Bi-volution: A Static and Dynamic Coupled Filter

Xiwei Hu,[1*] **Xuanhong Chen,**[1*] **Bingbing Ni,**[1†] **Teng Li,**[2] **Yutian Liu**[1]

[1] Shanghai Jiao Tong University
[2] Anhui University
{huxiwei, chen19910528, nibingbing, stau7001}@sjtu.edu.cn, tenglwy@gmail.com

Abstract

Dynamic convolution has achieved significant gain in performance and computational complexity, thanks to its powerful representation capability given limited filter number/layers. However, SOTA dynamic convolution operators are sensitive to input noises (*e.g.*, Gaussian noise, shot noise, *e.t.c.*) and lack sufficient spatial contextual information in filter generation. To alleviate this inherent weakness, we propose a lightweight and heterogeneous-structure (*i.e.*, static and dynamic) operator, named *Bi-volution*. On the one hand, Bi-volution is designed as a dual-branch structure to fully leverage complementary properties of static/dynamic convolution, which endows Bi-volution more robust properties and higher performance. On the other hand, the Spatial Augmented Kernel Generation module is proposed to improve the dynamic convolution, realizing the learning of spatial context information with negligible additional computational complexity. Extensive experiments illustrate that the ResNet-50 equipped with Bi-volution achieves a highly competitive boost in performance ($+2.8\%$ top-1 accuracy on ImageNet classification, $+2.4\%$ box AP and $+2.2\%$ mask AP on COCO detection and instance segmentation) while maintaining extremely low FLOPs (*i.e.*, ResNet50@2.7 GFLOPs). Furthermore, our Bi-volution shows better robustness than dynamic convolution against various noise and input corruptions. Our code is available at https://github.com/neuralchen/Bivolution.

Introduction

Dynamic convolution is a rapidly developing alternative computing primitive for Convolutional Neural Networks (CNNs) (Han et al. 2021), which achieves great success in various tasks, *e.g.*, image classification (Deng et al. 2009), object detection (Lin et al. 2014), and super-resolution (Xu et al. 2020) *e.t.c.*. Contrast to convolution, dynamic convolution exhibits spatial-anisotropy and content-adaptive properties, resulting in better performance and optimal feature learning. However, these properties also weaken the robustness of dynamic convolution and make it sensitive to input noises (*e.g.*, Gaussian noise, shot noise, *e.t.c.*). Furthermore, subject to rigorous complexity constraints, the squeeze-and-excitation module (Hu, Shen, and Sun 2018) (*i.e.*, consisting of 1×1 convolutions) becomes the de facto standard for dynamic convolution kernel generation, which excludes the interaction of spatially contextual information thus leads to sub-optimal estimation results.

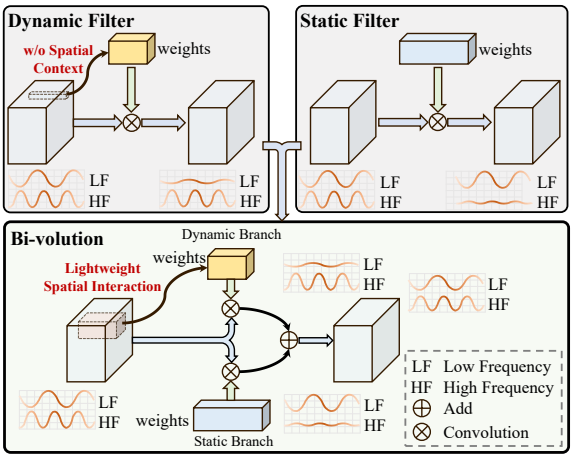

Figure 1: Static and dynamic coupled filter which unifies low-frequency/high-frequency semantic and spatial/channel-wise fusion to boost the accuracy performance and further improve model robustness.

Sensitivity to Noise Pattern. Spatial-anisotropy and content-adaptiveness are the prominent properties of a standard dynamic convolution. In detail, different filters are applied to each point in an image/feature, and at the same time, these filters are estimated online based on the input content. Such a dynamically adaptive design endows dynamic convolution the ability to deal with complex and changeable input images/features, and enables dynamic convolution to achieve better performance than static convolution. Unfortunately, it is these dynamic properties that make dynamic convolution more sensitive to noisy input. Specifically, content adaptability makes dynamic convolution misjudge the noise mixed in the input as part of the input content, resulting in the noisy generation of kernels that are mismatched with the real texture pattern. In addition, our experimental observations have found that dynamic convolution is very sensitive to various types of noise such as Gaussian noise, shot noise, impulse noise and speckle noise. In practical deployment scenarios (*e.g.*, video surveillance, industrial monitor-

[*] Equal contribution.
[†] Corresponding author: Bingbing Ni.

ing, *e.t.c.*), the input data inevitably contains various noise due to the complex data collection process. Unrobustness to input noise will lead to deployment difficulties and severely limit the application range of the model.

Inefficient Filters Generation. The most important component of dynamic convolution is the kernel estimation module, which predicts the corresponding convolution kernels based on the input content/features. As a fundamental building cell of CNNs, dynamic convolution should maintain a reasonable complexity, which requires the kernel estimation module to be extremely lightweight, *e.g.*, the squeeze-and-excitation module, *e.t.c.*. These modules (Hu, Shen, and Sun 2018) are usually composed of stacked 1×1 convolutional layers, however, such a topology fails to establish an effective and direct spatial context interaction. The spatial context interaction is crucial for the network/kernel to deal with complex textures such as corners and edges. Specifically, the 1×1 convolution only focuses on the center point of sliding windows, which prevents the module from effectively perceiving different texture patterns, resulting in generating pattern mismatched kernels. Increasing the receptive field is the easiest way to solve inefficient interaction, but the naive expansion will lead to serious computational complexity and parameter explosion.

