Understanding Deep Neural Network by Filter Sensitive Area Generation Network

Yang Qian[1, 2], Hong Qiao[1, 2, 3, 4], and Jing Xu[5]

[1] The State Key Lab of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Science, Beijing 100190, China
[2] University of Chinese Academy of Sciences
[3] CAS Center for Excellence in Brain Science and Intelligence Technology
[4] Cloud Computing Center, Chinese Academy of Sciences
{hong.qiao,qianyang2016}@ia.ac.cn
[5] Department of Mechanical Engineering, Tsinghua University
jingxu@tsinghua.edu.cn

Abstract. Deep convolutional networks have recently gained much attention because of their impressive performance on some visual tasks. However, it is still not clear why they achieve such great success. In this paper, a novel approach called Filter Sensitive Area Generation Network (FSAGN), has been proposed to interpret what the convolutional filters have learnt after training CNNs. Given any trained CNN model, the proposed method aims to figure out which object part each filter represents in a high conv-layer, through appropriate input image mask which filters out unrelated area. In order to obtain such a mask, a mask generation network is designed and the corresponding loss function is defined to evaluate the changes of feature maps before and after mask operation. Experiments on multiple datasets and networks show that FSAGN clarifies the knowledge representations of each filter and how small disturbance on specific object parts affects the performance of CNNs.

Keywords: Convolutional Neural Network, Interpretability, Knowledge Representations.

1 Introduction

Recent years have seen spectacular improvements in artificial intelligence. Particularly, deep neural networks (DNNs) has achieved superior performance in a variety of visual tasks, such as fine-grained classification [1,2], object detection [3,4] and semantic segmentation [5,6]. Although DNNs outperform previous machine learning techniques on the comparison of accuracy, we still have little knowledge about what they have learnt. When they fail on some cases, it is hard to explain what caused the DNNs to make such decisions. One Pixel Attack cheated the DNN successfully by changing value of a single pixel, which is impossible for human to make such mistakes. This lack of interpretability of DNNs is largely due to the end-to-end structure and learning strategy, which

lead to the difficulties of understanding the main role of individual neurons during the whole process of completing visual tasks.

Recently, a large number of researchers have realized the necessity of improving interpretability of DNNs and have proposed a variety of models to dig the interpretable knowledge representations learned by DNNs, especially by Convolutional Neural Networks (CNNs). Zeiler et al. [7] examine the pattern of every layer by visualization with a deconvnet and figure out whether a model is truly identifying the location of the object in the image by occluding different portion of the input image and observe the probability of the correct class. This approach finds the occlusion sensitive region of convolutional filters and classifiers, but the size of region is limited to a rectangle and the process is time-consuming. Yosinski et al. [8] visualized filters by finding an image that maximize the activation of this unit via regularized optimization. Much other work tries to leverage heatmaps to understand the decision-making process of networks. An approach called CLEAR [9] is invented to visualize attentive regions of DNNs during the decision-making process. These approaches change the original network structure or learning process more or less and give little insight about what each individual filter has learnt after a network is trained.

In this paper, we mainly focus on the question, which area of the input image does a convolutional filter mainly focuses on? Based on the observation that a specific filter has strong activations for certain parts of the object and keep silent for other areas, we expect to figure out the intrinsic activation mode of some filters and interpret what these filters have remembered after training.

To find out which parts each filter pays attention to automatically and efficiently, we propose a Filter Sensitive Area Generation Network (FSAGN) for generating input image mask to mask unimportant regions in an image. In consideration of sparse activation properties of neural network, we first statistically analyze average activation of every filter and filter out the silent filters. For each active filter, a network is designed to generate a mask of the input image and obtain a new input image by mask operation with the original image. Through a forward propagation, we can get new feature maps. By minimizing the difference between the original and new feature maps, FSAGN converges gradually and finally obtain the power to localize the key part that certain filter represents. Simultaneously, we also adopt an occlusion strategy to generate occlusion sensitive area. After we have a clear insight about which parts each filter focuses on, adversarial samples can be designed to cheat the original network.

