Rich feature hierarchies for accurate object detection and semantic segmentation

by Ross Girshick(UC Berkeley),
Jeff Donahue(UC Berkeley),
Trevor Darrell(UC Berkeley),
Jitendra Malik(UC Berkeley)

Index

Abstract

연구 계기 : 1. 저자들이 연구할 당시 Object Detection 분야의 알고리즘 성능은 큰 발전 없이 정체 상태.

2. 이미지의 저 차원 정보(Fine information) + 고 차원 정보(Coarse information)를 활용한 아주 복잡한 형태의 앙상블 모델이 주류.

간단하면서도 여러 개념을 도입할 수 있는 확장 가능한 방법이 없을까.

키 포인트: 1. 비전 분야의 과제에서 효율적이라고 알려진 대용량의 CNN을 Object Detection 혹은 Semantic segmentation에서 Region proposals에 적용 가능.

2. 타겟 도메인의 데이터가 부족할 때, Transfer learning + Fine-tuning을 적용하여 문제점 완화.

Introduction

키포인트 1

연구 계기(Detailed): 1. 이 연구가 진행되었을 당시 다음과 같은 성과를 낸 방법들의 앙상블이거나 약간 변형된 알고리즘은 계속 제안되었으나 큰 진전은 없었음.

- SIFT - D. Lowe. Distinctive image features from scale-invariant keypoints. IJCV, 2004
- HOG - N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In CVPR, 2005
- Neocognitron - K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological cybernetics, 36(4):193–202, 1980

Introduction

키포인트 1

연구 계기(Detailed): 2. LeCun 등이 역전파를 통해서 SGD를 수행하는 것이 CNN을 훈련시키는 것에 효율적이라는 것을 보여줌.

3. 그러나 SVM이 출현하면서 다시 딥러닝 기반 방법들의 이용 빈도가 줄어들게 됨.

4. Krizhevsky 등이 ILSVRC에서 매우 높은 정확도를 보이는 CNN 모델을 선보이면서 딥러닝 기반(CNN) 방법이 재주목을 받게 되었다(CNN + ReLU, Dropout 등).

· SIFT - A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In NIPS, 2012.

Introduction

키포인트 1

연구 계기(Detailed): 5. 저자들은 Classification task를 해결한 CNN을 어떻게 Object Detection에 적용할 수 있는지 고민함.

저자들이 고민한 두 가지 관점

(1) 딥러닝 네트워크(CNN)을 통한 이미지 안의 객체들의 위치 추정

(2) 적은 양의 Annotated된 Detection dataset으로 대용량의 모델에서 훈련시키는 방법

Introduction

키포인트 1

연구 계기(Detailed):

6. 처음에는 슬라이딩 윈도우 방식의 Detector를 만들어서 객체들의 위치 추정을 하려고 했으나 네트워크의 용량이 커지고, 입력 이미지의 크기가 커지면서 정확한 위치 추정이 어려워짐.

7. 따라서 이미지의 부분 부분을 살펴보는 방식으로 Object Detection이나 Semantic segmentation task를 해결하고자 함.

Introduction

키포인트 2

연구 계기(Detailed): 1. Object detection에서 대용량의 CNN을 학습시킬 만한 충분한 Dataset이 부족함.

2. 이를 해결하기 위한 전통적인 방법으로는 비지도 학습으로 모델을 Pre-training 시키고 나서 타겟 도메인 데이터로 Fine-tuning 시키는 것이다.

- P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun. Pedestrian detection with unsupervised multi-stage feature learning. In CVPR, 2013

3. 그러나 저자들은 비지도 학습이 아니라 지도 학습 기반의 Pre-trainin으로 학습한 CNN(ILSVRC classification)에 Fine-tuning 시키는 방법이 효율적이라는 것을 제시하고자 함.

Introduction

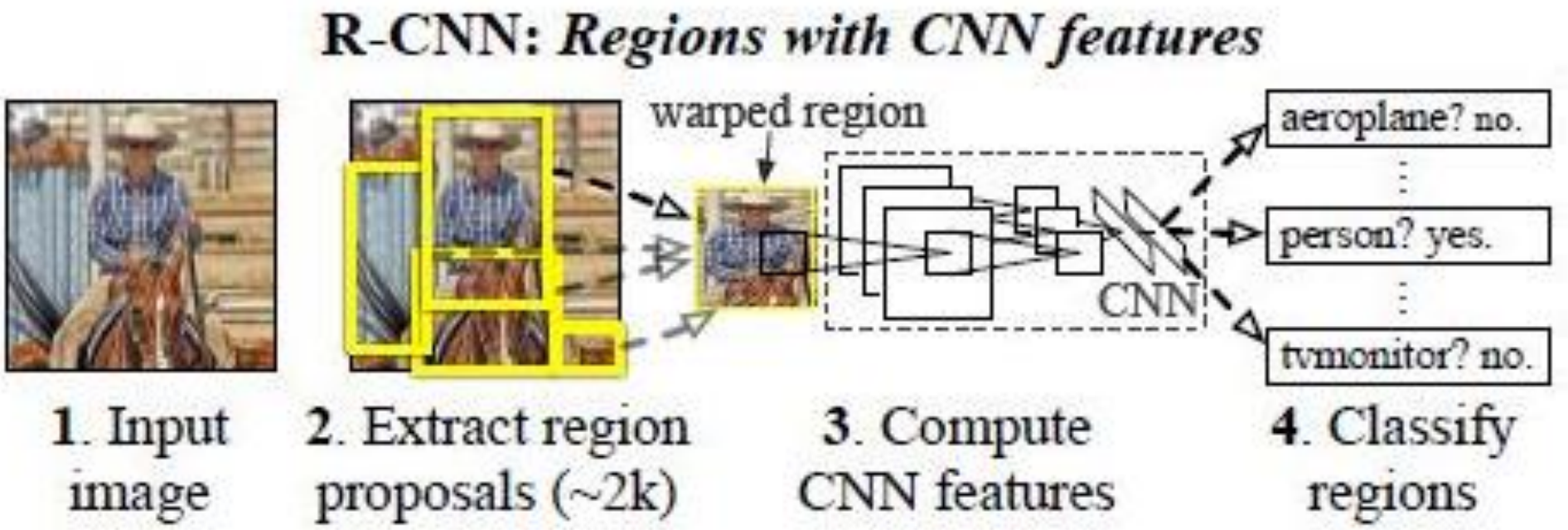
기타

연구 계기(Detailed): 1. 시스템을 개선하기 위해서 실패 사례를 분석하는 것이 중요하다고 생각하여 Hoiem 등이 만들어낸 분석 도구로 분석함.

- D. Hoiem, Y. Chodpathumwan, and Q. Dai. Diagnosing error in object detectors. In ECCV. 2012

2. 그 결과 실패의 주요 원인인 이미지 상의 객체의 잘못된 위치 추정을 Bounding box regression으로 상당히 줄일 수 있음을 확인함.

