Deformable Convolution

Abstract

CNN은 고정된 모듈의 구조로 인하여 기하하적 모양에 제한이 있습니다. 이 논문에서 우리는 CNN의 모델 수용성을 확장시켜줄 두개의 새로운 모델인 deformable convolution과 deformable ROI pooling을 제안합니다. 둘 다 offset을 이용하여 위치 샘플링을 보강하는 기법을 베이스로 합니다. 새로운 모듈은 쉽게 존재하는 CNN을 대체할 수 있으며 쉽게 end-to-end로 학습이 가능합니다. 많은 실험은 우리의 접근의 성능을 검증합니다. 처음으로 우리는 공간 변형을 CNN에서 학습하는 것이 object detection과 semantic segmentation에서 효과적인 것을 보여줍니다.

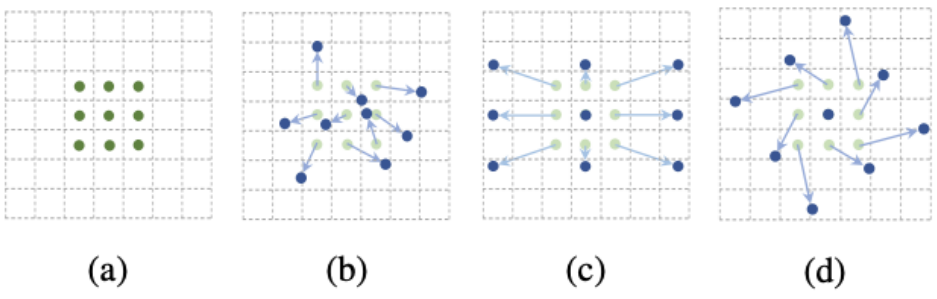
Introduction

시각인식에서 중요한 문제는 어떻게 기하학적 다양성과 모델의 크기, 포즈, 관점 그리고 일부분을 수용하는가 입니다. 대게 두개의 방법이 있습니다. 첫번째는 충분히 다양한 학습 데이터셋 준비입니다. 이는 존재하는 데이터를 다양한 데이터 증강 기법을 통해 해결할 수 있습니다. 데이터를 통해 잘 학습할 수 있지만 이는 많은 학습과 복잡한 모델을 필요로 합니다.

두번째 방식은 transformation-invariant 피쳐와 알고리즘을 사용하는 것이 있습니다. 대표적으로 SIFT, sliding window가 있습니다. 여기에는 두가지 문제가 있는데, 기하하적 모형이 알려지거나 고정되어야합니다. 사전 지식은 데이터를 증강하는데, features와 알고리즘을 디자인하는데 사용됩니다. 이러한 가정은 새로운 테스크(알려지지 않은 모형)에 일반화를 할 수 없습니다. 그리고 사전 지식이 있더라도 복잡한 모형의 경우 feature와 알고리즘을 제작하는 것은 불가능합니다.

최근 CNN은 시각 인식 테스크에서 많은 성공을 거두었습니다. 그러나 기하학적 다양성을 해결하기 위해서는 어려움이 있고, 해결하기 위해서라도 큰 모델을 사용해야 한다는 것과 알려지지 않은 모형에 대한 어려움이 있습니다. 이러한 한계는 CNN의 고정된 모듈 구조에서 있습니다. Convolution은 고정된 위치에서 sample을 추출하게 됩니다. pooling layer는 고정된 비율로 해상도를 감소시킵니다. 이들은 기하하적 모양을 다루는데 결함이 존재합니다.

예를 들어, 초기 CNN layers는 공간적 위치에 대해 설명을 하게 되는데, 고정적인 위치의 Convolution과 pooling layer는 기하하적 다양성으로 인해 다른 위치가 물체와 연관지어질 수 있습니다.

