MASC-Units:Training Oriented Filters for Segmenting Curvilinear Structures

Chapter · September 2021

DOI: 10.1007/978-3-030-87231-1_57

2 authors, including:

Zewen Liu
The University of Manchester
2 PUBLICATIONS **1** CITATION

SEE PROFILE

MASC-Units:Training Oriented Filters for Segmenting Curvilinear Structures

Zewen Liu and Timothy Cootes

Division of Informatics, Imaging & Data Science, Stopford Building, The University of Manchester, Manchester, M13 9PT, United Kingdom
zewen.liu@manchester.ac.uk

Abstract. Many medical and biological applications involve analysing vessel-like structures. Such structures often have no preferred direction and a range of possible scales. We take advantage of this self-similarity by demonstrating a CNN based segmentation system that requires far fewer parameters than conventional approaches. We introduce the Multi Angle and Scale Convolutional Unit (MASC) with a novel training approach called Response Shaping. In particular, by reflecting and rotating a single oriented kernel we can generate four versions at different angles. We show how two basis kernels can lead to the equivalent of eight orientations. This introduces a degree of orientation invariance by construction. We use Gabor functions to guide the training of the kernels, and demonstrate that the resulting kernels generally form rotated versions of the same pattern. Invariance to scale can be added using a pyramid pooling layer. A simple model containing a sequence of five such blocks was tested on CHASE-DB1 dataset, and achieved better performance comparing to the benchmark with only 0.6% of the parameters and 25% of the training examples. The resulting model is fast to compute, converges more rapidly and requires fewer examples to achieve a given performance than more general techniques such as U-Net.

Keywords: Steerable filter · Image Segmentation.

1 Introduction

Many CNN-based approaches have been proposed for solving image segmentation tasks in medicine and biology. Such models are usually huge, with millions of parameters. However some tasks involve segmenting structures with a large degree of local self similarity. For instance, many curvilinear structures (such as blood vessels or neuron fibrils) can be thought of as being constructed from rotated and scaled versions of a small number of cannonical templates. We propose a system which explicitly re-uses kernels at multiple scales and orientations and is trained in a way that encourages invariance to rotation. This leads to a model with less than 1% of the parameters of more general approaches, such as U-Net, yet which gives performance at least as good.

We introduce a novel structure, called Multi-Angle Convolution (MAC) Unit, which is designed to encourage rotation invariance.

Unlike many steerable networks [2, 6, 15], a MAC Unit kernel is not manipulated by combining basis filters, but learns rotated versions during training.

The MAC Unit involves applying a set of filters, corresponding to different orientations, to a patch to compute a vector of outputs. The shape of these outputs (response vs orientation) is compared against an expected shape for each orientation in order to select the best angle. The output is then given by a weighted sum of the individual outputs, with the weights being angle specific (full details are given below). The advantage of this approach is that rather than just choose the best response, all filters contribute to the final output, which makes training more stable.

Since this approach involves training filters to achieve a particular shape of output responses, we call it "Response Shaping".

We can make the unit robust to scale changes by using a pyramid representation [5]. A MAC Unit is applied to different downsampled versions of the input, the results are upsampled and a max operation performed to select the best fitting scale.

In the following we describe the approach in detail, and demonstrate the approach on the the CHASE-DB1 dataset [13]. We show that we can achieve equivalent performance to the benchmark U-Net with only 0.6% of the parameters and 25% of the training samples.

2 Related Works

Steerable filters have been popular in image analysis for tasks such as edge detection and texture feature extraction [4]. A core property is rotational invariance, as the filters are evenly-distributed across orientations. Typically steerable filters require only a few parameters. One of the most popular is the Gabor filter [12]. They combine a cosine wavelet function with a 2D Gaussian.

