CBAM: Convolutional Block Attention Module

Abstract

infers attention maps along two separate dimensions, channel and spatial
a lightweight and general module

Introduction

기존 - depth, width, and cardinality.
cardinality 이점
saves the total number of parameters
stronger representation power

Attention
where to focus
representation of interests.

Our goal - increase representation power by using attention mechanism: focusing on important features and suppressing unnecessary ones.

emphasize meaningful features along those two principal dimensions: channel and spatial axes
each of the branches can learn ‘what’ and ‘where’ to attend in the channel and spatial axes respectively

Contribution.
1. We propose a simple yet effective attention module (CBAM) that can be widely applied to boost representation power of CNNs.
2. We validate the effectiveness of our attention module through extensive ablation studies.
3. We verify that performance of various networks is greatly improved on the multiple benchmarks (ImageNet-1K, MS COCO, and VOC 2007) by plugging our light-weight module.

Related Work

Network Engineering
most of recent network engineering methods - depth [19, 9, 10, 5], width [10, 22, 6, 8], and cardinality [7, 11]
we focus on the other aspect, ‘attention’, one of the curious facets of a human visual system.

Attention Mechanism
attention is important at human perception

Residual Attention Network
we decompose the process that learns channel attention and spatial attention separately.
-> less compute, parameter, overhead -> plug-and-play

Squeeze-and-Excitation
suboptimal features in order to infer fine channel attention
miss the spatial attention

Convolutional Block Attention Module

$$\mathbf{F}' = \mathbf{M_c}(\mathbf{F}) \otimes \mathbf{F},$$

$$\mathbf{F''} = \mathbf{M_s}(\mathbf{F'}) \otimes \mathbf{F'},$$

Channel Attention Module

channel attention focuses on 'what' is meaningful given an input image.

squeeze the spatial dimension

average-pooling

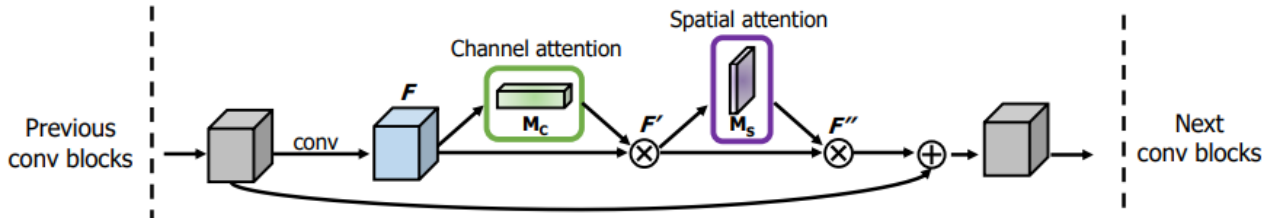
max-pooling

exploiting both

$$\begin{aligned}\mathbf{M_c}(\mathbf{F}) &= \sigma(MLP(AvgPool(\mathbf{F})) + MLP(MaxPool(\mathbf{F}))) \\ &= \sigma(\mathbf{W_1}(\mathbf{W_0}(\mathbf{F^c_{avg}})) + \mathbf{W_1}(\mathbf{W_0}(\mathbf{F^c_{max}}))),\end{aligned}$$

Spatial attention module

average-pooling and max-pooling operations along the channel axis and concatenate them to generate an efficient feature descriptor



ResBlock + CBAM

$$\mathbf{M_s}(\mathbf{F}) = \sigma(f^{7\times7}([AvgPool(\mathbf{F}); MaxPool(\mathbf{F})])) \\ = \sigma(f^{7\times7}([\mathbf{F^s_{avg}}; \mathbf{F^s_{max}}])),$$

Arrangement of attention modules

sequential arrangement > parallel arrangement.

channel-firs > spatial-first.

Experiments

Ablation Studies

We first search for the effective approach to computing the channel attention, then the spatial attention.

