2021 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-3864-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICME51207.2021.9428397

LEARNING CONNECTED ATTENTIONS FOR CONVOLUTIONAL NEURAL NETWORKS

Xu Ma[1], *Jingda Guo*[1], *Sihai Tang*[1], *Zhinan Qiao*[1], *Qi Chen*[1], *Qing Yang*[1], *Song Fu*[1],
Paparao Palacharla[2], *Nannan Wang*[2], *Xi Wang*[2]

[1]University of North Texas, [2]Fujitsu Network Communications
{xuma, jingdaguo, sihaitang, zhinanqiao, qichen}@my.unt.edu; {qing.yang, Song.Fu}@unt.edu;
{Paparao.Palacharla, Nannan.Wang, Xi.Wang}@fujitsu.com

ABSTRACT

While self-attention mechanism has shown promising results for many vision tasks, it only considers the current features at a time. We show that such a manner cannot take full advantage of the attention mechanism. In this paper, we present Deep Connected Attention Network (DCANet), a novel design that boosts attention modules in a CNN model without any modification of the internal structure. To achieve this, we interconnect adjacent attention blocks, making information flow among attention blocks possible. With DCANet, all attention blocks in a CNN model are trained jointly, which improves the ability of attention learning. Our DCANet is generic. It is not limited to a specific attention module or base network architecture. Experimental results on ImageNet and MS COCO benchmarks show that DCANet consistently outperforms the state-of-the-art attention modules with a minimal additional computational overhead in all test cases. The code is available at: https://github.com/13952522076/DCANet .

Index Terms— Convolutional neural network, self-attention mechanism, computer vision

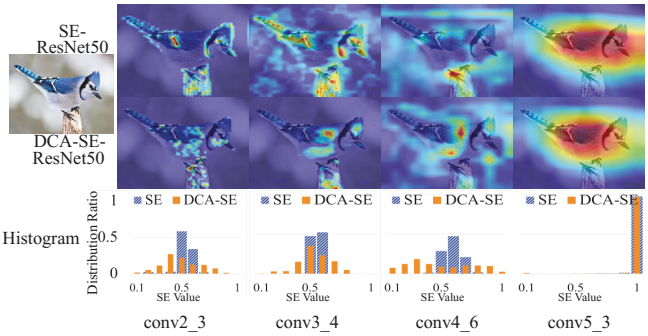

Fig. 1: Illustration of our DCANet. Top and middle lines: we visualize the class activation maps [2]. Vanilla SE-ResNet50 varies its focus dramatically at different stages. In contrast, our DCA enhanced SE-ResNet50 **progressively and recursively** adjusts focus, and closely pays attention to the target object. Bottom line: Corresponding histogram of SE attention values. Clearly, the values of SE are concentrated around 0.5, resulting in little discrimination. With DCANet, the distribution becomes relatively uniform.

1. INTRODUCTION

In the last few years, we have witnessed a flourish of self-attention mechanism in the vision community. As a common practice in self-attention design, the attention modules are integrated sequentially with each block in a base CNN architecture, in pursuit of an easy and efficient implementation. Benefiting from the inherent philosophy and this simple design, self-attention mechanism performs well in a diverse range of visual tasks. In spite of the improvement achieved by the existing designs, a question we ask is: do we take full advantage of self-attention mechanism? We can address this question from two aspects: human visual attention system and empirical insights from SENet [1].

Previous studies in the literature provided deep insights into the human visual attention system. In [3], experimental results indicate that two stimuli present at the same time in the human cortex are not processed independently. Instead they interact with each other. Moreover, research in physiology discovers that human visual representations in the cortex are activated in a parallel fashion, and the cells participating in these representations are engaged by interacting with each other [4]. These works show the *important interaction among attention units*. However, this critical property of human visual attention has not been considered in the existing designs of self-attention modules. Existing attention networks only include an attention block following a convolutional block, which makes the attention block only learn from current feature maps without sharing information with others. As a result, the independent attention blocks cannot effectively decide what to pay attention to.