Recently encouraging results have been achieved by combining steerable filters with convolutional networks. Optimisation can lead to steerable models that are more flexible, and have less filter redundancy [6]. In [2], Cohen and Welling designed an operator that can rotate a kernel by a chosen angle. The work had lower error rate comparing to the state-of-arts at the time. Worral *et al.* [16] proposed a new approach of compositing steerable filters using some atomic filters. Based on their results, Weiler *et al.* [15] proposed a model of arbitrary directions, with the corresponding combining process being trainable. However, these methods are computationally expensive as the kernel size is constrained by the number of predefined circular harmonic patterns. To bring more variance to the kernel pattern, large kernels are preferred but this brings increased computation. Ghosh and Gupta [6] proposed a generative kernel which is also scalable. This enables filters to detect a pattern at different scales. However, the kennel pattern is still not as flexible as desired and can be hard to optimize. In [1], Bekkers et al. enabled CNNs to deal with rotation and translation effects via bi-linear interpolation. In other works, the idea of using combined kernel is investigated. For example, in [9, 10], the authors combined a bank of generative Gabor filters

with some random initialized kernel. The resulting assembled neuron showed good performance on MNIST dataset.

Most steerable CNN models are based on a steering operator and the rotation basis is pre-defined to some degree. This is theoretically favored as it gives perfect rotation, but could put implicit constraints on training.

Our proposed MAC Unit variants achieve approximate rotation invariance by response shaping. This enables the model to be fully trainable, and also can be applied to any number of directions.

Unlike conventional matched filters for detecting particular signals in radar and images, both the optimal template and the response shape is unknown for MASC model before training.

In recent projects, multi-scale pooling has been shown to be beneficial when extracting features. Studies such as [7] suggested using a pyramid representation in analyzing multi-scale information. [5] introduced cross connections between different scale layers, and found they bring improvement to pose detection and image segmentation. Using pyramid pooling is much more computational efficient than the filter-scaling strategy used by the previous generative-based steerable models. For efficiency we use the invertible bottleneck [14]. Its narrow-wide-narrow shape can achieve comparable accuracy with fewer parameters.

3 Methods

Limitations of conventional steerable convolutional models include their computational expense, their inflexiblity/indifferentiablity in pattern generation and direction selection, and most are not fully trainable. Our method abandoned operator-based rotating, instead we use response shaping to solve these problems.

3.1 Rotatable MAC Unit and Response Shaping

A simple approach to achieving approximate rotational invariance would be to construct a set of n filters which are rotated versions of the basis kernel. Each would be convolved with the target image, and the strongest response taken at each pixel. In CNN terms, this is equivalent to performing a maximum operator over channels, where each channel is the output of one oriented filter.

Let $\mathbf{w}_i$ be a vector containing all the elements of the filter at orientation i (of n), which has been normalised so that $|\mathbf{w}_i| = 1$. If $\mathbf{x}$ is the vector of the input patch, then the response of the filter to the patch is given by $v_i = \frac{\mathbf{w}_i \cdot \mathbf{x}}{|\mathbf{x}|}$. The output of the simple approach would then be $\max_i(\frac{\mathbf{w}_i \cdot \mathbf{x}}{|\mathbf{x}|})$.

Let $M_{ij} = \mathbf{w}_i \cdot \mathbf{w}_j$ be the response of filter i to an image patch equal to filter j. Typically this will have a strong response when i is near j, and weak response when they are more mismatched. Let $\mathbf{m}_i = (M_{i1}|...|M_{in})^T$. The elements of $\mathbf{m}_i$ give the shape of the response of filter i to the other filters. See, for instance, Fig.2.