Object detection with R-CNN



이 연구에서 R-CNN 시스템의 세 부분

(1) 클래스에 상관 없이 적절한 Region proposal들을 만들어 내는 부분

(2) 각 Region마다 고정된 길이의 특징 벡터를 추출하기 위한 CNN

(3) 특정 클래스의 유무를 판단하는 선형 SVM들의 집합

Object detection with R-CNN

Region proposals

이미지에 대하여 Selective search를 수행해 약 2000개의 Region proposal을 추출한다.

※ Selective search

Efficient Graph-Based Image Segmentation(Felzenszwalb 등)을 사용하여 초기 지역 후보 영역을 다음과 같이 다양한 크기와 비율로 생성한다.

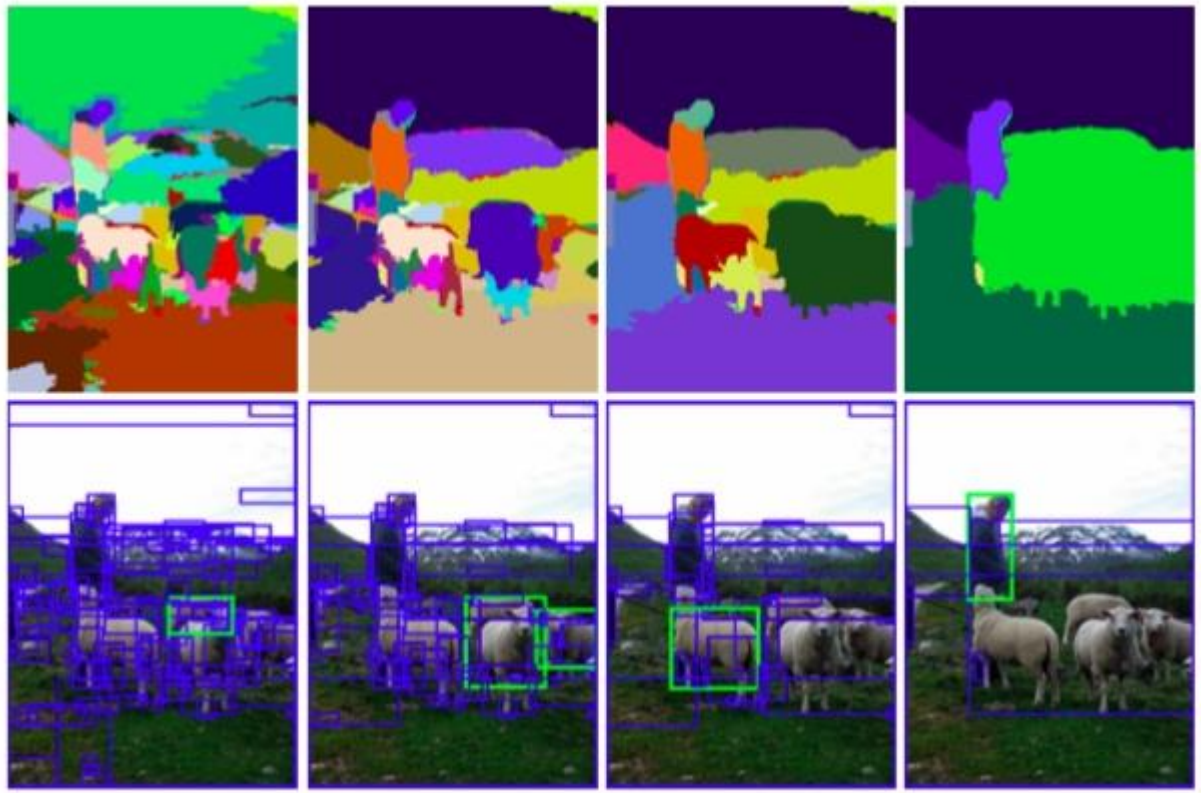
Object detection with R-CNN

Region proposals

이미지에 대하여 Selective search를 수행해 약 2000개의 Region proposal을 추출한다.

※ Selective search

그리디 알고리즘을 통해 비슷한 영역을 반복적으로 통합한다.



Object detection with R-CNN

Region proposals

이미지에 대하여 Selective search를 수행해 약 2000개의 Region proposal을 추출한다.

※ Selective search

그리디 알고리즘을 통해

비슷한 영역을 반복적으로

통합한다.

Algorithm 1: Hierarchical Grouping Algorithm

Input: (colour) image
Output: Set of object location hypotheses L

Obtain initial regions $R = \{r_1, \cdots, r_n\}$ using [13]
Initialise similarity set $S = \emptyset$
foreach *Neighbouring region pair* (r_i, r_j) **do**
    Calculate similarity $s(r_i, r_j)$
    $S = S \cup s(r_i, r_j)$

