

객체탐지

object detection

Dr. Rhee
Feb 2020

R-CNN

컴퓨터 비전 작업

- 비전 작업의 종류

Classification



CAT

No spatial extent

Semantic Segmentation



GRASS, CAT, TREE,
SKY

No objects, just pixels

Object Detection



DOG, DOG, CAT

Multiple Objects

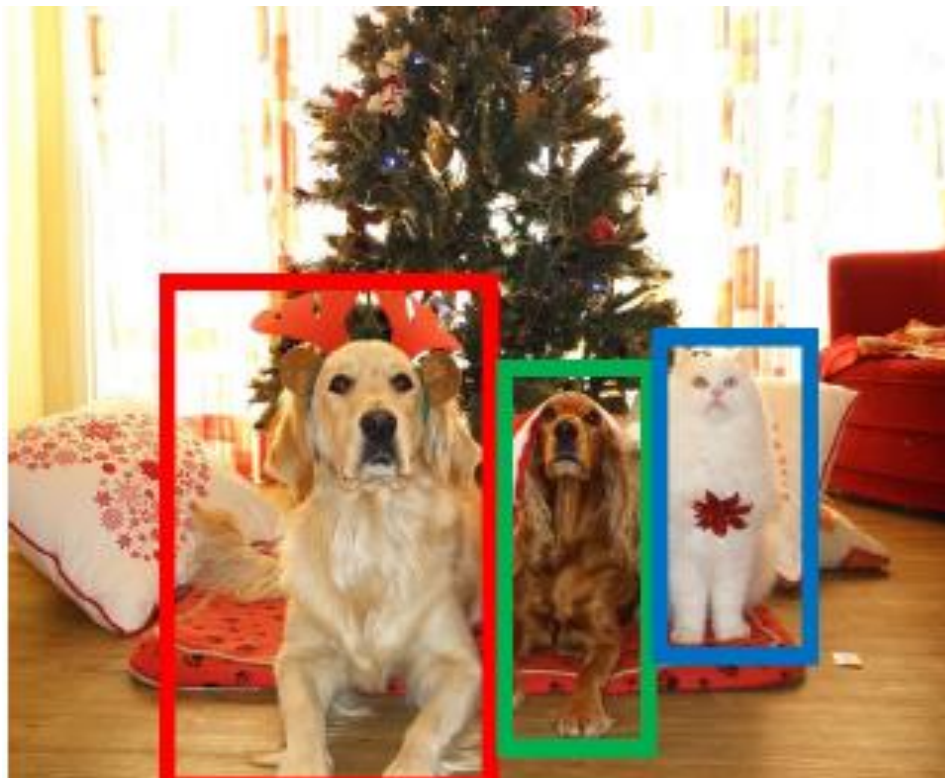
Instance Segmentation



DOG, DOG, CAT

객체 탐지:작업 정의

- 입력: 단일 RGB 이미지
- 출력: 탐지된 객체
 1. 카테고리 레이블 (고정 수의 알려진 카테고리)
 2. 바운딩 박스 (4개의 수치: x , y , 너비, 높이)



객체 탐지:작업 정의

- 단일 객체의 탐지

Detecting a single object

“What”

Correct label:

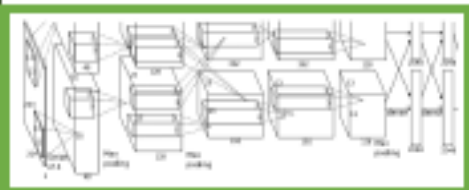
Cat

Often pretrained
on ImageNet
(Transfer learning)



This image is CC0 public domain

Treat localization as a
regression problem!



Fully
Connected:
4096 to 1000

Class Scores

Cat: 0.9
Dog: 0.05
Car: 0.01
...

Softmax
Loss

Multitask
Loss

Weighted
Sum

Loss

Vector:
4096

Fully
Connected:
4096 to 4

Box
Coordinates
(x, y, w, h)

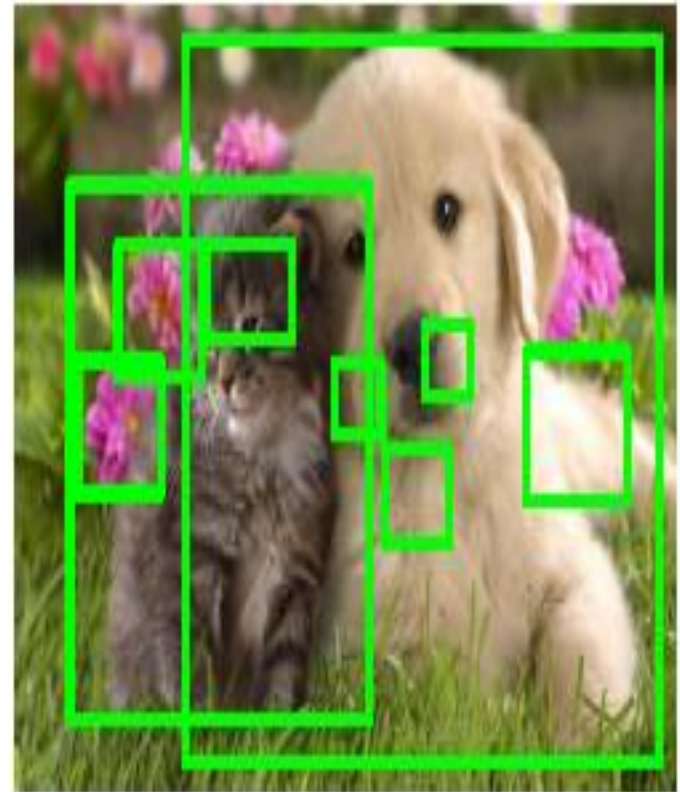
L2 Loss

Correct box:
(x', y', w', h')

“Where”