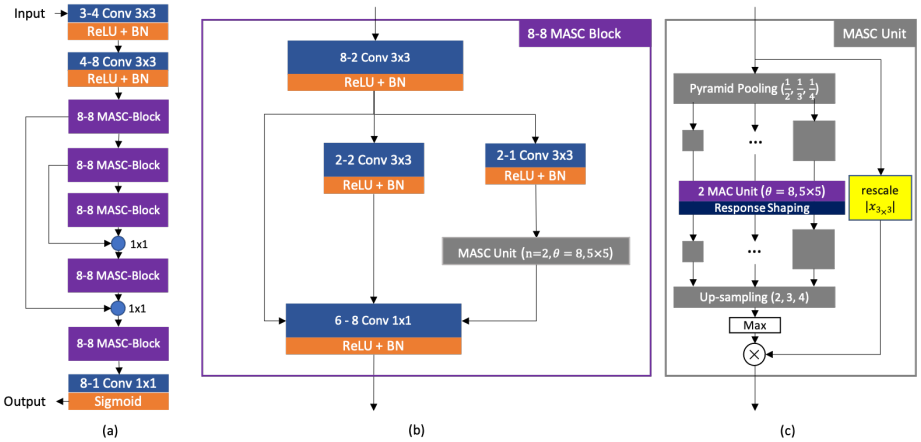

Fig. 1. (a) Model MASC-5-2-8, with 5 stacking MASC Blocks, each with 2 parallel MAC Units of 8 5x5 directional kernels; (b) 8-8 MASC Block, with 8 input and 8 output channels and a thin bottleneck; (c) MASC Unit, including 2 MAC Units applied on pyramid representations, where only the maximum is taken from across the scale channels. MASC is a normalized calculation, the result value will be rescaled with a 3×3 L2pooling node

In order to use information from all filters, the Multi-Angle Convolutional Unit performs the following operations to get its output for a patch $\mathbf{x}$;

$$v_i = \frac{\mathbf{w}_i \cdot \mathbf{x}}{|\mathbf{x}|} \; \forall i \tag{1}$$

$$R(\mathbf{x}) = \max_i\left(\frac{\mathbf{v} \cdot \mathbf{m}_i}{|\mathbf{m}_i|}\right) \text{ where } \mathbf{v} = (v_1|...|v_n)^T \tag{2}$$

Thus it is comparing the shape of $\mathbf{v}$ with that of each $\mathbf{m}_i$, and choosing the one with the largest similarity - the orientation used is that which provides an output most similar to the shape given by comparing each angle filter with all the others (rather than the strongest response from the individual filters).

If we initialise each $\mathbf{w}_i$ as oriented filters, such as Gabor kernels at equally spaced angles, they will have a particular shape of output when compared to one another.

During training we modify the filters, but seek to retain the relationship between them by encouraging the output of all of them to follow a particular shape using (2).

The first step uses normalised cross correlation, with the output dependent only on the angle between $\mathbf{x}$ and $\mathbf{w}_i$. This would discard the overall intensity, and would tend to exaggerate noise in near flat regions. In order to retain some information about the overall intensity we rescaled the output using

$$\text{MAC Unit}(\mathbf{x}) = |\mathbf{x}_{3\times3}|R(\mathbf{x}) \tag{3}$$

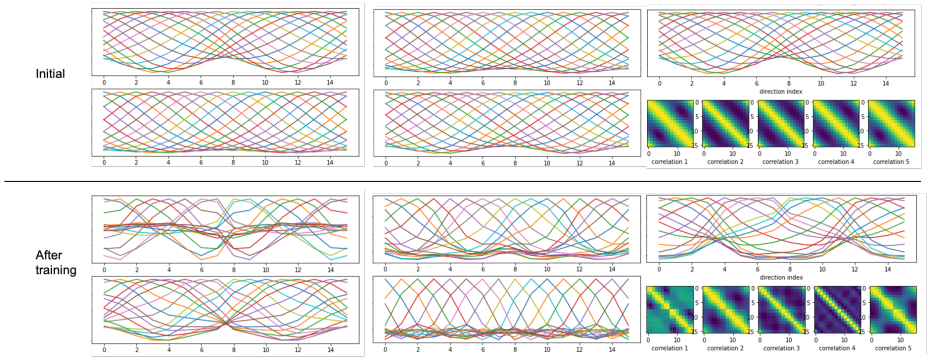

Fig. 2. The correlation curves of the first block in a MASC5-5-16 9x9 model, in which 5 parallel MAC units are used. Each curve represents a row in M which is illustrated bottom right. It shows that the correlation between two close kernels with similar directions is strong. This pattern is retained after training.

where $\mathbf{x}_{3\times3}$ is a vector containing the pixel information in the central 3×3 part of the input patch.

Instead of taking the maximum signal from $\mathbf{v}$ like [6, 11, 16], we combine all the signals in a way that is dependent on the estimate of the best angle. Extracting information by simple maximum selection can lead to poor training for directional kernels, kernels for rare directions can be underfitted. One the advantage of response shaping is that it includes all the directional parameters in the forward propagation. Also, using an n-length directional response vector better depicts the input in the feature space than a single maximum scalar.

During training the values of M_{ij} are updated before every forward propagation iteration.

Though MASC is not inspired by the idea of 'matched filters', they shared some common characteristics.