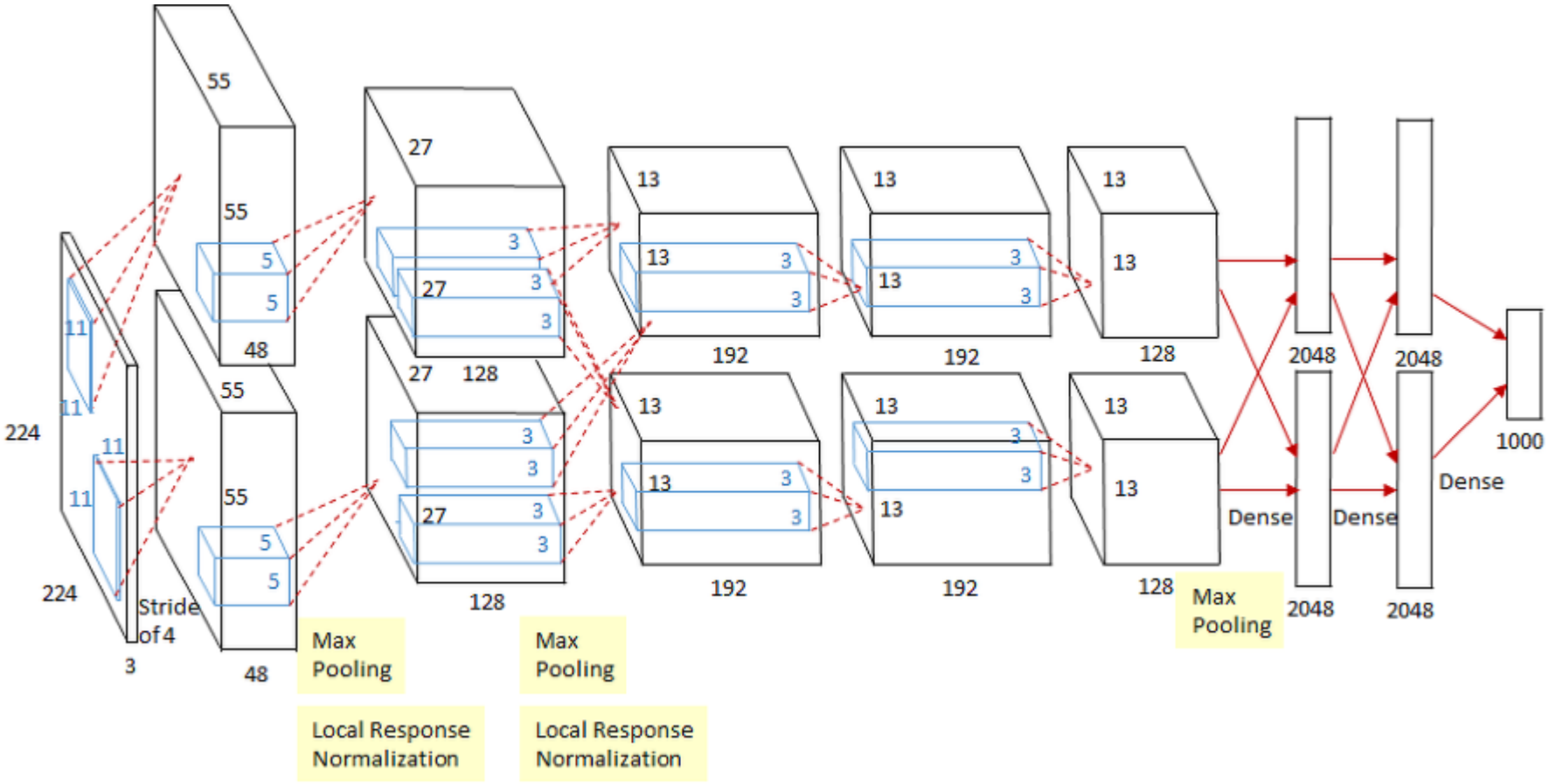
while $S \neq \emptyset$ **do**
    Get highest similarity $s(r_i, r_j) = \max(S)$
    Merge corresponding regions $r_t = r_i \cup r_j$
    Remove similarities regarding $r_i : S = S \setminus s(r_i, r_*)$
    Remove similarities regarding $r_j : S = S \setminus s(r_*, r_j)$
    Calculate similarity set S_t between r_t and its neighbours
    $S = S \cup S_t$
    $R = R \cup r_t$

Extract object location boxes L from all regions in R

Object detection with R-CNN

Feature extraction

Krizhevsky 등이 구현한 AlexNet(Pre-trained)을 통해서 각 Region의 특징을 나타내는 벡터를 추출한다.



Object detection with R-CNN

Feature extraction

Krizhevsky 등이 구현한 AlexNet(Pre-trained)을 통해서 각 Region의 특징을 나타내는 벡터를 추출한다.

이 연구에서 사용된 CNN의 입력 크기는 227x227로 고정되어 있다. 따라서 임의의 사이즈의 입력 Region을 CNN의 입력 크기에 딱 맞도록 변환 하는 과정이 필요하다(Warp : Region을 원하는 크기의 Bounding box로 워프 시킨다.).

Object detection with R-CNN

Test-time detection

1. 테스트 이미지에 대해서 Selective search를 수행해 약 2000개의 Region proposal을 생성.

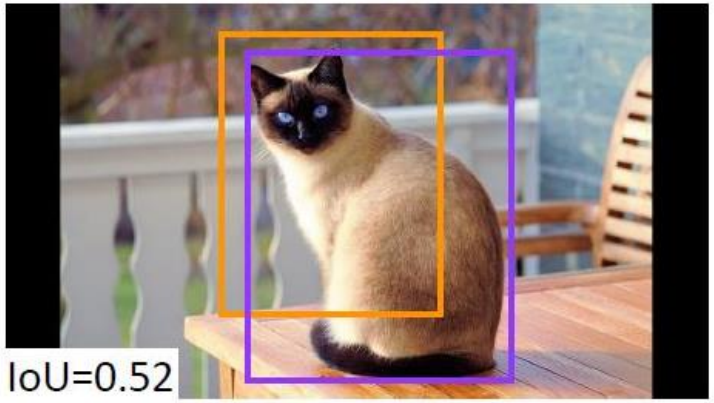
2. Region proposal에 대해 Warp를 수행하고 CNN에서 Forward propagation을 수행해 Fixed-length feature vector(4096 Dimensional)을 추출.

3. 각 벡터에 대해서 특정 클래스인지 여부를 판단하는 SVM을 통해서 Scoring 수행.

4. Scored된 지역들에 대해서 Greedy NMS(Non-maximum suppression)을 수행하여 IoU Threshold보다 낮거나 Threshold보다 높은 Prediction box 중에 몇 가지 제거.

※ IoU(Intersection-over-Union)

실제 정답 박스와 예측 박스가 얼마나 영역이 겹치는지를 평가하는 척도.



$$\mathrm{IoU}(b_{\mathrm{pred}}, b_{\mathrm{gt}}) = \frac{\mathrm{Area}(b_{\mathrm{pred}} \bigcap b_{\mathrm{gt}})}{\mathrm{Area}(b_{\mathrm{pred}} \bigcup b_{\mathrm{gt}})} \qquad \mathrm{Prediction} = \begin{cases} \mathrm{Positive} & c_{\mathrm{pred}} = c_{\mathrm{gt}} \text{ and } \mathrm{IoU}(b_{\mathrm{pred}}, b_{\mathrm{gt}}) > \Omega \\ \mathrm{Negative} & \text{otherwise} \end{cases}$$

Object detection with R-CNN

Training

1. AlexNet을 ILSVRC2012 Classification task 데이터셋으로 Pre-training함. Bounding box label이 이 데이터에 없기 때문에 이미지의 클래스에 관련된 특징을 추출할 수 있는 능력 학습.
2. CNN을 Object Detection 목적에 맞게 하기 위해서 SGD Fine-tuning. 이때 클래스의 개수 : N + 1(N은 객체 클래스 개수, 1은 Background). VOC의 N=20, ILSVRC2013의 N=200
3. Region proposal은 Ground-truth 박스와 IoU 0.5 이상인 Region을 Positive, 그렇지 않으면 Negative로 설정.
4. SGD의 Learning rate를 0.001로 시작하여 미리 학습한 가중치 정보를 최대한 손상시키지 않도록 함.
5. 각 Iteration마다 클래스가 균등하게 Positive 윈도우를 32개 샘플링하고 Background 윈도우를 96개 샘플링하여 128의 배치 사이즈를 구성. Positive 윈도우의 수가 더 적기 때문에 학습 시 가중을 둠.

