Dynamic Region-Aware Convolution

Jin Chen,* Xijun Wang,* Zichao Guo, Xiangyu Zhang, Jian Sun

MEGVII Technology

{chenjin, wangxijun, guozichao, zhangxiangyu, sunjian}@megvii.com

Abstract

We propose a new convolution called Dynamic Region-Aware Convolution (DRConv), which can automatically assign multiple filters to corresponding spatial regions where features have similar representation. In this way, DR-Conv outperforms standard convolution in modeling semantic variations. Standard convolutional layer can increase the number of filers to extract more visual elements but results in high computational cost. More gracefully, our DR-Conv transfers the increasing channel-wise filters to spatial dimension with learnable instructor, which not only improve representation ability of convolution, but also maintains computational cost and the translation-invariance as standard convolution dose. DRConv is an effective and elegant method for handling complex and variable spatial information distribution. It can substitute standard convolution in any existing networks for its plug-and-play property, especially to power convolution layers in efficient networks. We evaluate DRConv on a wide range of models (MobileNet series, ShuffleNetV2, etc.) and tasks (Classification, Face Recognition, Detection and Segmentation). On ImageNet classification, DRConv-based ShuffleNetV2-0.5× achieves state-of-the-art performance of 67.1% at 46M multiply-adds level with 6.3% relative improvement.

1. Introduction

Benefiting from powerful representation ability, convolutional neural networks (CNNs) have made significant progress in image classification, face recognition, object detection and many other applications. The powerful representation ability of CNNs stems from that different filters are responsible for extracting information at different level of abstraction. However, current mainstream convolutional operations perform in filter sharing manner across spatial domain, so that more effective information can only be captured when these operations are applied repeatedly (e.g., in-
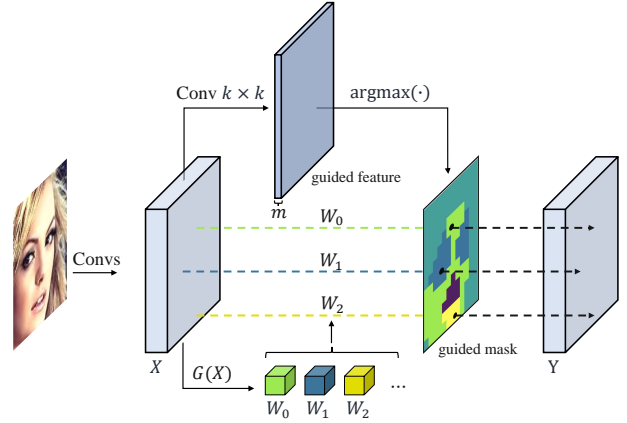
Figure 1. Illustration of DRConv with kernel size $k \times k$ and region number m. We get guided feature from X with standard $k \times k$ convolution and get m filters from filter generator module $G(\cdot)$. The spatial dimension is divided into m regions as guided mask shows. Every region has individual filter W_i which is shared in this region and we execute $k \times k$ convolution with corresponding filter in these regions of X to output Y.

creasing channels and depth with more filters). This repeating manner has several limitations. First, it is computationally inefficient. Second, it causes optimization difficulties that need to be carefully addressed [12, 31].

Different from the filter sharing methods, to model more visual elements, some studies focus on making use of the diversity of semantic information by using multiple filters in spatial dimension. [9, 28] came up with alternative convolutions to possess individual filter at each pixel in spatial dimension, and we collectively call them local convolution for convenience. Therefore, the feature of each position will be treated differently, which is more effective to extract the spatial feature than standard convolution. [28, 26, 25] have shown local convolution's power on face recognition task. Although local convolution doesn't increase the computation complexity compared with standard convolution, it has two fatal drawbacks. One is to bring a large amount of parameters, which is proportional to the

1

spatial size. The other is that local convolution destroys translation-invariance, which is unfriendly to some tasks requiring translation-invariant features (e.g., local convolution doesn't work for classification task). Both of them make it hard to be widely used in neural networks. Besides, local convolution still shares filters across different samples, which makes it insensitive to the specific feature of each sample. For example, there are samples with different poses or viewpoints on face recognition and object detection tasks. Therefore, shared filters across different samples can not be effective to extract customized features.

Considering above mentioned limitations, in this paper, we put forward a new convolution named Dynamic Region-Aware Convolution (DRConv), which can automatically assign filters to corresponding spatial-dimension regions with learnable instructor. As a consequence, DRConv has powerful semantic representation ability and perfectly maintains translation-invariance property. In detail, we design a learnable guided mask module to automatically generate the filters' region-sharing-pattern for each input image according to their own characteristic. The region-sharing-pattern means that we divide spatial dimension into several regions and only one filter is shared within each region. The filters for different samples and different regions are dynamically generated based on the corresponding input features, which is more effective to focus on their own vital characteristic.