The MASC approach can also be thought of as a variant of an 'attention' mechanism. The directional response vector V can be regarded as the query vector, and the optimal response matrix M can be treated as the keys. Comparing to ViT(vision transformer) [3] on ImageNet, a medical task typically does not contain arbitrary objects, and useful features tend to be local. Unlike the common self-attention unit, here, the encoder for query and key are the same, and the model has n candidate keys for each query. Also, it does not refer to context information to compute keys but uses a learned bank of patterns. Since the mechanism is different from the 'self-attention', in the paper, we named it as response shaping.

3.2 Filter re-use

Suppose that we have a $k \times k$ (k odd) filter which gives a strong response to structures at orientation θ, with $0 < \theta < \pi/2$. If we rotate the kernel 90 degree,

we get a new filter which would give a strong response at angle $\pi/2 + \theta$. If we transpose the filter (reflect in the line $y = x$), then the new filter gives a strong response at angle $\pi/2 - \theta$. Thus a single set of filter weights can be re-used in four filters at angles $\theta, \pi/2 - \theta, \pi/2 + \theta, \pi - \theta$. So we need only two basis filters, at angles $\pi/16, 3\pi/16$ to generate a set of 8 filters equally spread over all angles ($\theta_i = (1+2i)\pi/16$). In general, for filters at $4a$ orientations we need a basis filters at angles $\theta_j = (1 + aj)\pi/(8a), j = 0..a - 1$. This enables a significant reduction in the total number of parameters in the model.

3.3 Initialization Strategy

We initialise the basis filters using Gabor filters at the appropriate angles, but with the other parameters (amplitude, σ, aspect ratio) chosen randomly.

We have tried two approaches to train the filters.

1. Initialise the basis filters with Gabor filters and optimise;
2. Use a Gabor function to weight the filter elements.

In the second case the basis filter, W_i, is given by the per-element (Hadamard) product of a Gabor kernel, G, with a filter B_i, so $W_i = G \circ B_i$. During optimisation we vary the parameters defining the Gabor kernel and the elements of B_i. We find that this is often more stable than simply initialising with a Gabor.

3.4 Multi-scale Processing with Pyramids

In the works of [17], multi-scale pooling has been proven effective in against scale variance and signal discontinuity. Here, instead of using different filters, we apply the same MAC Unit on each pyramid representation of the input. The outputs are rescaled via bilinear interpolation. Then only the maximum signal among scale channels is taken. The scale ratio selection depends on the nature of task. In our model, ratios of $(\frac{1}{2}, \frac{1}{3}, \frac{1}{4})$ are used, see Fig. 1(c).

The method of pyramid pooling takes less computation than using scaled kernels [6], as the kernel is applied on smaller feature maps. We use max pooling during the downsampling, for efficiency. We call the resulting network a Multi-Angle and Scale Convolutional (MASC) Unit, see Fig. 1(c).

We create a MASC-Block, as shown in Fig. 1(b). The input is to two nodes, a standard Conv layer and a MASC Unit. The combined feature map is concatenated with the residuals. A summarization layer combines the information.

More than one MAC Unit can be used in a MASC Unit. They are managed in parallel when processing the pooled features, and their results are concatenated. In experiments, we found the marginal profit of using more MAC Units tapers off gradually, and the speed of convergence depends on initialization. The channel numbers are arranged as wide-narrow-wide-narrow with a thin bottleneck.

By stacking 5 MASC blocks, each includes 2 independent 8-directional MAC Unit arranged in parallel (same input), a MASC-5-2-8 model is illustrated in Fig. 1(a). Here, a single MASC Unit can work like 24 filters (in 8 directions and 3 scales).

4 Experiments

In this paper, we tested the MASC-5-2-8 model on retina vessel images in the CHASE-DB1 dataset [13] (see Fig.2(b)). The Ground truth of 1stHO is used.

The dataset contains 30 color images. The first 20 were taken as training set, and the last 10 as test set as stated in [8]. Then 8000 50x50 patches were randomly cropped from the training images.

The MASC-5-2-8 model was initialized as described above. An Adam optimizer was used with plateau learning rate scheduler(start=0.01, ratio=0.5, patience=10). The initialization noise was picked from range $(0, 0.1]\times$Gabor amplitude. In pilot experiments, we found setting Gabor pattern to be more peaked (high γ, e.g. 15, exaggerates the difference among directions) can bring significant improvements in training, but the optimum choise may vary for different tasks.