Object detection with R-CNN

Training

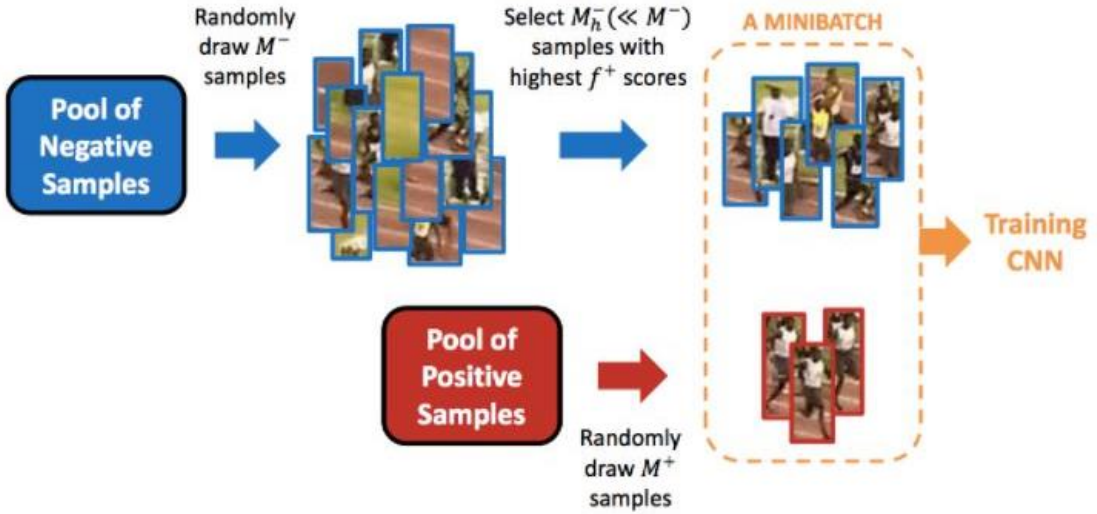
6. 최적의 IoU값을 찾기 위해서 IoU Threshold를 {0, 0.1, ..., 0.5}에서 그리드서치 수행.

7. 클래스의 객체가 들어 있는 Positive region보다 없거나 부분적으로 있는 Negative region이 수가 훨씬 많음. 따라서 이 Background에 대해서 Hard Negative mining 수행.

(8.) Fine-tuning 후에 SVM을 따로 훈련시키는 방법 말고도 Fine-tuning한 CNN의 마지막 계층의 Softmax 회귀 분류기를 달아서 Dector로 활용하는 옵션도 실험해 보았으나 전자에 비해 mAP가 약 4%정도 하락해서 쓰지 않음.

Hard Negative Mining

Hard Negative Mining은 positive example과 negative example을 균형적으로 학습하기 위한 방법입니다. 단순히 random하게 뽑은 것이 아니라 confidence score가 가장 높은 순으로 뽑은 negative example을 (random하게 뽑은 positive example과 함께) training set에 넣어 training합니다.



(그림 출처: 한보형 님의 슬라이드 "Lecture 6: CNNs for Detection, Tracking, and Segmentation")

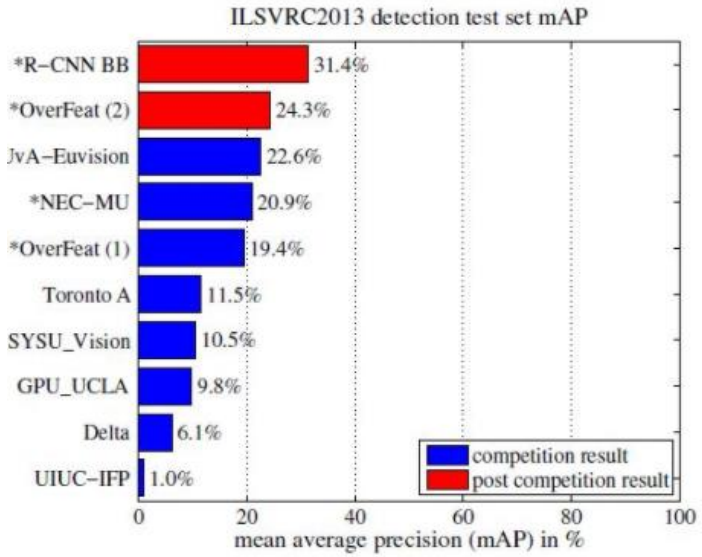
Object detection with R-CNN

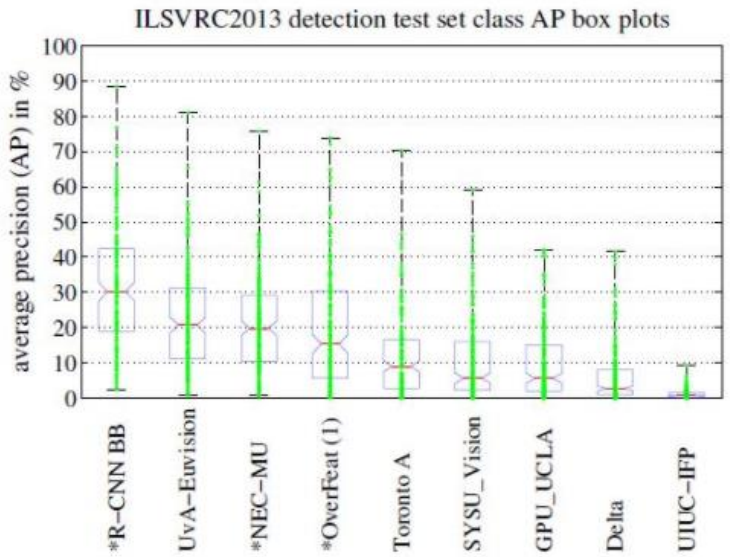
Results on PASCAL VOC 2010-2012

VOC 2010 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
DPM v5 [20]†	49.2	53.8	13.1	15.3	35.5	53.4	49.7	27.0	17.2	28.8	14.7	17.8	46.4	51.2	47.7	10.8	34.2	20.7	43.8	38.3	33.4
UVA [39]	56.2	42.4	15.3	12.6	21.8	49.3	36.8	46.1	12.9	32.1	30.0	36.5	43.5	52.9	32.9	15.3	41.1	31.8	47.0	44.8	35.1
Regionlets [41]	65.0	48.9	25.9	24.6	24.5	56.1	54.5	51.2	17.0	28.9	30.2	35.8	40.2	55.7	43.5	14.3	43.9	32.6	54.0	45.9	39.7
SegDPM [18]†	61.4	53.4	25.6	25.2	35.5	51.7	50.6	50.8	19.3	33.8	26.8	40.4	48.3	54.4	47.1	14.8	38.7	35.0	52.8	43.1	40.4
R-CNN	67.1	64.1	46.7	32.0	30.5	56.4	57.2	65.9	27.0	47.3	40.9	66.6	57.8	65.9	53.6	26.7	56.5	38.1	52.8	50.2	50.2
R-CNN BB	**71.8**	**65.8**	**53.0**	**36.8**	**35.9**	**59.7**	**60.0**	**69.9**	**27.9**	**50.6**	**41.4**	**70.0**	**62.0**	**69.0**	**58.1**	**29.5**	**59.4**	**39.3**	**61.2**	**52.4**	**53.7**

Table 1: Detection average precision (%) on VOC 2010 test. R-CNN is most directly comparable to UVA and Regionlets since all methods use selective search region proposals. Bounding-box regression (BB) is described in Section C. At publication time, SegDPM was the top-performer on the PASCAL VOC leaderboard. †DPM and SegDPM use context rescoring not used by the other methods.