The structure of our DRConv is shown in Fig. 1. We apply standard convolution to generate guided feature from the input. According to the guided feature, the spatial dimension is divided into several regions. As can be seen, pixels with same color in the guided mask are attached to the same region. In each shared region, we apply filter generator module to produce a filter to execute 2D convolution operation. So the parameters needed to be optimized are mainly in filter generator module, and its amount has nothing to do with spatial size. Therefore, apart from significantly improving networks' performance, our DRConv can greatly reduce the amount of parameters compared with local convolution, and nearly doesn't increase the computation complexity compared with standard convolution.

To verify the effectiveness of our method, we conduct a series of empirical studies on several different tasks, including image classification, face recognition, object detection and segmentation by simply replacing standard convolution with our DRConv. The experimental results show that DRConv can achieve excellent performance on these tasks. We also offer adequate ablation studies for analyzing the effectiveness and robustness of our DRConv.

In brief, this work makes the following contributions,

1. We present a new Dynamic Region-Aware Convolution, which not only has powerful semantic representation ability but also perfectly maintains translation-invariance property.

2. We specially design the backward propagation process for learnable guided mask, so that our region-sharing-pattern is determined and updated according to the gradient of the overall task loss through backward propagation, which means our method can be optimized in an end-to-end manner.

3. Our DRConv can achieve excellent performance on image classification, face recognition, detection and segmentation tasks by simply replacing standard convolution without increasing much computation cost.

2. Related Work

We distinguish our work from other methods in term of spatial related work and dynamic mechanism.

Spatial Related Convolution. From the perspective of spatial related convolution design, the earliest enlightenment is local convolution. To effectively utilize the semantic information in image data, local convolution [9] applies individual unshared filters to each pixel, which has great potential in tasks that don't require translation-invariance. DeepFace [28] and DeepID series [26, 25] demonstrate the advantages of local convolution on face recognition task. These works illustrate that local distribution of spatial dimension is important.

On other tasks, such as detection, R-FCN [5] uses region-based fully convolutional networks to extract the local representations. It enlarges output channels to 3×3 times, and then selects corresponding subtensor in different channels to put together to form 3×3 blocks. On person re-identification, Sun et al. [27] applies part-based convolution to learn discriminative part-informed features, which can also be viewed as a kind of spatial related convolution.

Besides the mentioned methods above, some studies try to change the spatial feature to better model the semantic variations. Spatial Transform Networks [22] learns transformation to warp the feature map but difficult to be trained. Jeon et al. [17] introduces a convolution unit called active convolution unit (ACU), which produces unfixed shape because they can learn to any form through back propagation during training. ACU augments the sampling locations in the convolution with the learning offsets, and the offsets become static after training. Deformable Convolutional Networks [6] further makes the location offsets dynamic and then add the offsets to the regular grid sampling locations in the standard convolution.

Compared with the mentioned studies above, our method adaptively divides the spatial dimension into several regions and shares one filter within each region. What's more, our design can maintain translation-invariance and extract more plentiful information.

Dynamic Mechanism. With the prevalence of data de-

pendency mechanism [1, 16, 29] , which emphasizes to extract more customized feature [22]. Studies about dynamic mechanism have promoted many tasks to new state-of-the-art. Benefiting from data dependency mechanism, networks can flexibly adjust themselves, including the structure and parameters, to fit the diverse information automatically and improve representation ability of neural networks.

Some methods [2, 32] indicate that different regions in the spatial dimension are not equally important in representation learning. For instance, activation in important regions needs to be amplified so that it can play dominant role in the forward propagation. SKNet [19] designs an efficacious module to channel-wisely select suitable receptive fields on the basis of channel attention and achieves better performance. It dynamically restructures the networks for the sake of different receptive fields in dilated convolutions [34, 35]. In semantic segmentation, [36] imposes a pixel-group attention to remedy the deficiency of spatial information in SENet and [15] builds a link between each pixel and its surrounding pixels to capture important information. Attention mechanism is designed to dynamically calibrate the information flow in the forward propagation by learnable method.

From the aspect of dynamic weights, CondConv [33] obtains dynamic weights by dynamical linear combination of several weights. And the specialized convolution kernels for each sample are learned in a way similar to mixture of experts. In spatial domain, to handle object deformations, Deformable Kernels [8] directly resamples the original kernel space to adapt the effective receptive field (ERF). Local Relation Networks [14] adaptively determines aggregation weights for spatial dimension based on the compositional relationship of local pixel pairs. Non-local [31] operation computes the response at each position by weighted sum of the features at all positions, which can make it to capture long-range dependencies.

Different from above dynamic methods, DRConv applies a dynamic guided mask to automatically determine the distribution of multiple filters so that it can handle variable distribution of spatial semantics.

3. Our Approach

The weight sharing mechanism inherently limits standard convolution to model semantic variations due to single filter's poor capacity. Therefore, standard convolution has to violently increase the number of filters across channel dimension to match more spatial visual elements, which is inefficient. Local convolution makes use of the diversity of spatial information but sacrifices translation-invariance. To deal with the above limitations once for all, we go a step further and propose a feasible solution called DRConv as Fig. 1 shows, which not only increases the diversity of statistic by using more than one filter across spatial dimension, but also

maintains translation-invariance for these positions having similar feature.