The rest of this paper is organized as follows. The proposed framework and design of network are introduced in Sect. 2. Section 3 presents experimental results and corresponding analysis. Section 4 make a conclusion of the paper.

2 Filter Sensitive Area Generation Network

This section describes the proposed network for finding which parts contribute most to the response of certain filter.

2.1 Filters Selection

Thanks to the sparsity of feature maps, only a few filters response strongly to some parts of objects, while others remain inactivated. If we select a filter randomly to generate its concerned part, we may fail because it has small probability to represent specific part of objects theoretically.

In order to find which filters are sensitive for part discovery and are valuable to analyze, we first test all samples and record their activation for every channel. For better measuring the importance of filters, we calculate the sum of each feature map. Then we visualize the response map over all samples and channels, where the vertical axis is channel number, and the horizontal axis is the sample number. Due to the sparse response distributions of CNNs, some filters are always activated for all samples, while others keep silent no matter what images are selected as input.

As is seen in Fig. , there are several bright lines in the map, where most areas are dark, which indicates that these filters are potential to have strong response for some specific parts of objects. Therefore, we refer to these filters as the target filters for sensitive area discovery.

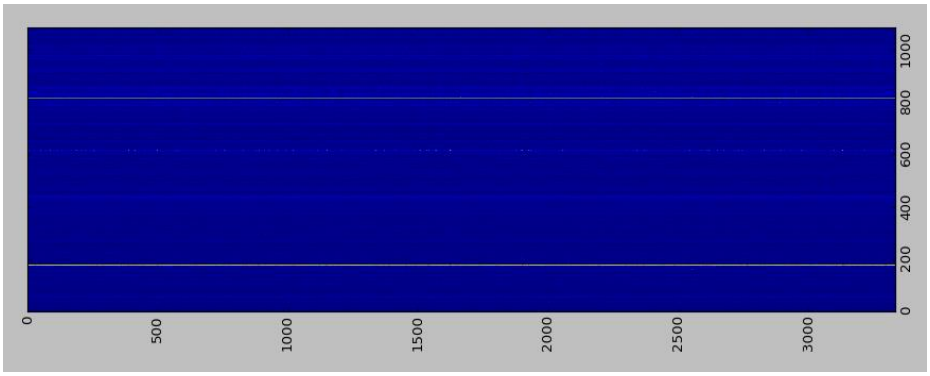

Fig. 1. Average response of each filter in CNN over multiple samples

2.2 Filter Sensitive Area Generation Network

Inspired by the observation that some convolutional filters only response to a small specific area on the input image, which means that when we occlude other areas, the filter activation is not affected dramatically. Zeiler. et al [7] manually adopt a gray rectangle window for occlusion test by sliding window over the whole image to generate an occlusion sensibility map, which is limited for the fixed shape and size of part area and the whole process is time-consuming because of sliding window strategy.

In this section we proposed a Filter Sensitive Area Generation Network (FSAGN) to locate the area that a filter focuses on. The network structure is shown in Fig. . Taking feature maps of the last convolutional layer in a trained CNN as input, the FSAGN outputs a mask with the same size as input image through a deconvolution structure [6]. Then the new image generated by mask operation is input to the original network and

the new feature maps are obtained too. By comparing the original and the new feature maps, we can evaluate the influence of different regions in images on the response of filters. Two criterions are adopted to define filter sensitive area. When the image except the sensitive area of a filter is set to zero, the response distribution of this filter will keep unchanged compared with the response from the intact image. This strategy is called as filter sensitive area reservation.

On the contrary, when we occlude the sensitive area on the image, the filter response distribution will change significantly, which is called filter sensitive area occlusion. In either case, the sensitive area should be as small as possible to avoid the area from converging to the whole image.

Next, two methods will be introduced in detail respectively.