Results on ILSVRC 2013 detection





Visualization, ablation, and modes of error

Visualizing learned features

저자들은 네트워크가 무엇을 학습하는지 위해서 계층의 각 unit의 activation을 계산해서 클래스 스코어 순으로 정렬하고 NMS를 수행한 다음 가장 높은 N개의 Region을 Receptive field와 함께 보고자 했다.

Figure 4: Top regions for six pool$_5$ units. Receptive fields and activation values are drawn in white. Some units are aligned to concepts, such as people (row 1) or text (4). Other units capture texture and material properties, such as dot arrays (2) and specular reflections (6).

두 번째 행은 검은 점의 배열, 세 번째 행은 붉은 덩어리, 다섯 번째 행은 창문과 삼각형 구조, 마지막 행은 빛 반사 점...

Visualization, ablation, and modes of error

Ablation studies – Performance layer-by-layer, without fine-tuning

저자들은 계층 별로 탐지 성능에 대한 중요도를 알아보고자 CNN의 마지막 3개의 계층에 대한 출력을 분석했다.

- CNN은 PASCAL 데이터로 실험을 진행했으며, ILSVRC 2012 데이터셋에서 Pre-trained됨.
- Fc6는 Pool5에 Dense하게 연결되어 있고 Flatten된 Pool5의 9216 차원의 벡터를 4096차원으로 변환한다.
- Fc7은 Fc6와 Dense하게 연결되어 있고 4096차원을 4096으로 출력하되 비선형성과 편향을 더한다.

결과:

Fc7까지의 성능이 Fc6까지의 성능보다 떨어짐. => 네트워크 파라미터의 29%에 해당하는 Fc7 제거 가능

Pool5까지의 성능이 Fc가 없을 때보다 성능이 크게 떨어지지 않음 => 이미지 특징성은 거의 컨볼루션에서...

VOC 2007 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
R-CNN $pool_5$	51.8	60.2	36.4	27.8	23.2	52.8	60.6	49.2	18.3	47.8	44.3	40.8	56.6	58.7	42.4	23.4	46.1	36.7	51.3	55.7	44.2
R-CNN fc_6	59.3	61.8	43.1	34.0	25.1	53.1	60.6	52.8	21.7	47.8	42.7	47.8	52.5	58.5	44.6	25.6	48.3	34.0	53.1	58.0	46.2
R-CNN fc_7	57.6	57.9	38.5	31.8	23.7	51.2	58.9	51.4	20.0	50.5	40.9	46.0	51.6	55.9	43.3	23.3	48.1	35.3	51.0	57.4	44.7

Visualization, ablation, and modes of error

Ablation studies – Performance layer-by-layer, with fine-tuning

결과:

Fine-tuning을 하지 않았을 경우보다 Map가 대략 8% 증가.

Fc6, Fc7에서 두드러짐 -> 컨볼루션은 General한 특징을, Fc는 Domain-specif하고 Non-linear한 특징을 잡음.

R-CNN FT $pool_5$	58.2	63.3	37.9	27.6	26.1	54.1	66.9	51.4	26.7	55.5	43.4	43.1	57.7	59.0	45.8	28.1	50.8	40.6	53.1	56.4	47.3
R-CNN FT fc_6	63.5	66.0	47.9	37.7	29.9	62.5	70.2	60.2	32.0	57.9	47.0	53.5	60.1	64.2	52.2	31.3	55.0	50.0	57.7	63.0	53.1
R-CNN FT fc_7	64.2	69.7	50.0	41.9	32.0	62.6	71.0	60.7	32.7	58.5	46.5	56.1	60.6	66.8	54.2	31.5	52.8	48.9	57.9	64.7	54.2

Ablation studies – Comparison to recent feature learning methods

R-CNN 계열 모델들이 전반적으로 뛰어난 성능을 보임.

DPM v5 [20]	33.2	60.3	10.2	16.1	27.3	54.3	58.2	23.0	20.0	24.1	26.7	12.7	58.1	48.2	43.2	12.0	21.1	36.1	46.0	43.5	33.7
DPM ST [28]	23.8	58.2	10.5	8.5	27.1	50.4	52.0	7.3	19.2	22.8	18.1	8.0	55.9	44.8	32.4	13.3	15.9	22.8	46.2	44.9	29.1
DPM HSC [31]	32.2	58.3	11.5	16.3	30.6	49.9	54.8	23.5	21.5	27.7	34.0	13.7	58.1	51.6	39.9	12.4	23.5	34.4	47.4	45.2	34.3

Visualization, ablation, and modes of error

Network architectures

VOC 2007 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
R-CNN T-Net	64.2	69.7	50.0	41.9	32.0	62.6	71.0	60.7	32.7	58.5	46.5	56.1	60.6	66.8	54.2	31.5	52.8	48.9	57.9	64.7	54.2
R-CNN T-Net BB	68.1	72.8	56.8	43.0	36.8	66.3	74.2	67.6	34.4	63.5	54.5	61.2	69.1	68.6	58.7	33.4	62.9	51.1	62.5	64.8	58.5
R-CNN O-Net	71.6	73.5	58.1	42.2	39.4	70.7	76.0	74.5	38.7	71.0	56.9	74.5	67.9	69.6	59.3	**35.7**	62.1	64.0	66.5	**71.2**	62.2
R-CNN O-Net BB	**73.4**	**77.0**	**63.4**	**45.4**	**44.6**	**75.1**	**78.1**	**79.8**	**40.5**	**73.7**	**62.2**	**79.4**	**78.1**	**73.1**	**64.2**	35.6	**66.8**	**67.2**	**70.4**	71.1	**66.0**

Table 3: Detection average precision (%) on VOC 2007 test for two different CNN architectures. The first two rows are results from Table 2 using Krizhevsky et al.'s architecture (T-Net). Rows three and four use the recently proposed 16-layer architecture from Simonyan and Zisserman (O-Net) [43].