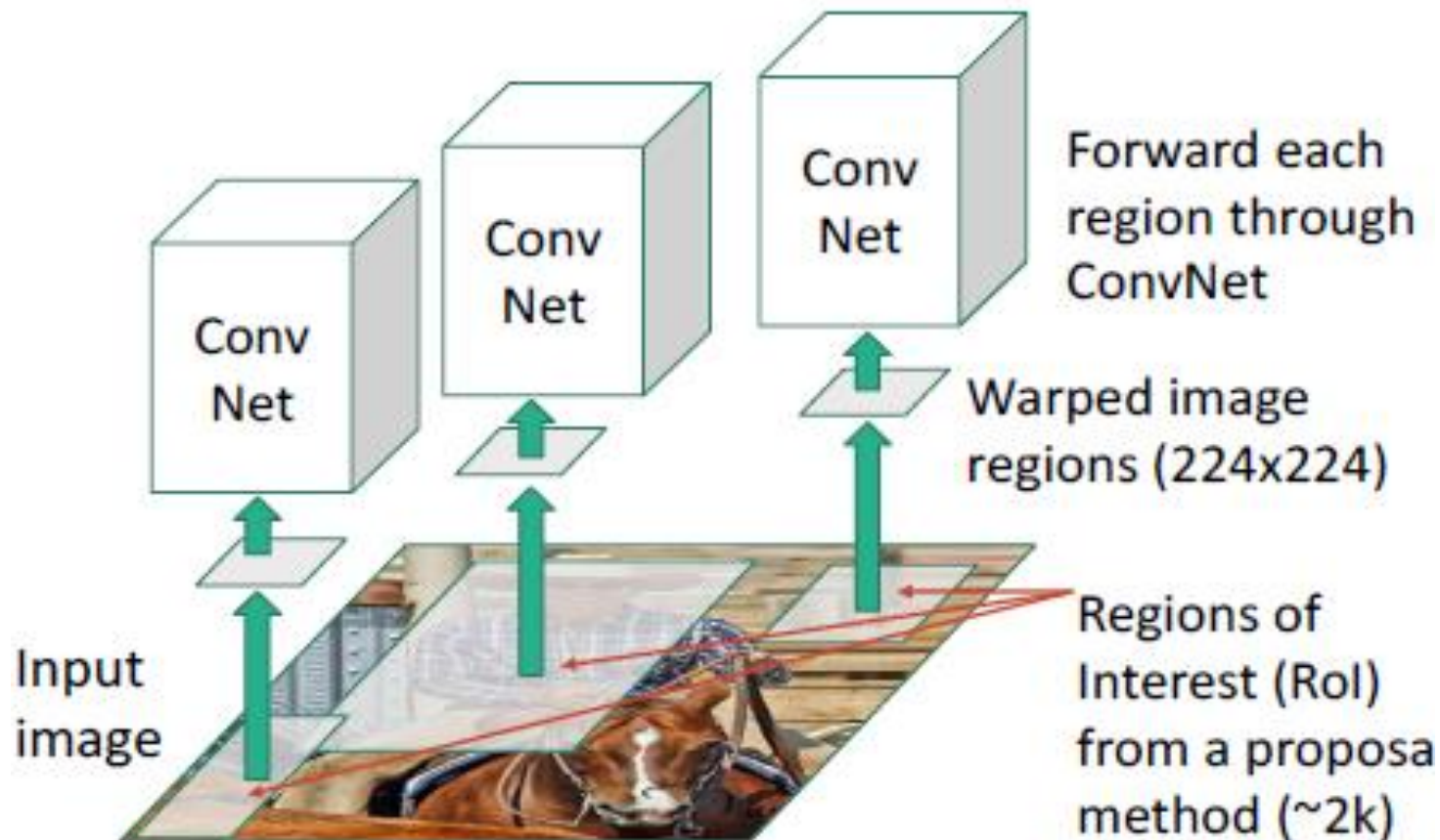
객체 탐지: 영역 프로포잘

- 영역 프로포잘: 모든 객체를 커버할 박스 집합을 발견



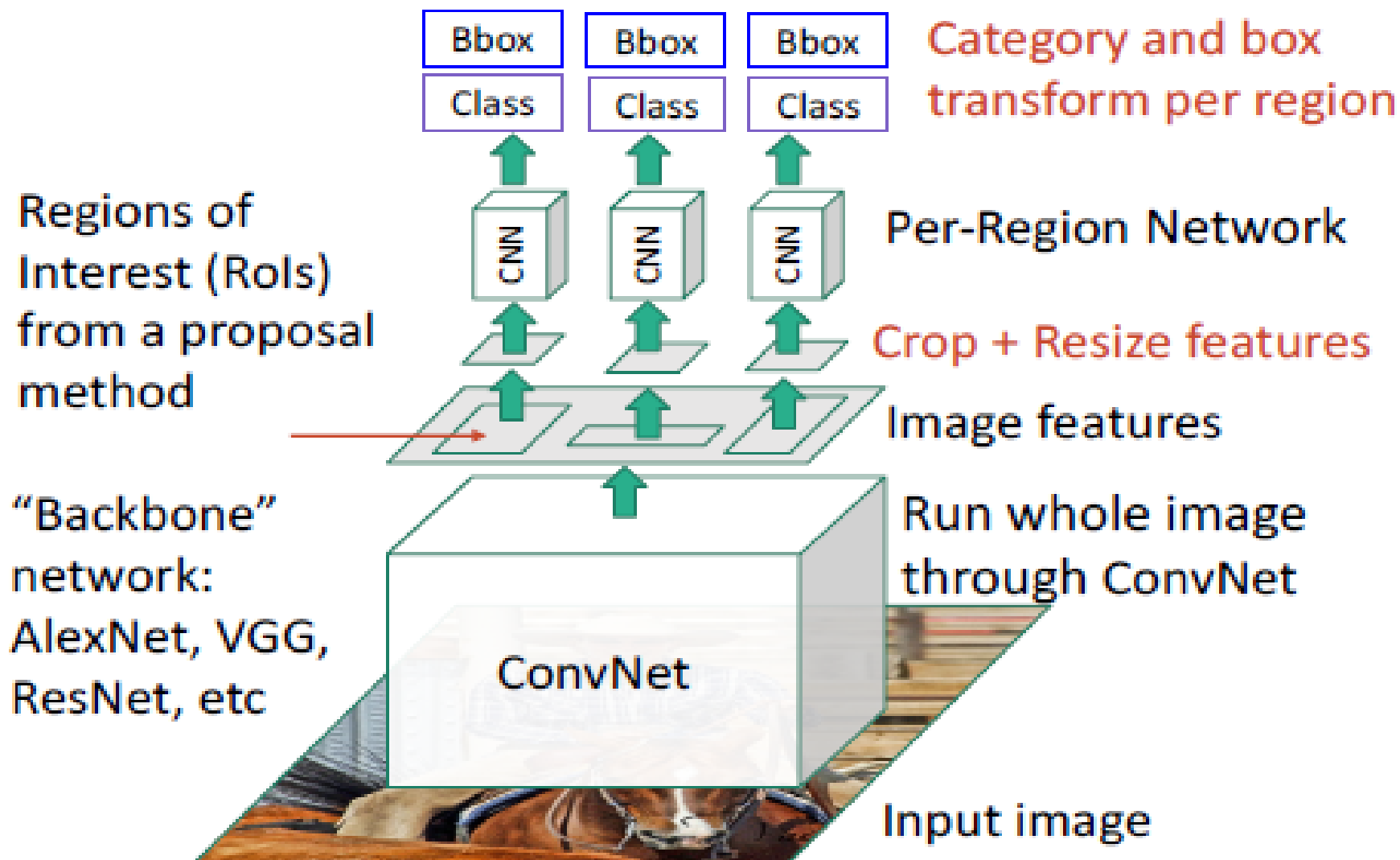
R-CNN: 영역 기반 CNN

- R-CNN (Region based CNN)



R-CNN: 영역 기반 CNN

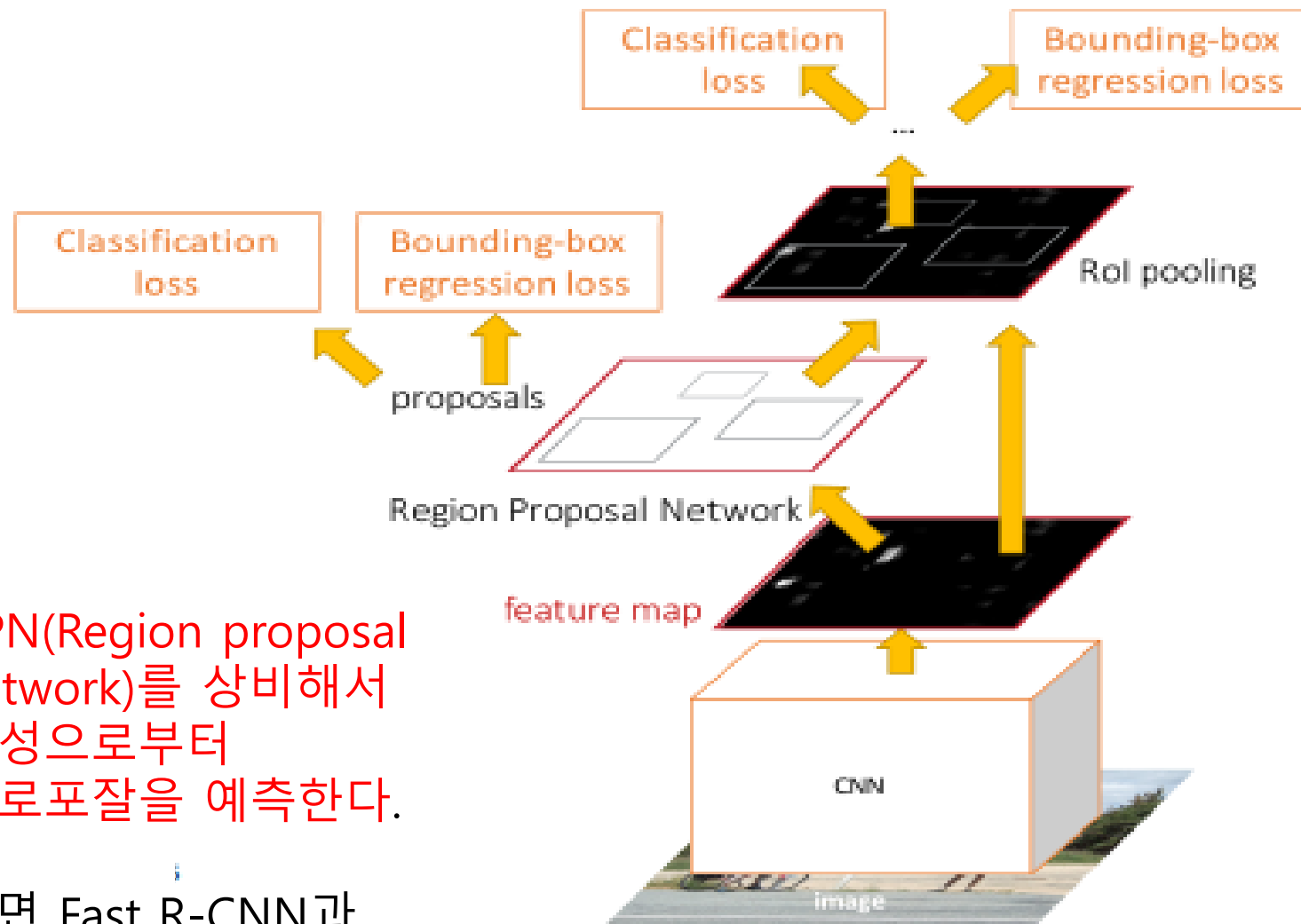
- Fast R-CNN



Girshick, "Fast R-CNN", ICCV 2015. Figure copyright Ross Girshick, 2015; [source](#). Reproduced with permission.