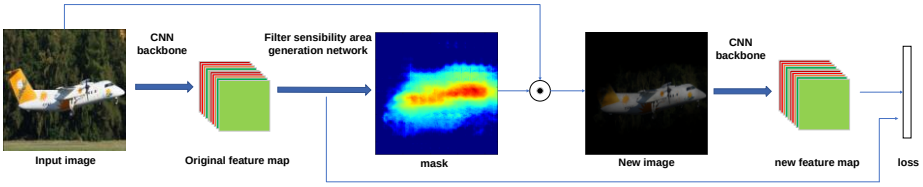

Fig. 2. Framework of FSAGN.

Filter Sensitive Area Reservation. For a filter in certain layer, it represents a specific part for some objects, which means activation of the filter mainly originates from a subarea of the whole input image I. Our target is to find a corresponding mask $M \in [0,1]$ for input image to generate filter sensitive area reservation image I'.

$$I' = I \odot M \tag{1}$$

Given a trained CNN f, the original response of the c-th filter in layer l when inputting the original image is denoted as $r_{l,c} = f(I)[l,c]$. Then the new image I' is fed into the same CNN, and we get the new feature map of the c-th filter in layer l denoted as $r'_{l,c} = f(I)[l,c]$. To find some object parts that contribute the most to the response of specific filter, the optimization goal of FSAGN is to minimize the difference between old feature maps and new feature maps. However, the generated mask is usually sparse. To encourage a compact distribution of mask, we introduce a new constraint in the loss function as follows:

$$\text{Loss}(M) = L_{dif}(r, r') + \lambda * L_{area}(M) \tag{2}$$

where L_{dif} and L_{area} represents the feature map difference loss and the mask generation loss respectively. The feature map difference loss is used to describe the difference between the original and new feature maps. In order to focus on the consistency of response distribution rather than the concrete value of activation, two feature maps are firstly normalized to [0,1], then the feature map difference loss is given by:

$$L_{dif}(r, r') = \|r - r'\|_F \tag{3}$$

To introduce compact distribution constraint on generated mask, $L_{area}(M)$ is formulated as follows [12]:

$$L_{area}(M) = \sum_{(x,y)\in M} m(x,y)\,[(x-t_x)^2 + (y-t_y)^2] \tag{4}$$

where $m(x, y)$ is the concrete value located at (x, y) on mask M, and t_x, t_y is the coordinate corresponding to the location of peak response of the selected filter. With the constraint, the FSAGN will discover the most sensitive part to some filters.

Filter Sensitive Area Occlusion. Contrary to the Filter Sensitive Area Reservation, we select filter sensitive area by observing the change rate of the corresponding feature map after adding occlusion on the input image, which is called Filter Sensitive Area Occlusion. Our target is developing a Filter Sensitive Area Generation Network to find some areas in input image so that when these areas are occluded, the response of the related filter changes dramatically. This method helps us better understand what the filters have learnt and which part they focus on.

The optimization function is given as follows:

$$\text{Loss(M)} = L_{sim}(r, r') + \lambda L_{area}(M) \tag{5}$$

where L_{sim} and L_{area} represents the feature map similarity loss and the mask generation loss respectively. Different from the Filter Sensitive Area Reservation, the similarity between new feature map and original feature map should be as small as possible.

Herein the activation function is selected as ReLU, thus the feature map is non-negative. When occluding some parts of the object, the new response of this filter will drop rapidly and even decrease to zero. Therefore, the similarity loss is designed as follows:

$$L_{sim}(r, r') = \|r'\|_F \tag{6}$$

Meanwhile, the mask generation loss L_{area} keeps the same as that in Filter Sensitive Area Reservation.

3 Experiments

In this section, we will illustrate the efficiency of Filter Sensitive Area Generation Network and show some examples to figure out which parts the specific filters pay attention to. Experiments were conducted on two public datasets, including MNIST and FGVC-Aircraft [11]. Next, more implementation details and experimental results are explained.