Additionally, we study self-attention using SENet [1] which is a simple module that investigates the channel relationships. We visualize the intermediate attention maps as shown in Fig. 1 (top line) at each stage in SE-ResNet50 [5]. Interestingly, we observe that SE block can hardly adjust the

attention to the key regions, and it even changes focus dramatically at different stages. We plot a histogram of SE's attention value for each block, as shown in Fig. 1 (bottom line). We find that SE's values cluster around 0.5, showing an insufficient learning ability of the attention modules. A reasonable explanation is that a lack of extra information in learning from self-attention affects its discrimination ability. This in turn motivates us to connect attention blocks.

Both human visual attention and our study of SENet show an insufficient exploitation of self-attention, and a new design that allows attention blocks to cooperate with each other is desirable. In this paper, we present a Deep Connected Attention network (DCANet) to address the problem. DCANet gathers information from precedent attention and transmits it to the next attention block, making attention blocks cooperate with each other, which improves attention's learning ability.

DCANet is conceptually simple and generic and empirically powerful. We apply DCANet to multiple state-of-the-art attention modules and a number of base CNN architectures to evaluate its performance for visual tasks. Without bells and whistles, the DCA-enhanced networks **outperform all of the original counterparts**. For ImageNet 2012 classification [6], DCA-SE-MobileNetV2 outperforms SE-MobileNetV2 by 1.19%, with negligible parameters and FLOPs increase. We also employ the DCA-enhanced attention network as a backbone for object detection on the MS COCO dataset [7]. Experimental results show that the DCANet-enhanced attention networks outperform the vanilla networks with different detectors.

2. RELATED WORK

Self-attention mechanisms. Self-attention mechanism explores the interdependence within the input features for a better representation. To the best of our knowledge, applying self-attention to explore global dependencies was first proposed in [8] for machine translation. More recently, self-attention has gathered much more momentum in the field of computer vision. SENet [1] leverages self-attention to investigate channel interdependencies. For global context information, NLNet [9] and GCNet [10] introduce self-attention to capture long-range dependencies in non-local operations. CBAM [11] considers both channel-wise and spatial attentions. Beyond channel and spatial dependencies, SKNet [12] applies self-attention to kernel size selection.

Residual connections. By introducing a shortcut connection, neural networks are decomposed into biased and centered subnets to accelerate gradient descent. ResNet [5, 13] adds an identity mapping to connect the input and output of each convolutional block, which drastically alleviates the degradation problem [5] and opens up the possibility for deep convolutional neural networks. Instead of connecting adjacent convolutional blocks, DenseNet [14] connects each block to every other block in a feed-forward fashion. Despite the fact that residual connections have been well studied for base network architectures, they are still fairly new when it comes to integration with attention mechanisms. For example, RANet [15] utilizes residual connections in attention block. In contrast to leveraging residual connection *in* attention blocks, we explore residual connections *between* attention blocks.

Connected Attention. Recently, there has been a growing interest for building connections in attention blocks. In [16], a new network structure named RA-CNN is proposed for fine-grained image recognition; RA-CNN recurrently generates attention region based on current prediction to learn the most discriminative region. By doing so, RA-CNN obtains an attention region from coarse to fine. In GANet [17], the top attention maps generated by customized background attention blocks are up-sampled and sent to bottom background attention blocks to guide attention learning. Different from the recurrent and feed-backward methods, our DCA module enhances attention blocks in a feed-forward fashion, which is more computation-friendly and easier to implement.

3. DEEP CONNECTED ATTENTION

By analyzing the inner structure of various attention blocks, we design a generic connection scheme that is not confined to any particular attention block. Fig. 2 illustrates the pipeline.

