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

https://arxiv.org/pdf/1506.01497

0. Introduction

- 기존 객체 탐지 모델들은 지역 제안(region proposal) 알고리즘을 외부에 의존했음
- 제안 속도가 전체 탐지 파이프라인의 병목이 됐음
- 본 논문은 Region Proposal Network (RPN) 을 도입해, 이미지 전체 convolution feature를 공유하면서 region 제안을 거의 비용 없이 수행 가능하게 함
- 핵심 기여
  - RPN과 detection 네트워크 간 feature 공유 설계
  - end-to-end 학습 가능한 region proposal & detection 통합 시스템
  - VGG-16 기반 구현에서 5fps 속도 유지하면서 PASCAL VOC 2007/2012 기준 SOTA 정확도 달성 (mAP 73.2%, 70.4%)

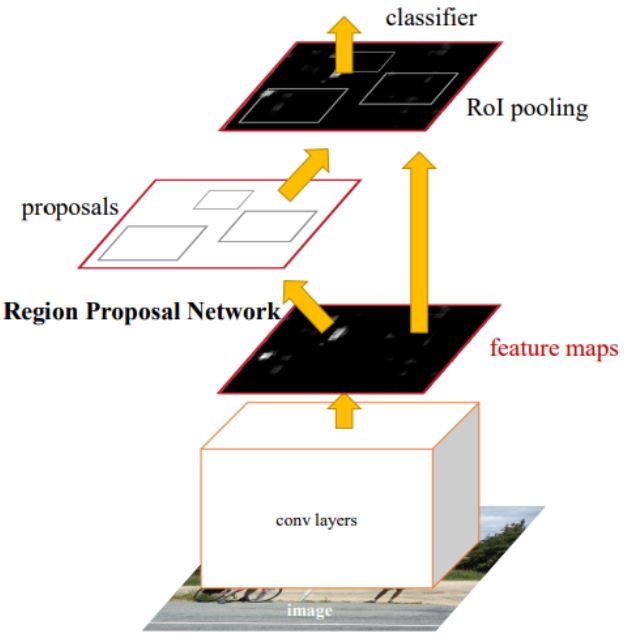
1. Overview

- Faster R-CNN은 two-stage object detector 구조
- 첫 번째 단계 : RPN이 feature map 위에서 후보 박스(proposals)를 생성
- 두 번째 단계 : 제안된 박스들을 기준으로 classification 및 bounding-box regression 수행
- 네트워크 설계에서는 backbone CNN (예: VGG-16) 위에 RPN 모듈과 detection 모듈을 쌓고, convolution feature를 공유
- 학습 시 RPN과 detection 네트워크를 번갈아 최적화하는 alternating training 방식 사용

2. Challenges

- region 제안 속도 및 비용 문제: 전통적 방법은 느리고 비효율적
- 후보 박스의 질과 다양성 확보 어려움
- 공유된 feature가 서로 다른 목적(RPN vs detection)에 사용될 때의 균형 문제
- end-to-end 학습 안정성 확보
- 소형 객체, 복잡한 배경, 다양한 스케일 처리의 어려움

3. Method



- Region Proposal Network (RPN)
  - feature map 위의 sliding window 방식으로 anchor boxes 생성
  - 각 anchor에 대해 objectness score + bounding-box offset 예측
  - fully convolutional 구조이며 backbone feature 공유
- Detection Network (Fast R-CNN 기반)
  - 제안된 region을 RoI pooling 레이어로 고정 크기 feature로 변환
  - 이후 classification과 bounding-box regression 분기로 처리

- Feature Sharing & Alternating Training
  - RPN과 detection 네트워크가 convolution층을 공유
  - 두 모듈을 번갈아 학습: RPN 고정 → detection 학습 → detection 고정 → RPN 학습 반복
- Anchor 설정 및 hyperparameter
  - 다양한 스케일과 비율의 anchor 사용
  - proposal 수 제한 (예: 300 proposals per image)
  - non-max suppression과 객체 없는(anchor negative) 샘플링 전략

---

4. Experiments

- 데이터셋 : PASCAL VOC 2007, VOC 2012 기준 benchmark 사용
- 백본 네트워크 : VGG-16 모델을 주로 사용
- 평가 지표 : mAP (mean Average Precision)
- 속도 측정: 전체 파이프라인 포함한 프레임 처리 속도 약 5fps 수준 (GPU 기준)
- 비교 대상 : Fast R-CNN + 외부 proposal 방법 (Selective Search 등)
- 실험 변수 : proposals 수, anchor 수, feature 공유 유무 등

5. Results

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

Table 3: Detection results on **PASCAL VOC 2007 test set**. The detector is Fast R-CNN and VGG-16. Training data: "07": VOC 2007 trainval, "07+12": union set of VOC 2007 trainval and VOC 2012 trainval. For RPN, the train-time proposals for Fast R-CNN are 2000. †: this number was reported in [2]; using the repository provided by this paper, this result is higher (68.1).

method	# proposals	data	mAP (%)
SS	2000	07	66.9†
SS	2000	07+12	70.0
RPN+VGG, unshared	300	07	68.5
RPN+VGG, shared	300	07	69.9
RPN+VGG, shared	300	07+12	**73.2**
RPN+VGG, shared	300	COCO+07+12	**78.8**