3.1 Implementation Details

Before analyzing the sensitive area for some filters, CNN models for object recognition should be trained first. Specifically, a small-scale convolutional neural network is designed for MNIST classification. It has two convolutional layers and two fully

connected layers, taking $28 * 28$ gray images as inputs, as shown in Fig. (a). We achieved an accuracy of 99.18% for MNIST datasets. Then we consider the feature maps of the last convolutional layer as reference. A deconvolutional network is adopted as the Filter Sensitive Area Generation Network, as shown in Fig. (b). It takes the feature maps of the last convolutional layer as inputs, and adds a sigmoid layer to the output, which generates a single-channel mask $M \in (0,1)$.

For FGVC-Aircraft benchmark, a VGG-16 [10] model pre-trained on ImageNet [13] with inputs of size $224 * 224$ are used for better recognition performance, which gained 74% accuracy. We removed the last three fully-connected layers and augmented with a deconvolutional network for filter sensibility area generation. The structure of FSAGN is shown in Fig. .

When training the whole network, the parameters of basic recognition network remain fixed, with only the Filter Sensitive Area Generation Network updated.

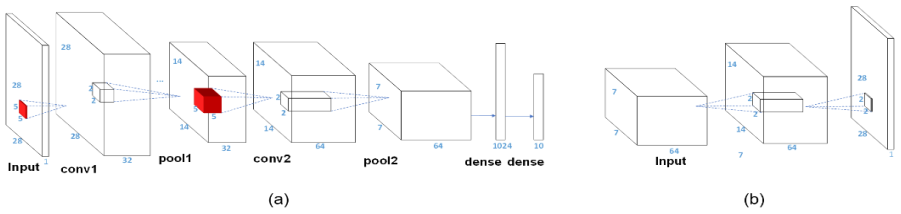

Fig. 3. Network structure for MNIST. (a) Network for classification. (b) Network for sensitive area generation.

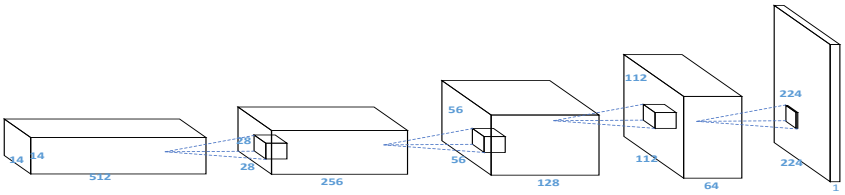

Fig. 4. Structure of FSAGN for VGG-16 trained on FGVC-Aircraft dataset.

3.2 Experiments on MNIST

Filter Selection. We get a collection of filter response distribution in the last convolutional layer tested on randomly selected 1K samples and plot the filter response diagram, which looks like sparse stripes. Following the method described in Section 2.1, the 18-th and 13-th channels are finally chosen as the target filters.

Sensitive Area of the 13-th filter. We adopt filter sensitive area reservation strategy to generate the sensitive areas for the 13-th filter shown in Fig. . The figure shows the original images, original feature maps, generated sensitive areas, new input images after mask operation and new feature maps corresponding to the new image. From the

results, some observations can be made: 1) This filter mainly focuses on a small region of the whole image, which means that removing other parts does not have dramatic effects on the activation of this filter. 2) From the similarity of sensitive areas on different samples, the concerned part of the 13-th filter is the slash of handwritten numeral. After training for MNIST recognition, this filter has learned to capture the inclined part of images.

Sensitive Area of the 18**-th filter.** The same experimental process is applied on the 18-th filter. Results are shown in Fig. . Apparently, we can get similar conclusions with the 13-th filter. However, the 18-th filter tends to pay more attention to the vertical line in images. Therefore, these two filters both have their own sensitive areas, and they detect different parts of input images during object recognition.