3.1. Revisiting Self-Attention Blocks

As a common practice, we boost the base CNN architecture by adding extra attention blocks laterally. However, different attention blocks are tailored for different purposes, their implementations are also diverse. For instance, SE block composes of two fully-connected layers, while GC block includes several convolutional layers. Therefore, it is not easy to directly provide a standard connection schema that is generic enough to cover most attention blocks. To tackle this problem, we study the state-of-the-art attention blocks and summarize their processing and components.

Inspired by recent works [10, 9] that formulate attention modules and their components, we study various attention modules and develop a generalized attention framework, in which an attention block consists of three components: context extraction, transformation, and fusion. These components are generic and not confined to a particular attention block. Figure 3 exemplifies four well-known attention blocks and their modeling by using the three components.

Extraction is designed for gathering feature information from a feature map. For a given feature map $\mathbf{X} \in \mathbb{R}^{C\times W\times H}$ produced by a convolutional block, we extract features from $\mathbf{X}$ by an extractor g: $\mathbf{G} = g(\mathbf{X}, w_g)$, where w_g is the parameter for the extraction operation and $\mathbf{G}$ is the output. When g is a parameter-free operation, w_g is not needed (like pooling operations). The flexibility of g makes $\mathbf{G}$ take different shapes depending on the extraction operation. For instance, SENet

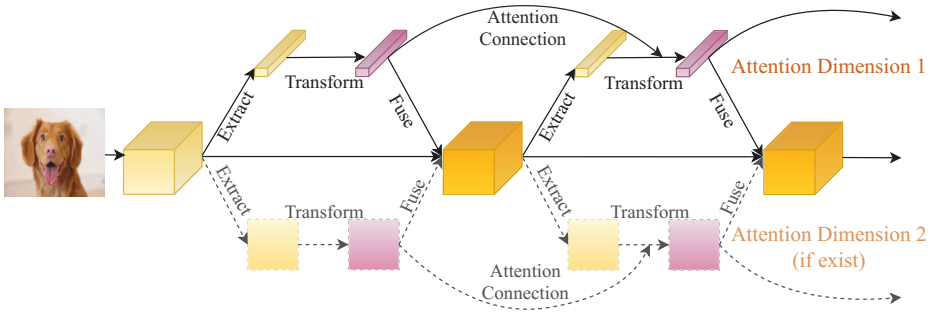

Fig. 2: **An overview of our Deep Connected Attention Network.** We connect the output of transformation module in the previous attention block to the output of extraction module in the current attention block. In the context of multiple attention dimensions, we connect attentions along each dimension. Here we show an example with two attention dimensions.

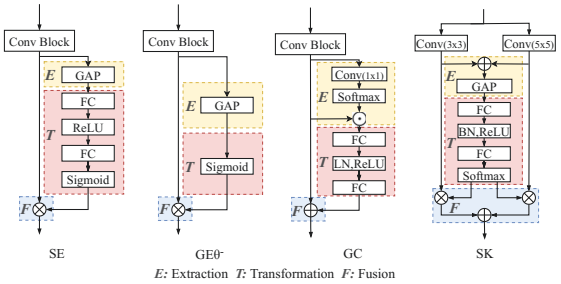

Fig. 3: We model an attention block by three components: Feature extraction, Transformation and Fusion. "$\oplus$" denotes element-wise summation, "$\otimes$" represents element-wise multiplication, and "$\odot$" performs matrix multiplication.

and GCNet gather feature map $\mathbf{X}$ as a vector ($\mathbf{G} \in \mathbb{R}^C$) while the spatial attention module in CBAM gathers feature map to a tensor ($\mathbf{G} \in \mathbb{R}^{2\times W\times H}$).

Transformation processes the gathered features from extraction and transforms them into a non-linear attention space. Formally, we define t as a feature transformation, and the output of an attention block can be expressed as $\mathbf{T} = t(\mathbf{G}, w_t)$. Here w_t denotes the parameters used in the transform operation, and $\mathbf{T}$ is the output of the extraction module.