R-CNN: 영역 기반 CNN

- Faster R-CNN



- RPN(Region proposal network)를 상비해서 특성으로부터 프로포절을 예측한다.

(아니면 Fast R-CNN과 동일)

R-CNN: 영역 기반 CNN

- 2단계 탐지기

Faster R-CNN: Learnable Region Proposals

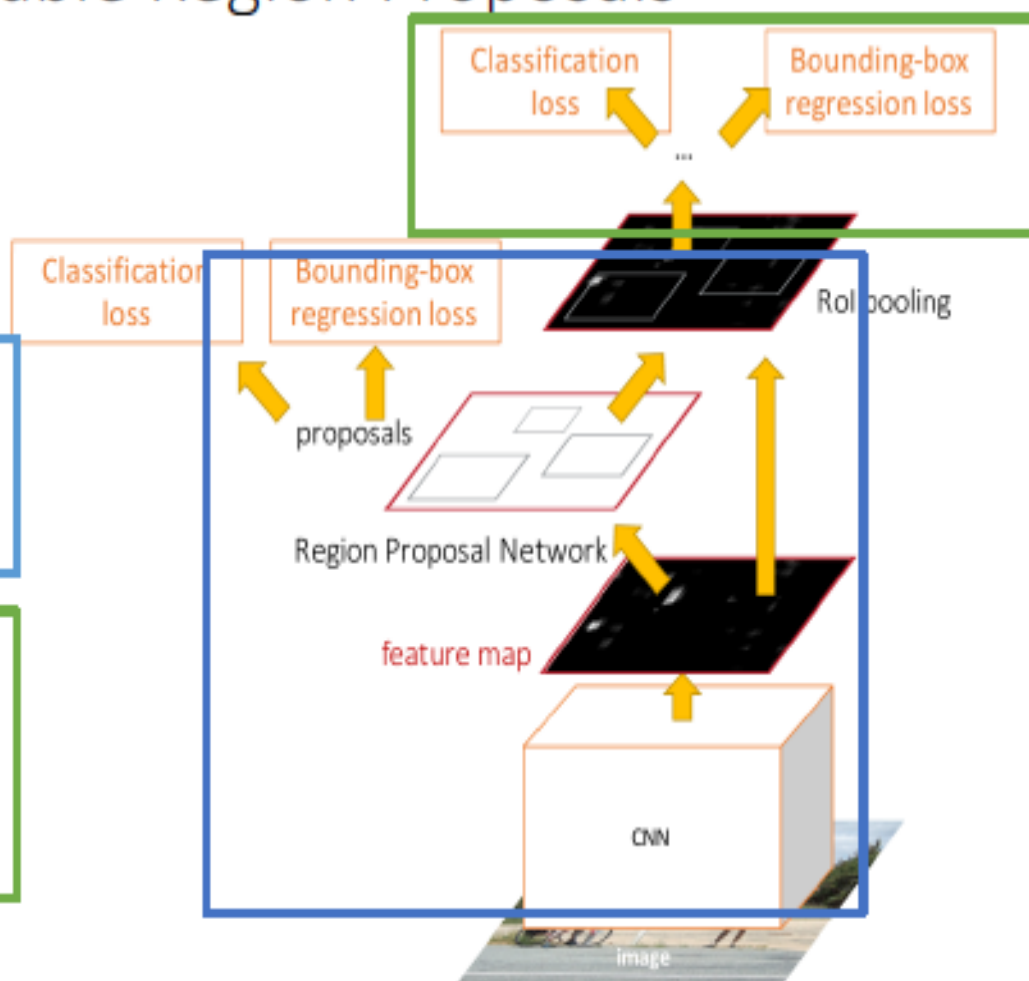
Faster R-CNN is a
Two-stage object detector

First stage: Run once per image

- Backbone network
- Region proposal network

Second stage: Run once per region

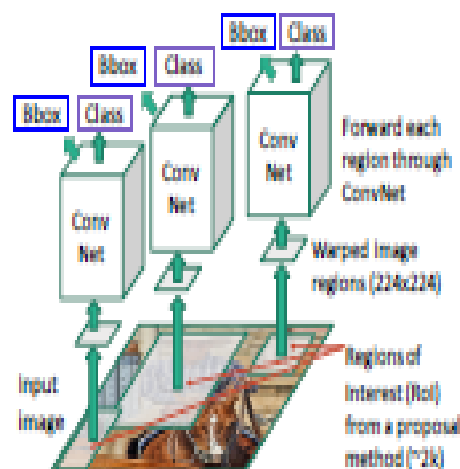
- Crop features: RoI pool / align
- Predict object class
- Prediction bbox offset



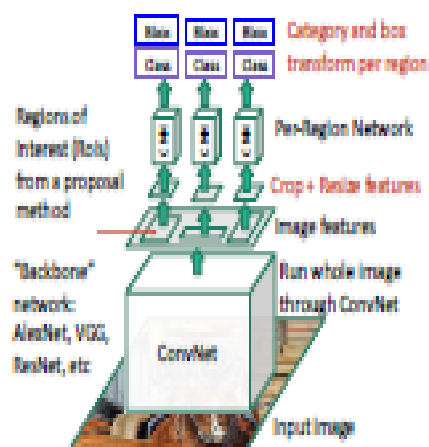
R-CNN: 영역 기반 CNN

- R-CNN 요약

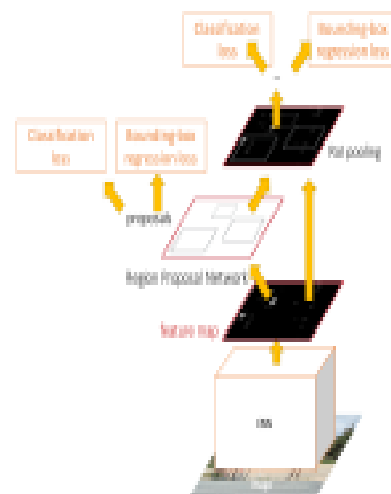
“Slow” R-CNN: Run CNN independently for each region



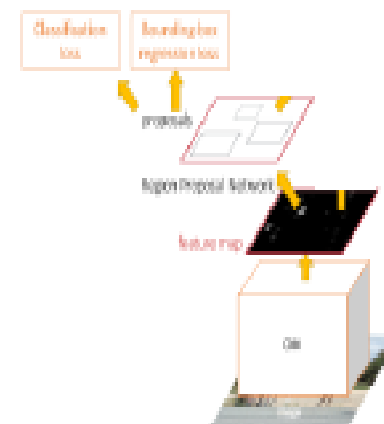
Fast R-CNN: Apply differentiable cropping to shared image features



