Paper Review 3

Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks

0. Abstract

객체 탐지(Object Detection)의 SOTA 모델인 SPPnet과 Fast R-CNN 모델은 영역 제안(region proposal) 알고리즘을 사용한다. 이들은 실행 시간을 줄였지만, 영역 제안 계산 단계에서 병목(bottleneck) 현상이 있다는 단점이 있다.

본 논문은 객체 탐지 네트워크와 합성곱 특징들을 공유하여 cost가 거의 없이 영역 제안을 가능하게 하는 RPN(Region Proposal Network)을 소개한다. RPN은 객체의 경계(와 사물 유무(Objectness score)를 동시에 예측하는 완전 연결 네트워크이다. 또한, RPN과 Fast R-CNN을 결합하여 단일 네트워크를 만들었으며, 이는 GPU에서 매우 깊은 VGG-16 모델을 사용하여 우수한 객체 탐지 성능을 보였다.

1. Introduction

Fast R-CNN은 합성곱을 공유하여 R-CNN의 단점인 속도를 빠르게 개선한 모델이다. 하지만 이 모델은 기본적으로 GPU를 이용하지만, 영역 제안은 CPU에서 수행하기 때문에 영역 제안 단계에서 병목 현상이 일어난다.

본 논문은 객체 탐지 네트워크와 합성곱 계층을 공유하는 영역 제안 네트워크인 RPN을 사용하여 계산 단계에서 cost 가 거의 들지 않게 하였다. 이를 통해 이미지당 10ms만큼 제안 속도를 크게 개선하였다. Convolution feature map 위에 두 개의 추가적인 레이어를 추가하여 RPN을 구성하였다.

RPN은 FCN의 한 종류로서 end-to-end 방식으로 훈련이 가능하다. 이는 제안 작업에 대해 fine-tuning을 한 후, 물체 탐지를 위해 fine-tuning을 하는 것을 번갈아 진행함으로써, 두 작업 간 Convolution feature가 공유되는 통합된 단일 네트워크를 생성하는 방식이다.

2. Related work

- Selective Search, CPMC, MCG
- Objectness in windows, EdgeBoxes
- Selective Search object detectors, R-CNN, Fast R-CNN
- OverFeat method
- MultiBox method

3. Faster R-CNN

Faster R-CNN은 2가지 모듈로 구성된다.

1. 영역을 제안하는 Conv 네트워크 (Deep fully convolutional network that proposes region)
2. 제안된 영역을 활용하는 Fast R-CNN detector(Fast R-CNN detector that uses proposed regions)

이러한 Faster R-CNN의 구조는 아래와 같다.