The results are summarised in table 1. The MASC model with only 3,292 parameters outperforms the UNet benchmark almost on all columns, and is generally similar to other benchmarks despite having so few parameters. On the metric such as averaged specificity and F1 score, our method achieves best results.

Ablation Experiments As a baseline we replaced all MASC blocks with 8-8 convolutional layers with 3×3 kernels (using more parameters) - "MASC-replaced" in the table. This demonstrates the benefits of the MASC over simpler convolutions. The MASC5-2-8-init represents the case in which we initialize kernels W_i with Gabor distribution plus noise, rather than using a Hadamard product ($W_i = G \circ B_i$) during optimisation. It is not as stable as when using the product, occasionally failing to converge to a sensible form. MASC5-2-8-w/o-rescale is a version without the intensity rescaling (Equation 3) - the rescaling helps.

In model MASC5-2-8-max, the response shaping is replaced by a max function. This can be regarded as a special case of response shaping, where all directional kernels are mutually orthogonal and the correlations are equal to 0 except to itself. In this case, the max of combined responses is equivalent to the max of single-direction response. In the experiment, we found the performance of MASC5-2-8-max is close to the full version but is less stable when training.

We also explored how well the model performance varied with the size of the training set. Fig.3(a) shows how the Area Under the ROC Curve (AUC) declines much more slowly as the training set shrinks, compared to a U-Net. The MASC model can achieve good results even with relatively small numbers of training examples.

In Fig. 3 (c) and (d), some intermediate outputs from MASC Units with 16 directions are illustrated, together with the 9×9 kernel patterns and the maximum index map generated by equation (2). The response shaping approach has encouraged the kernels to represent rotated versions of the same pattern, which can also be read from the index map. For example, in the index map, the horizontal vessels are mostly identified by the yellow response shape. The indices for the background area are generally random. The intensity map shows the vessel area has a higher shaped response value.

Table 1. Comparison with other methods on CHASE_DB1(*results obtained from [8])

Methods	F1	Se	Sp	Acc	AUC
U-Net	79.2	79.1	97.4	95.4	95.3
Residual U-Net*	78.0	77.3	98.2	95.5	97.8
Reccurent U-Net*	78.1	74.6	**98.4**	96.2	98.0
R2U-Net*	79.3	77.6	98.2	**96.3**	**98.1**
LadderNet*	79.0	78.6	98.0	96.2	97.7
VesselNet*	79.1	78.2	98.1	96.2	97.6
MASC5-2-8-w/o-rescale	79.8	80.3	97.4	95.4	97.4
MASC5-2-8-max	79.7	78.6	97.7	95.5	97.2
MASC5-2-8-init	79.1	76.7	97.9	95.5	97.3
MASC-replaced	76.4	72.1	97.9	95.1	97.1
MASC5-2-8	**80.5**	**81.3**	97.4	95.5	97.6