Fusion integrates the attention map with the output of the original convolutional block. An attention guided output $\mathbf{X}'$ can be presented as $\mathbf{X}'_i = \mathbf{T}_i \circledast \mathbf{X}_i$, where i is the index in a feature map and "$\circledast$" denotes a fusion function; "$\circledast$" performs element-wise multiplication when the design is scaled dot-product attention [12, 1, 11], and summation otherwise [10].

3.2. Attention Connection

We present a generalized attention connection schema by using the preceding attention components. Regardless of the implementation details, an attention block can be modeled as:

$$\mathbf{X}' = t\left(g\left(\mathbf{X}, w_g\right), w_t\right) \circledast \mathbf{X}. \tag{1}$$

As explained in the previous section, the attention maps generated by the transformation component is crucial for attention learning. To construct connected attention, we feed the previous attention map to the current transformation component, which merges previous transformation output and the current extraction output. **This connection design ensures the current transformation module learns from both extracted features and previous attention information.** The resulting attention block can be described as:

$$\mathbf{X}' = t\left(f\left(\alpha\mathbf{G}, \beta\tilde{\mathbf{T}}\right), w_t\right) \circledast \mathbf{X}, \tag{2}$$

where $f(\cdot)$ denotes the connection function, α and β are learnable parameters, and $\tilde{\mathbf{T}}$ is the attention map generated by the previous attention block. In some cases (e.g., SE block and GE block), $\tilde{\mathbf{T}}$ is scaled to the range of $(0, 1)$. For those attention blocks, we multiply $\tilde{\mathbf{T}}$ by $\tilde{\mathbf{E}}$ to match the scale, where $\tilde{\mathbf{E}}$ is the output of the Extraction component in the previous attention block. We also note that if α is set to 1 and β is set to 0, the attention connections are not used and the DCA enhanced attention block is reduced to the vanilla attention block. That is the vanilla network is a special case of our DCA enhanced attention network. Next, we present two schemas that instantiate the connection function $f(\cdot)$.

Direct Connection. We instantiate $f(\cdot)$ by adding the two terms directly. The connection function can be presented as: $f\left(\alpha\mathbf{G}_i, \beta\tilde{\mathbf{T}}_i\right) = \alpha\mathbf{G}_i + \beta\tilde{\mathbf{T}}_i$, where i is the index of a feature. $\tilde{\mathbf{T}}$ can be considered as an enhancement of $\mathbf{G}$.

Weighted Connection. Direct connection can be augmented by using weighted summation. To avoid introducing extra parameters, we calculate weights using $\alpha\mathbf{G}$ and $\beta\tilde{\mathbf{T}}$. The connection function is represented as $f\left(\alpha\mathbf{G}_i, \beta\tilde{\mathbf{T}}_i\right) = \frac{|\alpha\mathbf{G}_i|^2}{\alpha\mathbf{G}_i+\beta\tilde{\mathbf{T}}_i} + \frac{|\beta\tilde{\mathbf{T}}_i|^2}{\alpha\mathbf{G}_i+\beta\tilde{\mathbf{T}}_i}$. Compared to the direct connection, the weighted connection introduces a competition between $\alpha\mathbf{G}$ and $\beta\tilde{\mathbf{T}}$. Besides, it can be easily extended to a softmax form, which is more robust and less sensitive to trivial features. By default, we use a direct connection in our method.

Table 1: Single-crop classification accuracy (%) on the ImageNet validation set. The best performances are marked as **bold**. "-" means no experiments since our DCA is designed for attention blocks, which are not existent in base networks.

