Revisiting the Deformable Convolution by Visualization

Yuqi Zhang, Yuyang Xie, Linfeng Luo and Fengming Cao
Pingan International Smart City Technologies Ltd., Shenzhen, China

Keywords: Deformable Convolution, Object Detection, Visualization.

Abstract: The deformable convolution improves the performance by a large margin across various tasks in computer vision. The detailed analysis of the deformable convolution attracts less attention than the application of it. To strengthen the understanding of the deformable convolution, the offset fields of the deformable convolution in object detectors are visualized with proposed visualizing methods. After projecting the offset fields to the feature map coordinates, we find that the displacement condenses the features of each object to the object center and it learns to segment objects even without segmentation annotations. Meanwhile, projecting the offset fields to the kernel coordinates demonstrates that the displacement inside each kernel is able to predict the size of the object on it. The two findings indicate the offset field learns to predict the location and the size of the object, which are crucial in understanding the image. The visualization in this work explicitly shows the power of the deformable convolution by decoding the information in the offset fields. The ablation studies of the two projections of the offset fields reveal that the projection in the kernel viewpoint contributes mostly in current object detectors.

1 INTRODUCTION

The wide applications of deformable convolution operation (Dai et al., 2017; Zhu et al., 2019; Yang et al., 2019a; Chen et al., 2020; Thomas et al., 2019; Yang et al., 2019b; Kong et al., 2020; Vu et al., 2019) show its importance in computer vision. The deformable convolution operation is proposed to improve the anchor-based object detector and semantic segmentation initially (Dai et al., 2017; Zhu et al., 2019), then is applied in several anchor-free object detectors (Wang et al., 2019; Yang et al., 2019a; Chen et al., 2020; Yang et al., 2019b; Kong et al., 2020), and recently it helps in modelling the 3D point clouds (Thomas et al., 2019). The deformable convolution operation is believed to have two functionalities (Wang et al., 2019): (1) it improves the representation of features, (2) it encodes the object geometry inside its parameters. The deformable convolution has demonstrated its superior performance in the computer vision tasks. However, the understanding of the deformable convolution is not thoroughly enough, and one challenge remains in the visualization of it. The visualization, as the most straightforward approach, can demystify how the black boxes learns and why the operation works.

For the deformable convolution (Dai et al., 2017), the output feature map y for the location p_0,

$$y(p_0) = \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n + \Delta p_n) \quad (1)$$

where w is the weight of the kernel and x is the value from feature map. As a 3×3 kernel with dilation of 1, $R \in \{(-1,-1),(-1,0),...,(0,1),(1,1)\}$, and Δp_n is the offset with respect to each p_n. The main difference between the deformable convolution and the normal convolution is that the deformable convolution contains the dynamic offset field Δp_n. The dynamic Δp_n enables the network to recognize objects with various geometric variations in images. The dynamic offset field is predicted from the feature maps with a lightweight neural networks. Therefore, the explanation of the deformable convolution relies on the offset field Δp_n. Unfortunately, previous attempts to visualize the deformable convolution failed to concentrate on the offset field due to the high dimensions of the offset field. Although many works provide visualization methods for convolution neural networks such as gradient visualization (Simonyan and Zisserman, 2014), perturbation (Ribeiro et al., 2016), class activation map (Zhou et al., 2016; Wang et al., 2020), and the deconvolutions related methods (Zeiler and Fergus, 2014), the deformable convolutions differ from basic convolutions because of the use of the offset field.

In this work, to understand the deformable convolution further, we visualize the offset field using the

Zhang, Y., Xie, Y., Luo, L. and Cao, F.
Revisiting the Deformable Convolution by Visualization.
DOI: 10.5220/0010200801900195
In *Proceedings of the 10th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2021)*, pages 190-195
ISBN: 978-989-758-486-2

vector analysis. In the visualization, the offset field is projected to the feature map coordinate and the kernel coordinate respectively. The two projections effectively separate the learned information into a global context in image scale and a local context in kernel scale. The visualization results straightforwardly reveal why the deformable convolution surpasses the normal convolution in object detectors: (1) the deformable convolution learns to condense the features of the object to the object center, (2) the deformable convolution learns to segment objects even without segmentation annotations, (3) the deformable convolution learns the size of the object for each feature point. The ablation studies of the two projections of the offset fields are also carried out and show that the projection in the kernel viewpoint has more contributions.