Faster R-CNN: Compute proposals with CNN



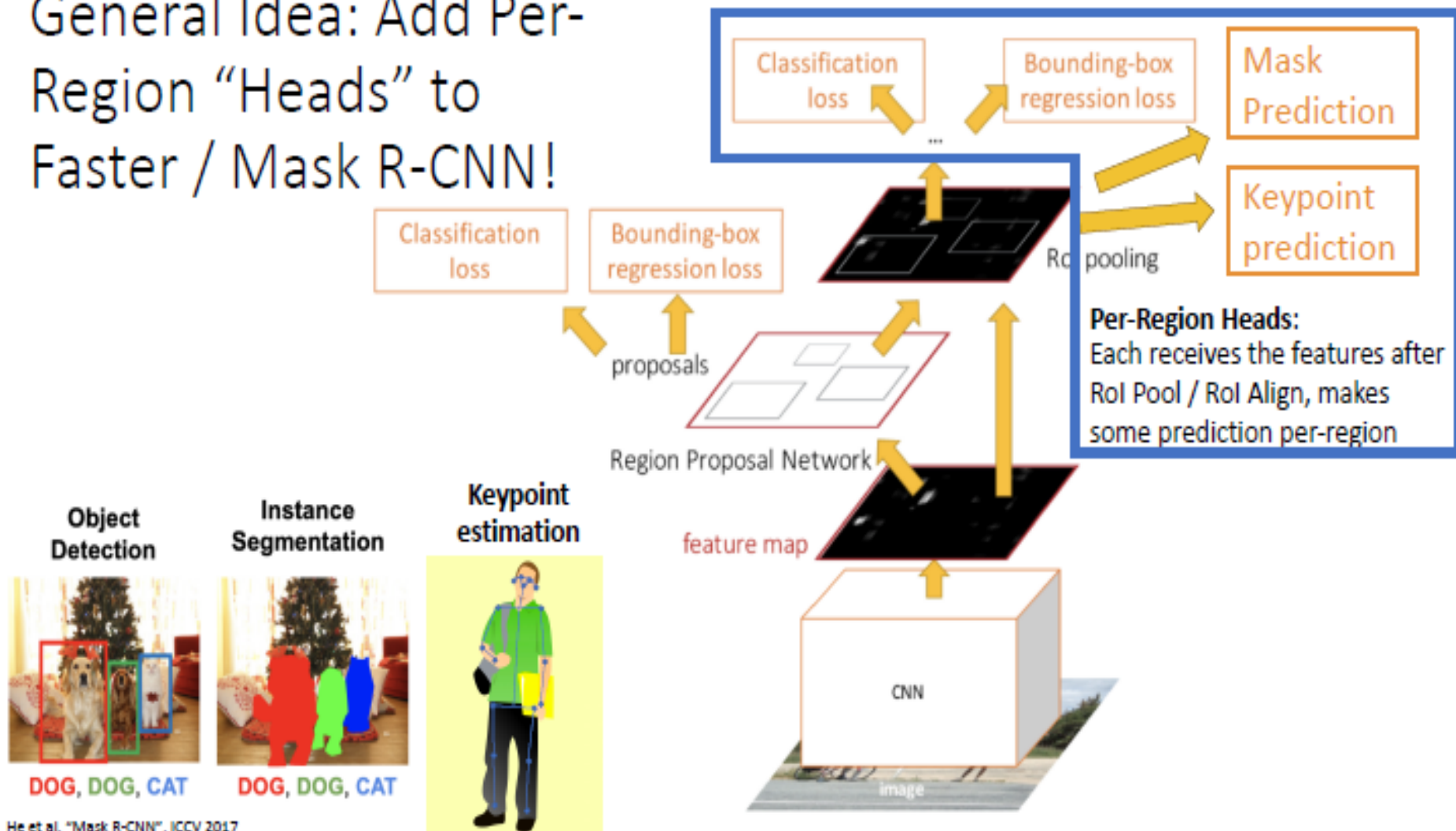
Single-Stage: Fully convolutional detector



이미지 분할

- Mask R-CNN

General Idea: Add Per-Region “Heads” to Faster / Mask R-CNN!



He et al, "Mask R-CNN", ICCV 2017

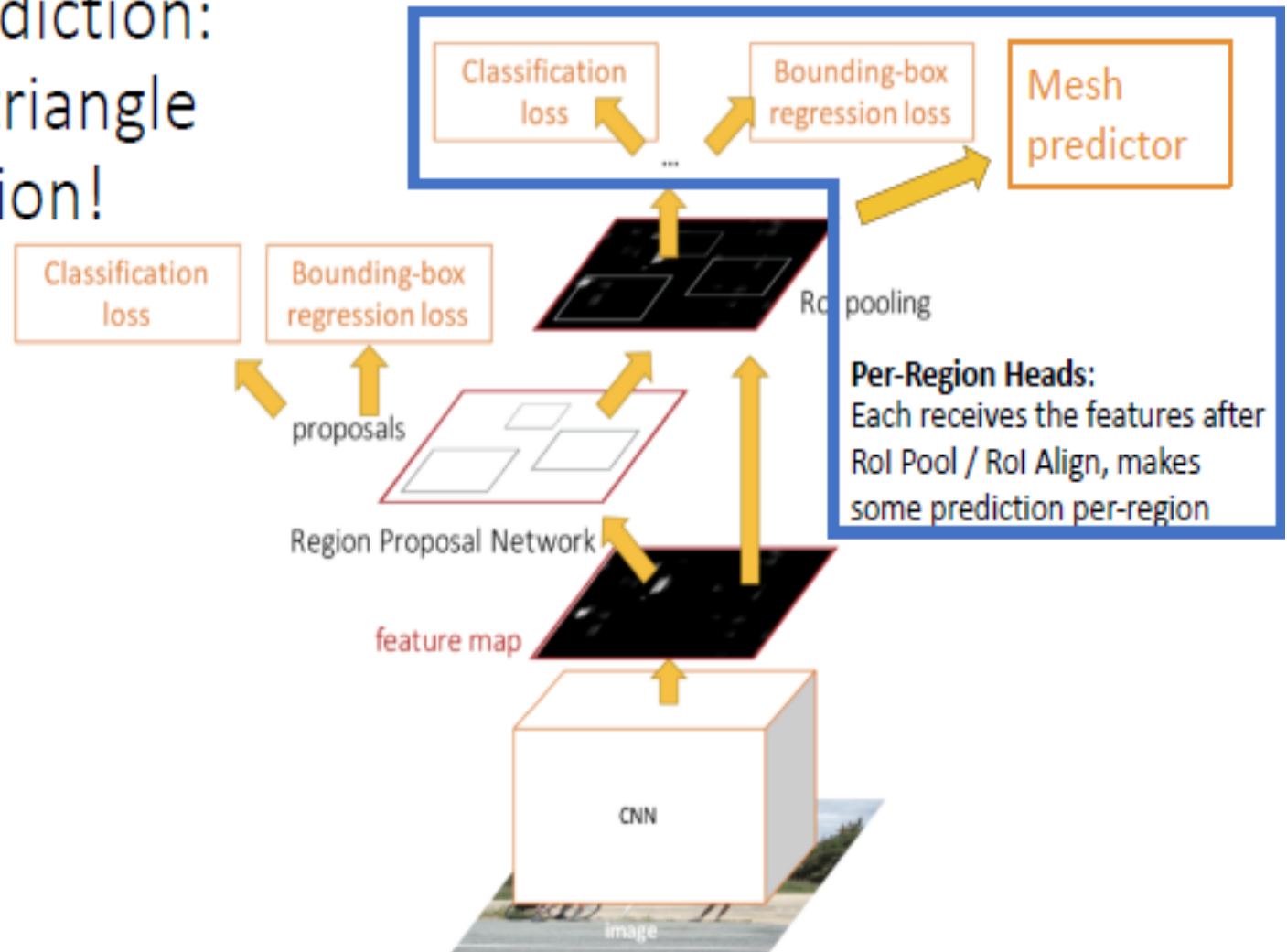
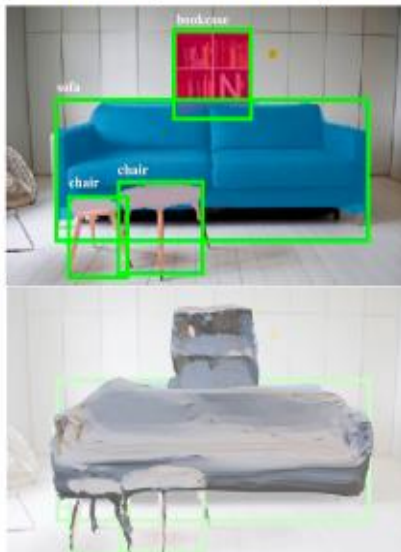
3D 모양 예측

- Mask R-CNN + Mesh Head

3D Shape Prediction:
Predict a 3D triangle
mesh per region!