Finally, we consider how to combine both channel and spatial attention modules

Channel Attention

Description	Parameters	GFLOPs	Top-1 Error(%)	Top-5 Error(%)
ResNet50 (baseline)	25.56M	3.86	24.56	7.50
ResNet50 + AvgPool (SE [28])	25.92M	3.94	23.14	6.70
ResNet50 + MaxPool	25.92M	3.94	23.20	6.83
ResNet50 + AvgPool & MaxPool	25.92M	4.02	**22.80**	**6.52**

max-pooled - encode the degree of the most salient part can compensate

average-pooled - encode global statistics softly

Spatial Attention

Description	Param.	GFLOPs	Top-1 Error(%)	Top-5 Error(%)
ResNet50 + channel (SE [28])	28.09M	3.860	23.14	6.70
ResNet50 + channel	28.09M	3.860	22.80	6.52
ResNet50 + channel + spatial (1x1 conv, k=3)	28.10M	3.862	22.96	6.64
ResNet50 + channel + spatial (1x1 conv, k=7)	28.10M	3.869	22.90	6.47
ResNet50 + channel + spatial (avg&max, k=3)	28.09M	3.863	22.68	6.41
ResNet50 + channel + spatial (avg&max, k=7)	28.09M	3.864	**22.66**	**6.31**

Arrangement of the channel and spatial attention

Description	Top-1 Error(%)	Top-5 Error(%)
ResNet50 + channel (SE [28])	23.14	6.70
ResNet50 + channel + spatial	**22.66**	**6.31**
ResNet50 + spatial + channel	22.78	6.42
ResNet50 + channel & spatial in parallel	22.95	6.59

Final module design

average- and max-pooling

convolution with a kernel size of 7 in the spatial attention module

the channel and spatial submodules sequentially

Image Classification on ImageNet

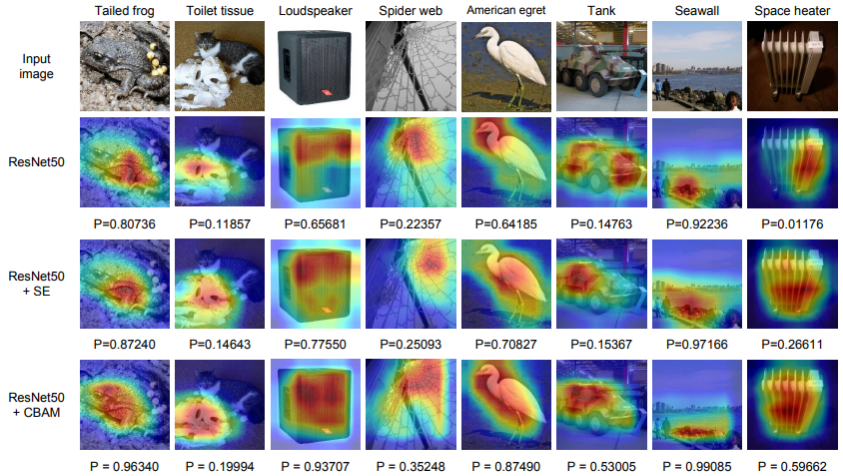
Architecture	Param.	GFLOPs	Top-1 Error (%)	Top-5 Error (%)
ResNet18 [5]	11.69M	1.814	29.60	10.55
ResNet18 [5] + SE [28]	11.78M	1.814	29.41	10.22
ResNet18 [5] + CBAM	11.78M	1.815	**29.27**	**10.09**
ResNet34 [5]	21.80M	3.664	26.69	8.60
ResNet34 [5] + SE [28]	21.96M	3.664	26.13	8.35
ResNet34 [5] + CBAM	21.96M	3.665	**25.99**	**8.24**
ResNet50 [5]	25.56M	3.858	24.56	7.50
ResNet50 [5] + SE [28]	28.09M	3.860	23.14	6.70
ResNet50 [5] + CBAM	28.09M	3.864	**22.66**	**6.31**
ResNet101 [5]	44.55M	7.570	23.38	6.88
ResNet101 [5] + SE [28]	49.33M	7.575	22.35	6.19
ResNet101 [5] + CBAM	49.33M	7.581	**21.51**	**5.69**
WideResNet18 [6] (widen=1.5)	25.88M	3.866	26.85	8.88
WideResNet18 [6] (widen=1.5) + SE [28]	26.07M	3.867	26.21	8.47
WideResNet18 [6] (widen=1.5) + CBAM	26.08M	3.868	**26.10**	**8.43**
WideResNet18 [6] (widen=2.0)	45.62M	6.696	25.63	8.20
WideResNet18 [6] (widen=2.0) + SE [28]	45.97M	6.696	24.93	7.65
WideResNet18 [6] (widen=2.0) + CBAM	45.97M	6.697	**24.84**	**7.63**
ResNeXt50 [7] (32x4d)	25.03M	3.768	22.85	6.48
ResNeXt50 [7] (32x4d) + SE [28]	27.56M	3.771	**21.91**	6.04
ResNeXt50 [7] (32x4d) + CBAM	27.56M	3.774	21.92	**5.91**
ResNeXt101 [7] (32x4d)	44.18M	7.508	21.54	5.75
ResNeXt101 [7] (32x4d) + SE [28]	48.96M	7.512	21.17	5.66
ResNeXt101 [7] (32x4d) + CBAM	48.96M	7.519	**21.07**	**5.59**