2 RELATED WORK

In the beginning, the deformable convolutions were used in the backbone of the object detectors in (Dai et al., 2017; Gao et al., 2019) by substituting the normal convolution operation with the deformable convolution. Amazed by its prevailing results on the COCO detection benchmark (Lin et al., 2014), many other researchers tried to apply the deformable convolution in the region proposal networks (Wang et al., 2019; Vu et al., 2019) and the bounding box heads (Yang et al., 2019a; Chen et al., 2020; Kong et al., 2020; Yang et al., 2019b). With so many applications of the deformable convolution, the straightforward visualization of it demands more attention.

The receptive fields and the sampling locations are used to visualize the effect of the offset field in (Dai et al., 2017), the visualization connects the sampling locations to the activation units and shows the power of the dynamic sampling locations. After that, the effective receptive fields and effective sampling locations are used to further analyze the deformable convolution in (Zhu et al., 2019; Gao et al., 2019), still, the visualization results fail to focus on the core of the deformable convolution, which is the offset field. In (Wang et al., 2019; Yang et al., 2019a; Chen et al., 2020; Yang et al., 2019b; Kong et al., 2020), the offset field of the deformable convolutions are correlated directly with the size and the position of the object, which cast a light on the potential use of the deformable convolution. To promote the future application of the deformable convolution, we believe the detailed visualization of it will increase the transparency to humans and provide promising insights of its potential use.

3 VISUALIZE THE OFFSET FIELD

For a deformable convolution operation, the $\Delta p_n(p_0)$ has 9 separate offset maps, corresponding to the offset field for 9 kernel points. In this work we focus on a single deformable convolution operation and visualize the offset field inside of it. When a kernel point p_n slides through the feature map, the offset maps in Figure 1(a)-(i) records its learned offset field. In the beginning of the training process, the offset fields are initialized with zero vectors, and after the learning of the detector, most of the offset fields evolve into a quite ordering state as the red arrows in each sub figures have similar vector value.

Apart from visualizing the offset field as a whole, the offset field can be projected into two perspectives. One is from the feature map coordinate and the other one is from the kernel coordinate. In the feature map coordinate, the average displacement $d^f(p_0, p_n)$ over p_n is,

$$d^f(p_0) = \frac{1}{9} \sum_{p_n \in R} \Delta p_n(p_0) \tag{2}$$

where $R \in \{(-1,-1),(-1,0),...,(0,1),(1,1)\}$. Then in the kernel coordinate, the displacement field inside each kernel, the d^k, is,

$$d^k(p_0, p_n) = \Delta p_n(p_0) - d^f(p_0) \tag{3}$$

The distribution of $d^f(p_0)$ over the entire feature map can be seen in Figure 2. Since observing that the arrows of $d^f(p_0)$ point to the objects' centers, we borrow the concept of the divergence from the vector analysis here to visualize the $d^f(p_0)$ better. The divergence $div(p_0)$ of the $d^f(p_0)$ is,

$$div(p_0) = \frac{\partial}{\partial x} d_x^f(p_0) + \frac{\partial}{\partial y} d_y^f(p_0) \tag{4}$$

where $d_x^f(p_0)$ and $d_y^f(p_0)$ are the displacement components in the x and y axis respectively, and $\partial/\partial x$ and $\partial/\partial y$ are the partial derivative operators in the x and y axis respectively. The divergence is a vector operator that operates on a vector field, producing a scalar field representing the volume density of the outward flux of a vector field from an infinitesimal volume around a given point. The positive value means the vector field outward flux and the negative value means the inward flux.