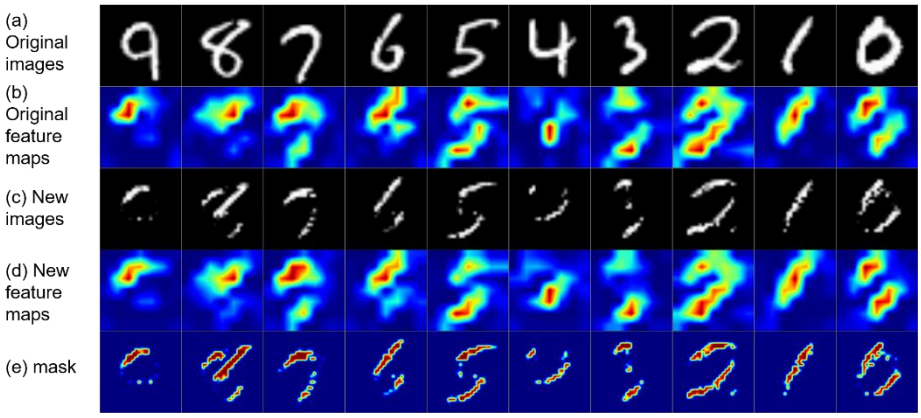

Fig. 5. The sensitive area of the 13-th filter.

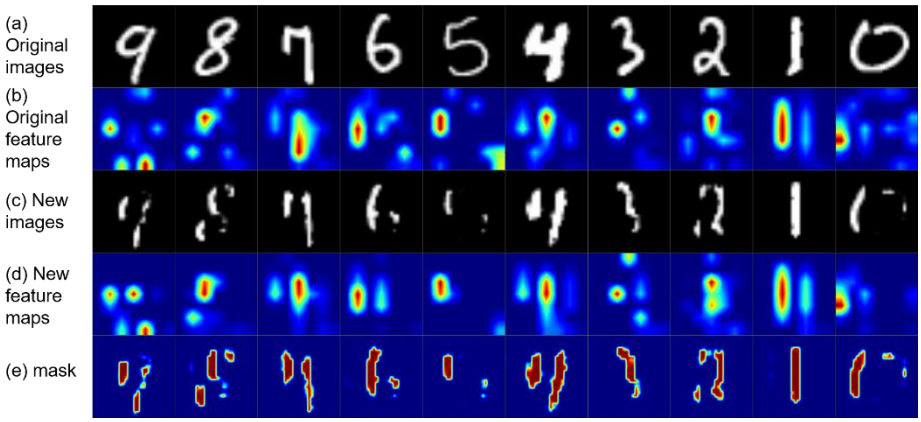

Fig. 6. The sensitive area of the 18-th filter.

3.3 Experiments on FGVC-Aircraft

Filter Selection. We randomly choose 200 aircraft images from FGVC-Aircraft dataset and record feature maps of every channel to form average filter response diagram shown in Fig. . We can observe from the diagram that filters which keeping active all the time (bright vertical lines in the image) account for a rather small part of all channels. In the statistical sense, it is consistent with our intuition that the response of CNN is sparse. Following the method described in Section 2.1, the 26-th and 262-th filter in the last convolutional layer of the VGG-16 model are selected for finding the key parts that the filters represent.

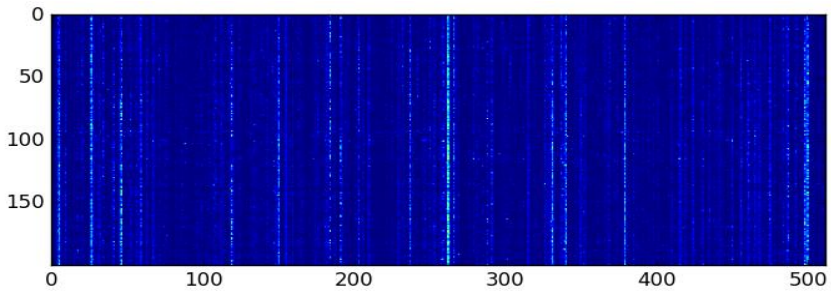

Fig. 7. Response diagram of each filter in VGG-16 for FGVC-Aircraft over multiple samples.