In this paper, we propose a lightweight and heterogeneous-structure operator, named Bi-volution, which simultaneously addresses both the above weaknesses of the dynamic convolution operator. Similar to bilateral filters (Tomasi and Manduchi 1998) and guided filters (He, Sun, and Tang 2010), content-adaptive properties make dynamic convolution powerful in processing high-frequency information (*i.e.*, edges, corners, *e.t.c.*). In contrast, static convolution is essentially a low-pass filter (*i.e.*, Wx), which focuses on processing low-frequency information. Through experiments, we found that noise (*e.g.*, Gaussian noise, shot noise) mainly destroys the high-frequency components of the input image, leading to severe model performance degradation (Wang et al. 2020a). The key to improving input noise robustness is to process low-frequency components more effectively. Interestingly, the convolutions aggregated structure (*i.e.*, ResNeXt (Xie et al. 2017)) exhibits excellent noise robustness, indicating that convolution aggregation is beneficial to processing low-frequency signals. Based on this understanding, we design Bi-volution as a dual-branch topology to make full use of complementary properties (*i.e.*, low-frequency and high-frequency) of static/dynamic convolution, which endows our Bi-volution with full-band processing capabilities and thus leads to excellent input noise robustness. More importantly, such a dual-branch structure possesses better optimization properties (*i.e.*, multi-branch alleviates the gradient problems (He et al. 2016; Ding et al. 2021)), resulting in better performance than a single dynamic convolution. Furthermore, we propose the Spatial Augmented Kernel Generation module to improve the dynamic convolution, which enables the interaction of spatial context information with negligible additional computational complexity. Inspired by literature of lightweight static convolution (Chollet 2017; Howard et al. 2017; Sandler et al. 2018; Howard et al. 2019) and efficient channel interaction methods (Hu, Shen, and Sun 2018; Wang et al. 2020c), we design the module with spatial context extractor branch and channel information aggregation branch, which can be applied in any dynamic convolution kernel generation to boost the model performance with limited computational costs. With these two unique designs, our Bi-volution gains highly competitive performance and excellent input noise robustness, while maintaining extremely low complexity, and is ready to be embedded into the mainstream backbone as the computational primitive.

We experiment with the proposed Bi-volution in terms of qualitative and quantitative evaluations on mainstream vision tasks. Extensive results demonstrate that our Bi-volution operator achieves highly competitive improvements (*i.e.*, image classification on ImageNet (Deng et al. 2009): $+2.8\%$ top1 accuracy, object detection on COCO (Lin et al. 2014): $+2.4\%$ box AP, instance segmentation on COCO: $+2.2\%$ mask AP) while maintaining an extremely low FLOPs (*i.e.*, ResNet50@2.7GFLOPs). Furthermore, our Bi-volution shows better input corruption robustness than dynamic convolution in a wide range of noise and other corruptions.

Related Works

Static convolution is a standard operator of modern neural networks. Recent years witness an astonishing variety of deep CNN architectures with diverse aggregation methods which achieve impressive performance in various tasks(Deng et al. 2009; He et al. 2016; Xie et al. 2017; Szegedy et al. 2017). However, the convolution kernels of static models are fixed once trained, limiting their model complexity and representation power (Graves 2016; Huang et al. 2017; Yang et al. 2019).

In contrast to the static ones, dynamic convolution adapts the filters to the input feature, boosting the representation power and thus the performance of CNNs. One kind of dynamic convolution is designed to adjust the sampling grid of convolution kernel (Dai et al. 2017; Jeon and Kim 2017; Zhu et al. 2019) while another kind of approach generates directly filter values with regard to input features (Ha, Dai, and Le 2016; Jia et al. 2016; Yang et al. 2019). Regarding the latter category, some works predict the coefficients of different convolution filters to combine them dynamically (Ma et al. 2020; Yang et al. 2019; Zhang et al. 2020; Chen et al. 2020). However, these methods only add additional computational complexity to static convolution and thus can hardly be applied to large network models. Recently, Li *et al.*propose the involution (Li et al. 2021) whose kernel is entirely predicted from input features. As its operation is in a depth-wise separable convolution manner, the computation complexity is relatively competitive but fails to encode channel-specific information. The Decoupled Dynamic Filter proposed by Zhou *et al.* (Zhou et al. 2021) further design a channel filter branch, adding channel-wise information to dynamic convolution kernels in an efficient way. However, dynamic convolution tends to capture high-frequency components of input image, which leads to robustness reduction facing noise pattern (Wang et al. 2020a).

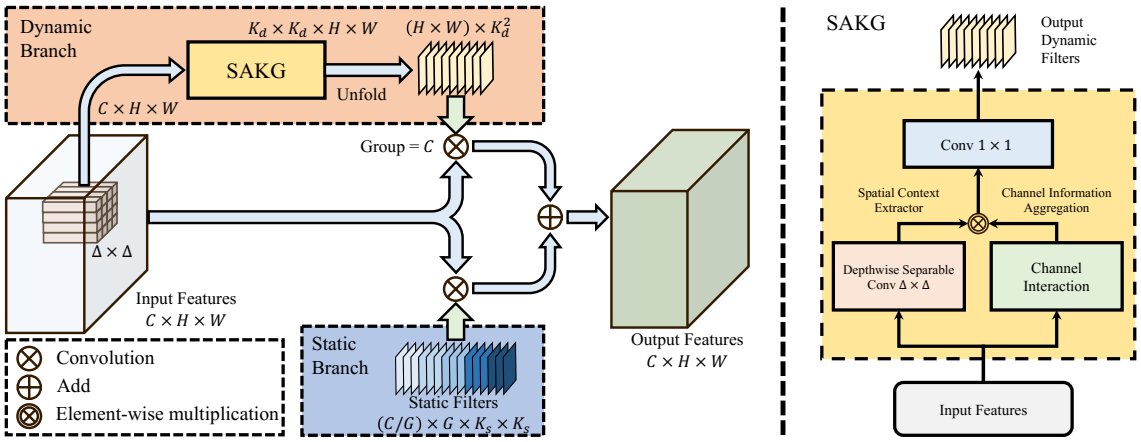

Figure 2: Illustration of our proposed Bi-volution. The input feature is processed by a static branch and a dynamic one. The static branch is responsible for the low-frequency and local features while the dynamic branch equipped with our lightweight Spatial Augmented Kernel Generation (SAKG) module specializes in high-frequency and long-term feature extraction.