3.1. Dynamic Region-Aware Convolution

We first briefly formulate standard convolution and basic local convolution, then transfer to DRConv. For convenience, we omit kernel size and stride of filters. The input of standard convolution can be denoted as $X \in \mathbb{R}^{U \times V \times C}$, where U, V, C mean height, width and channel respectively. And $S \in \mathbb{R}^{U \times V}$ denotes spatial dimension, $Y \in \mathbb{R}^{U \times V \times O}$ for the output, and $W \in \mathbb{R}^C$ for standard convolution filters. For the o-th channel of output feature, the corresponding feature map is

$$Y_{u,v,o} = \sum_{c=1}^{C} X_{u,v,c} * W_c^{(o)} \quad (u,v) \in S, \qquad (1)$$

where $*$ is 2D convolution operation.

For basic local convolution, we use $W \in \mathbb{R}^{U \times V \times C}$ to denote the filters which don't share across spatial dimension. Therefore the o-th output feature map can be expressed as

$$Y_{u,v,o} = \sum_{c=1}^{C} X_{u,v,c} * W_{u,v,c}^{(o)} \quad (u,v) \in S, \qquad (2)$$

where $W_{u,v,c}^{(o)}$ represents individual unshared filter at pixel (u,v) which is different from standard convolution.

Following above formulations, we define a guided mask $M = \{S_0, \cdots, S_{m-1}\}$ to represent the regions divided from spatial dimension, in which only one filter is shared in region S_t, $t \in [0, m-1]$. M is learned from the input features according to the data dependency mechanism. We denote the filters of the regions as $W = [W_0, \cdots, W_{m-1}]$, where the filter $W_t \in \mathbb{R}^C$ is corresponding to the region S_t. The o-th channel of this layer's output feature map can be expressed as

$$Y_{u,v,g} = \sum_{c=1}^{C} X_{u,v,c} * W_{t,c}^{(o)} \quad (u,v) \in S_t, \qquad (3)$$

where $W_{t,c}^{(o)}$ represents the c-th channel of the $W_t^{(o)}$ and (u,v) is one of the points in region S_t. It needs to be noted that the point (u,v) we use here is corresponding to the center of convolutional filter if the kernel size is larger than 1×1. That means a filter with kernel size 3×3 or 5×5 will extract features in adjacent regions on the border.

In general, our method is mainly divided into two steps. First, we use a learnable guided mask to divide the spatial features into several regions across spatial dimension. As Fig. 1 shows, pixels with same color in the guided mask are attached to same region. And from the prospective of

image semantics, the semantically similar features will be assigned to the same region.

Second, in each shared region, we use filter generator module to produce a customized filter to execute normal 2D convolution operation. The customized filter can be adjusted automatically according to the important characteristic of the input images. To better explain our method, we mainly introduce the following two modules: Learnable guided mask and Filter generator module. Learnable guided mask decides which filter will be assigned to which region. Filter generator module generates the corresponding filters which will be assigned to different regions.

3.2. Learnable guided mask

As one of the most important parts of our proposed DRConv, learnable guided mask decides the distribution (which filter will be assigned to which region) of filters across spatial dimension and is optimized by loss function. So that the filters can automatically adapt to the variance of spatial information for each input and the filter distribution will vary accordingly. In detail, as for a $k \times k$ DRConv with m shared regions, we apply a $k \times k$ standard convolution to produce guided feature with m channels (k means kernel size). We use $F \in \mathbb{R}^{U \times V \times m}$ to denote guided feature, $M \in \mathbb{R}^{U \times V}$ for guided mask. For each position (u, v) in spatial domain, we have

$$M_{u,v} = argmax(\hat{F}_{u,v}^0, \hat{F}_{u,v}^1, \cdots, \hat{F}_{u,v}^{m-1}), \quad (4)$$

where $argmax(\cdot)$ outputs the maximum value's index and $F_{u,v}$ denotes the vector of guided feature at position (u, v) and has m elements. So values in guided mask vary from 0 to $m - 1$ and indicate the index of filters which should be used in corresponding positions.

To make guided mask learnable, we must get the gradient for weights which produce guided feature. However, there is no gradient for guided feature, resulting in that the related parameters can not be updated. Therefore, we design approximate gradient for guided feature in an alternative way as shown in Fig. 2(a). We will explain the forward and backward propagation in detail.

Forward propagation: Since we have the guided mask as Eq.(4), we can get the filter $\hat{W}_{u,v}$ for each position (u, v) as flowing:

$$\hat{W}_{u,v} = W_{M_{u,v}} \quad M_{u,v} \in [0, m-1] = W * M_{u,v}, \quad (5)$$

where $W_{M_{u,v}}$ is one of the filters $[W_0, \cdots, W_{m-1}]$ generated by $G(\cdot)$ and $M_{u,v}$ is the index of the maximum across the channel dimension of guided feature F at position (u, v). In this way, m filters will build corresponding relationship with all positions and the entire spatial pixels can be divided into m groups. These pixels using the same

filter are of similar context because a standard convolution with translation-invariance conveys their information to guided feature.