Mesh R-CNN:

2D Image -> 3D shapes

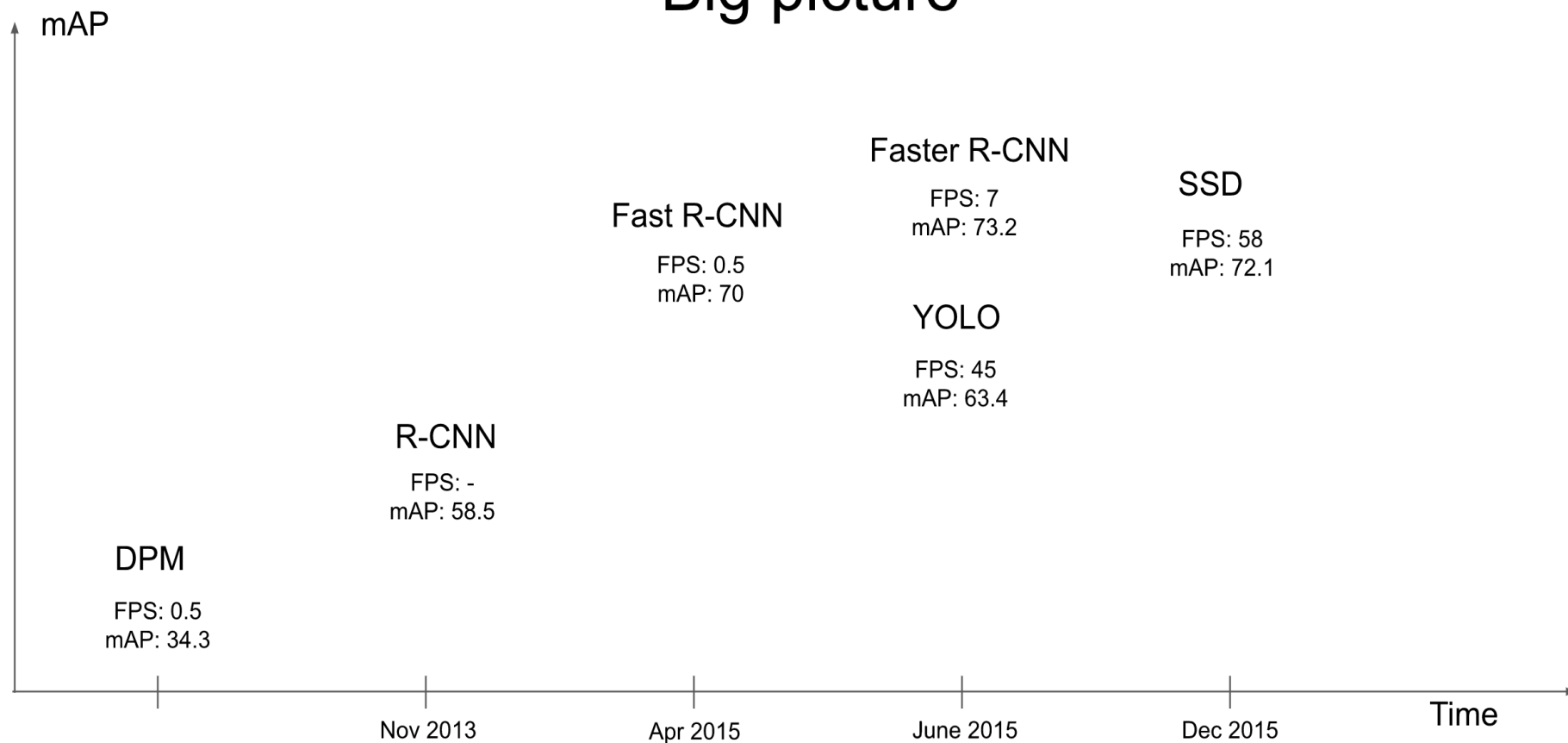


실무적 1단계 객체 탐지

객체 탐지

- 객체 탐지의 역사

Big picture

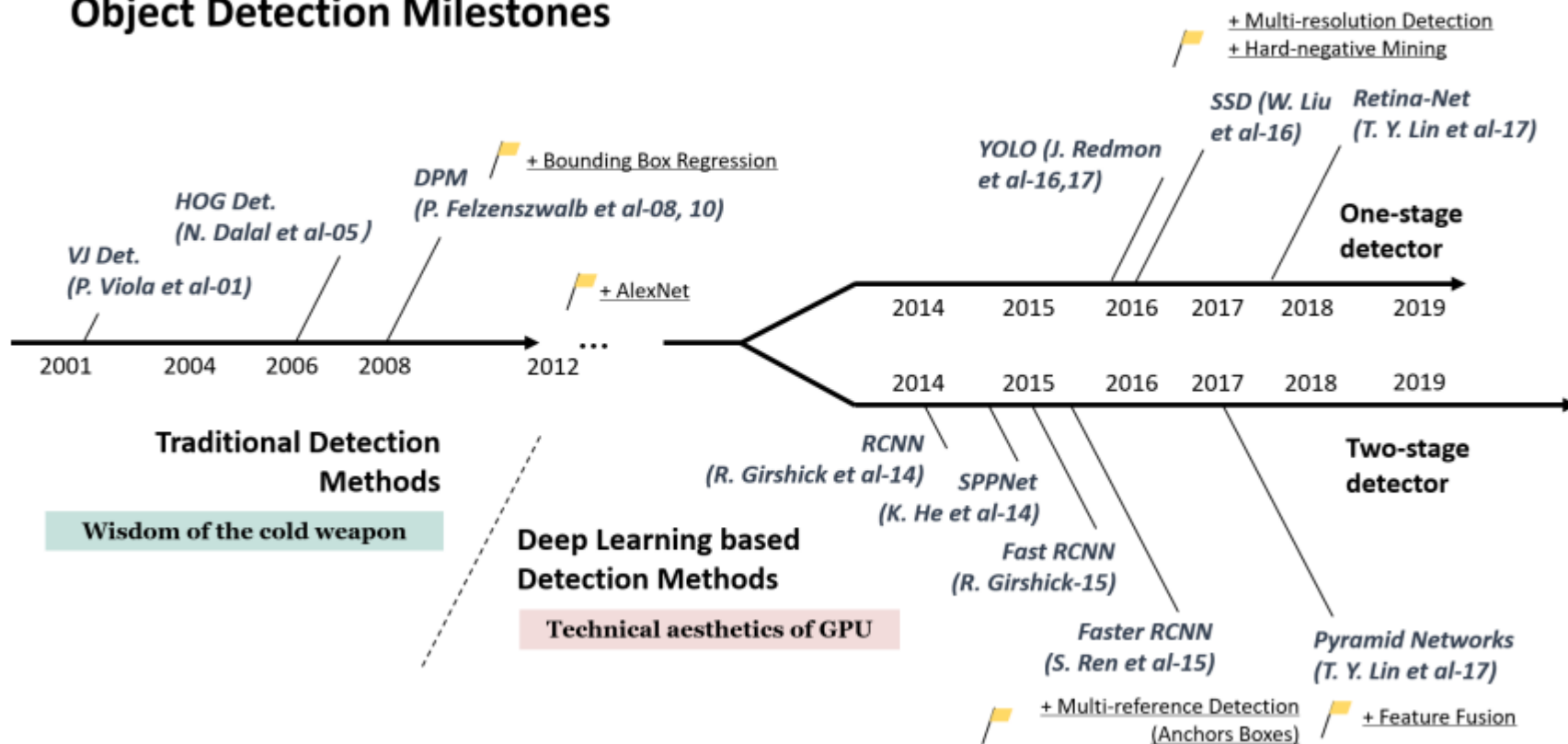


Результаты на тестовой выборки Pascal VOC 2007. Обучение на trainval sets 2007+2012

객체 탐지:작업 정의

- 영역 프로포잘: 모든 객체를 커버할 박스 집합을 발견

Object Detection Milestones



YOLO (You only look once)