Methodology

Preliminaries

Static Convolutions. Given the input feature map $\mathbf{X} \in \mathbb{R}^{H \times W \times C_i}$, where H and W represent the height and width of the feature map and C_i indicates the number of input channels. The standard static convolution is a linear operation with a fixed $K_s \times K_s$ convolution kernel $\mathbf{W^s} \in \mathbb{R}^{C_o \times C_i \times K_s \times K_s}$, where C_o and C_i are the number of output channels and input channels. For the static convolution at certain pixel in input feature, its corresponding output pixel $Y_{i,j,c}$ in the output $\mathbf{Y} \in \mathbb{R}^{H \times W \times C_o}$ can be computed as

$$Y_{i,j,c} = \sum_{k=1}^{c_i} \sum_{(u,v) \in \Omega_{i,j}} \mathbf{W^s}_{c,k,u,v} \mathbf{X}_{i+u-\lfloor K_s/2 \rfloor, j+v-\lfloor K_s/2 \rfloor, k}, \tag{1}$$

where $\Omega_{i,j}$ denotes the set of positions in the $K_s \times K_s$ kernel window, written under Cartesian product as

$$\Omega_{i,j} = [0, 1, \cdots, 2\lfloor K_s/2 \rfloor] \times [0, 1, \cdots, 2\lfloor K_s/2 \rfloor]. \tag{2}$$

The standard static convolution aggregates local spatial information and channel-wise information. It is essentially a low-pass filter, which estimates the result by weighting the information in the average window.

Dynamic Convolutions. As opposed to static convolution, dynamic convolution predicts the kernel weight from the input features. Previous works (Ma et al. 2020; Yang et al. 2019; Zhang et al. 2020; Chen et al. 2020) generate the filters by predicting coefficients of several expert static convolution kernels, which still works in a spatially shared convolution manner and adds additional computation burden to existing static convolution. Recent works (Li et al. 2021; Zhou et al. 2021) propose to generate kernels whose values are spatially adapted to the input features and are channel-wise shared to reduce the computation consumption. The output can be expressed as

$$Y_{i,j,c} = \sum_{(u,v) \in \Omega_{i,j}} \mathbf{W}^d_{u,v} \mathbf{X}_{i+u-\lfloor K_d/2 \rfloor, j+v-\lfloor K_d/2 \rfloor, c}. \tag{3}$$

To generate the dynamic convolution kernel $\mathbf{W}^d \in \mathbb{R}^{H \times W \times K_d \times K_d}$, where H and W denote the height and width of input features, K_d denotes the kernel size, they apply a mapping function f as

$$\mathbf{W}^d_{u,v} = f(\mathbf{X}_{\Psi_{u,v}}), \tag{4}$$

where $\Psi_{u,v}$ is the set of indexes of pixels $\mathbf{W}^d_{u,v}$ is conditioned on. The mapping function f consists of standard convolution layers with kernel size 1×1 in consideration of computational costs. As a trade-off, the kernel generation process will miss the spatial information from input features. In addition, channel-wise shared property makes dynamic convolution fail to aggregate channel-specific information. To address the latter issue, Zhou *et al.* (Zhou et al. 2021) propose to encode channel-specific information by an additional channel branch via a squeeze-and-excitation structure. However, the former problem of the absence of spatial information still remains. In addition, similar to bilateral filters (Tomasi and Manduchi 1998) and guided filters (He, Sun, and Tang 2010), content-adaptive properties make dynamic convolution powerful in processing high-frequency information to achieve better accuracy. Simultaneously, common input noise and corruption usually destroy high-frequency components, reducing the noise robustness of dynamic convolution.

The Hybrid Filter: Bi-volution

Combining with characteristics of static convolution, we propose an augmented dynamic convolution operator named Bi-volution. Specifically, we design, on the one hand, a dual-branch structure to complement the dynamic convolution with robust static convolution branch. On the other hand, we propose a Spatial Augmented Kernel Generation (SAKG) module which aggregates the spatial context information in input features in an efficient way.

Dual-branch Structure. Given the input feature $\mathbf{X}$ and the kernel generation function f, the dynamic convolution

result $\mathbf{Y}$ can be written as

$$\mathbf{Y} = f(\mathbf{X}) * \mathbf{X}, \tag{5}$$

where $*$ indicates the convolution operation. As an input-conditional execution, dynamic convolution extracts the second-order information from input features, exploiting higher-order statistics. It is demonstrated that high-order feature statistics induce more discriminative representations, which improves the model performance in large-scale classification and other computer vision tasks (Li et al. 2017; Basri et al. 2020; Tancik et al. 2020; Wang et al. 2020a). From the perspective of *spectral bias* (Rahaman et al. 2019; Basri et al. 2020), networks with static convolution tend to learn lower frequencies, which leads to insufficient performance. However, it is well-known that the high-order operators are more sensitive to perturbation while the first-order operators (*i.e.*, static convolution) are not. Better high-frequency processing capabilities result in better accuracy and worse noise robustness (Wang et al. 2020a). In short, the dynamic convolution and the static convolution can generate representations of different levels which are complementary to each other. It is obviously insufficient to focus only on one band. Aggregating different features will significantly improve the network robustness (Hendrycks and Dietterich 2019) and optimization properties (Ding et al. 2021; Chen, Wang, and Ni 2021). Inspired by the above issue, we propose the dual-branch structure fusing the dynamic and static convolution to hybrid the respective advantages.

Given the static convolution function g, our dual-branch structure with first-order and second-order information can be expressed as

$$\mathbf{Y}' = f(\mathbf{X}) * \mathbf{X} + g(\mathbf{X}). \tag{6}$$

Note that the batch normalization layer and nonlinearity are placed outside our dual branch. We apply group convolution and small kernel size (*i.e.*, 1×1 or 3×3) to better reduce the computation. The overall structure is shown in Fig. 2. From the perspective of signal analysis, the first-order static convolution is robust to noise and the second-order dynamic convolution is sensitive to fine changes between different input features. Accordingly, the proposed structure is able to balance model robustness and accuracy. In terms of feature aggregation, we separate the static and dynamic convolution in two branches, which encourages them to extract features at different levels from input. Our hybrid method aggregates the representations, fusing the complementary information from the results of two kinds of convolution operations. By increasing the feature aggregation, our proposed structure endows stronger representation power, benefiting both accuracy performance and noise robustness.