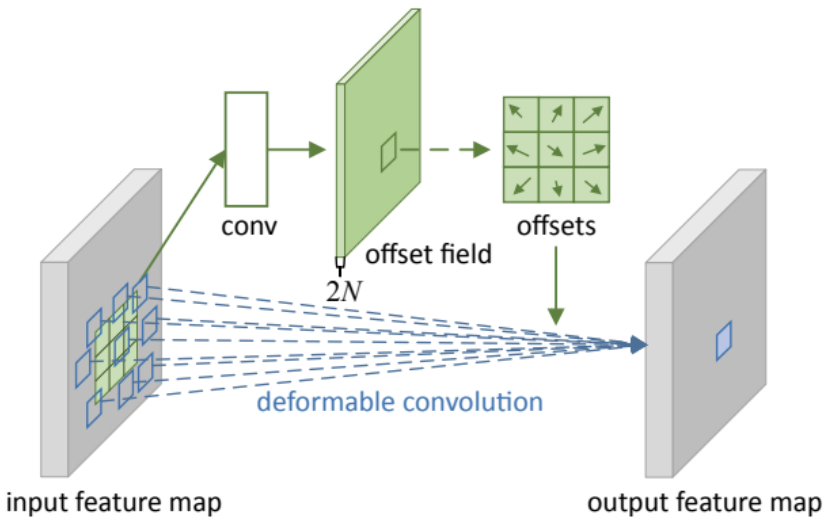


(a) 기존의 standard convolution sampling

(b), (c), (d) deformable convolution offset sampling (object에 따라서 다양한 모습으로 변화한다.)

아래의 두 모듈은 offset learning을 위해 적은 파라미터를 추가한다. 또한, 이전의 CNN을 대체하며 end-to-end 학습이 가능하다.

Deformable convolution



deformable convolution은 2D offset을 기존 CNN의 grid sampling에다가 더한 것이다. 이는 기존의 sampling 형식에서 자유롭게 해준다. offset은 이전의 피쳐맵으로부터 추가적인 convolution layer를 통해 학습이 진행된다. 그래서 deformable convolution은 입력된 피쳐에 따라 학습이 진행됩니다.

ex) $R = (-1,-1),(-1,0),(-1,1),\ldots,(0,1),(1,1)$ 3*3kernel , dilation = 1이라 할 때

Standard CNN | $y(p0) = \sum w(pn) * x(p0 + pn)(pn \in R\,)$

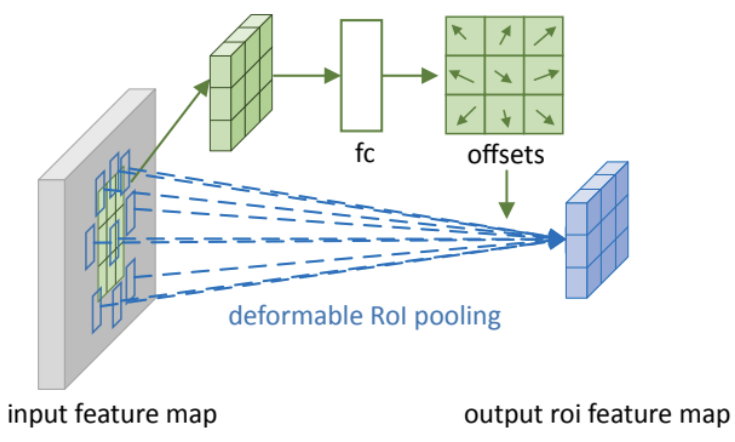
Deformable CNN | $y(p0) = \sum w(pn) * x(p0 + pn + \triangle pn)(pn \in R\,)$

offset은 convolution layer에 의해 학습이 진행된다. 이 때 convolution layer를 지나면 resolution이 같으며 채널이 2배가 되는 피쳐맵을 생성하는데, 이는 offset의 상하, 좌우를 의미한다.

여기서 사용하는 convolution layer는 3kernel 1padding을 사용하여 input과 output이 동일한 피쳐맵이 생성된다.

이렇게 생성된 offset은 위의 Deformable CNN 공식의 $\triangle pn$이 된다. $\triangle pn$의 경우 정수가 아닌 경우가 많은데, 이는 blinear interpolation으로 도출한다.

Deformable RoI pooling



RoI pooling은 오브젝트 디텍션에서 region proposal을 고정된 벡터의 크기로 변환시켜주기 위해 사용한다. 위의 Deformable convolution과 비슷한 양식을 따르며, 다른점이 있다면 offset을 학습할 때 fully connected layer를 사용하여 진행한다는 것이다.

deformable roi pooling은 offset을 각각의 bin position에 더해주는 것이다. 비슷하게 offset은 이전 feature map과 RoIs를 진행함에 따라 객체에 마다 다른 모양으로 학습된다.