Figure 2 plots the distributions of the d^f for the feature maps, which are learned in the Faster RCNN object detector with deformable convolutions in the backbone as stated in (Gao et al., 2019) and trained with COCO bounding boxes annotations only. Two fundamental insights can be found in them. The first

Figure 1: The offset fields for 3×3 kernel points for a deformable convolution with respect to an example image in the validation dataset from COCO. The small arrows in (a)-(i) represent the offset field for each kernel point and the thick arrows in central show the trends of offset for each kernel point.

insight is that the arrows d^f from Equation 2 are converging to the centers of each objects, which are represented by the negative value of divergence from Equation 4 and the negative divergence from the feature points are shown with red masks in Figure 2. Converging feature points to the object center is a key function of the deformable convolution. The converging of the feature points can be understood as an information condensation process when the feature map stride is smaller than the object size. The information from the feature map with only a part of a zebra in it is less than the information from the feature map with the whole zebra. Therefore, after the deformable convolution, the feature maps are learned with the condensed information which helps the network to understand the image. The second insight is that though the network is trained without any mask information, the visualization of the offset field shows that it is able to learn the segmentation inside the deformable convolution operation. In different scale of the feature maps, the deformable convolution focuses on different scale of information. For example, the deformable convolution in the feature map generated with smaller stride locates the strips of the zebra and the branches of the tree in Figure 2(c) while it locates the zebra and the tree as a whole with larger stride, as shown in Figure 2(d).

To find out what information is inside the kernel viewpoint, Figure 3 is shown for the learned distribution of the d^k. The radius of the circle is calculated as $r \propto \sqrt{width \times height}$ where *width* is the offset difference in x axis between left and right kernel points, and *height* is the offset difference in y axis between top and bottom kernel points. Therefore, the areas of each circles in Figure 3 indicate the expansion or shrinkage status of the d^k. In other words, the circles reflect how large each 3×3 kernel wants to cover. Figure 3 shows that the d^k has learned the size of the object for each 3×3 kernel since the radius of the circle accords with the size of the object behind the feature point. In detail, in Figure 3(a) with a dog, a person, a Frisbee, and a tree, the largest circle is inside the tree while the smallest circle is inside the Frisbee, which corresponds to the sizes of the objects, $Size_{tree} > Size_{Frisbee}$. It should be noted that not all feature points inside the object have the same size, especially when the object is across many feature points. Not every feature point inside the object

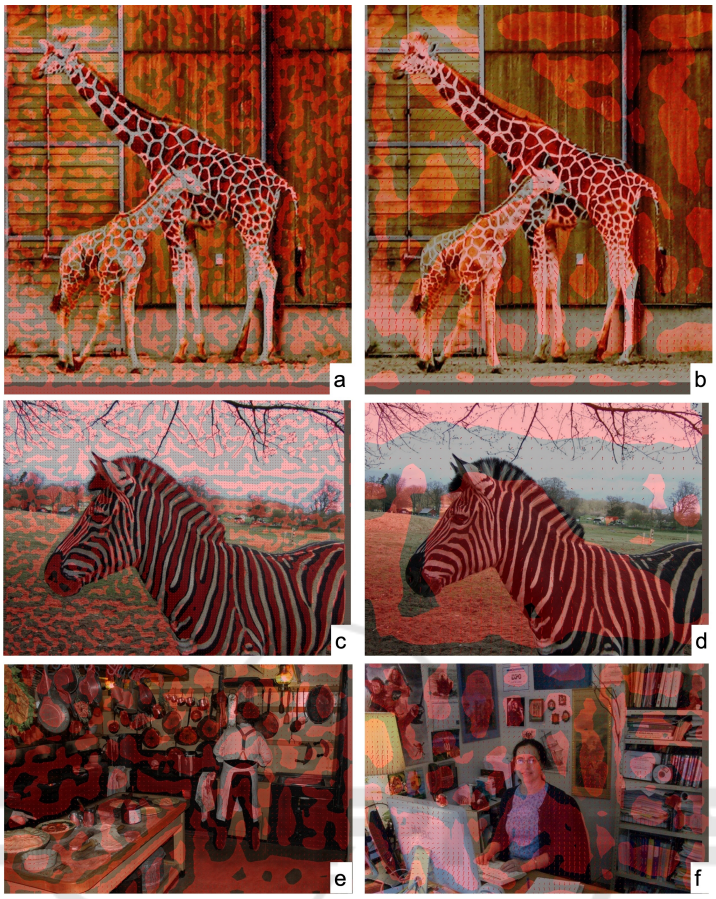

Figure 2: The visualization of the offset field in deformable convolution for the feature map point of view. (a) and (c) are from stage 3, (b), (e), and (f) are from stage 4, and (d) is from stage 5. All are examples of displacement d^f converging to the centers of the object.

reflects the real size of the object except the feature point with the largest circle.