	Re-implementation				DCANet			
	Top-1	Top-5	GFLOPs	Params	Top-1	Top-5	GFLOPs	Params
MoibleNetV2 [18]	71.03	90.07	0.32	3.50M	-	-	-	-
+ SE [1]	72.05	90.58	0.32	3.56M	**73.24**	91.14	0.32	3.65M
+ SK [12]	74.05	91.85	0.35	5.28M	**74.45**	91.85	0.36	5.91M
+ GEθ^- [19]	72.28	90.91	0.32	3.50M	**72.47**	90.68	0.32	3.59M
+ CBAM [11]	71.91	90.51	0.32	3.57M	**73.04**	91.18	0.34	3.65M
Mnas1_0 [20]	71.72	90.32	0.33	4.38M	-	-	-	-
+ SE [1]	69.69	89.12	0.33	4.42M	**71.76**	90.40	0.33	4.48M
+ GEθ^- [19]	72.72	90.87	0.33	4.38M	**72.82**	91.18	0.33	4.48M
+ CBAM [11]	69.13	88.92	0.33	4.42M	**71.00**	89.78	0.33	4.56M
ResNet50 [5]	75.90	92.72	4.12	25.56M	-	-	-	-
+ SE [1]	77.29	93.65	4.13	28.09M	**77.55**	93.77	4.13	28.65M
+ SK [12]	77.79	93.76	5.98	37.12M	**77.94**	93.90	5.98	37.48M
+ GEθ^- [19]	76.24	92.98	4.13	25.56M	**76.75**	93.36	4.13	26.12M
+ CBAM [11]	77.28	93.60	4.14	28.09M	**77.83**	93.72	4.14	30.90M

3.3. Size Matching

Feature maps produced at different stages in a CNN model may have different sizes. We match the shape of attention maps along the channel and spatial dimensions adaptively. For the channel, we match sizes using a fully-connected layer to convert C' channels to C channels, where C' and C refer to the number of previous and current channels, respectively. To further reduce the number of parameters in attention connections, we re-formulate the direct fully-connected layer by two lightweight fully-connected layers; the output sizes are C/r and C, respectively, where r is reduction ratio. In all our experiments, we use two fully-connected layers with $r = 16$ to match channel size, unless otherwise stated. To match the spatial resolutions, a simple yet effective strategy is to adopt an average-pooling layer. We set stride and receptive field size to the scale of resolution reduction.

3.4. Multi-dimensional attention connection

We note that some attention blocks focus on more than one attention dimension, like CBAM [11]. Inspired by MobileNet [21], we design attention connections for one attention dimension at a time. To build a multi-dimensional attention block, we connect attention maps along with each dimension and assure connections in different dimensions are independent of one another (as shown in Fig. 2). This decoupling of attention connections brings two advantages: 1) it reduces the number of parameters and computational overhead; 2) each dimension can focus on its intrinsic property.

4. EXPERIMENTS

We evaluate DCANet for image recognition and object detection. Experimental results on ImageNet [6] and MS-COCO [7] benchmarks demonstrate the effectiveness.

4.1. Classification on ImageNet

We apply our DCANet to a number of state-of-the-art attention blocks, including SE [1], SK [12], GE [19], and CBAM [11]. We train all models on the ImageNet 2012 training set and measure the single-crop (224×224 pixels) top-1 and top-5 accuracy on the validation set. We train models for 100 epochs on 8 Tesla V100 GPUs with 32 images per GPU (the batch size is 256). All models are trained using synchronous SGD with Nesterov momentum of 0.9 and a weight decay of 0.0001. The learning rate is set to 0.1 initially and lowered by a factor of 10 every 30 epochs. For lightweight models like MnasNet and MobileNetV2, we take cosine decay method [22] to adjust the learning rate and train the models for 150 epochs with 64 images per GPU.

Table 1 presents the results on the validation set. We observed that integrating the DCA module improves the classification accuracy in all cases when compared to the vanilla attention models. Of note is that we are comparing with corresponding attention networks, which is stronger than the base networks. Among the tested networks, DCA-CBAM-ResNet50 improves the top-1 accuracy by 0.51% compared with CBAM-ResNet50, and DCA-SE-MobileNetV2 improves the top-1 accuracy by 1.19% compared with SE-MobileNetV2, but the computation overhead is comparable.

