



# [논문 리뷰] SSD [2016]

## Abstract

SSD 접근법은 경계상자의 출력 공간을 다양한 가로,세로 비율에 걸쳐 기본 상자 집합으로 이산화하고, feature map을 위치별로 조정합니다.

예측 시, 각각의 기본 상자에서 각각의 객체에 대한 점수를 매기고, 객체 모양과 더 잘 맞는 box를 생성하고, 조정합니다. 추가적으로, 객체의 다양한 size를 다루기 위해서, different resolution과 multiple feature map으로부터, 예측을 앙상블합니다.

제안영역을 제거하고, feature를 resampling 하고, 단일 Network를 캡슐화하기 때문에, SSD는 object proposals이 필요한 비교적 간단한 방법입니다.

## Introduction

지금까지의 Object Detection은, 속도가 증가하면, 탐지의 정확도가 크게 저하된다는 문제점이 있습니다.

이러한 문제로, Realtime Object Detection을 하는 경우, 성능이 저하되는 문제점이 있었습니다.

SSD는 경계 상자 가설을 위한 픽셀 또는 기능을 샘플링하지 않고, 접근방식만큼 정확한 성능을 보여줍니다.

경계 상자를 제안하고, subsequent pixel 또는 feature resampling stage 단계를 제거함으로써 속도가

상당히 개선되었습니다.

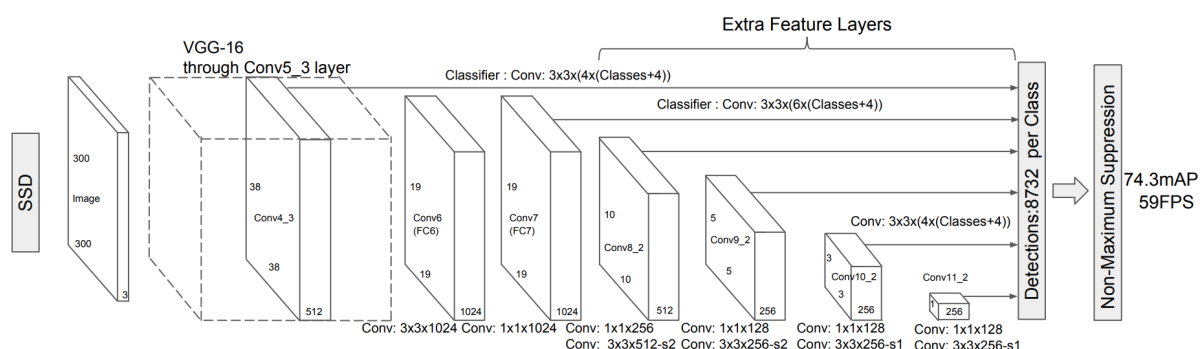
Accuracy를 개선하기 위해, object categories를 small convolution filter를 활용하였으며, 다양한 측면에서 예측을 위해 separate predictors를 활용하였습니다. 그리고, 다중 스케일에서 탐지를 수행하기 위해서 Network의 말단에서 multiple feature maps를 filter로 적용하였습니다.

상대적으로 낮은 해상도를 input으로 활용하여 높은 정확도를 성취하였습니다 뿐만 아니라, 속도 또한 매우 향상되었다는 것을 알 수 있었습니다.

## Summarize

- SSD(Single Shot detector)을 도입하였습니다. 이전에 YOLO1보다 성능이 좋으며, 정확도 또한 개선되었습니다.
- **SSD(Single Shot detector)**의 핵심은 category score를 예측합니다. 그리고 **small convolutional filter**를 사용함으로써, **bounding box**에 대한 **fixed set**을 활용하였습니다.
- 높은 정확도를 가지고, **빠른 성능**을 가지고 있으며, 다양한 **scale**의 **feature map**으로부터, **예측**을 진행합니다.

## Model



**SSD approach**는 **feed-forward 경계 상자의 고정된 크기의 집합**을 생성하고, 객체의 존재에 관하여 score를 매기는 **Convolutional Network**를 기반 하에 사용됩니다.

초기의 network layer들은 일반적인 high quality image classification을 위해 standard architecture를 사용합니다. 이후, **Network에 여러가지 다양한 Network**를 추가합니다.

### Multi-scale feture maps for detection

CNN feature layer의 끝에 base Network의 질린 마지막 부분을 추가합니다. 그리고, layer의 size를 점진적으로 감소시키며, 다양한 측면에서 탐지 및 예측을 허용합니다.