Deformable ConvNets

위 두가지 기법은 기본 버젼에서는 같은 인풋과 아웃풋의 사이즈를 가진다. 그래서 기존 CNN을 쉽게 대체할 수 있다. 학습시에는, offset vector는 0의 값을 시작으로 학습이 진행된다.

우리는 deformable convnet을 기존의 state-of-the-art model에 통합시키기 위해 두 가지 방식을 따랐다. 먼저 전체 이미지를 deep filly convolution layer에 넣었다. 두번째로 간단한 task 특정 네트워크로 결과를 생성한다.

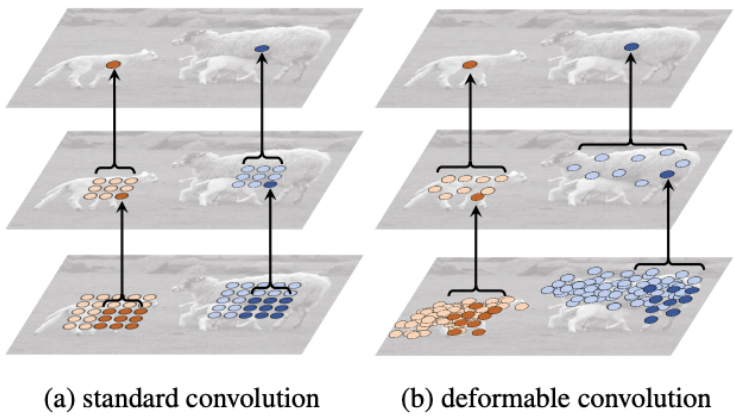
Deformable convolution for Feature Extraction

Feature extraction을 위해서 sota 모델인 ImageNet 데이터셋으로 학습된 ResNet-101과 Inception-ResNet을 사용한다. average pooling과 fc layer를 삭제하였다. 그리고 채널을 변경하기 위해 1*1 convolution layer를 추가하였고, 몇몇 파라미터를 변경하였다. 그리고 model에 마지막 레이어 부근에 deformable convolution layer로 수정하였으며 마지막에서 1, 2, 3, 6개를 바꿔가며 성능을 확인하였고, 3개, 6개 를 사용했을 때 가장 좋은 성능을 보였다.

usage of deformable convolution (# layers)	DeepLab		class-aware RPN		Faster R-CNN		R-FCN	
	mIoU@V (%)	mIoU@C (%)	mAP@0.5 (%)	mAP@0.7 (%)	mAP@0.5 (%)	mAP@0.7 (%)	mAP@0.5 (%)	mAP@0.7 (%)
none (0, baseline)	69.7	70.4	68.0	44.9	78.1	62.1	80.0	61.8
res5c (1)	73.9	73.5	73.5	54.4	78.6	63.8	80.6	63.0
res5b,c (2)	74.8	74.4	74.3	56.3	78.5	63.3	81.0	63.8
res5a,b,c (3, default)	**75.2**	**75.2**	74.5	57.2	78.6	63.3	81.4	64.7
res5 & res4b22,b21,b20 (6)	74.8	75.1	**74.6**	**57.7**	**78.7**	**64.0**	**81.5**	**65.4**

Table 1: Results of using deformable convolution in the last 1, 2, 3, and 6 convolutional layers (of 3×3 filter) in ResNet-101 feature extraction network. For *class-aware RPN*, *Faster R-CNN*, and *R-FCN*, we report result on VOC 2007 test.