Spatial Augmented Kernel Generation. As discussed above, previous state-of-the-art works (Li et al. 2021; Zhou et al. 2021) generate dynamic convolution kernels using 1×1 convolution to reduce their model complexity while giving up the spatial context in input features. To tackle this problem, we carefully design the Spatial Augmented Kernel Generation (SAKG) module which has a larger receptive field with limited additional computational complexity. Inspired by lightweight static convolution literature (Chollet 2017; Howard et al. 2017; Sandler et al. 2018; Howard et al. 2019), we effectively expand the receptive field with little computation complexity by replacing the standard convolution with the depth-wise separable convolution. However, such simple replacement usually results in a significant degradation in performance due to the absence of effective channel interaction. Therefore, we consider to equip the depth-wise separable convolution with a lightweight squeeze-and-excitation efficient channel attention (Wang et al. 2020c) to complement the channel interaction information. As shown in Fig. 2, our SAKG module is designed with a spatial context extractor branch and a channel information aggregation branch. The spatial context extractor branch is a simple $\Delta \times \Delta$ depth-wise separable convolution. The channel information aggregation branch consists of a global average pooling following by a convolution mapping layer.

To specify the time complexity of previous dynamic convolution generation methods and our SAKG module, we assume that the kernel in previous works is generated by a single 1×1 convolution. Given the input feature $\mathbf{X} \in \mathbb{R}^{h \times w \times c}$, generating a kernel of size $K_d \times K_d$ through a 1×1 convolution takes $hwcK_d^2$ FLOPs. If adding our SAKG module, it will take additional $\Delta^2 hwc$ FLOPs for spatial context extractor and $hwc + c^2$ for channel information aggregation, which is in total $((\Delta^2 + 1)hw + c)c$ FLOPs with time complexity of $O(hwc + c^2)$. Since $hw >> c$, time complexity of our proposed module is approximately equals to $O(hwc)$, which only adds constant FLOPs to the original kernel generation computation complexity of $O(hwcK_d^2)$.

Experiments

To evaluate our proposed method, we implement *BiNet* equipped with Bi-volution with three series of experiments. **(a)** Basic experiments on ImageNet classification (Deng et al. 2009), COCO object detection and instance segmentation (Lin et al. 2014). **(b)** Robustness experiments with input corruptions to verify the robustness of different convolution structure. **(c)** Ablation study analyzing the effectiveness of several components in the Bi-volution.

Main Results

Image Classification. ImageNet (Deng et al. 2009) dataset is considered as one of the most challenging object recognition datasets in computer vision, consisting of 1.28M training images and 50K validation images of 1000 different classes. We follow the Inception-style data augmentation (Szegedy et al. 2015). Specifically, input images are randomly cropped to 224×224 with horizontal flipping. For fair comparisons, we compare our Bi-volution network with static convolution networks, dynamic convolution networks and their variants within similar **model scale** (*i.e.*, number of parameters and FLOPs) and **training schedule** (*i.e.*, data augmentation and training epochs), and report the top-1 accuracy on the validation set. We embed our Bi-volution in ResNet (He et al. 2016) backbones to demonstrate its effectiveness. Concretely, the Bi-volution in our BiNet is de-

Architecture	Params	FLOPs	Top-1 Acc.
ResNet-38 (2016)	19.6M	3.2G	76.0
Stand-Alone ResNet-38 (2019)	14.1M	3.0G	76.9
SAN15 (2020)	14.1M	3.0G	77.1
RedNet-38 (2021)	12.4M	2.2G	77.3
BiNet-38 (dc)	13.2M	2.2G	**78.2**
DDF-ResNet-38 (2021)	13.1M	1.9G	78.3
BiNet-38 (dc-ca)	14.8M	2.1G	**78.7**
ResNet-50 (2016)	25.6M	4.1G	76.8
ResNeXt-50 (32×4d) (2017)	25.0M	4.3G	77.8
SE-ResNet-50 (2018)	28.1M	4.1G	77.6
Res2Net-50 (14w-8s) (2019)	25.7M	4.2G	78.0
LR-Net-50 (2019)	23.3M	4.3G	77.3
AA-ResNet-50 (2019)	25.8M	4.2G	77.7
Stand-Alone ResNet-50 (2019)	18.0M	3.6G	77.6
SAN19 (2020)	17.6M	3.8G	77.4
ECA-ResNet-50 (2020c)	25.6M	4.1G	77.5
Axial ResNet-S (2020b)	12.5M	3.3G	78.1
Fca-ResNet-50 (2020)	28.1M	4.1G	78.5
RedNet-50 (2021)	15.5M	2.7G	78.1
BiNet-50 (dc)	17.6M	2.8G	**78.8**
DDF-ResNet-50 (2021)	16.8M	2.3G	79.1
BiNet-50 (dc-ca)	19.3M	2.7G	**79.6**
ResNet-101 (2016)	44.6M	7.9G	78.5
ResNeXt-101 (32×4d) (2017)	44.2M	8.0G	78.8
SENet ResNet-101 (2018)	49.3M	7.9G	77.6
BAM-ResNet-101 (2018)	49.3M	7.8G	78.3
CBAM-ResNet-101 (2018)	49.3M	7.8G	78.5
LR-Net-101 (2019)	42.0M	8.0G	78.5
AA-ResNet-101 (2019)	45.4M	8.1G	78.7
Res2Net-101 (26w-4s) (2019)	45.2M	8.1G	79.2
ECANet ResNet-101 (2020c)	44.6M	7.9G	78.7
FcaNet ResNet-101 (2020)	49.3M	7.9G	79.6
RedNet-101 (2021)	25.6M	4.7G	78.8
BiNet-101 (dc)	29.5M	4.9G	**79.7**
DDF-ResNet-101 (2021)	28.1M	4.1G	80.2
BiNet-101 (dc-ca)	33.0M	4.7G	**80.7**