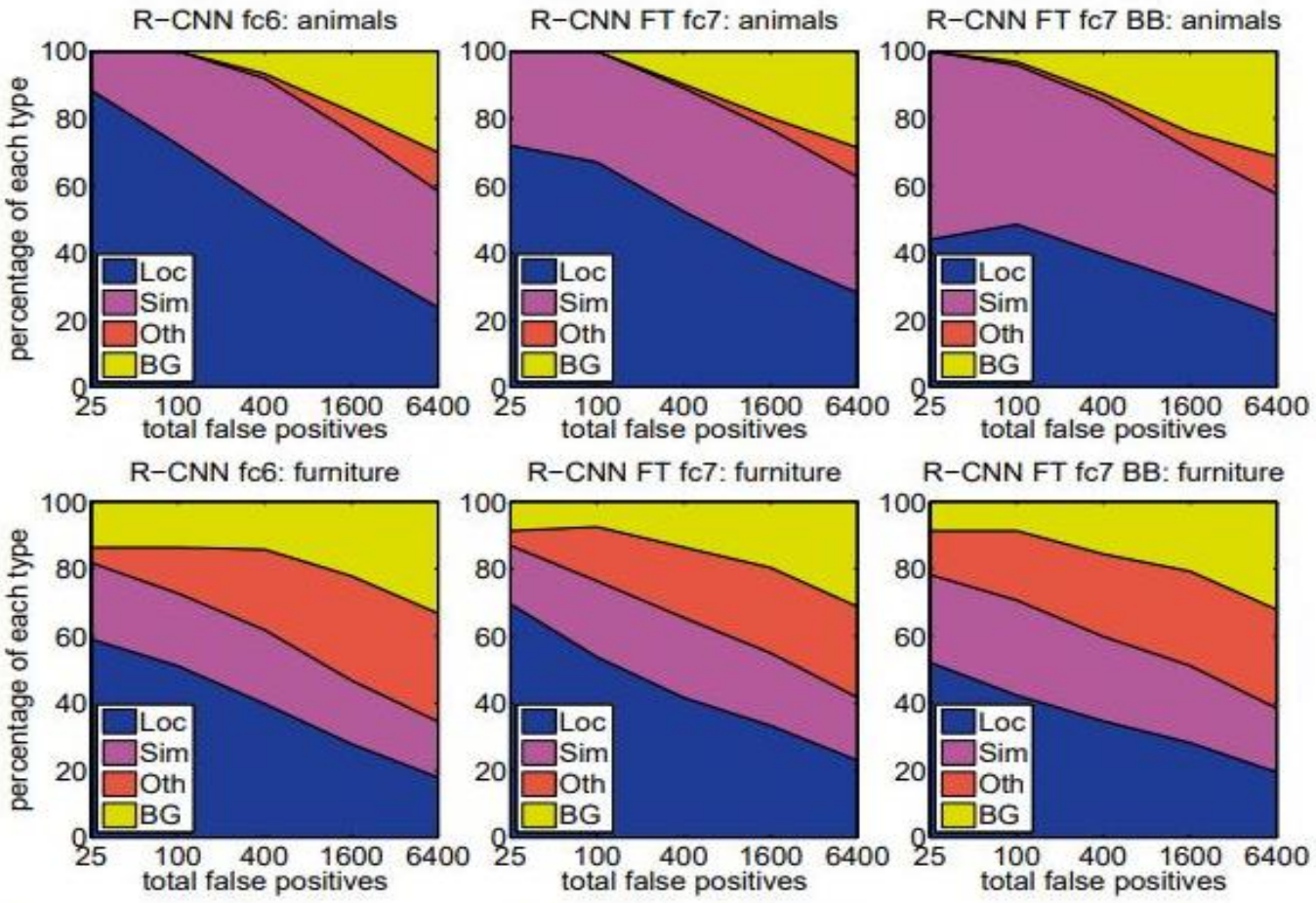
저자들은 아키텍처가 R-CNN의 탐지 성능에 크게 영향을 끼친다는 것을 알아냄.

OxfordNet(O-Net), TorontoNet(T-Net)에서 성능을 비교했을 때, O-Net이 Map에서는 T-Net을 압도하지만 학습 순전파시에 7배의 시간이 더 걸린다고 함.

Visualization, ablation, and modes of error

Detection error analysis

Hoiem 등의 분석도구로 R-CNN의 오류의 비율을 분석함.



FT : Fine-tuing
BB : Bound box regression

FP types:
Loc : 잘못된 위치 추정(IoU 0.1에서 0.5 혹은 아예 중복)
Sim: 비슷한 클래스 혼동 오류
Oth: 비슷하지 않은 클래스 혼동 오류
BG: Background와 관련된 오류

· 클래스 관련된 오류보다 위치 추정 오류가 많은 것을 보아 HOG보다는 CNN에서 추출한 특징이 변별력이 비교적 높다.

· Bounding box regression이 위치 추정과 관련된 오류를 줄이는데 효과가 있다.

Visualization, ablation, and modes of error

Bounding-box regression

N training pairs	Proposal P	Ground-truth G
$\{(P^i, G^i)\}_{i=1,\ldots,N}$	$P^i = (P^i_x, P^i_y, P^i_w, P^i_h)$	$G = (G_x, G_y, G_w, G_h)$

여기서 Px, Py, Pw, Ph는 각각 Proposal P의 중앙 x, y 좌표, 넓이, 높이이고 Gx, Gy, Gw, Gh는 Ground truth의 중앙 x, y 좌표, 넓이, 높이이다. 이 모델의 목적은 P를 G로 매핑하는 변환을 학습하는 것이다.

$$d_x(P),\ d_y(P),\ d_w(P),\ \text{and}\ d_h(P)$$

저자들은 Proposal P에 대한 값을 인자로 하는 변환을 다음과 같이 함수로 만들었다. 앞의 두 개는 Scale-invariant하게 P 박스의 중앙 좌표 값을 변환하고 뒤의 두 개는 P 박스의 넓이와 높이를 log 값으로 변환한다.

Visualization, ablation, and modes of error

Bounding-box regression

$$\hat{G}_x = P_w d_x(P) + P_x \quad (1)$$
$$\hat{G}_y = P_h d_y(P) + P_y \quad (2)$$
$$\hat{G}_w = P_w \exp(d_w(P)) \quad (3)$$
$$\hat{G}_h = P_h \exp(d_h(P)). \quad (4)$$

위의 개념들을 이용해서 P를 이용해서 Ground-truth의 예측 값 G^hat을 위와 같이 나타낼 수 있다.

$$d_\star(P) = \mathbf{w}_\star^{\mathrm{T}} \boldsymbol{\phi}_5(P)$$

d*(P)는 P의 pool5의 출력 특징에 대한 선형 함수로서 모델링할 수 있다. w*는 학습하는 파라미터 벡터,

$\boldsymbol{\phi}_5(P)$ 는 pool5의 특징 출력

Visualization, ablation, and modes of error

Bounding-box regression

$$\mathbf{w}_\star = \underset{\hat{\mathbf{w}}_\star}{\operatorname{argmin}} \sum_i^N (t_\star^i - \hat{\mathbf{w}}_\star^{\mathrm{T}} \boldsymbol{\phi}_5(P^i))^2 + \lambda \|\hat{\mathbf{w}}_\star\|^2 . \quad (5)$$

w*를 Ridge regression을 통해서 최적화 시킨다.

$$t_x = (G_x - P_x)/P_w \quad (6)$$
$$t_y = (G_y - P_y)/P_h \quad (7)$$
$$t_w = \log(G_w/P_w) \quad (8)$$
$$t_h = \log(G_h/P_h). \quad (9)$$

훈련셋(P,G)에 대한 타겟 t*는 위와 같이 나타낼 수 있다.

