of the Gradient-weighted Class Activation Map (Grad-CAM) [22] on FGVC samples with ResNet50 as the backbone model. The model trained with CDB tends to focus on multiple discriminative patterns.

4.3 Visualization

In order to draw insights of the proposed method, we apply Grad-CAM [22] to visualize class activations of fine-grained samples on ResNet50 trained with and without CDB. We select test images from CUB-200-2011, Stanford-Cars, and FGVC-Aircraft, respectively. The second row of Figure 3 shows the class activations of the baseline model and the third and fourth rows are visualization of the proposed method with two proposed correlation metrics. Consistent observations demonstrate that models trained with CDB not only dilate activations of the baseline model, but also distil other visual patterns to improve feature representations.

5 Conclusion

In this paper we introduce a novel regularization technique, *Channel DropBlock* (CDB), which destructs feature channels from co-adaptation by clustering channels with correlation metrics and dropping a correlated channel group randomly during training. We demonstrate that CDB is more lightweight and effective to enhance feature representations and distill multiple discriminative patterns than existing FGVC methods. We conduct experiments on three widely tested fine-grained datasets, which confirm the superiority of our method. Two particularly interesting directions for future work include exploring the method that group channels with adaptive size, and measuring channel correlations with integrated metrics.

6 Acknowledgements

This work was supported in part by the National Key R&D Program of China under Grant 2019YFF0303300 and under Subject II No. 2019YFF0303302, in part by the National Natural Science Foundation of China (NSFC) under Grant 61773071, Grant 61922015, Grant U19B2036, and Grant 62076031, in part by the Beijing Natural Science Foundation Project No. Z200002.

References

[1] Thomas Berg and Peter N Belhumeur. Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 955–962, 2013.

[2] Yuning Chai, Victor Lempitsky, and Andrew Zisserman. Symbiotic segmentation and part localization for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 321–328, 2013.

[3] Dongliang Chang, Yifeng Ding, Jiyang Xie, Ayan Kumar Bhunia, Xiaoxu Li, Zhanyu Ma, Ming Wu, Jun Guo, and Yi-Zhe Song. The devil is in the channels: Mutual-channel loss for fine-grained image classification. *IEEE Transactions on Image Processing*, 29: 4683–4695, 2020.

[4] Yue Chen, Yalong Bai, Wei Zhang, and Tao Mei. Destruction and construction learning for fine-grained image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5157–5166, 2019.

[5] Junsuk Choe and Hyunjung Shim. Attention-based dropout layer for weakly supervised object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2219–2228, 2019.

[6] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[7] Yifeng Ding, Zhanyu Ma, Shaoguo Wen, Jiyang Xie, Dongliang Chang, Zhongwei Si, Ming Wu, and Haibin Ling. Ap-cnn: weakly supervised attention pyramid convolutional neural network for fine-grained visual classification. *IEEE Transactions on Image Processing*, 30:2826–2836, 2021.

[8] Ruoyi Du, Dongliang Chang, Ayan Kumar Bhunia, Jiyang Xie, Zhanyu Ma, Yi-Zhe Song, and Jun Guo. Fine-grained visual classification via progressive multi-granularity training of jigsaw patches. In *European Conference on Computer Vision*, pages 153–168. Springer, 2020.

[9] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4438–4446, 2017.

[10] Yu Gao, Xintong Han, Xun Wang, Weilin Huang, and Matthew Scott. Channel interaction networks for fine-grained image categorization. In *AAAI*, pages 10818–10825, 2020.

[11] Weifeng Ge, Xiangru Lin, and Yizhou Yu. Weakly supervised complementary parts models for fine-grained image classification from the bottom up. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3034–3043, 2019.

[12] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Dropblock: A regularization method for convolutional networks. In *Advances in Neural Information Processing Systems*, pages 10727–10737, 2018.

[13] 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.

[14] Shaoli Huang, Zhe Xu, Dacheng Tao, and Ya Zhang. Part-stacked cnn for fine-grained visual categorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1173–1182, 2016.

[15] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013.

[16] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[17] Jianjun Lei, Jinhui Duan, Feng Wu, Nam Ling, and Chunping Hou. Fast mode decision based on grayscale similarity and inter-view correlation for depth map coding in 3d-hevc. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(3):706–718, 2016.

[18] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 1449–1457, 2015.

[19] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[20] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[21] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[22] 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.

[23] Sidney Siegel. Nonparametric statistics for the behavioral sciences. 1956.

[24] Marcel Simon and Erik Rodner. Neural activation constellations: Unsupervised part model discovery with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1143–1151, 2015.

[25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[26] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[27] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. Efficient object localization using convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 648–656, 2015.

[28] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[29] Ze Yang, Tiange Luo, Dong Wang, Zhiqiang Hu, Jun Gao, and Liwei Wang. Learning to navigate for fine-grained classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 420–435, 2018.

[30] Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell. Part-based r-cnns for fine-grained category detection. In *European conference on computer vision*, pages 834–849. Springer, 2014.

[31] Xiaopeng Zhang, Hongkai Xiong, Wengang Zhou, Weiyao Lin, and Qi Tian. Picking deep filter responses for fine-grained image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1134–1142, 2016.

[32] Heliang Zheng, Jianlong Fu, Tao Mei, and Jiebo Luo. Learning multi-attention convolutional neural network for fine-grained image recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 5209–5217, 2017.

[33] Peiqin Zhuang, Yali Wang, and Yu Qiao. Learning attentive pairwise interaction for fine-grained classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13130–13137, 2020.