Table 1: The architecture profiles on ImageNet validation set. We test with 224×224 crop size. We compare with improved re-implementations if available and extract the other results from their original publications.

signed as two types, one based on single dynamic convolution (Li et al. 2021), named *BiNet (dc)*, and the other one based on the combination of dynamic convolution and channel attention (Zhou et al. 2021), named *BiNet (dc-ca)*. We train our models using the same recipe as (Li et al. 2021) and (Zhou et al. 2021) with SGD optimizer (the momentum of 0.9 and the weight decay of 1×10^{-4}). The initial learning rate is set as 0.1 per batch size 256 and decays to 1×10^{-5} following the cosine schedule for 130 epochs in total. The detailed network architecture and training setup can be found in the supplementary material. We use 8 NVIDIA Tesla V100 GPUs for training. Our model is implemented with the PyTorch (Paszke et al. 2019) framework, and the source code will be released for reproducibility.

We compare BiNet with the state-of-the-art variants of ResNet-38, ResNet-50 and ResNet-101 (He et al. 2016), including static/dynamic-convolution-based models, as shown in Table 1. Specifically, our BiNet effectively outperforms other models at similar model size. At the tiny level, BiNet-38(dc-ca) obtains a boost of 2.7% higher accuracy over ResNet-38 with 34.4% lower FLOPs. With ResNet-50 backbone, BiNet attains a compelling 79.6% top-1 accuracy, which is 0.5% higher than previous SOTA dynamic convolution network combined with channel attention. Within reasonable cost of model size, our proposed method boosts the performance of dynamic convolution in an efficient way.

Object Detection and Instance Segmentation. Beyond the classification task, we further evaluate our proposed operator on object detection and instance segmentation to exploit its versatility. For object detection, we adopt the representative detector Faster R-CNN (Ren et al. 2015) with FPN (Lin et al. 2017) as our base architecture. For instance segmentation, we employ Mask R-CNN (He et al. 2017) framework with FPN neck. We finetune these two models with ResNet-50 backbone on the COCO 2017 (Lin et al. 2014) dataset containing 115K training images. To evaluate our method, we test on 5K validation images in COCO 2017 and report the standard COCO metrics (Lin et al. 2014) for mean Average Precision (mAP). Our models adopt the same training protocol as (Li et al. 2021). Concretely, all models are trained for 12 epochs using SGD optimizer with the momentum of 0.9 and weight decay of 1×10^{-5}. The learning rate initiates from 0.02 and decays by 0.1 at 8^{th} and 11^{st} epoch. All detectors are trained with a total batch size 16 on 8 Tesla V100 GPUs with 2 samples per GPU. More training details can be found in supplementary materials.

In table 2, we compare our models against ResNet-50 (He et al. 2016) with static convolution and RedNet-50 (Li et al. 2021) with dynamic one. It is observed that with our BiNet backbone, both Faster R-CNN and Mask R-CNN yield an impressive improvement, *i.e.*, $+2.3\%$, $+2.2\%$ higher over ResNet-50 and $+0.5\%$, $+0.4\%$ higher over RedNet-50 in bounding box AP. We replace the convolution in the FPN neck and task specific heads of Faster R-CNN to build fully Bi-volution-based detectors. In this case, we observe further performance gains, *i.e.*, $+2.4\%$ over ResNet-50 and $+0.9\%$ over fully involution-based detector.

Robustness Analysis

Setup. In real-world applications of vision system, the human vision system is robust to images corrupted in various ways while computer vision system can be easily fooled by subtle changes in query images (Madry et al. 2017; Azulay and Weiss 2018). Achieving robustness to common corruption is an important goal for computer vision and it is also essential in safety-critical applications. Hendrycks and Dietterich (Hendrycks and Dietterich 2019) create a dataset named ImageNet-C which introduces 75 common visual corruptions and applies them to validation images of ImageNet. These corruptions fall into four main categories, noise, blur, weather and digital, and each corruption type consists of five levels severity. Especially, the noise corruption includes Gaussian, shot, impulse and speckle noise.

To compare the robustness of various classifier structure and our proposed dual-branch structure, we choose static convolution network ResNet-50 (He et al. 2016), static

Detector	Backbone	Neck & Head	AP^{bbox}	AP^{bbox}_{50}	AP^{bbox}_{75}	AP^{bbox}_{S}	AP^{bbox}_{M}	AP^{bbox}_{L}
Faster R-CNN	ResNet-50	convolution	37.7	58.7	40.8	21.7	41.6	48.4
	RedNet-50	convolution	$39.5_{(+1.8)}$	$60.9_{(+2.2)}$	$42.8_{(+2.0)}$	$23.3_{(+1.6)}$	$42.9_{(+1.3)}$	$52.2_{(+\mathbf{3.8})}$
	BiNet-50 (dc)	convolution	$40.0_{(+\mathbf{2.3})}$	$61.4_{(+\mathbf{2.7})}$	$43.4_{(+\mathbf{2.6})}$	$23.5_{(+\mathbf{1.8})}$	$43.5_{(+\mathbf{1.9})}$	$52.1_{(+3.7)}$
	RedNet-50	involution	$39.2_{(+1.5)}$	$61.0_{(+2.3)}$	$42.4_{(+1.6)}$	$23.1_{(+1.4)}$	$43.0_{(+1.4)}$	$50.7_{(+2.3)}$
	BiNet-50 (dc)	Bi-volution	$40.1_{(+\mathbf{2.4})}$	$61.8_{(+\mathbf{3.1})}$	$43.8_{(+\mathbf{3.0})}$	$23.4_{(+\mathbf{1.7})}$	$43.9_{(+\mathbf{2.3})}$	$52.2_{(+\mathbf{3.8})}$