Visualization, ablation, and modes of error

Bounding-box regression

주의해야 할 두 가지 이슈사항

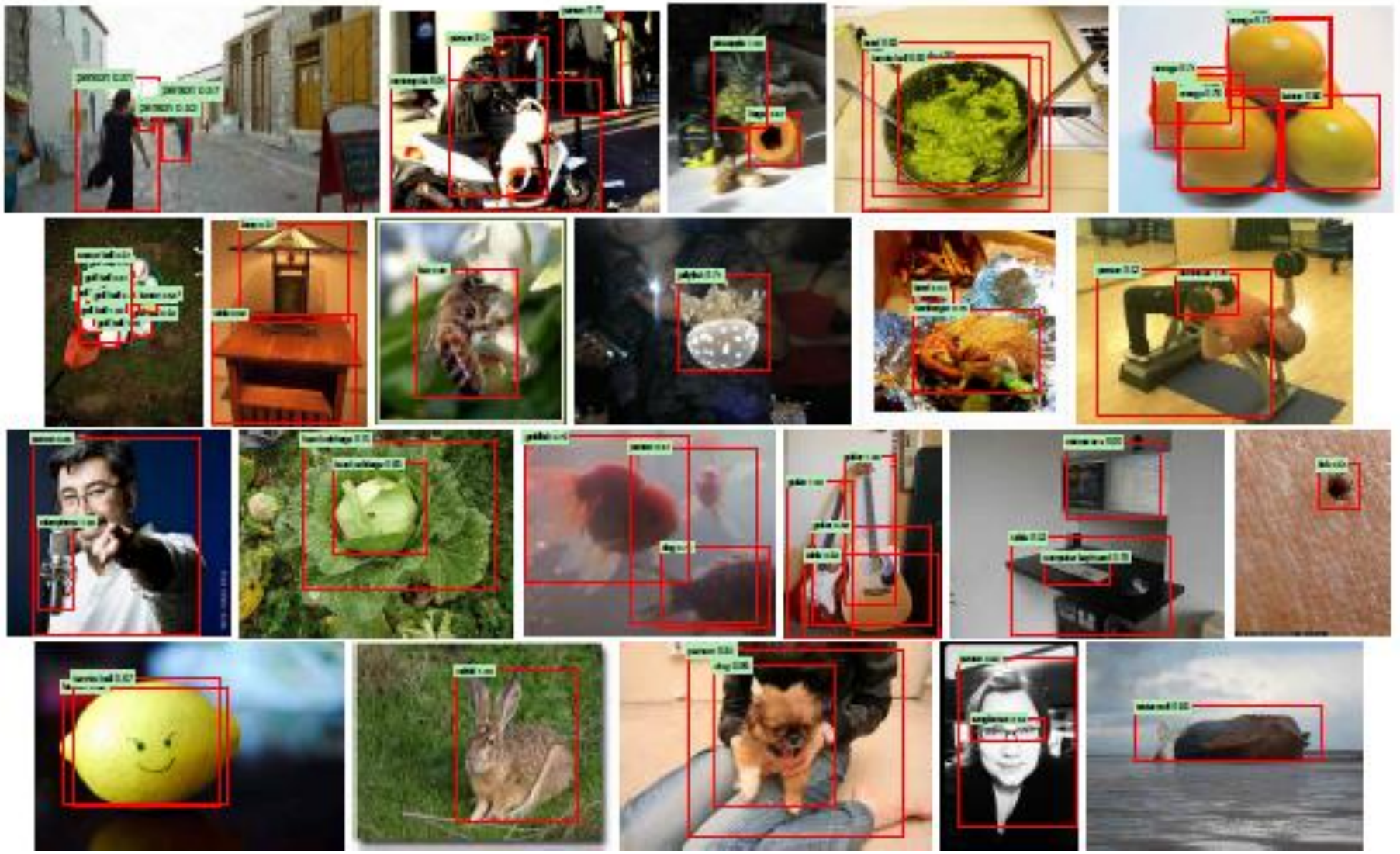
(1) 규제의 정도 - 검증 셋에 근거해서 λ를 1000으로 설정했다.

(2) (P, G) 설정의 중요성 - 만약에 P가 그 어떤 G와도 가까이 있지 않다면 P를 G로 근사화 시키는 것은 애초에 말이 안되는 작업이 된다. 따라서 P가 최소 하나의 G와 가까이 있을 때 P와 G를 묶었다. 가까움의 정도는 P가 G와의 IoU가 가장 높으면서 특정 임계 값보다(검증 셋을 이용하여 0.6으로 설정) 높으면 그 P를 G와 묶었다. 나머지 G와 묶이지 않는 P는 버린다.

테스트 시에는 이런 과정을 한 번씩만 수행했다. 여러 번 수행할 수도 있으나 여러 번 수행하는 것이 결과 개선에 효과가 없었다고 한다.

Visualization, ablation, and modes of error

Qualitative results – ILSVRC2013 검증 셋 테스트 결과 일부



The ILSVRC2013 detection dataset

Dataset overview

ILSVRC2013 탐지용 데이터셋은 PASCAL VOC 데이터셋보다 Less homogeneous(훈련셋과 검증 + 테스트 셋의 처리에 있어서 차이가 있음)하기 때문에 분석이 중요.

ILSVRC2013 탐지 셋이 어떻게 수집되었고 annotated되었는지는 다음을 참고

- J. Deng, O. Russakovsky, J. Krause, M. Bernstein, A. C. Berg, and L. Fei-Fei. Scalable multi-label annotation. In CHI, 2014.

- H. Su, J. Deng, and L. Fei-Fei. Crowdsourcing annotations for visual object detection. In AAAI Technical Report, 4th Human Computation Workshop, 2012

val + test : 같은 Image distribution 샘플링. PASCAL VOC 이미지 셋과 비슷함. 이미지 안의 거의 모든 객체(200 클래스)에 대해서 바운딩 박스 레이블링 되어 있음.

train : ILSVRC2013 Classification 데이터셋의 Image distribution에서 샘플링. 이미지 안의 객체에 대해서 Annotation이 있는 경우도 있고 없는 경우도 있음. train의 경우 철저하게 레이블링 되어 있지 않기 때문에 Background 레이블에 대한 Hard negative mining 적용 불가.