Sensitive Area of the 26-th filter. Firstly, filter sensitive area reservation strategy is adopted to generate the sensitive areas of aircrafts, as shown in Fig. . From the result, the key observations are the following: 1) The generated mask can filter out the background and localize the object coarsely, which is unsupervised without any bounding box labels. 2) Occluding most of the background will take little effect on the activation distribution of this filter. 3) We select the region with the biggest value on the mask (red circle on the new images) and find that this filter tends to be the most sensitive to the nose of aircraft. After training on the FGVC-Aircraft dataset, the 26-th filter has remembered the pattern of aircrafts' nose.

Next, occlusion strategy-based experiments are conducted to figure out which area has dramatical effect on the response of the filter when it is occluded. As shown in Fig. , some interesting observations are made as following: 1) The occlusion region consists of discrete points and lines rather than a whole continuous area. Although human can still recognize the aircraft after such occlusion, the response of the filter weakens rapidly. 2) The occlusion sensitive area of the filter tends to cover the whole object, which is apparently different from that in reservation strategy.

Sensitive Area of the 262-th filter. Similar experiments are repeated for the 262-th filter. From the results of sensitive area reservation strategy (see Fig. 1), we observe

that: 1) Unsurprisingly, this filter has the same preliminary ability to localize the object without supervision. 2) The main part that the 262-th filter focuses on is the fuselage close to the engine, which is different from the 26-th filter. It implies the diversity of filters and these filters have learnt the key parts of aircraft.

From the results of sensitive area occlusion strategy (see Fig. 1), some unexpected observations are the following: 1) The occlusion region degenerates to multiple parallel vertical lines. This confirms that small disturbance can lead to the network's failure.

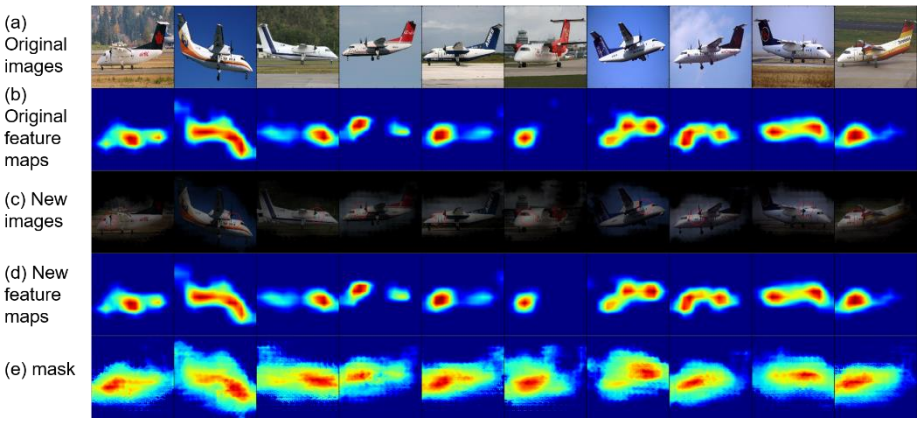

Fig. 8. The sensitive area of the 26-th filter of VGG-16 by reservation strategy.