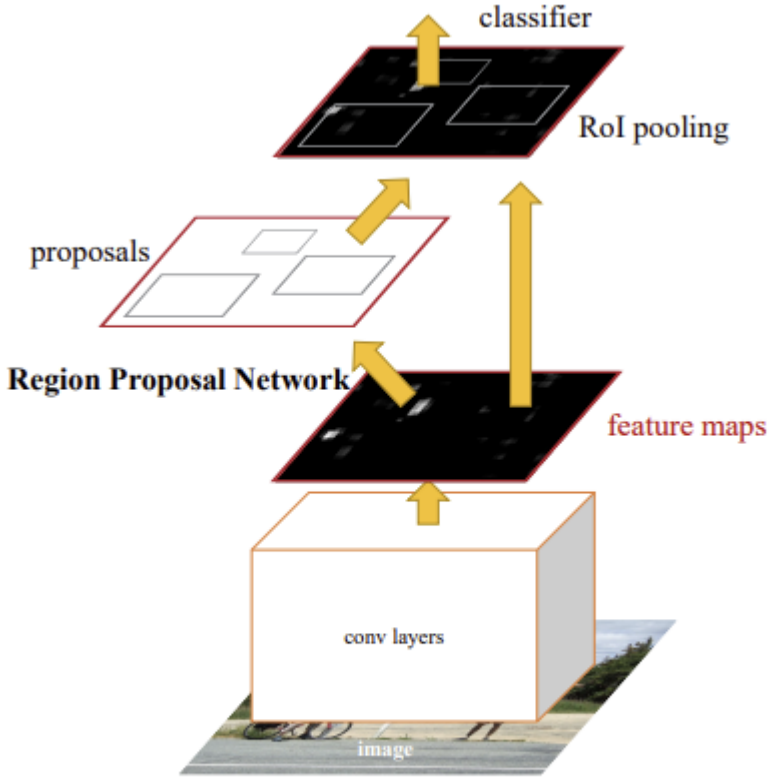


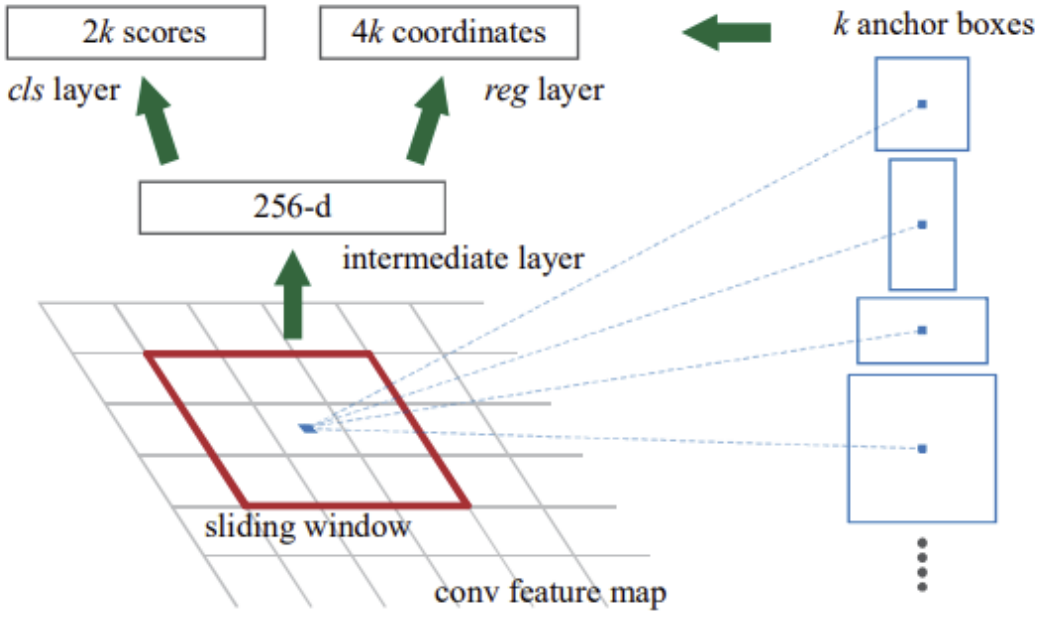
Figure 2: Faster R-CNN is a single, unified network for object detection. The RPN module serves as the 'attention' of this unified network.

Faster R-CNN은 우선 이미지를 입력받아 conv layers에서 합성곱 연산을 수행한다. RPN은 Convolution feature map을 기반으로 영역들을 제안하고, RoI(Region of Interest) pooling layer에서 feature map과 제안된 영역들을 입력 받아 classifier인 Fast R-CNN에 전달하는 것이다. 이 과정을 통해 RPN은 Fast R-CNN detector가 어디에 주목해야 하는 지를 알려줌으로써 객체 탐지를 수행한다.

이는 Faster R-CNN이 Attention mechanism 과 유사하다고 할 수 있다.

3.1 Region Proposal Networks

RPN은 영역 제안 경계 박스를 출력하기 위해 Sliding windows 방식을 적용한다. 각 슬라이딩 윈도우의 위치마다 최대로 예측할 수 있는 영역 제안의 개수를 k개라고 하면, 회귀 레이어(reg layer)는 4k개의 coordinates, 분류 레이어(cls layer)는 2k개의 scores을 가진다. 이러한 k개의 제안은 k개의 reference box인 anchors로 매개변수화 된다.



그림과 같이 앵커는 sliding window의 중앙에 위치하고, 여러 크기의 anchor box를 이용한다. 이는 각 sliding 위치마다 만들어내기 때문에 WxHxk개의 앵커를 생성한다. 본 논문에서는 3개의 scale, 3개의 ratio를 이용한 k=9의 anchor box를 만들었다. 3x3 sliding을 진행하여 WxHx9의 앵커가 생성되었다.

이러한 sliding windows 방식은 앵커와 제안을 계산하는 함수에 대해 이동 불변성(translation invariant)를 가진다. 따라서 출력 레이어의 차원이 작고, proposal layer의 매개변수의 개수가 더 적기 때문에 과적합의 위험이 적다.

Multi-Scale Anchors as Regression Referneces

Multi-scale prediction에서는 image/feature pyramids method와 pyramid of filters method가 유명하다. 본 논문에서 사용된 anchor-based method는 Bounding boxes에 대한 분류와 회귀가 모두 anchor box 안에서 진행되기 때문에, 비용 면에서 훨씬 효율적이다.