Backward propagation: As shown in Fig. 2(a), we first introduce $\hat{F}$, which is the substitution of guided mask's one-hot-form(e.g., $M_{u,v} = 2, m = 5$, and $M_{u,v}$'s one-hot-form is $[0, 0, 1, 0, 0]$) in backward propagation,

$$\hat{F}_{u,v}^j = \frac{e^{F_{u,v}^j}}{\sum_{n=0}^{m-1} e^{F_{u,v}^n}} \quad j \in [0, m-1], \quad (6)$$

Eq.(6) is $softmax(\cdot)$ function, which is applied to guided feature F across channel dimension. With $softmax$ operation, we expect $\hat{F}_{u,v}^j$ to approximate 0 or 1 as close as possible. As a result, the gap between $\hat{F}_{u,v}^j$ and guided mask's one-hot-form becomes very small. Moreover, $\hat{W}_{u,v}$ in Eq.(5) can be viewed as the filters $[W_0, \cdots, W_{m-1}]$ multiplied by the one-hot-form of $M_{u,v}$ which can be approximated by $[\hat{F}_{u,v}^0, \cdots, \hat{F}_{u,v}^{m-1}]$. Then the gradient of $\hat{F}_{u,v}^j$ can be got by

$$\bigtriangledown_{\hat{F}_{u,v}^j} \mathcal{L} = \langle \bigtriangledown_{\hat{W}_{u,v}} \mathcal{L}, W_j \rangle \quad j \in [0, m-1], \quad (7)$$

where $\langle, \rangle$ denotes dot product and $\bigtriangledown . \mathcal{L}$ means the tensor's gradient with respect to loss function. As Fig. 2(a) shows, Eq.(7) is the approximate backward propagation of Eq.(5).

$$\bigtriangledown_{F_{u,v}} \mathcal{L} = \hat{F}_{u,v} \odot (\bigtriangledown_{\hat{F}_{u,v}} \mathcal{L} - \mathbf{1} \langle \hat{F}_{u,v}, \bigtriangledown_{\hat{F}_{u,v}} \mathcal{L} \rangle), \quad (8)$$

where $\odot$ denotes element-by-element multiplication and Eq.(8) is exactly the backward propagation of Eq.(6). If we don't design special backward propagation, SGD can't optimized relevant parameters because the function $argmax(\cdot)$ is non-differentiable and will stop the propagation of gradient. Therefore, $softmax(\cdot)$ function is used to be an approximate replacement of $argmax(\cdot)$ in backward propagation, which is differentiable and will minify the gap between the two function's outputs. More importantly, we can utilize it to transfer the gradient to guided feature so that guided mask can be optimized.

3.3. Dynamic Filter: Filter generator module

In our DRConv, multiple filters will be assigned to different regions. Filter generator module is designed to generate filters for these regions. Due to the diversity of characteristics among different images, shared filter across images is not effective enough to focus on their own characteristics. Such as images with different poses and viewpoints in face recognition and object detection tasks, which customized features are needed to focus on specific characteristic of each image.

4
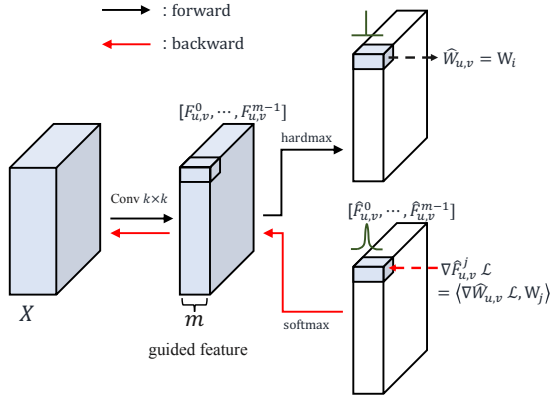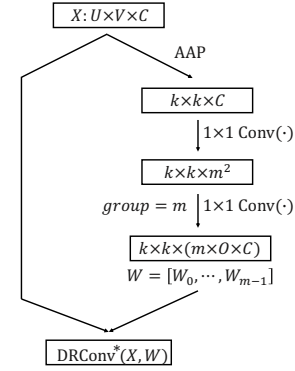
Figure 2. (a) is the optimization process of learnable guided mask. Symbols are defined as Section 3.2. We assign $\hat{W}_{u,v} = W_i$ when $F_{u,v}^i$ is the maximum across channel dimension. (b) is the architecture of DRConv for illustrating filter generator module. DRConv* denotes part of DRConv, AAP for adaptive average pooling and Conv($\cdot$) for standard convolution

Following the symbols we use above, we denote the input as $X \in \mathbb{R}^{U \times V \times C}$ and $G(\cdot)$ for filter generator module which mainly includes two convolution layers. The m filters are denoted as $W = [W_0, \cdots, W_{m-1}]$ and each filter is only shared in one region R_t. As shown in Fig. 2(b), to get m filters with kernel size $k \times k$, we use adaptive average pooling to downsample X to size $k \times k$. Then we apply two consecutive 1×1 convolution layers: the first uses $sigmoid(\cdot)$ as activation function and the second with $group = m$ doesn't use activation function. Filter generator module can enhance the ability of capturing specific characteristics of different images. As Fig. 2(b) shows, the filters for convolution are predicted based on the feature of each sample respectively. So each filter can be adjusted automatically according to their own characteristic.