Detector	Backbone	Neck & Head	AP	AP_{50}	AP_{75}	AP_{S}	AP_{M}	AP_{L}
Mask R-CNN	ResNet-50	convolution	38.4 35.1	59.2 56.3	41.9 37.3	21.9 18.5	42.3 38.6	49.7 46.9
	RedNet-50	convolution	$40.2_{(+1.8)}$ $36.1_{(+1.0)}$	$61.4_{(+2.2)}$ $58.1_{(+1.8)}$	$43.7_{(+1.8)}$ $38.2_{(+0.9)}$	$24.2_{(+\mathbf{2.3})}$ $19.9_{(+1.4)}$	$43.4_{(+1.1)}$ $39.3_{(+0.7)}$	$52.5_{(+2.8)}$ $48.9_{(+2.0)}$
	BiNet-50 (dc)	convolution	$40.6_{(+\mathbf{2.2})}$ $36.4_{(+\mathbf{1.3})}$	$61.7_{(+\mathbf{2.5})}$ $58.3_{(+\mathbf{2.0})}$	$44.3_{(+\mathbf{2.4})}$ $38.8_{(+\mathbf{1.5})}$	$23.8_{(+1.9)}$ $20.0_{(+\mathbf{1.5})}$	$43.9_{(+\mathbf{1.6})}$ $39.6_{(+\mathbf{1.0})}$	$53.1_{(+\mathbf{3.4})}$ $49.3_{(+\mathbf{2.4})}$

Table 2: Performance comparison of object detection (based on Faster R-CNN (2015)) and instance segmentation (based on Mask R-CNN (2017)) on COCO validation set 2017. In the parentheses are the gaps to the ResNet-based model.

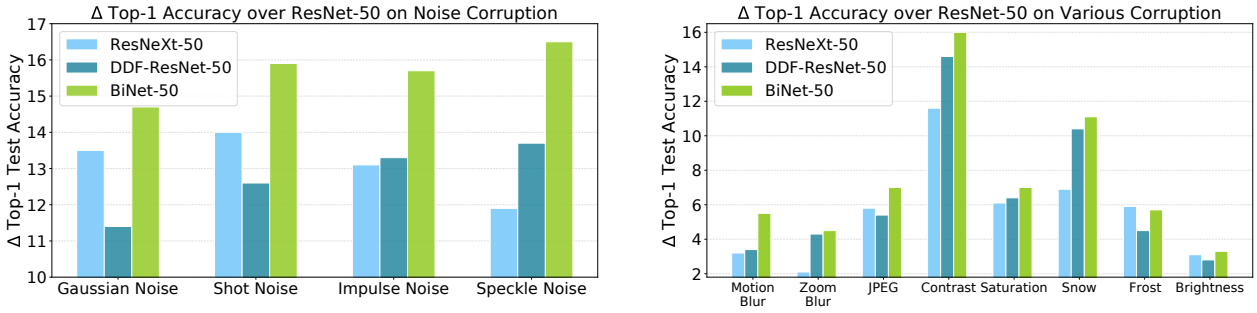

Figure 3: A comparison of the Δ Top-1 accuracy ResNeXt-50, DDF-ResNet-50 and our BiNet-50 over ResNet-50 on ImageNet-C corruptions. Each bar represents an average over five corruption severity for a given corruption category.

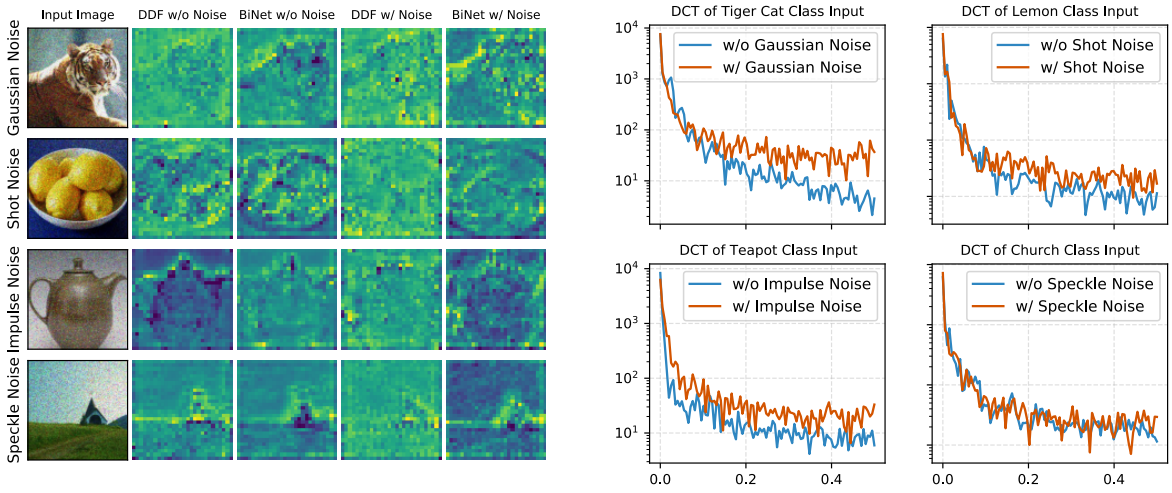

Figure 4: **Left:** Visualization of features extracted by the last block of the third stage of ResNet-50 backbone. The first two columns of the heat maps indicate extracted features of input images without noise and the last two columns correspond to the case that input images are corrupted by different noise. Input images are randomly drawn from ImageNet validation set. **Right:** Discrete Cosine Transform power spectrum of four selected input images on the left with/without noise. The axis of x indicates the frequency and the axis of y represents the corresponding value.

convolution aggregation network ResNeXt-50 (Xie et al. 2017), previous state-of-the-art dynamic convolution network DDF-ResNet-50 (Zhou et al. 2021) and our dual-branch network. All models are trained on ImageNet dataset without any fine-tuning on the corruptions and test on ImageNet-C dataset. We use ResNet-50 as backbone and the same training recipe as mentioned in previous sections. We set ResNet-50 as baseline and the average performance gains of different networks over ResNet-50 on noise corruptions and other diverse corruptions are reported in Fig. 3.