4.2. Ablation Evaluation

Connection Schema. As shown in Table 2a, all three connection schemas outperform vanilla SE-ResNet50. This indicates the improvements come from the connections between attention blocks, rather than particular connection schema. Besides, only minimal differences are observed in the top-1 and

Table 2: Ablation studies on the ImageNet 2012 validation set.

(a) DCA connection schemas.

Model	Top-1	Top-5	GFLOPs	Params
SE	77.29	93.65	4.13	28.09M
+Direct	**77.55**	**93.77**	4.13	28.65M
+Softmax	77.52	93.71	4.13	28.65M
+Weighted	77.49	93.69	4.13	28.65M

(b) Multiple Attention dimensions.

Model	Top-1	Top-5	GFLOPs	Params
CBAM	77.28	93.60	4.14	28.09M
+DCA-C	77.79	93.71	4.14	30.90M
+DCA-S	77.58	**93.80**	4.14	28.09M
+DCA-All	**77.83**	93.72	4.14	30.90M

(c) Spatial size matching.

Model	Top-1	Top-5	GFLOPs	Params
CBAM	77.28	93.60	4.14	28.09M
Max Pooling	77.43	93.77	4.14	28.09M
Avg Pooling	**77.58**	**93.80**	4.14	28.09M

(d) Channel matching based on SE-ResNet50.

Model	Top-1	Top-5	GFLOPs	Params
SE	77.29	93.66	4.13	28.09M
1 FC	**77.64**	93.74	4.13	30.90M
2 FC (r=16)	77.55	**93.77**	4.13	**28.65M**
2 FC (r=8)	77.50	93.72	4.13	29.87M
2 FC (r=4)	77.42	93.75	4.13	32.31M

top-5 accuracy of these three connection schemas (77.55% *vs.* 77.52% *vs.* 77.49%). By default we use direct connection which eases the implementation compared to others.

Size matching. For matching the number of channels, we use SE-ResNet50 for illustration due to its pure concerns on channel dependencies. Table 2d presents the results. Directly applying one FC layer can achieve the best top-1 accuracy, while, on the other hand, setting reduction rate r to 16 in two FC layers can reduce the number of parameters and achieve a comparable result. For spatial resolution, we adopt average pooling to reduce the resolution. We also compare with max pooling and present the results in Table 2c. The performance of max pooling is slightly inferior compared to the performance of average pooling, indicating that all attention information should be passed to the succeeding attention blocks.

Multiple Attention dimensions. For illustration, we use CBAM-ResNet50 as a baseline. We use DCA-C/DCA-S to present applying DCANet on channel/spatial attention, and DCA-All indicates we apply DCA module on both attention dimensions for CBAM-ResNet50. Table 2b shows the results of DCANet applied on two dimensions. From the table, we notice that applying DCANet on either dimension will certainly improve the accuracy. When enhancing both attention dimensions, we achieve a 0.54% improvement. When we work on spatial and channel dimensions separately, the improvement is 0.51% and 0.29%, respectively.

4.3. Object Detection on MS COCO

We further evaluate the performance of DCANet for object detection. We measure the average precision of bounding box detection on the challenging COCO 2017 dataset [7]. We adopt the settings used in [23] and train all models with a total of 16 images per mini-batch (2 images per GPU). We employ two state-of-the-art detectors: RetinaNet [23] and Cascade R-CNN [24] as the detectors, with SE-ResNet50, GC-ResNet50 and their DCANet variants as the corresponding backbone respectively. All backbones are pre-trained using ImageNet and are directly taken from Table 1. The detection models are trained for 24 epochs using synchronized SGD with a weight decay of 0.0001 and a momentum of 0.9. The results are reported in Table 3. Although DCANet introduces almost no additional calculations, we observe that DCANet achieves the best performance for all IoU threshold values and most object scales (DCA-SE-ResNet50 obtains +1.5% $AP_{50:95}$ on ResNet50 and +0.3% $AP_{50:95}$ on SE-ResNet50 in RetinaNet; DCA-GC-ResNet50 obtains +0.8% $AP_{50:95}$ on ResNet50 and 0.3% $AP_{50:95}$ on GC-ResNet50 in Cascade R-CNN).