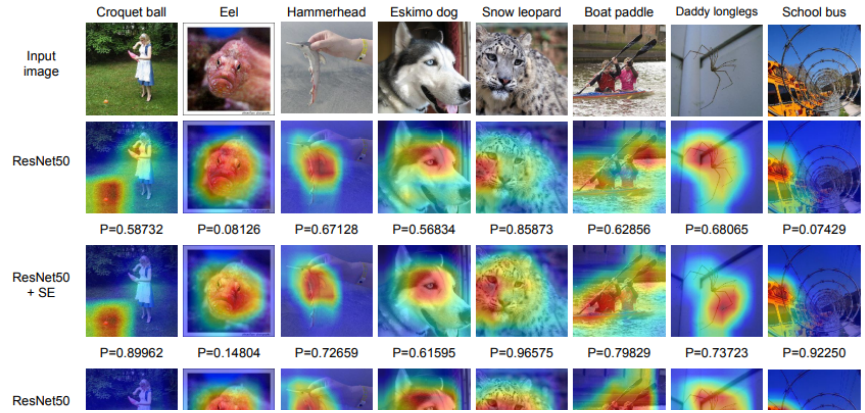
* all results are reproduced in the PyTorch framework.

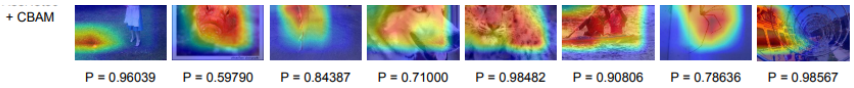
Architecture	Parameters	GFLOPs	Top-1 Error (%)	Top-5 Error (%)
MobileNet [34] $\alpha = 0.7$	2.30M	0.283	34.86	13.69
MobileNet [34] $\alpha = 0.7$ + SE [28]	2.71M	0.283	32.50	12.49
MobileNet [34] $\alpha = 0.7$ + CBAM	2.71M	0.289	**31.51**	**11.48**
MobileNet [34]	4.23M	0.569	31.39	11.51
MobileNet [34] + SE [28]	5.07M	0.570	29.97	10.63
MobileNet [34] + CBAM	5.07M	0.576	**29.01**	**9.99**

* all results are reproduced in the PyTorch framework.

Network Visualization with Grad-CAM







Conclusion

channel and spatial

keeping the overhead small

max-pooled features along with the average-pooled features