- 통합된 탐지: 영역 프로포잘과 분류를 통합

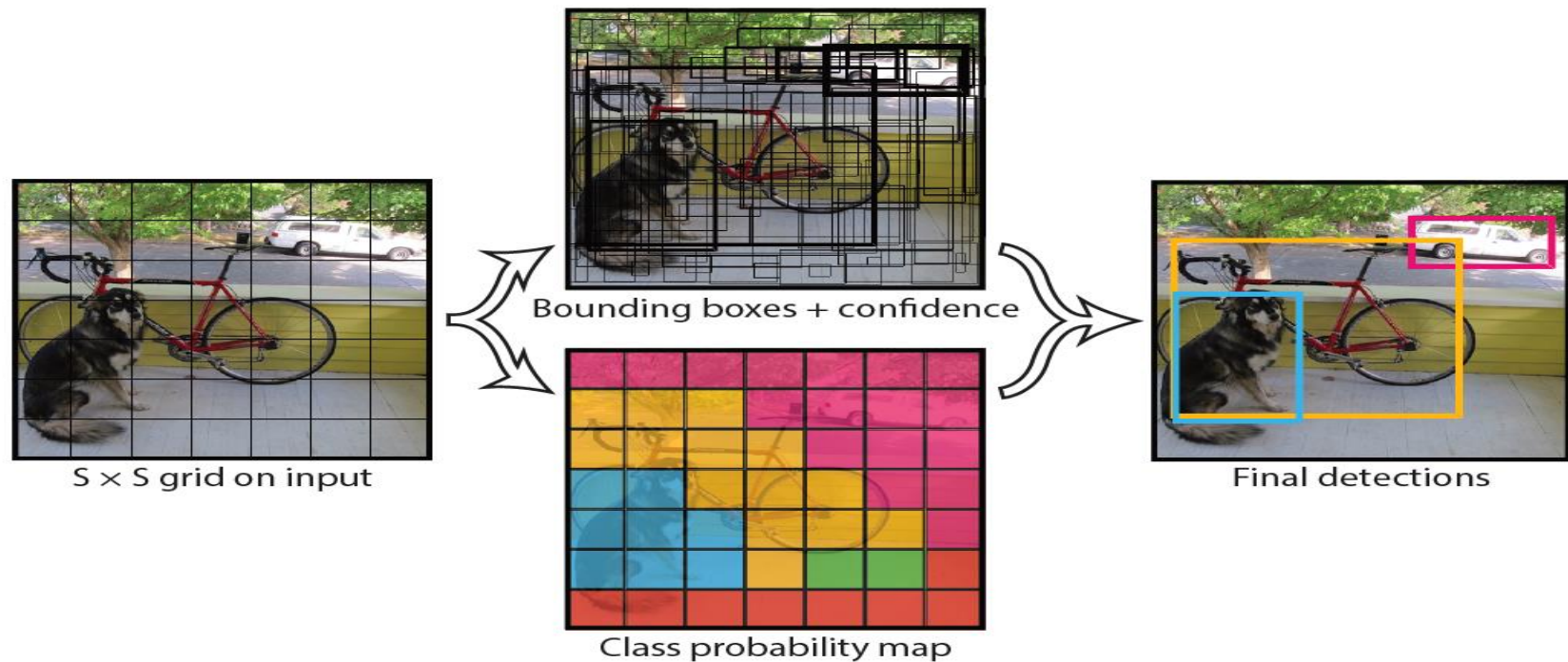
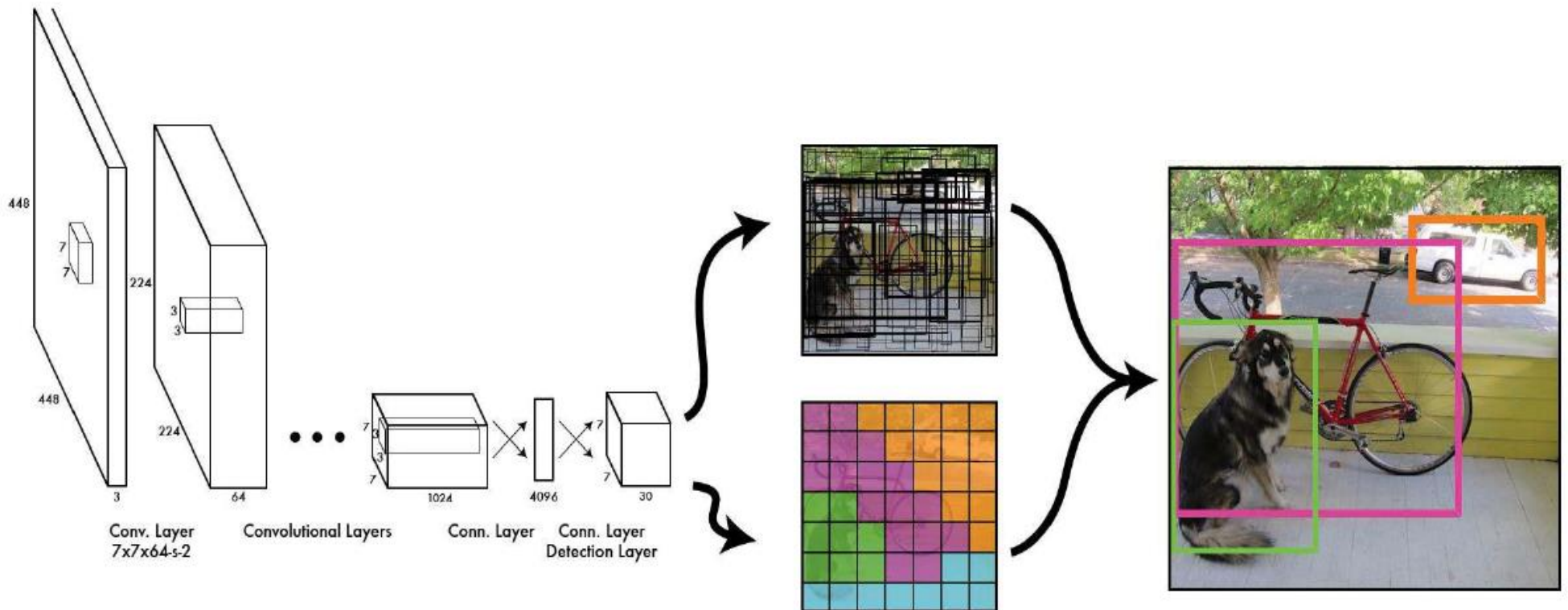


Figure 2: The Model. Our system models detection as a regression problem. It divides the image into an $S \times S$ grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. These predictions are encoded as an $S \times S \times (B * 5 + C)$ tensor.

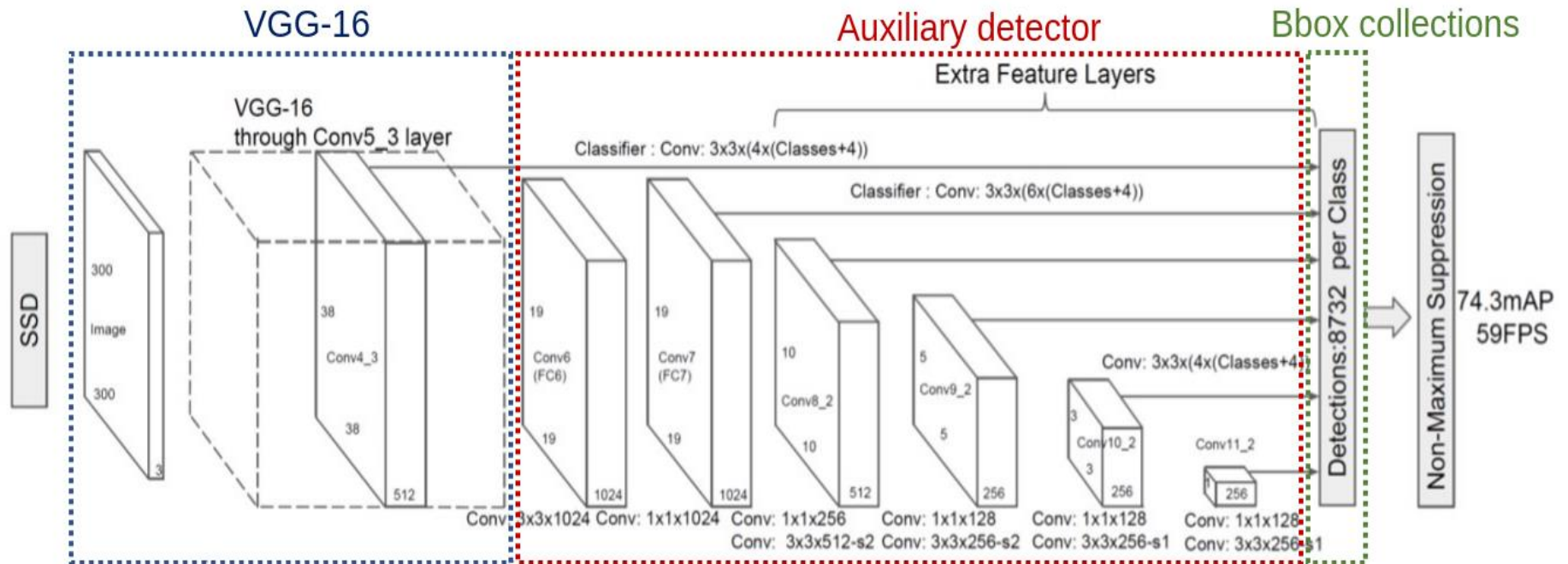
YOLO (You only look once)