The separate visualizations of the offset field into the feature map coordinate and the kernel coordinate straightforwardly show that the offset field has learned the position information and the size information of the objects in the basic learning of the object detectors without any direct supervision on the offset fields. The position of objects is global information, so it can be represented by $d^f(p_0)$ as shown in Figure 2. In the meanwhile, the size of the object is rather a local information, as shown in Figure 3.

4 ABLATION STUDY OF d^f AND d^k

We trained our detectors on the 118k images of the COCO 2017 train dataset and evaluated the performance on the 5k images of the validation dataset. The standard mean average-precision over IoU=0.5:0.05:0.95 is used to measure the performance of the detectors.

The Faster R-CNN and the Mask R-CNN are chosen as two baselines representing the use of the deformable convolution in object detectors. The implementation is based on the MMDetection framework. The Faster R-CNN detector is trained with stochastic gradient descent optimizer over 2 GPUs with a total batch size of 8 images per mini batch. The Mask R-CNN detector is trained over 2 GPUs with a total batch size of 4 images per mini batch. The "1×" schedule is adopted for learning rate. No test time augmentation is used and non-maximum suppression IoU threshold of 0.5 is employed for both detectors.

Both the d^f and the d^k adjust how the convolution sees the feature map. The d^f changes the receptive field globally while the d^k changes it locally inside each kernel. To investigate the performance of the d^f and the d^k in the offset fields of deformable convolution, experiments are conducted in Mask RCNN and Faster RCNN detectors with either d^f or d^k. The

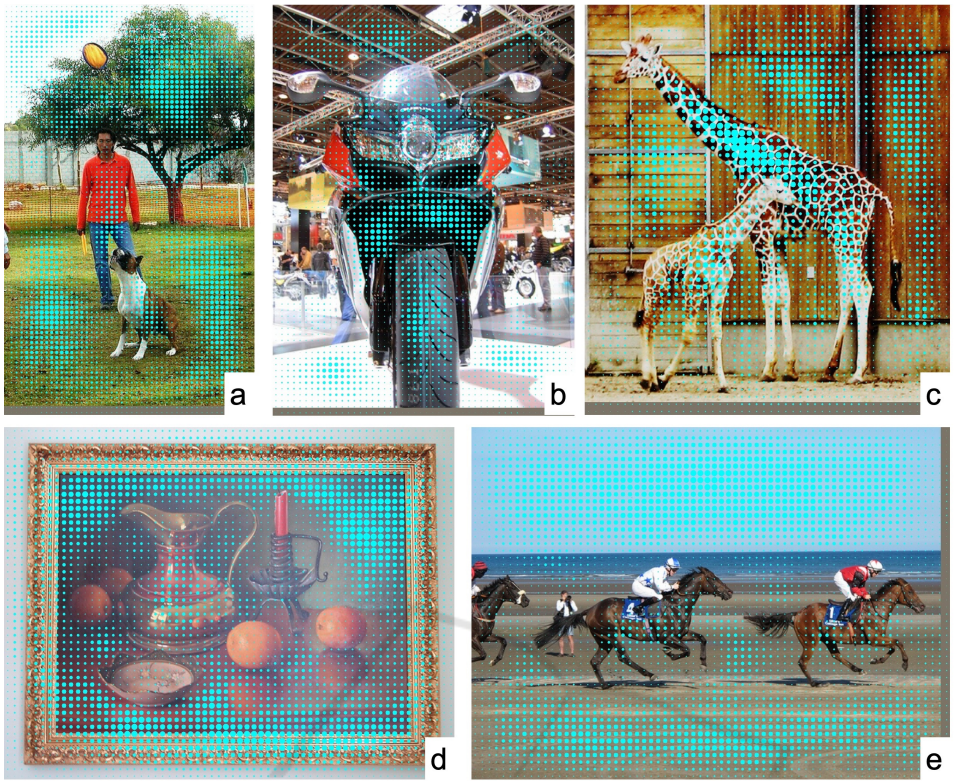

Figure 3: The visualization of the offset field in deformable convolution for the kernel point of view. The size of the circle for each point represents the expansion or shrinkage inside each, in other words, the size of the circle represents the area of the offset field inside each 3×3 kernel.