5. CONCLUSION

In this paper, we aim to address a critical issue, that is the capacity of self-attention mechanism is not fully exploited. To achieve a higher utilization, we present Deep Connection Attention Network, which adaptively propagates information among attention blocks via attention connections. We have demonstrated that DCANet consistently improves various attention designs and base CNN architectures on the ImageNet benchmark with a minimal computational overhead. Moreover, experimental results on the MS-COCO dataset show that DCANet generalizes well for other vision tasks, such as object detection. The novel design and feed-forward approach make DCANet easy to be integrated with various attention designs using the mainstream frameworks.

6. ACKNOWLEDGMENT

This work has been supported in part by the National Science Foundation grants CNS-1852134, OAC-2017564, ECCS-2010332, CNS-2037982, and CNS-1563750.

Table 3: Detection performances (%) with different backbones on the MS-COCO validation dataset.

Detector	Backbone	$AP_{50:95}$	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
RetinaNet [23]	ResNet50	36.2	55.9	38.5	19.4	39.8	48.3
	+ SE	37.4	57.8	39.8	20.6	40.8	50.3
	+ DCA-SE	**37.7**	**58.2**	**40.1**	**20.8**	**40.9**	**50.4**
Cascade R-CNN [24]	ResNet50	40.6	58.9	44.2	22.4	43.7	**54.7**
	+ GC	41.1	59.7	44.6	**23.6**	44.1	54.3
	+ DCA-GC	**41.4**	**60.2**	**44.7**	22.8	**45.0**	54.2

7. REFERENCES

[1] Jie Hu, Li Shen, and Gang Sun, "Squeeze-and-Excitation networks," in *CVPR*, 2018.

[2] 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 *ICCV*, 2017.

[3] Sabine Kastner Ungerleider and Leslie G, "Mechanisms of visual attention in the human cortex," *Annual Review of Neuroscience*, 2000.

[4] Leonardo Chelazzi, John Duncan, Earl K Miller, and Robert Desimone, "Responses of neurons in inferior temporal cortex during memory-guided visual search," *Journal of Neurophysiology*, 1998.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[6] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al., "Imagenet large scale visual recognition challenge," *IJCV*, 2015.

[7] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick, "Microsoft COCO: Common objects in context," in *ECCV*, 2014.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *NeurIPS*, 2017.

[9] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He, "Non-local neural networks," in *CVPR*, 2018.

[10] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu, "GCNet: Non-local networks meet Squeeze-Excitation networks and beyond," *arXiv preprint arXiv:1904.11492*, 2019.

[11] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon, "CBAM: Convolutional block attention module," in *ECCV*, 2018.

[12] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang, "Selective kernel networks," in *CVPR*, 2019.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Identity mappings in deep residual networks," in *ECCV*, 2016.

[14] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger, "Densely connected convolutional networks," in *CVPR*, 2017.

[15] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang, "Residual attention network for image classification," in *CVPR*, 2017.

[16] Jianlong Fu, Heliang Zheng, and Tao Mei, "Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition," in *CVPR*, 2017.

[17] Yuanqiang Cai, Dawei Du, Libo Zhang, Longyin Wen, Weiqiang Wang, Yanjun Wu, and Siwei Lyu, "Guided attention network for object detection and counting on drones," *arXiv preprintarXiv:1909.11307*, 2019.

[18] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *CVPR*, 2018.

[19] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Andrea Vedaldi, "Gather-Excite: Exploiting feature context in convolutional neural networks," in *NeurIPS*, 2018.

[20] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le, "MnasNet: Platform-aware neural architecture search for mobile," in *CVPR*, 2019.

[21] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

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

[23] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár, "Focal loss for dense object detection," in *ICCV*, 2017.

[24] Zhaowei Cai and Nuno Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *CVPR*, 2018.