- YOLO 파이프 라인: 24개의 conv 층과 2개의 FC층
 - 분류 성능은 떨어지지만 속도 향상을 달성



SSD (Single Shot MultiBox Detector)

- YOLO 이후에 나온 것으로 당시 최고 기술인 2단계 Faster R-CNN을 초월

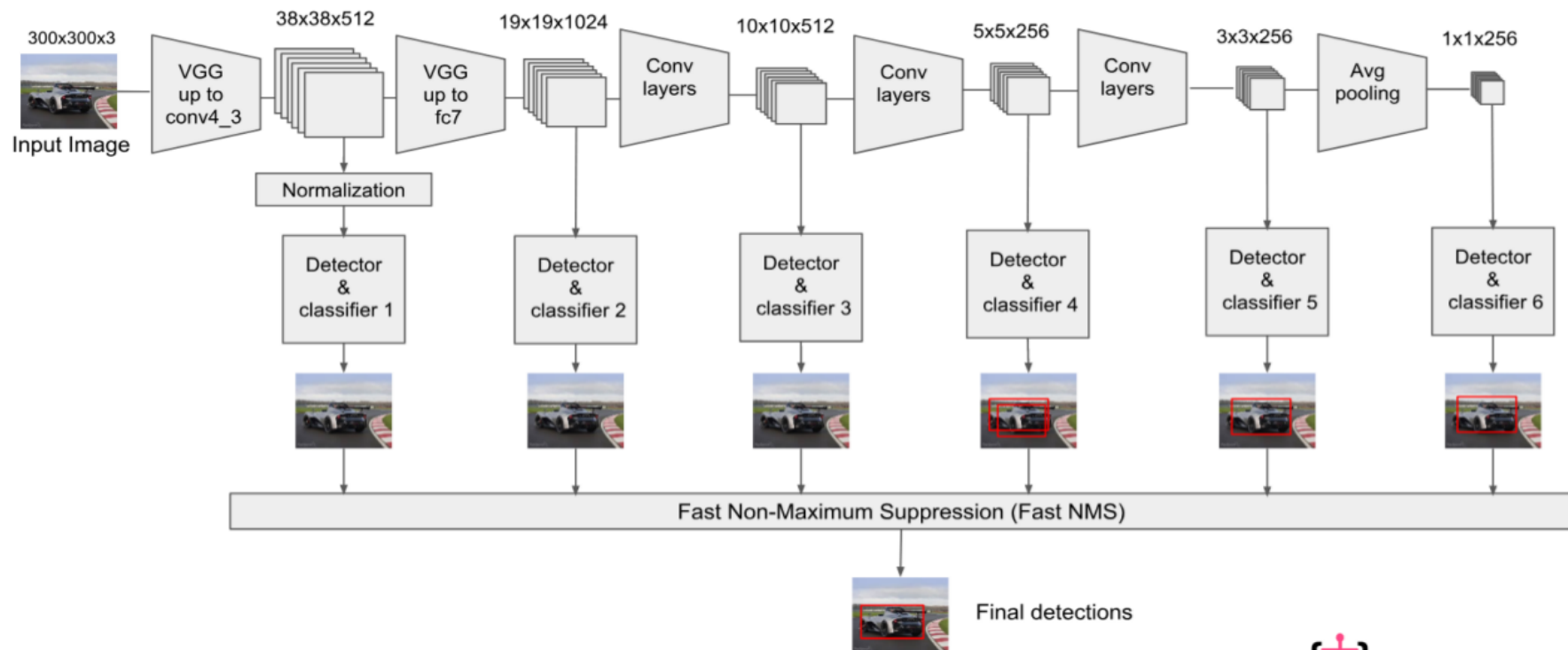


- 3등분 구조
 - VGG-16
 - 보조 탐지기로 정보 추출하고
 - 2에서 추출된 정보를 사용해 bbox 예측(회귀)

SSD (Single Shot MultiBox Detector)

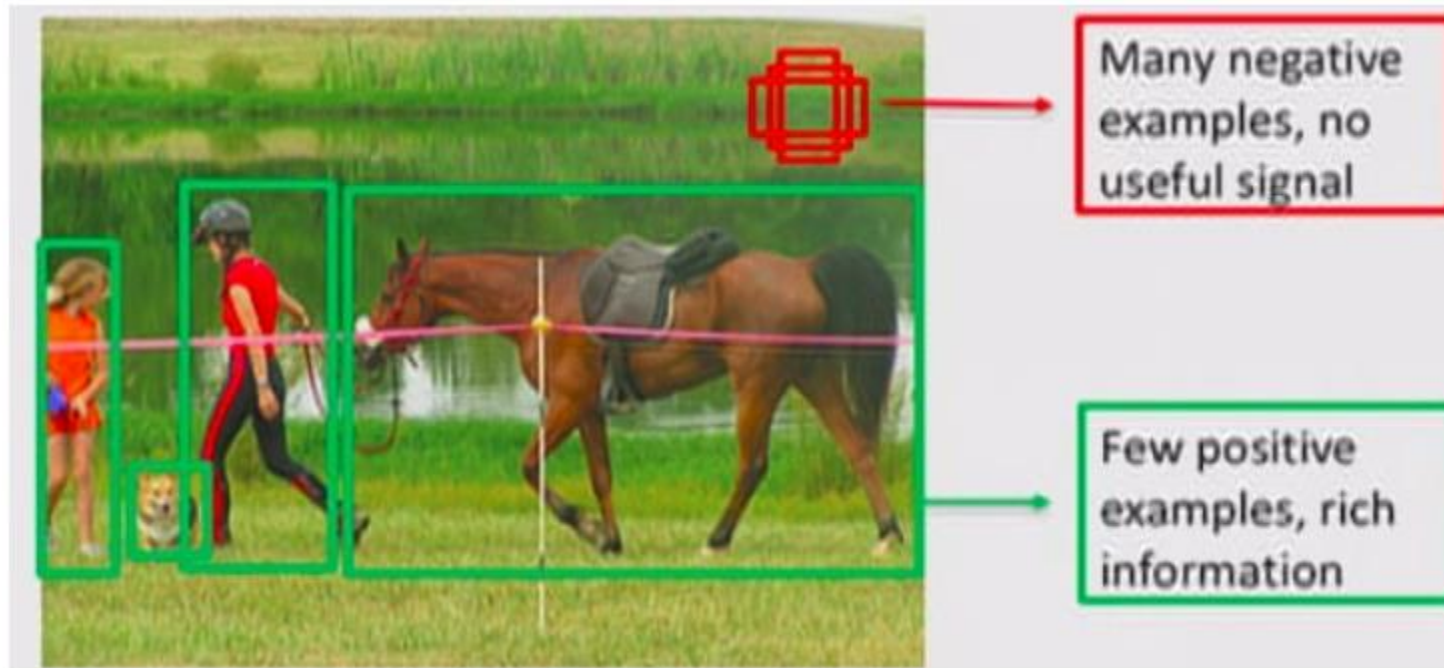
- 핵심 아이디어: 한개의 물체를 다양한 크기의 bbox를 사용해 다해상도(multi-resolution) 환경에서 예측하는 것