4. Experiments

In this section, we will prove the effectiveness of our proposed DRConv by embedding it into the existing popular neural networks including ShuffleNetV2 [21] and MobileNet series [13, 24]. We compare DRConv with existing state-of-the-art on ImageNet [23], MS1M-V2 [10], and COCO in terms of image classification, face recognition, object detection and segmentation. Unless otherwise specified, all the experiments of DRConv are based on 8-learnable-region (i.e. $m = 8$).

4.1. Classification

The ImageNet 2012 dataset [23] is a widely accepted and authoritative image classification dataset, consisting of 1.28 million training images and 50k validation images from 1000 classes. Following the mainstream works, all the models are trained on the entire training dataset and evaluated by the single-crop top-1 validation set accuracy. And for both training and evaluation, the input image resolution is 224×224. The training setting follows [21], all models in our experiments are trained for 240 epochs, with learning rate which starts from 0.5 and linearly reduces to 0.

To prove the effectiveness of DRConv, we compare DRConv with state-of-the-art methods including [8, 33]. The results are shown in Table 1. In the first column, for example, CondConv-ShuffleNetV2 means that all 1×1 standard convolutions in ShuffleNetV2 are replaced by CondConv [33]. For DRConv-based model, we replace all the 1×1 standard convolutions in the backbone with DRConv. As can be seen, with comparable computational cost, DRConv-ShuffleNetV2 achieves 6.3% and 3.6% gain over ShuffleNetV2 for $0.5\times$ and $1\times$ scale respectively. DRConv-MobileNetV2 achieves a 3.7% gain over MobileNetV2 and DRConv-MobileNetV1 achieves a 4.9% gain over the baseline MobileNetV1. We also evaluate our approach by using fewer computational cost, and found that we still had an advantage over CondConv in using fewer calculations as shown in model tagged with $*$. These experimental results show that DRConv-based networks not only have a considerable improvement over the baselines, but also a great improvement over the state-of-the-art methods, demonstrating the effectiveness of our method.

As a basic of some other tasks, classification needs to extract as ample information as possible for predicting image's label because the large number of categories in ImageNet dataset. Traditional large networks can realize state-of-the-art due to their huge depth and width. As for efficient networks which are expected to be used in practice, they need to improve the efficiency of extracting useful information under the constraint of limited depth and width.

Table 1. Comparisons with state-of-the-art in terms of Top-1 classification accuracy (%) on ImageNet. DRConv outperforms previous methods (global 2×2 & local 4×4 means global 1×1 Deformable kernels with scope size 2×2 and local 3×3 Deformable kernels with scope size 4×4. ∗ means using fewer computational cost. MADDs refers to the number of multiply-adds operations)

Model	Setting	MADDs	Top-1 ACC.(%)
ShuffleNetV2 0.5×	baseline	42M	60.8
CondConv-ShuffleNetV2 0.5×	8 weights	43M	65.0
DRConv-ShuffleNetV2 0.5×* (Ours)	8 regions	**39M**	**64.9**
DRConv-ShuffleNetV2 0.5× (Ours)	8 regions	46M	**67.1**
ShuffleNetV2 1×	baseline	147M	69.5
CondConv-ShuffleNetV2 1×	8 weights	152M	72.0
DRConv-ShuffleNetV2 1×* (Ours)	8 regions	**109M**	**72.2**
DRConv-ShuffleNetV2 1× (Ours)	8 regions	157M	**73.1**
MobileNetV2 [24]	baseline	300M	72.0
CondConv-MobileNetV2 [33]	8 weights	329M	74.6
DK-MobileNetV2 [8]	global 2×2 & local 4×4	760M	74.8
DRConv-MobileNetV2* (Ours)	8 regions	**201M**	**74.7**
DRConv-MobileNetV2 (Ours)	8 regions	328M	**75.7**
MobileNetV1 [13]	baseline	569M	70.6
CondConv-MobileNetV1 [33]	8 weights	600M	73.7
DRConv-MobileNetV1* (Ours)	8 regions	**344M**	**74.4**
DRConv-MobileNetV1 (Ours)	8 regions	610M	**75.5**

Table 2. Results of DRConv on Megaface. "ACC." refers to the rank-1 face identification accuracy with 1M distractors. (Training dataset: MS1M-V2)

Model	MADDs ($\times 10^6$)	ACC.(%)
MobileFaceNet	189	91.3
Local-MobileFaceNet	189	94.9
CondConv-MobileFaceNet	195	94.8
DRConv-MobileFaceNet	201	**96.2**

Therefore, we design DRConv to augment representation capacity by making full use of the diversity of spatial information without extra computation cost. The multi-filter strategy for spatial information means it can match more information pattern.