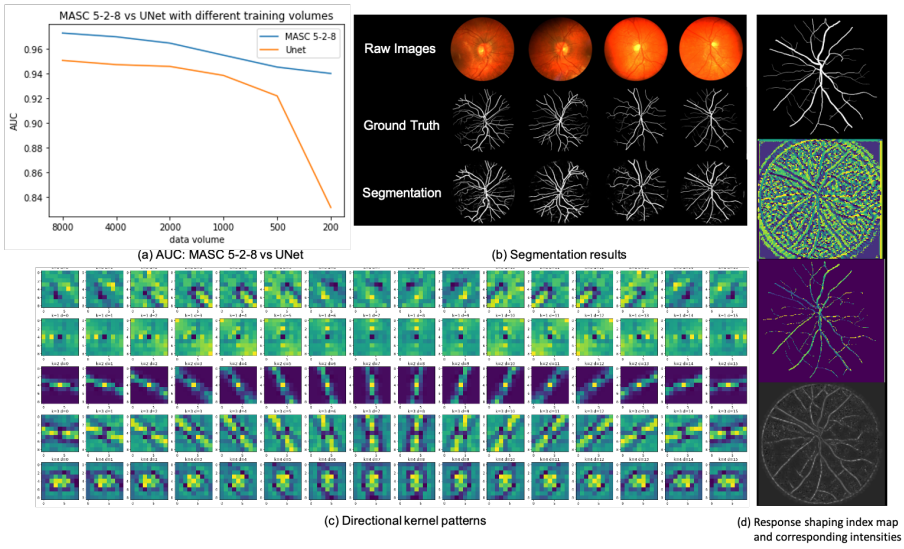

Fig. 3. (a) AUC comparison between MASC5-2-8 and UNet with different training set sizes; (b) Segmentation results; (c) Illustration of MASC5-5-16-9x9 directional kernels, MASC with response shaping can achieve a pseudo-steerable effect via gradient decent; (d) The maximum index map generated by equation (2), the colors represent the chosen indices. From top to bottom, i. ground truth, ii. response shaping index map, iii. masked index map by ground truth, iv. corresponding shaped response intensity map. The index map shows that structures with particular directions are picked out with specific index values.

5 Conclusion

We have introduced novel model for analysing curvilinear structures which are composed of self similar elements at arbitrary orientation and scale. The system learns a set of filters which can be transformed easily to produce responses at a range of angles. We show how this can be extended to include a range of scales. The resulting model is very parameter efficient.

On the task of retina vessel segmentation the model achieves accuracy equivalent to of the benchmark U-Net model with only 0.6% of the parameters and 25% of the training set. It is thus potentially very useful where limited numbers of training examples are available. Though this work focuses on retinal images, we have also applied it successfully to tracking growing axons in microscopy images. In future work we will extend it to 3D volume data.

References

1. Bekkers, E.J., Lafarge, M.W., Veta, M., Eppenhof, K.A., Pluim, J.P., Duits, R.: Roto-translation covariant convolutional networks for medical image analysis. In: International conference on medical image computing and computer-assisted intervention. pp. 440–448. Springer (2018)
2. Cohen, T.S., Welling, M.: Steerable cnns. arXiv preprint arXiv:1612.08498 (2016)
3. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
4. Freeman, W.T., Adelson, E.H., et al.: The design and use of steerable filters. IEEE Transactions on Pattern analysis and machine intelligence **13**(9), 891–906 (1991)
5. Ghiasi, G., Lin, T.Y., Le, Q.V.: Nas-fpn: Learning scalable feature pyramid architecture for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7036–7045 (2019)
6. Ghosh, R., Gupta, A.K.: Scale steerable filters for locally scale-invariant convolutional neural networks. arXiv preprint arXiv:1906.03861 (2019)
7. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
8. Liu, B., Gu, L., Lu, F.: Unsupervised ensemble strategy for retinal vessel segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 111–119. Springer (2019)
9. Liu, Z., Cootes, T., Ballestrem, C.: An end to end system for measuring axon growth. In: International Workshop on Machine Learning in Medical Imaging. pp. 455–464. Springer (2020)
10. Luan, S., Chen, C., Zhang, B., Han, J., Liu, J.: Gabor convolutional networks. IEEE Transactions on Image Processing **27**(9), 4357–4366 (2018)
11. Marcos, D., Volpi, M., Tuia, D.: Learning rotation invariant convolutional filters for texture classification. In: 2016 23rd International Conference on Pattern Recognition (ICPR). pp. 2012–2017. IEEE (2016)
12. Mehrotra, R., Namuduri, K.R., Ranganathan, N.: Gabor filter-based edge detection. Pattern recognition **25**(12), 1479–1494 (1992)

13. Owen, C.G., Rudnicka, A.R., Mullen, R., Barman, S.A., Monekosso, D., Whincup, P.H., Ng, J., Paterson, C.: Measuring retinal vessel tortuosity in 10-year-old children: validation of the computer-assisted image analysis of the retina (caiar) program. Investigative ophthalmology & visual science **50**(5), 2004–2010 (2009)
14. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4510–4520 (2018)
15. Weiler, M., Hamprecht, F.A., Storath, M.: Learning steerable filters for rotation equivariant cnns. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 849–858 (2018)
16. Worrall, D.E., Garbin, S.J., Turmukhambetov, D., Brostow, G.J.: Harmonic networks: Deep translation and rotation equivariance. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5028–5037 (2017)
17. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2881–2890 (2017)