detectors are still trained end to end, and the new offset field are calculated by Equation 2 and Equation 3 respectively. The performance are shown in Table 1. The major contribution of deformable convolution comes from the d^k while the d^f also improves the performance compared with the baseline detectors without the deformable convolution. Although from the visualization of d^k and d^f in 3 and 2, both components have learned the information of the object, the ablation studies show that the local information encoded in d^k is more significant than the d^f and the global information in d^f may not be well utilized in the following parts of the networks.

Table 1: The effect of the d^f and the d^k.

baseline	d^f	d^k	AP_{bbox}	AP_{segm}
Mask RCNN			0.382	0.347
	✓		0.388	0.351
		✓	0.418	0.375
	✓	✓	0.420	0.376
Faster RCNN			0.374	NA
	✓		0.380	NA
		✓	0.413	NA
	✓	✓	0.416	NA

5 FUTURE WORK

The visualization and analysis in this work show the effects of the deformable convolution in predicting the size and the position of the object. It should be noted that the deformable convolution learns the two sources of information without direct supervision and the automatically learned information is all from its natural usage of the offset field. Therefore, the future work includes (1) the study of the behaviour of the deformable convolution when direct supervision is enforced on the offset field; (2) utilization of the two properties of the offset fields in the prediction branches of bounding box and mask.

6 CONCLUSION

In summary, the detailed visualization and analysis of the offset field are made to promote the straightforward understanding of the deformable convolution. For the object detector, even a single deformable convolution has convey the information of object position

and object size. We directly prove this by visualizing the offset field in the feature map viewpoint and the kernel viewpoint separately. The position of the object is global information which is in the feature map viewpoint while the size of the object is local information which is in the kernel viewpoint. The effect of the offset field in the two viewpoints is investigated separately and the results show the components in the kernel viewpoint improves the deformable convolution more.

REFERENCES

Chen, Y., Zhang, Z., Cao, Y., Wang, L., Lin, S., and Hu, H. (2020). Reppoints v2: Verification meets regression for object detection. *arXiv preprint arXiv:2007.08508*.

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

Gao, H., Zhu, X., Lin, S., and Dai, J. (2019). Deformable kernels: Adapting effective receptive fields for object deformation. *arXiv preprint arXiv:1910.02940*.

Kong, T., Sun, F., Liu, H., Jiang, Y., Li, L., and Shi, J. (2020). Foveabox: Beyound anchor-based object detection. *IEEE Transactions on Image Processing*, 29:7389–7398.

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

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., and Guibas, L. J. (2019). Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6411–6420.

Vu, T., Jang, H., Pham, T. X., and Yoo, C. (2019). Cascade rpn: Delving into high-quality region proposal network with adaptive convolution. In *Advances in Neural Information Processing Systems*, pages 1432–1442.

Wang, H., Wang, Z., Du, M., Yang, F., Zhang, Z., Ding, S., Mardziel, P., and Hu, X. (2020). Score-cam: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 24–25.

Wang, J., Chen, K., Yang, S., Loy, C. C., and Lin, D. (2019). Region proposal by guided anchoring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2965–2974.

Yang, Z., Liu, S., Hu, H., Wang, L., and Lin, S. (2019a). Reppoints: Point set representation for object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9657–9666.

Yang, Z., Xu, Y., Xue, H., Zhang, Z., Urtasun, R., Wang, L., Lin, S., and Hu, H. (2019b). Dense reppoints: Representing visual objects with dense point sets. *arXiv preprint arXiv:1912.11473*.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929.

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