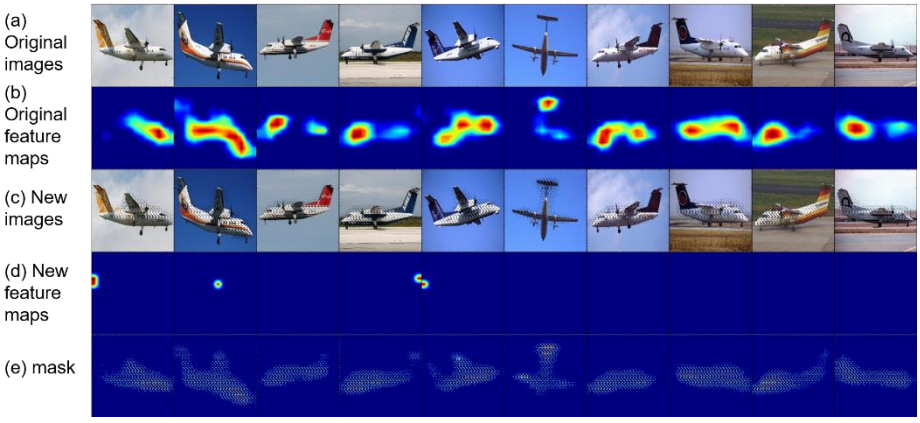

Fig. 9. The sensitive area of the 26-th filter of VGG-16 by occlusion strategy.

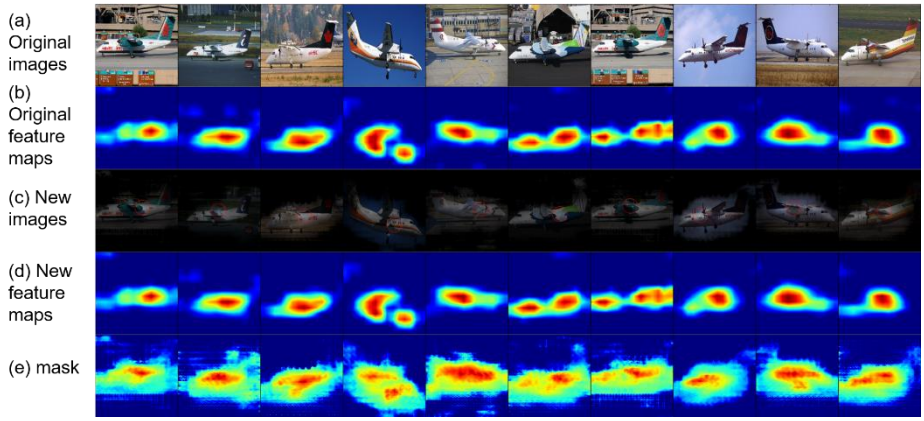

Fig. 10. The sensitive area of the 262-th filter of VGG-16 by reservation strategy.

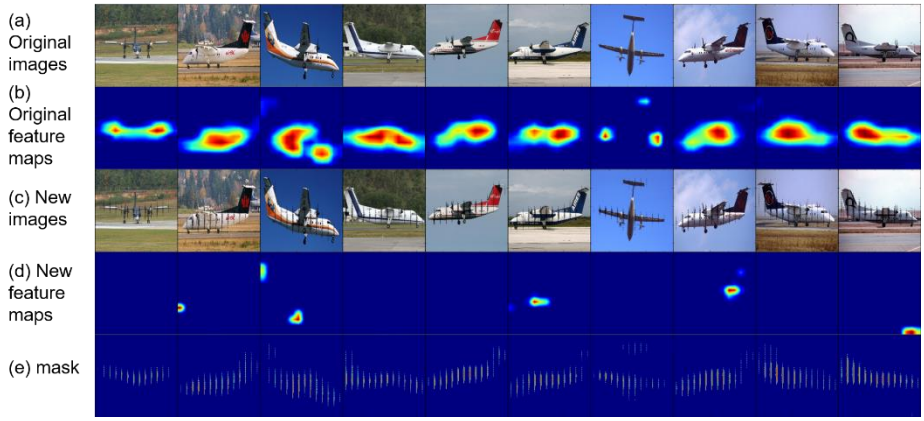

Fig. 11. The sensitive area of the 262-th filter of VGG-16 by occlusion strategy.