4.2. Face Recognition

We use MobileFaceNet [3] as our backbone which has only 1M parameters and 189M MADDs with input size 112×96. To keep stability of training, we replace the Arc-face loss [7] with AM-Softmax loss [30] in our implementation. The dataset we use for training is MS1M-V2, which is introduced as a large-scale face dataset with 5.8M images from 85k celebrities. It is a semi-automatic refined version of the MS-Celeb-1M dataset [10] which consists of 1M photos from 100k identities and has a large number of noisy image or wrong ID labels. The dataset we use for evaluation is MegaFace [18], which includes 1M images of 60k identities as the gallery set and 100k images of 530 identities

from FaceScrub as the probe set. Due to the same reason, it is also a refined version by manual clearing.

Training and Evaluation: We use SGD with momentum 0.9 to optimize the model and the batch size is 512. We train all the models for 420k iterations. The learning rate begins with 0.1 and is divided by 10 at 252k, 364k and 406k iterations. The setting of weight decay follows [3]. For evaluation, we use face identification metric which refers to the rank-1 accuracy on MegaFace as the evaluation indicator.

In order to verify the effectiveness of our DRConv, we compare DRConv with several related methods. Based on the MobileFaceNet backbone, we simply replace 1×1 standard convolution in all bottleneck blocks with our DRConv. As Table 2 shows, DRConv-MobileFaceNet outperforms the baseline by a margin of 4.9%, and achieves a 1.4% gain over CondConv. For further comparison, we choose local convolution which works for face recognition but needs huge amount of parameters. Under the limitation of device memory, we apply local convolution in the last three layers. DRConv-MobileFaceNet achieves 1.3% higher accuracy than Local-MobileFaceNet(using local convolution in MobileFaceNet), further indicating the superiority of our proposed DRConv. Due to spatial stationarity of local statistics in face dataset, DRConv's guided mask module can learn clear semantic pattern. As shown in Fig. 3, the facial components appear in these guided masks.

4.3. COCO Object Detection and Segmentation

We further evaluate the effectiveness of our DRConv on object detection and segmentation. We use the COCO
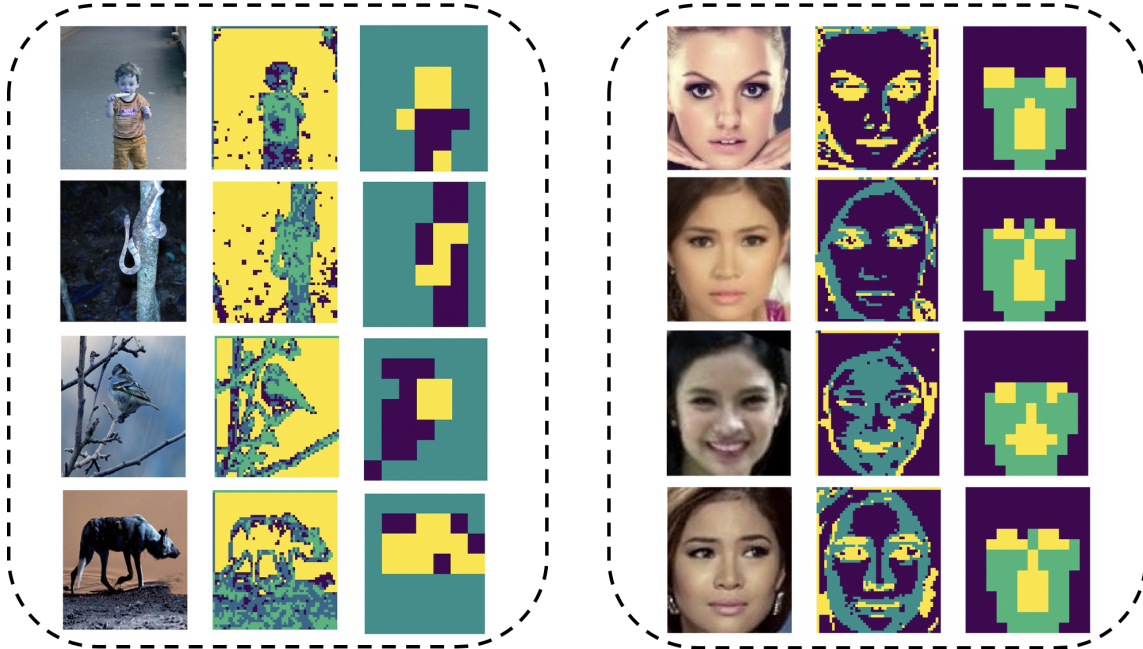
Figure 3. Visualizations: guided mask of image on classification and face recognition. The first column represents the original images. The second and third columns represent visualizations of different layers' guided masks, respectively

Table 3. Results of DRConv on COCO object detection and segmentation. "R" in 8R denotes region number. We replace 1×1 standard convolutions in the backbone of DetNAS-300M and only *two-layer* in FPN of Mask R-CNN with DRConvs.