3.2 Sharing Features for RPN and Fast R-CNN

RPN과 Fast R-CNN이 Convolution layer를 공유하기 위해서, Fast R-CNN과 RPN이 Convolution feature를 공유하면서 분리해서 사용할 수 있는 Alternating training 방식을 사용한다.

4-Step Alternating Training

1. RPN을 학습한다.
2. 1단계에서 생성된 제안을 사용하여 Fast R-CNN에 의한 별도의 검출 네트워크를 학습한다.
3. Detection 네트워크로 RPN 학습을 초기화한다. 이때 공유된 conv layer는 고정하고, RPN에만 고유한 레이어를 fine-tuning 하여 conv layer를 공유한다.
4. 공유된 conv layer를 고정한 채로 Fast R-CNN의 fc layer를 fine-tuning 한다.

→ RPN과 Fast R-CNN은 같은 Convolution layer를 공유하면서 단일 네트워크를 생성한다.

4. Experiments

정확도 평가에 mAP(mean Average Precision)을 사용하였다.

4.1 PASCAL VOC

Table 2: Detection results on **PASCAL VOC 2007 test set** (trained on VOC 2007 trainval). The detectors are Fast R-CNN with ZF, but using various proposal methods for training and testing.

train-time region proposals		**test-time region proposals**		
method	# boxes	method	# proposals	mAP (%)
SS	2000	SS	2000	58.7
EB	2000	EB	2000	58.6
RPN+ZF, shared	2000	RPN+ZF, shared	300	**59.9**
ablation experiments follow below				
RPN+ZF, unshared	2000	RPN+ZF, unshared	300	58.7
SS	2000	RPN+ZF	100	55.1
SS	2000	RPN+ZF	300	56.8
SS	2000	RPN+ZF	1000	56.3
SS	2000	RPN+ZF (no NMS)	6000	55.2
SS	2000	RPN+ZF (no *cls*)	100	44.6
SS	2000	RPN+ZF (no *cls*)	300	51.4
SS	2000	RPN+ZF (no *cls*)	1000	55.8
SS	2000	RPN+ZF (no *reg*)	300	52.1
SS	2000	RPN+ZF (no *reg*)	1000	51.3
SS	2000	RPN+VGG	300	59.2

→ RPN+ZF, shared모델이 SS(Selective Search)와 EB(EdgeBoxes)보다 제안 개수는 훨씬 적지만 정확도는 더 높은 것을 알 수 있다. 300개의 제안으로 59.9%의 정확도를 보인다.

Table 5: **Timing** (ms) on a K40 GPU, except SS proposal is evaluated in a CPU. "Region-wise" includes NMS, pooling, fully-connected, and softmax layers. See our released code for the profiling of running time.

model	system	conv	proposal	region-wise	total	rate
VGG	SS + Fast R-CNN	146	1510	174	1830	0.5 fps
VGG	RPN + Fast R-CNN	141	**10**	47	**198**	**5 fps**
ZF	RPN + Fast R-CNN	31	**3**	25	**59**	**17 fps**

→ SS+Fast R-CNN 에 비해 RPN+Fast R-CNN이 훨씬 높은 fps를 보여주는 것을 알 수 있다.

4.2 MS COCO

Table 11: Object detection results (%) on the **MS COCO** dataset. The model is VGG-16.

			COCO val		COCO test-dev	
method	proposals	training data	mAP@.5	mAP@[.5, .95]	mAP@.5	mAP@[.5, .95]
Fast R-CNN [2]	SS, 2000	COCO train	-	-	35.9	19.7
Fast R-CNN [impl. in this paper]	SS, 2000	COCO train	38.6	18.9	39.3	19.3
Faster R-CNN	RPN, 300	COCO train	41.5	21.2	42.1	21.5
Faster R-CNN	RPN, 300	COCO trainval	-	-	**42.7**	**21.9**

→ Faster R-CNN이 Fast R-CNN보다 더 정확도가 높다는 것을 알 수 있다.

4.3 From MS COCO to PASVAL VOC

PASCAL VOC 에 없는 MS COCO의 카테고리는 무시하고, 20개의 카테고리와 배경만을 대상으로 softmax layer에 통과시킴으로써 객체 탐지를 수행하였다. 그 결과 PASCAL VOC 2007 test set에서 fine-tuning 없이 정확도 76.1%를 기록하였다. Fine-tuning 을 거친 후 정확도는 78.8%로 향상되었고, 각 개별 카테고리에서 가장 좋은 AP를 기록하였다.