Results and analysis. The clean dataset top-1 accuracy of previous state-of-the-art model based on dynamic convolution achieves 79.1% and the average accuracy over all corruption at all levels of severity reduces to 47.5%. Our proposed static/dynamic dual-branch network attains the accuracy of 79.6% and substantially improves the robustness by 1.2% to 48.7% in average accuracy on corruption. Especially, when facing noise corruption, the dual-branch design significantly improves the performance by 3.0% for 4 types of noise in average accuracy. Although the dynamic convolution network owns an attractive clean accuracy, it has an obvious performance degradation for Gaussian and shot noise compared to static convolution network (*i.e.*, ResNeXt-50). In contrast, our hybrid network maintains high performance over static/dynamic convolution ones.

To have better insight, we visualize features extracted by dynamic convolution and Bi-volution in the last block of the third stage. As shown in Fig. 4, both dynamic convolution and Bi-volution are able to highlight diverse semantic concepts from original input image. However, once the input is corrupted by noise, dynamic convolution fails to distinguish meaningful information under the perturbation. We also inspect frequency distribution change of the input when adding different noise. As seen in Fig. 4, Gaussian and shot noise substantially influence the high-frequency components in input image while preserving the low-frequency parts. As a result, DDF-ResNet-50 has an significant accuracy reduction compared to ResNeXt-50 since the dynamic convolution focuses more on high-frequency features than the static one. For impulse noise, it affects both low/high-frequency parts, resulting in the similar performance degradation of ResNeXt-50 and DDF-ResNet-50. Speckle noise does not change the overall frequency distribution as other noise, thus the DDF-ResNet-50 preserves its accuracy advantage over ResNeXt-50. In contrast, our hybrid Bi-volution maintain its best performance among other models on all noise corruption. The dual-branch operator is capable to fuse both low/high-frequency features, which consequentially increases the representation power. Therefore, our hybrid method brings remarkable robustness to noise corruption along with the compelling clean accuracy improvement. Our method also shows performance gain over other kinds of corruption. For instance, compared to DDF-ResNet-50, our hybrid BiNet-50 boost the performance by 1.1% in average accuracy on blur inputs, 0.8% on weather and 0.5% on digital corruptions. These results demonstrate the consistent improvement of robustness performance when facing a wide variety of corruption.

Ablation Study

We perform ablation experiments to inspect the effect of different components in the Bi-volution. For both BiNet (dc) and BiNet (dc-ca), we employ ResNet-50 backbone with corresponding dynamic convolution design and analyse the effect of dual-branch structure and SAKG module on ImageNet classification accuracy. Table 3 reports the results of different modification of Bi-volution. By adding our proposed two components respectively to existing dynamic convolution, we observe the improvement of top-1 accuracy by up to $+0.4\%$ over previous dynamic base model. Once using the full Bi-volution with both dual-branch structure and SAKG module, the top-1 accuracy is improved by up to $+2.6\%$ over baseline ResNet-50 and $+0.7\%$ over previous network based on dynamic convolution.

Architecture	Dual-branch	SAKG	Top-1 Acc.
BiNet-50 (dc)	*Base Model*		78.1
	✓		78.2
		✓	78.5
	✓	✓	**78.8**
BiNet-50 (dc-ca)	*Base Model*		79.1
	✓		79.3
		✓	79.3
	✓	✓	**79.4**

Table 3: Ablation studies on different components in the design of Bi-volution on ImageNet dataset. We employ ResNet-50 backbone with default experimental settings. The kernel size of SAKG is set as 3×3.

Arch	Kernel Type	Params	FLOPs	Top-1 Acc.
BiNet-50 (dc)	1×1	15.5M	2.7G	78.1
	3×3	17.5M	2.8G	78.5
	3×3, 2	17.5M	2.8G	78.2
	5×5	17.6M	2.8G	78.6
	7×7	17.7M	2.9G	78.6
BiNet-50 (dc-ca)	1×1	16.8M	2.3G	79.1
	3×3	18.1M	2.3G	79.3
	3×3, 2	18.1M	2.3G	79.3
	5×5	18.2M	2.5G	79.4
	7×7	18.2M	2.6G	79.4

Table 4: Ablation studies on the kernel size of SAKG module with ResNet-50 backbone. We report the Top-1 Accuracy on ImageNet dataset with default experimental settings.

To evaluate the influence of the receptive field, we train BiNet-50 (dc)/(dc-ca) with different kernel types in SAKG module. Concretely, we test the kernel size of 3×3, 3×3 with dilation factor 2, 5×5 and 7×7. To better examine the effectiveness of the proposed SAKG module, we do not apply dual-branch structure in this experiment. Table 4 compares the performance of SAKG with different receptive fields. It is observed that the extracted spatial context substantially helps the kernel generation and improves the model performance with a very limited computational costs.

Conclusion

In this paper, we propose a lightweight and heterogeneous-structured Bi-volution to improve the robustness to noise and efficiency of kernel generation of dynamic convolution. With the proposed dual-branch structure, our operator fully leverages the complementary properties of static/dynamic convolution to increase feature aggregation and we found impressive promotion in robustness to input corruptions. In addition, the Spatial Augmented Kernel Generation module brings compelling gains in accuracy performance while maintaining low computational costs. Our Bi-volution demonstrates consistent improvements over noise robustness and accuracy performance with its rich representation power within reasonable costs.

Acknowledgements

This work was supported by National Science Foundation of China (U20B2072, 61976137).

References

Azulay, A.; and Weiss, Y. 2018. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv preprint arXiv:1805.12177*.

Basri, R.; Galun, M.; Geifman, A.; Jacobs, D. W.; Kasten, Y.; and Kritchman, S. 2020. Frequency Bias in Neural Networks for Input of Non-Uniform Density. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, Proceedings of Machine Learning Research, 685–694. PMLR.

Bello, I.; Zoph, B.; Vaswani, A.; Shlens, J.; and Le, Q. V. 2019. Attention augmented convolutional networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3286–3295.

Chen, X.; Wang, H.; and Ni, B. 2021. X-volution: On the unification of convolution and self-attention. *CoRR*, abs/2106.02253.

Chen, Y.; Dai, X.; Liu, M.; Chen, D.; Yuan, L.; and Liu, Z. 2020. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11030–11039.

Chollet, F. 2017. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1251–1258.

Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; and Wei, Y. 2017. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 764–773.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Ding, X.; Zhang, X.; Ma, N.; Han, J.; Ding, G.; and Sun, J. 2021. RepVGG: Making VGG-Style ConvNets Great Again. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, 13733–13742. Computer Vision Foundation / IEEE.

Gao, S.; Cheng, M.-M.; Zhao, K.; Zhang, X.-Y.; Yang, M.-H.; and Torr, P. H. 2019. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*.

Graves, A. 2016. Adaptive computation time for recurrent neural networks. *arXiv preprint arXiv:1603.08983*.

Ha, D.; Dai, A.; and Le, Q. V. 2016. Hypernetworks. *arXiv preprint arXiv:1609.09106*.

Han, Y.; Huang, G.; Song, S.; Yang, L.; Wang, H.; and Wang, Y. 2021. Dynamic neural networks: A survey. *arXiv preprint arXiv:2102.04906*.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

He, K.; Sun, J.; and Tang, X. 2010. Guided Image Filtering. In Daniilidis, K.; Maragos, P.; and Paragios, N., eds., *Computer Vision - ECCV 2010, 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part I*, volume 6311 of *Lecture Notes in Computer Science*, 1–14. Springer.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hendrycks, D.; and Dietterich, T. 2019. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*.

Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. 2019. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1314–1324.

Howard, A. G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; and Adam, H. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Hu, H.; Zhang, Z.; Xie, Z.; and Lin, S. 2019. Local relation networks for image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3464–3473.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132–7141.

Huang, G.; Chen, D.; Li, T.; Wu, F.; van der Maaten, L.; and Weinberger, K. Q. 2017. Multi-scale dense networks for resource efficient image classification. *arXiv preprint arXiv:1703.09844*.

Jeon, Y.; and Kim, J. 2017. Active convolution: Learning the shape of convolution for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4201–4209.

Jia, X.; De Brabandere, B.; Tuytelaars, T.; and Gool, L. V. 2016. Dynamic filter networks. *Advances in neural information processing systems*, 29: 667–675.

Li, D.; Hu, J.; Wang, C.; Li, X.; She, Q.; Zhu, L.; Zhang, T.; and Chen, Q. 2021. Involution: Inverting the inherence of convolution for visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12321–12330.

Li, P.; Xie, J.; Wang, Q.; and Zuo, W. 2017. Is second-order information helpful for large-scale visual recognition? In *Proceedings of the IEEE international conference on computer vision*, 2070–2078.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Ma, N.; Zhang, X.; Huang, J.; and Sun, J. 2020. Weightnet: Revisiting the design space of weight networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*, 776–792. Springer.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Park, J.; Woo, S.; Lee, J.-Y.; and Kweon, I. S. 2018. Bam: Bottleneck attention module. *arXiv preprint arXiv:1807.06514*.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32: 8026–8037.

Qin, Z.; Zhang, P.; Wu, F.; and Li, X. 2020. Fcanet: Frequency channel attention networks. *arXiv preprint arXiv:2012.11879*.

Rahaman, N.; Baratin, A.; Arpit, D.; Draxler, F.; Lin, M.; Hamprecht, F. A.; Bengio, Y.; and Courville, A. C. 2019. On the Spectral Bias of Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, Proceedings of Machine Learning Research, 5301–5310. PMLR.

Ramachandran, P.; Parmar, N.; Vaswani, A.; Bello, I.; Levskaya, A.; and Shlens, J. 2019. Stand-alone self-attention in vision models. *arXiv preprint arXiv:1906.05909*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28: 91–99.

Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; and Chen, L.-C. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4510–4520.

Szegedy, C.; Ioffe, S.; Vanhoucke, V.; and Alemi, A. A. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9.

Tancik, M.; Srinivasan, P. P.; Mildenhall, B.; Fridovich-Keil, S.; Raghavan, N.; Singhal, U.; Ramamoorthi, R.; Barron, J. T.; and Ng, R. 2020. Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Tomasi, C.; and Manduchi, R. 1998. Bilateral Filtering for Gray and Color Images. In *Proceedings of the Sixth International Conference on Computer Vision (ICCV-98), Bombay, India, January 4-7, 1998*, 839–846. IEEE Computer Society.

Wang, H.; Wu, X.; Huang, Z.; and Xing, E. P. 2020a. High-Frequency Component Helps Explain the Generalization of Convolutional Neural Networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 8681–8691. Computer Vision Foundation / IEEE.

Wang, H.; Zhu, Y.; Green, B.; Adam, H.; Yuille, A.; and Chen, L.-C. 2020b. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *European Conference on Computer Vision*, 108–126. Springer.

Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; and Hu, Q. 2020c. ECA-Net: efficient channel attention for deep convolutional neural networks, 2020 IEEE. In *CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE*.

Woo, S.; Park, J.; Lee, J.-Y.; and Kweon, I. S. 2018. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, 3–19.

Xie, S.; Girshick, R. B.; Dollár, P.; Tu, Z.; and He, K. 2017. Aggregated Residual Transformations for Deep Neural Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 5987–5995. IEEE Computer Society.

Xu, Y.; Tseng, S. R.; Tseng, Y.; Kuo, H.; and Tsai, Y. 2020. Unified Dynamic Convolutional Network for Super-Resolution With Variational Degradations. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 12493–12502. Computer Vision Foundation / IEEE.

Yang, B.; Bender, G.; Le, Q. V.; and Ngiam, J. 2019. Condconv: Conditionally parameterized convolutions for efficient inference. *arXiv preprint arXiv:1904.04971*.

Zhang, Y.; Zhang, J.; Wang, Q.; and Zhong, Z. 2020. Dynet: Dynamic convolution for accelerating convolutional neural networks. *arXiv preprint arXiv:2004.10694*.

Zhao, H.; Jia, J.; and Koltun, V. 2020. Exploring self-attention for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10076–10085.

Zhou, J.; Jampani, V.; Pi, Z.; Liu, Q.; and Yang, M.-H. 2021. Decoupled Dynamic Filter Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6647–6656.

Zhu, X.; Hu, H.; Lin, S.; and Dai, J. 2019. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9308–9316.