Model	$\mathrm{AP}^{\mathrm{bbox}}$	$\mathrm{AP}_{50}^{\mathrm{bbox}}$	$\mathrm{AP}_{75}^{\mathrm{bbox}}$	$\mathrm{AP}^{\mathrm{mask}}$	$\mathrm{AP}_{50}^{\mathrm{mask}}$	$\mathrm{AP}_{75}^{\mathrm{mask}}$
DetNAS-300M	36.6	57.4	39.3	\	\	\
DRConv-DetNAS-300M 8R	**38.4**	**59.6**	**41.6**	\	\	\
MaskRCNN	39.1	59.0	42.8	34.5	55.8	36.6
DRConv-MaskRCNN 4R	39.8	60.3	43.3	35.3	57.1	37.4
DRConv-MaskRCNN 8R	40.2	60.8	44.0	35.5	57.6	37.6
DRConv-MaskRCNN 16R	**40.3**	**61.2**	**44.2**	**35.6**	**58.0**	**37.6**

dataset which consists of 80k train images and 40k val images. As in many previous works, we train on the union of 80k train images and a 35k subset of val images excluding 5k minival images, on which we evaluate our DRConv.

In experiments, we use DetNAS-300M [4] and Mask R-CNN [11] framework with FPN [20] and a 4conv1fc box head as the basis to assess our method. Weights are initialized by the parameters of ClsNASNet [4] and ResNet50 [12] respectively which are trained on the ImageNet dataset [23] and used as the feature extractor. In DetNAS-300M, training settings follow [4]. In Mask R-CNN, the number of proposals in the head part for possible objects is set as 512. We train the detection and segmentation networks on 8 GPU with a batch size of 16 for 180k iterations. At beginning, we warm up the networks with factor 0.33 for 500 iterations. During the training process, we use the learning rate 0.2 and decay the learning rate by 0.1 times at 120k, 140k and 150k iterations.

Our goal is to assess the effect when we replace 1×1 standard convolutions in the backbone of DetNAS-300M and only *two-layer* in FPN of Mask R-CNN with DR-Convs, so that any improvement on performance can be attributed to the effectiveness of our DRConv. In addition, we apply 4-learnable-region, 8-learnable-region and 16-learnable-region settings to DRConv for analysing the influence of different number of regions.

The results of comparing our DRConv with standard convolution are shown in Table 3. From the results we can see that DRConv with 8 regions in DetNAS-300M can significantly improve the performance by 1.8% for detection, only two DRConv layers with 16 regions in FPN of Mask R-CNN can improve the performance by 1.2% for detection and 1.1% for segmentation on COCO's standard AP metric. DRConv utilizes guided mask to divide spatial dimension into groups so that each filter can focus on special context. On other hand, the noise like background can be
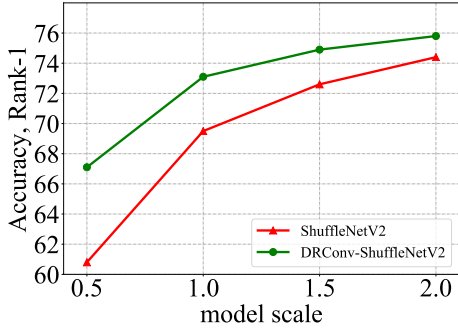
(a) DRConv-ShuffleNetV2
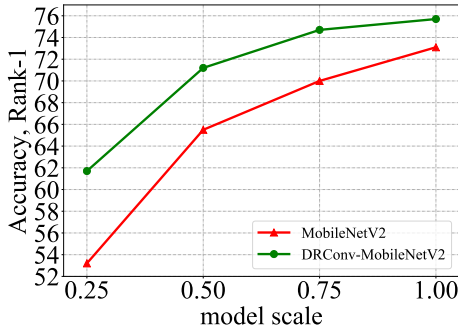
(b) DRConv-MobileNetV2

Figure 4. Results of DRConv-ShuffleNetV2 and DRConv-MobileNetV2 on different model size. Small models will achieve higher gain

easily separated from other regions of interest and most of the filters can concentrate on important regions. For different number of shared regions, the results are shown that DRConv can achieve better performance when we divide spatial dimension to more regions. More divided regions make every group's context more dedicated and each filter can be optimized more easily.

5. Ablation Study

The ablation experiments are conducted on classification (ImageNet 2012 [23]) and face recognition (MS1M-V2 [10]). The experimental settings are the same as that in section 4. In this section, we analyze the semantic information of learnable guided mask, the influence of different model size. The influence of different region number and different spatial size with respect to DRConv is analyzed in the *supplementary material*.

Visualization of dynamic guided mask. In order to

explore the mechanism of the learnable guided mask in our method, we visualize guided mask with $m = 4$ for image on classification task and face recognition task respectively. Fig. 3 shows that our method successfully assigns filters to regions with the same semantics. In other words, we have made it possible to learn that different regions use different filters according to image semantics, which is reasonable and effective. Due to more clear semantic representation, guided mask may automatically form less number of regions in deeper layer.