Table 4: Detection results on **PASCAL VOC 2012 test set**. The detector is Fast R-CNN and VGG-16. Training data: "07": VOC 2007 trainval, "07++12": union set of VOC 2007 trainval+test and VOC 2012 trainval. For RPN, the train-time proposals for Fast R-CNN are 2000. †: http://host.robots.ox.ac.uk:8080/anonymous/HZJTQA.html. ‡: http://host.robots.ox.ac.uk:8080/anonymous/YNPLXB.html. §: http://host.robots.ox.ac.uk:8080/anonymous/XEDH10.html.

method	# proposals	data	mAP (%)
SS	2000	12	65.7
SS	2000	07++12	68.4
RPN+VGG, shared†	300	12	67.0
RPN+VGG, shared‡	300	07++12	**70.4**
RPN+VGG, shared§	300	COCO+07++12	**75.9**

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**

Table 6: Results on PASCAL VOC 2007 test set with Fast R-CNN detectors and VGG-16. For RPN, the train-time proposals for Fast R-CNN are 2000. RPN* denotes the unsharing feature version.

method	# box	data	mAP	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
SS	2000	07	66.9	74.5	78.3	69.2	53.2	36.6	77.3	78.2	82.0	40.7	72.7	67.9	79.6	79.2	73.0	69.0	30.1	65.4	70.2	75.8	65.8
SS	2000	07+12	70.0	77.0	78.1	69.3	59.4	38.3	81.6	78.6	86.7	42.8	78.8	68.9	84.7	82.0	76.6	69.9	31.8	70.1	74.8	80.4	70.4
RPN*	300	07	68.5	74.1	77.2	67.7	53.9	51.0	75.1	79.2	78.9	50.7	78.0	61.1	79.1	81.9	72.2	75.9	37.2	71.4	62.5	77.4	66.4
RPN	300	07	69.9	70.0	80.6	70.1	57.3	49.9	78.2	80.4	82.0	52.2	75.3	67.2	80.3	79.8	75.0	76.3	39.1	68.3	67.3	81.1	67.6
RPN	300	07+12	73.2	76.5	79.0	70.9	65.5	52.1	83.1	84.7	86.4	52.0	81.9	65.7	84.8	84.6	77.5	76.7	38.8	73.6	73.9	83.0	72.6
RPN	300	COCO+07+12	**78.8**	**84.3**	**82.0**	**77.7**	**68.9**	**65.7**	**88.1**	**88.4**	**88.9**	**63.6**	**86.3**	**70.8**	**85.9**	**87.6**	**80.1**	**82.3**	**53.6**	**80.4**	**75.8**	**86.6**	**78.9**

Table 7: Results on PASCAL VOC 2012 test set with Fast R-CNN detectors and VGG-16. For RPN, the train-time proposals for Fast R-CNN are 2000.

method	# box	data	mAP	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
SS	2000	12	65.7	80.3	74.7	66.9	46.9	37.7	73.9	68.6	87.7	41.7	71.1	51.1	86.0	77.8	79.8	69.8	32.1	65.5	63.8	76.4	61.7
SS	2000	07++12	68.4	82.3	78.4	70.8	52.3	38.7	77.8	71.6	89.3	44.2	73.0	55.0	**87.5**	80.5	80.8	72.0	35.1	68.3	**65.7**	80.4	64.2
RPN	300	12	67.0	82.3	76.4	71.0	48.4	45.2	72.1	72.3	87.3	42.2	73.7	50.0	86.8	78.7	78.4	77.4	34.5	70.1	57.1	77.1	58.9
RPN	300	07++12	70.4	84.9	79.8	74.3	53.9	49.8	77.5	75.9	88.5	45.6	77.1	55.3	86.9	81.7	80.9	79.6	40.1	72.6	60.9	81.2	61.5
RPN	300	COCO+07++12	**75.9**	**87.4**	**83.6**	**76.8**	**62.9**	**59.6**	**81.9**	**82.0**	**91.3**	**54.9**	**82.6**	**59.0**	**89.0**	**85.5**	**84.7**	**84.1**	**52.2**	**78.9**	65.5	**85.4**	**70.2**

- Faster R-CNN은 외부 region proposal 알고리즘 의존도를 제거하면서도 detection 정확도 유지 또는 향상
- VOC 2007에서는 73.2% mAP, VOC 2012에서는 70.4% mAP 성과 달성
- 전체 시스템 속도 약 5fps 수준 유지하면서 실시간에 근접한 성능
- 다른 모델들과 비교 시 region proposal이 내장된 구조가 이전 방식보다 더 효율적이고 통합 가능

6. Insight

- region proposal을 네트워크 내부로 통합한 설계가 객체 탐지 분야에서 전환점
- 구조 공유 설계가 효율성과 정확도의 균형을 가능하게 함
- 소형 객체 처리, 스케일 다양성, anchor 설정 등이 여전히 중요한 설계 변수
- 실제 응용에서는 속도와 정확도 간 trade-off 고려 필수
- 후속 연구는 더 빠른 inference, lightweight design, multi-scale 개선 등이 주요 주제