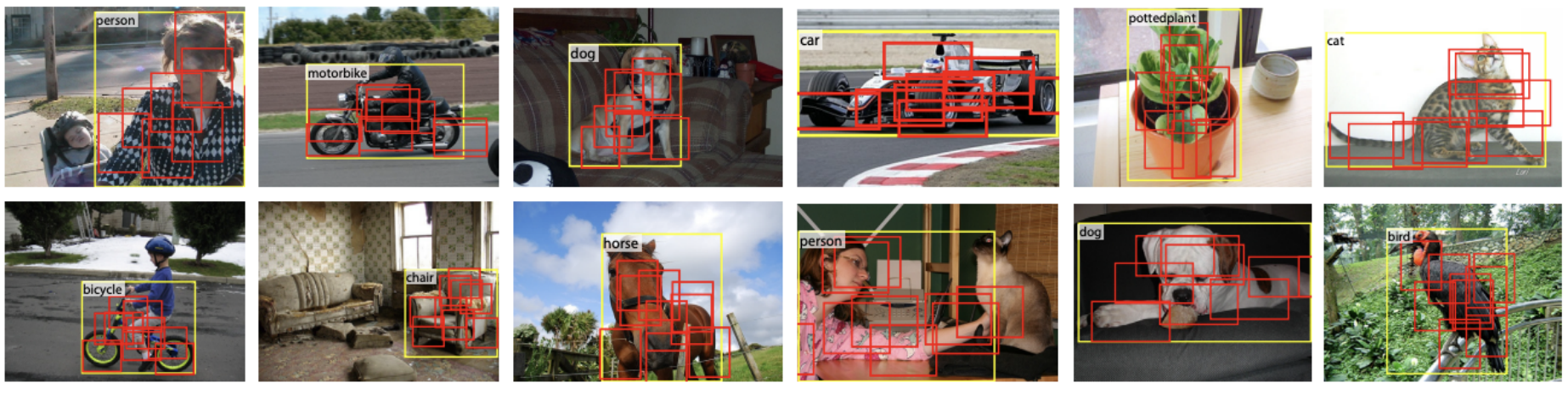
Understanding Deformable ConvNets



(a) standard convolution (b) deformable convolution

Deformable convolution이 쌓이게 되면 그 효과를 위의 사진에서 확인할 수 있다. 왼쪽과 다르게 오른쪽의 object를 표현하는데 사용된 이미지의 정보는 offset을 학습하면서 좀 더 객체의 모양에 알맞게 들어간 것을 확인할 수 있다.

추가적으로 RoI pooling에서도 위와 같은 효과를 확인할 수 있다. 아래 사진을 보면 grid structure가 더 이상 고정되어 있지 않으며, 객체가 아닌 부분에 대해 탐지 되지 않는다.



In Context of Related Works

Spatial Transform Networks (STN)

딥러닝 프레임워크에서 처음으로 데이터에서 모양 변경을 하여 진행한 네트워크이다. spatial transformation 기법을 적용하여 원하는 부분을 잘라내 객체 부분에 해당하여 학습을 집중적으로 한다.

Expriments

method	backbone architecture	M	B	mAP@[0.5:0.95]	mAP^r@0.5	mAP@[0.5:0.95] (small)	mAP@[0.5:0.95] (mid)	mAP@[0.5:0.95] (large)
class-aware RPN	ResNet-101			23.2	42.6	6.9	27.1	35.1
Ours				**25.8**	**45.9**	**7.2**	**28.3**	**40.7**
Faster RCNN	ResNet-101			29.4	48.0	9.0	30.5	47.1
Ours				**33.1**	**50.3**	**11.6**	**34.9**	**51.2**
R-FCN	ResNet-101			30.8	52.6	11.8	33.9	44.8
Ours				**34.5**	**55.0**	**14.0**	**37.7**	**50.3**
Faster RCNN	Aligned-Inception-ResNet			30.8	49.6	9.6	32.5	49.0
Ours				**34.1**	**51.1**	**12.2**	**36.5**	**52.4**
R-FCN	Aligned-Inception-ResNet			32.9	54.5	12.5	36.3	48.3
Ours				**36.1**	**56.7**	**14.8**	**39.8**	**52.2**
R-FCN	Aligned-Inception-ResNet	✓		34.5	55.0	16.8	37.3	48.3
Ours		✓		37.1	57.3	18.8	39.7	52.3
R-FCN		✓	✓	35.5	55.6	17.8	38.4	49.3
Ours		✓	✓	**37.5**	**58.0**	**19.4**	**40.1**	**52.5**

Conclusion

이 논문에서는 간단하고 효율적이며 깊고 ene-to-end deformable Convnets을 제시한다. 우리는 처음으로 CNN에서 공간 변형이 가능하며 효과적인 것을 증명했다.