It needs to be noted that the guided mask is totally decided by the spatial information distribution so that one region might be connected points or discrete points. The points of a region in shallow layers tend to be discrete because the feature is more relevant to the detailed context of the input image. The points of a region in deep layers tend to be connected because the points have a bigger receptive field which is more relevant to the semantic information.

Different model size. Besides the investigation above, we conduct the ablation study of DRConv's performance on different model size. On ImageNet dataset, we carry out experiments on $[0.5\times, 1\times, 1.5\times, 2\times]$ of ShuffleNetV2 and $[0.25\times, 0.5\times, 0.75\times, 1\times]$ of MobileNetV2 to analyse the effectiveness of our DRConv. From our experimental results shown in Fig. 4, smaller models with DRConv will gain more bonus than larger models. Obviously, small models are of less input channels and filters in each layer and they can't extract enough feature for next layer. By replacing standard convolution with DRConv, small models will conspicuously improve their capability of modeling semantic information, resulting in better performance.

6. Conclusion

In this paper, we propose a new convolution named Dynamic Region-Aware Convolution (DRConv), which is motivated by partial filter sharing in spatial domain and successfully maintains translation-invariance property. Therefore, our proposed DRConv can entirely become substitute of standard convolution in any existing networks. We design a small learnable module to predict the guided mask for instructing the filters' assignment, which guarantees similar feature in a region can match the same filter. Furthermore, we design filter generator module to produce the customized filters for each sample, which makes it possible that different inputs can use their own specialized filters. Comprehensive experiments on several different tasks have shown the effectiveness of our DRConv, which can outperform state-of-the-art and other superior manually designed methods in classification, face recognition, object detection and segmentation. And our experiments in ablation study manifest that learnable guided mask plays a key role in filter distribution for each sample, which can help to achieve better performance.

References

[1] Alan Allport. Visual attention. The MIT Press, 1989. 3

[2] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5659–5667, 2017. 3

[3] Sheng Chen, Yang Liu, Xiang Gao, and Zhen Han. Mobile-facenets: Efficient cnns for accurate real-time face verification on mobile devices. In *Chinese Conference on Biometric Recognition (CCBR)*, pages 428–438, 2018. 6

[4] Yukang Chen, Tong Yang, Xiangyu Zhang, GAOFENG MENG, Xinyu Xiao, and Jian Sun. Detnas: Backbone search for object detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6642–6652, 2019. 7

[5] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 379–387, 2016. 2

[6] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)*, pages 764–773, 2017. 2

[7] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4690–4699, 2019. 6

[8] Hang Gao, Xizhou Zhu, Steve Lin, and Jifeng Dai. Deformable kernels: Adapting effective receptive fields for object deformation. *arXiv preprint arXiv:1910.02940*, 2019. 3, 5, 6

[9] Karo Gregor and Yann LeCun. Emergence of complex-like cells in a temporal product network with local receptive fields. *arXiv preprint arXiv:1006.0448*, 2010. 1, 2

[10] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision (ECCV)*, pages 87–102, 2016. 5, 6, 8

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017. 7

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1, 7

[13] 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. 5, 6

[14] Han Hu, Zheng Zhang, Zhenda Xie, and Stephen Lin. Local relation networks for image recognition. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3464–3473, 2019. 3

[15] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 603–612, 2019. 3

[16] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2017–2025, 2015. 3

[17] Yunho Jeon and Junmo Kim. Active convolution: Learning the shape of convolution for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4201–4209, 2017. 2

[18] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. The megaface benchmark: 1 million faces for recognition at scale. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4873–4882, 2016. 6

[19] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 510–519, 2019. 3

[20] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2117–2125, 2017. 7

[21] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *European Conference on Computer Vision (ECCV)*, pages 116–131, 2018. 5

[22] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In *European Conference on Computer Vision (ECCV)*, pages 181–196, 2018. 2, 3

[23] 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. *International Journal of Computer Vision (IJCV)*, pages 211–252, 2015. 5, 7, 8

[24] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520, 2018. 5, 6

[25] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1988–1996, 2014. 1, 2

[26] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation from predicting 10,000 classes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1891–1898, 2014. 1, 2

[27] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In *European Conference on Computer Vision (ECCV)*, pages 480–496, 2018. 2

[28] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1701–1708, 2014. 1, 2

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008, 2017. 3

[30] Feng Wang, Jian Cheng, Weiyang Liu, and Haijun Liu. Additive margin softmax for face verification. *IEEE Signal Processing Letters (SPL)*, pages 926–930, 2018. 6

[31] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7794–7803, 2018. 1, 3

[32] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 3

[33] Brandon Yang, Gabriel Bender, Quoc V Le, and Jiquan Ngiam. Condconv: Conditionally parameterized convolutions for efficient inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1305–1316, 2019. 3, 5, 6

[34] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 3

[35] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 472–480, 2017. 3

[36] Zilong Zhong, Zhong Qiu Lin, Rene Bidart, Xiaodan Hu, Ibrahim Ben Daya, Jonathan Li, and Alexander Wong. Squeeze-and-attention networks for semantic segmentation. *arXiv preprint arXiv:1909.03402*, 2019. 3