4 Discussion

From the experiments on different datasets with different convolution neural networks, some interesting discussions are the following: 1) The activations of filters in CNNs are rather sparse. A small proportion of filters in a layer response strongly, while others keep silent all the time. 2) Each activated filter has a specific response pattern. For simple images and small networks, activation of filters may be sensitive to the vertical line or horizontal line. It indicates that when corresponding parts are occluded, the activation drop rapidly. By contrast, the feature map keeps unchanged when these areas are reserved. For complicated images and large networks, the response pattern of filter in high layer show stronger semantics. For example, a filter can represent the key part of object, like the nose of aircraft. It confirms that deep neural networks learned the key components of objects after training and we can establish a correspondence between parts of objects and filters by the proposed method.

5 Conclusion and Future Work

In this paper, we have proposed a general method to analyze the interpretability of the trained CNNs and better understand what the filters have learnt after training on certain dataset. Based on the observation that some filters could localize the key parts of objects, a Filter Sensitive Area Generation Network is designed and trained to generate the key area that every filter represents. To better describe the correlation between certain filter and the key part, reservation sensibility and occlusion sensibility are proposed respectively. Experiments have shown that the filters response to a certain part of the object and different filters have different fixed response pattern. Besides, small occlusion on the input image will take a significant effect on the activation of filters.

In future work, we will explore classifier sensitive area and make use of this interpretability to generate corresponding adversarial samples or improve the robustness of CNNs by adjusting the sensitive area of filters.

Acknowledgements. This work was supported in part by the National Key Research and Development Program of China (2017YFB1300203), in part by the National Natural Science Foundation of China under Grant 91648205.

References

1. Zhang, N., Donahue, J., Girshick, R., Darrell, T.: Part-Based R-CNNs for Fine-Grained Category Detection. In: European Conference on Computer Vision (ECCV 2014), vol. 8689, pp. 834-849. Springer, Cham (2014)
2. Zhang, X., Xiong, H., Zhou, W., Tian, Q.: Picking Deep Filter Responses for Fine-Grained Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), pp. 1134-1142. Las Vegas, NV (2016)
3. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: International Conference on Neural Information Processing Systems (NIPS 2015), vol. 39, pp. 91-99. MIT Press. (2015)
4. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You Only Look Once: Unified, Real-Time Object Detection. In: Computer Vision and Pattern Recognition (CVPR 2016), pp. 779-788. IEEE Computer Society (2016)
5. Shelhamer, E., Long, J., Darrell, T.: Fully Convolutional Networks for Semantic Segmentation. IEEE Trans. Pattern Analysis & Machine Intelligence 39(4), 640-651 (2017)
6. Noh, H., Hong, S., Han, B.: Learning Deconvolution Network for Semantic Segmentation. In: IEEE International Conference on Computer Vision (ICCV 2015), pp. 1520-1528. IEEE Computer Society. (2015)
7. Zeiler, M. D., Fergus, R.: Visualizing and Understanding Convolutional Networks. In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) Computer Vision – ECCV 2014. Lecture Notes in Computer Science, pp. 818-833. Springer, Cham (2014)
8. Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., Lipson, H.: Understanding Neural Networks Through Deep Visualization. In: International Conference on Machine Learning — Deep Learning Workshop, pp. 12. (2015)
9. Kumar, D., Wong, A., Taylor, G. W., Kumar, D., Wong, A., Taylor, G. W.: Explaining the Unexplained: A CLass-Enhanced Attentive Response (CLEAR) Approach to Understanding

Deep Neural Networks. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR 2017), pp. 1686-1694. IEEE (2017)
10. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint arXiv:1409.1556. (2014)
11. Maji, S., Rahtu, E., Kannala, J., Blaschko, M., Vedaldi, A.: Fine-Grained Visual Classification of Aircraft. arXiv preprint arXiv:1306.5151. (2013)
12. Zheng, H., Fu, J., Mei, T., Luo, J.: Learning Multi-attention Convolutional Neural Network for Fine-Grained Image Recognition. In: IEEE International Conference on Computer Vision, pp. 5219-5227. IEEE Computer Society (2017)
13. Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., Li, F. F.: ImageNet: A Large-Scale Hierarchical Image Database. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2009), pp. 248-255. IEEE (2009)