The ILSVRC2013 detection dataset

Dataset overview

ILSVRC2013 탐지용 데이터셋은 PASCAL VOC 데이터셋보다 Less homogeneous(훈련셋과 검증 + 테스트 셋의 처리에 있어서 차이가 있음)하기 때문에 분석이 중요.

· 저자들은 훈련 시에 검증 셋에 의존하되 몇 가지 훈련 셋을 Positive 샘플로서 추가하는 방법을 적용하기로 함.

· val을 각각 같은 크기로 val1, val2로 나눔. 나눌 때, 몇몇 클래스는 val에 적게 나타나므로 균등하게 나누도록 함.

Region proposals

· ILSVRC2013 데이터셋에도 Selective search를 val1, val2, 약간의 train, test 셋에 적용하여 Region proposal 생성

· 다만 이 데이터셋은 이미지의 크기의 범주가 크기 때문에 Selective search를 수행하기 전에 특정 길이로 이미지 크기를 조정한 후(여기서는 500픽셀) 수행.

· val 셋에서 Selective search를 수행했을 때 IoU가 0.5이상 되는 지역은 91.6% 정도의 Recall 수치를 보이는데 이는 PASCAL이 98%이라는 것과는 대조적으로 낮은 수치.

The ILSVRC2013 detection dataset

Training data

· 훈련용 데이터로 val1과 Ground truth 박스 정보, 그리고 ILSVRC2013 훈련 셋에서 클래스당 최대 N개의 Ground-truth 박스(만약에 클래스가 N개보다 적은 Ground truth 박스를 가지고 있다면 모두 사용)를 묶어서 val1+trainN이라고 함.

· CNN Fine-tuning할 때는 50K SGD Iteration, PASCAL 데이터 셋으로 훈련시킬 때와 동일 셋팅

· SVM을 훈련시킬 때는 val1+trainN의 모든 Ground-truth 박스들이 해당 클래스에 대한 Positive example, val1 데이터에서 무작위로 뽑은 5000장에 대해서 Hard negative mining을 수행해서 뽑은 데이터를 Negative example로 함.

· Bounding-box regressor는 val1으로 훈련.

Validation and evaluation

· 결과를 첫번째로, 평가 서버에 제출하기 전에 실험에서 적용했던 데이터 사용 패턴과 Fine-tuning의 효과 그리고 바운딩 박스 회귀를 val2 셋으로 검증. 훈련과 마찬 가지로 기타 셋팅(하이퍼 파라미터 셋팅 등)은 PASCAL 데이터 셋으로 훈련시킬 때와 동일 셋팅.

· val2에 맞는 최적의 셋팅을 찾아낸 후에는 With BBR, Without BBR 두 번 결과를 제출했고 이때는 SVM과 BBR 훈련 셋을 각각 val+train1k, val까지 확장해서 사용. CNN Fine-tuning은 val1+train1k로 수행.

The ILSVRC2013 detection dataset

Ablation study

test set	val_2	val_2	val_2	val_2	val_2	val_2	test	test
SVM training set	val_1	val_1+$train_{.5k}$	val_1+$train_{1k}$	val_1+$train_{1k}$	val_1+$train_{1k}$	val_1+$train_{1k}$	val+$train_{1k}$	val+$train_{1k}$
CNN fine-tuning set	n/a	n/a	n/a	val_1	val_1+$train_{1k}$	val_1+$train_{1k}$	val_1+$train_{1k}$	val_1+$train_{1k}$
bbox reg set	n/a	n/a	n/a	n/a	n/a	val_1	n/a	val
CNN feature layer	fc_6	fc_6	fc_6	fc_7	fc_7	fc_7	fc_7	fc_7
mAP	20.9	24.1	24.1	26.5	29.7	**31.0**	30.2	**31.4**
median AP	17.7	21.0	21.4	24.8	29.2	**29.6**	29.0	**30.3**

Table 4: ILSVRC2013 ablation study of data usage choices, fine-tuning, and bounding-box regression.

Semantic segmentation

저자들은 R-CNN을 PASCAL VOC segmentation 과제에 적용함.

저자들이 과제를 수행하기 위해서 참고한 연구 방법을 다음을 참조.

- P. Arbel´aez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik. Semantic segmentation using regions and parts. In CVPR, 2012

- J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In ECCV, 2012

또 PASCAL segementation 훈련 셋을 다음의 방법을 통해 추가적인 Annotation 정보가 추가될 수 있도록 함.

- B. Hariharan, P. Arbel´aez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In ICCV, 2011

Design decision과 하이퍼 파라미터 셋팅은 VOC 2011 검증 셋에서 교차 검증되었고 마지막 테스트 결과는 한 번 제출됨.

Semantic segmentation

CNN features for segmentation

저자들은 CPMP에서 생성된 지역의 특징을 계산하기 위한 세가지 전략을 평가했다. 이 세가지 전략 모두 직사각형 윈도우를 227x227 크기의 지역으로 워프 하는 것부터 시작한다.

첫 번째 전략(full)은 생성된 지역의 모양을 무시하고 워프된 윈도우를 입력으로 해서 CNN에서 특징 벡터를 추출한 뒤에 탐지와 관련된 작업을 수행하는 것이다. 그러나 이런 특징들은 각 지역의 직사각형이 아닌 모양은 무시한다.

두 개의 지역이 그다지 겹치지 않으면서도 매우 유사한 바운딩 박스를 가질 수 있다. 그러므로 두 번째 전략(fg)는 지역의 전경 마스크에 대해서만 CNN의 작업을 수행해서 특징을 계산해낸다. 이때 배경은 입력 값의 평균으로 교체해서 평균을 빼는 전처리를 할 때 0이 될 수 있도록 한다.

세번째 전략(full+fg)는 단순히 첫번째와 두 번째 전략의 특징을 Concatenate하는 것이다.

Results on VOC 2011

	full R-CNN		*fg* R-CNN		*full+fg* R-CNN	
O_2P [4]	fc_6	fc_7	fc_6	fc_7	fc_6	fc_7
46.4	43.0	42.5	43.7	42.1	**47.9**	45.8

Conclusion

Object Detection 과제에서 SOTA의 복잡한 시스템(주로 앙상블 모델)를 비교적 간단하면서도 확장 가능한 알고리즘으로 대체할 수 없을까?

두 가지 키 포인트

(1) 대용량의 Pre-trained된 CNN을 Object detection Task에 맞게(이미지 내 객체의 Localization 및 Segmentation) 변경하는 방법 (Region proposal generation, Extracting fixed-size feature vectors, Refinement of Localization)

(2) 대용량의 Pre-trained된 CNN을 Object detection Task에 맞게 변경할 때 관련 도메인 데이터(Object detection용 데이터)가 부족해도 이를 Fine-tuning을 통해서 해결하는 법