Choosing Scales and Aspect Ratios for Default Boxes



RetinaNet

- 데이터의 클래스 불균형 문제



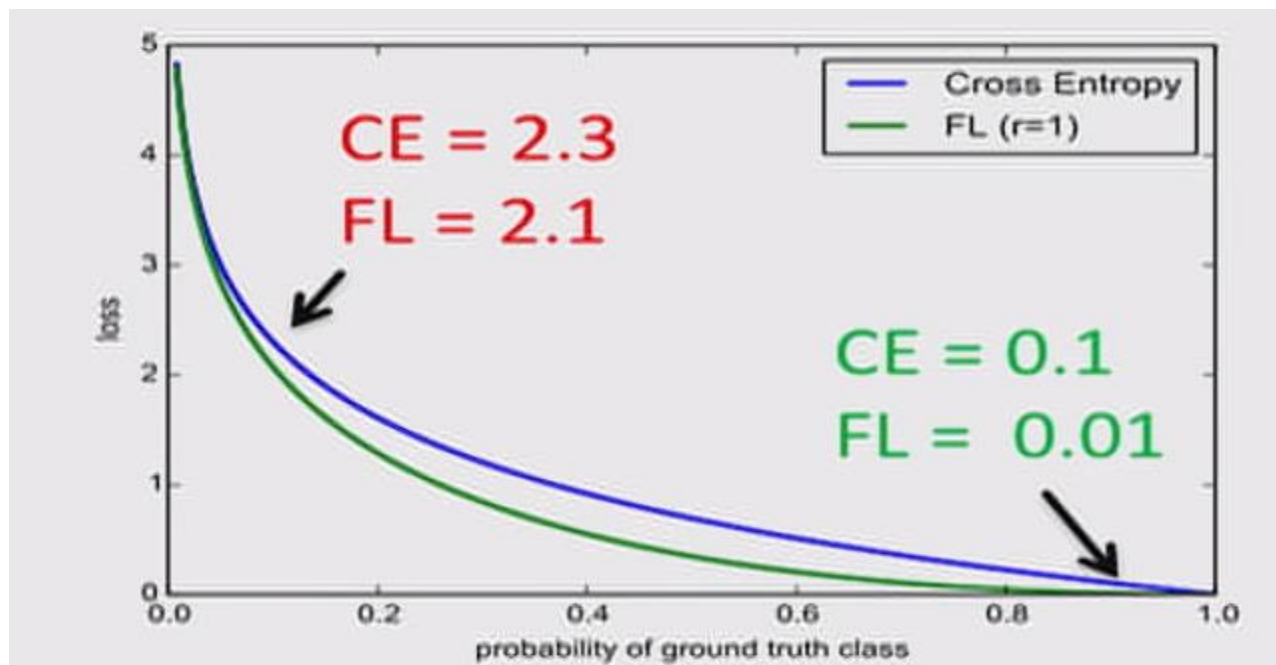
RetinaNet

- 교차 엔트로피(Cross Entropy)

$$\text{CE}(p, y) = \begin{cases} -\log(p) & \text{if } y = 1 \\ -\log(1 - p) & \text{otherwise.} \end{cases} \quad (1)$$

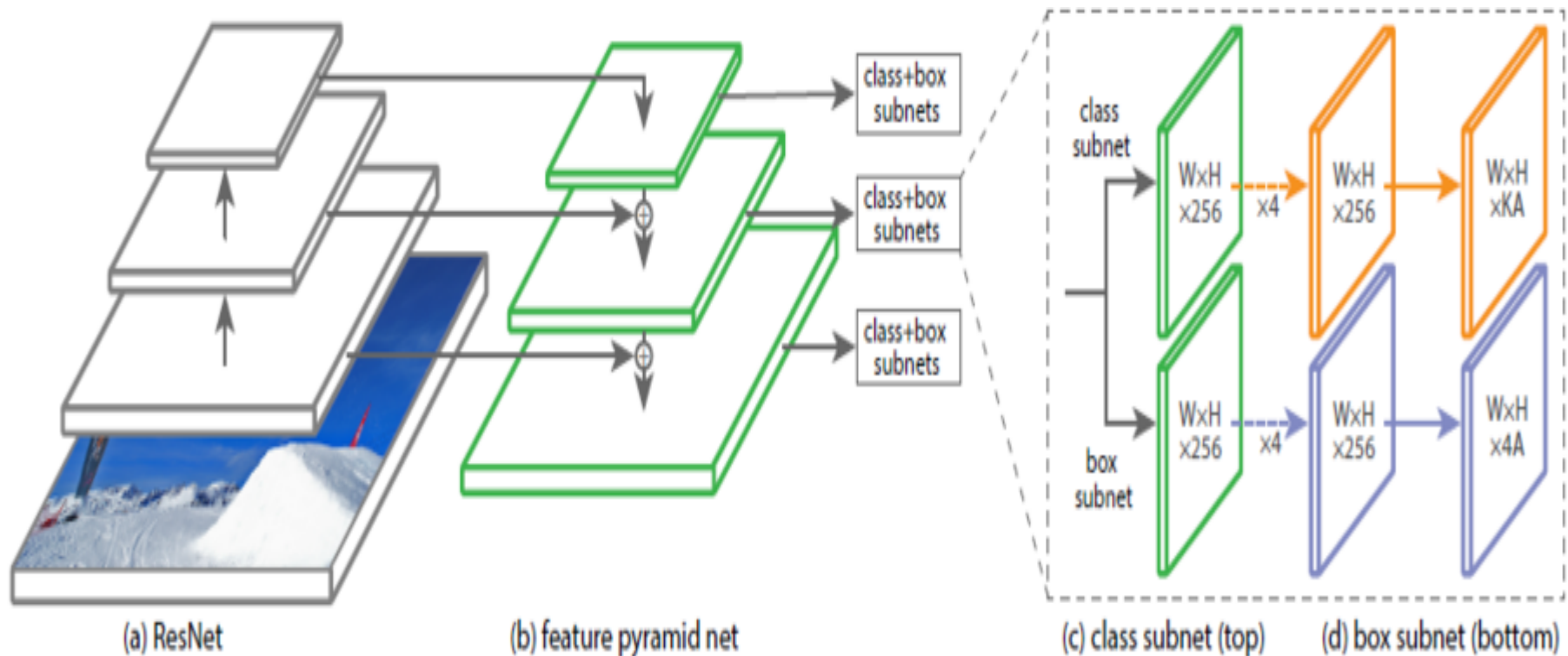
- Focal Loss

$$\text{FL}(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t)$$



RetinaNet

- 1단계 레티나넷 구조 (1 stage RetinaNet 구조):
 - 특성 피라미드 신경망(FPN: Feature Pyramid Network)



이미지 분할

이미지 분할

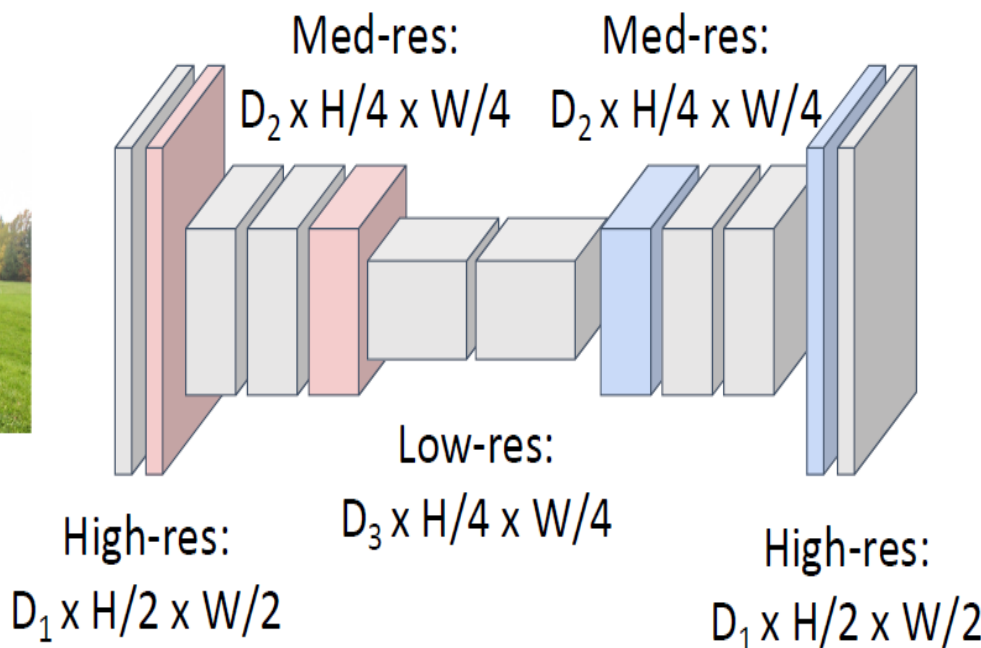
- 다운샘플링과 업샘플링의 합성곱층으로 구성된 네트워크 (Fully Convolutional Network)

Downsampling:
Pooling, strided
convolution



Input:
 $3 \times H \times W$

Design network as a bunch of convolutional layers, with **downsampling** and **upsampling** inside the network!



Upsampling:
Interpolation,
transposed conv



Predictions:
 $H \times W$

Loss function: Per-Pixel cross-entropy