### Convolutional predictors for detection

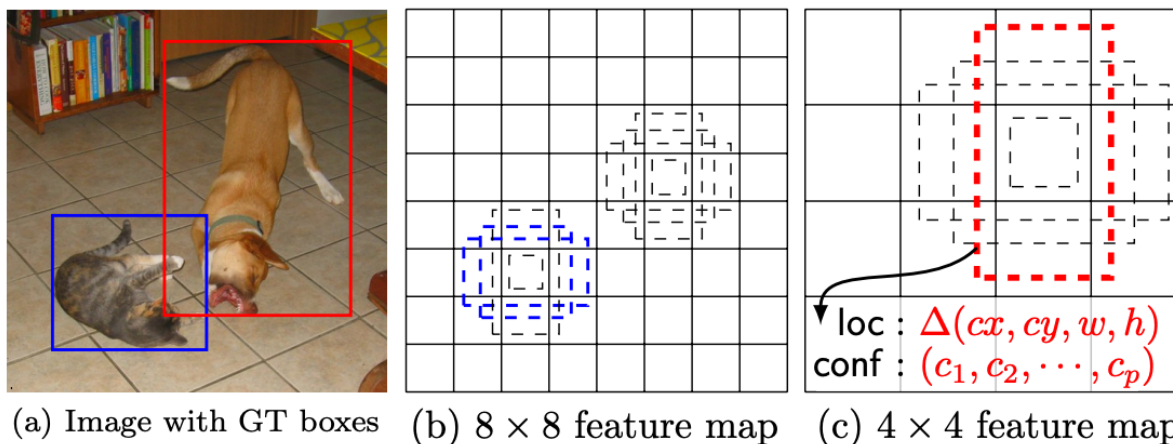
추가된 각 layer들은 Convolution filter를 사용하고, 고정된 예측 및 탐지를 하는 것이 가능합니다.

### Default boxes and aspect ratios

네트워크 상단에 있는 여러 feature map들은 각 feature map의 bounding box와 연결됩니다.

일반적인 box들은 Convolution 방식으로 feature map을 잘라냅니다.

우리는 default box shapes를 예측을 통하여 상쇄합니다.  
 Feature map cell에 다양한 크기를 사용하여, default box를 고정시킵니다.



각 feature map cell에는 default box가 위치하게 되고, default box에 상대적인 바운딩박스 offset과 class score를 예측합니다.  
 각 feature map cell에서  $3 \times 3$  크기에  $(c+4)k$ 개의 filter가 적용됩니다.

## Training

기존의 Model과의 차이점으로는, **ground truth information을 활용하여, detector outputs 내에 구체적인 output이 필요합니다.**

**Training시에는 default box와 detection을 위한 scale방식이 필요합니다.**  
 뿐만 아니라, hard negative mining 방식과 data augmentation 전략 또한 필요합니다.

## Matching Strategy

학습 시, Ground truth detection에 해당하는 default box를 선정해야 합니다.  
 뿐만 아니라 학습을 진행해야 합니다. IOU 0.5이상을 기준으로 ground truth를 default box를 매칭합니다. 뿐만 아니라, 가장 높은 IoU를 선택하는 것보다,  
 여러가지 **default box를 선정하는 것이 high score를 내는 것보다 학습을 간소화합니다.**

## Training Objective

SSD는 MultiBox objective로부터, 도출되었지만, object categories를 다루기 위해 확장되었습니다.

각 객체 클래스마다 default box와 경계 box가 매칭되는지 여부를 확인합니다.

**전반적인 objective loss function은 localization loss 와 confidence loss의 weighted sum으로 이루어져 있습니다.**

$$L(x, c, l, g) = \frac{1}{N} ( \underbrace{L_{conf}(x, c)}_{\text{신뢰도 손실(confidence loss)}} + \underbrace{\alpha L_{loc}(x, l, g)}_{\text{가중치 Localization 손실}} )$$

**N**은 **default box**의 개수입니다. 만약  $N=0$ 이라면, 전체 손실 값도 0이 됩니다. SSD에서는 가중치 파라미터를 1로 설정합니다.

**localization** 손실은 예측한 디폴트 박스(**l**)와 참 경계 박스(**g**) 사이의 **Smooth L1** 손실입니다.

**Faster R-CNN**과 비슷하게, 디폴트 박스의 좌표는 중앙 **x**값, 중앙 **y**값, 너비, 높이로 예측합니다.

$$L_{loc}(x, l, g) = \sum_{i \in Pos} \sum_{m \in \{cx, cy, w, h\}} x_{ij}^k \text{smooth}_{L1}(l_i^m - \hat{g}_j^m)$$

$$\hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx}) / d_i^w \quad \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy}) / d_i^h$$

$$\hat{g}_j^w = \log \left( \frac{g_j^w}{d_i^w} \right) \quad \hat{g}_j^h = \log \left( \frac{g_j^h}{d_i^h} \right)$$

## Choosing scales and aspect ratios for default boxes

다양한 크기의 객체를 다루기 위해서, 다양한 크기와 결과를 결합하는 방식으로 이미지를 접근하는 방식을 제안합니다.

그러나 예측을 위해서 단일 네트워크에서 여러 다른 layer로부터 **feature map**을 활용함으로써, 동일한 효과를 모방하는 것이 가능합니다. 뿐만 아니라, parameter를 공유하는 것이 가능합니다.

**Feature Map**으로부터 **global context pooled**을 추가한 것은 **segmentation results**를 **smooth**하는데 도움을 줍니다.

**Network**내에 서로 다른 수준의 **feature map**은 다양한 수용영역의 **field sizes**를 가지는 것으로 알려져 있습니다. 우리는 기본 상자를 자르는 방식으로 design하였습니다. 그리하여, 특정한 **feature map**은 객체의 각각의 **scale**에 대응하도록 학습합니다.

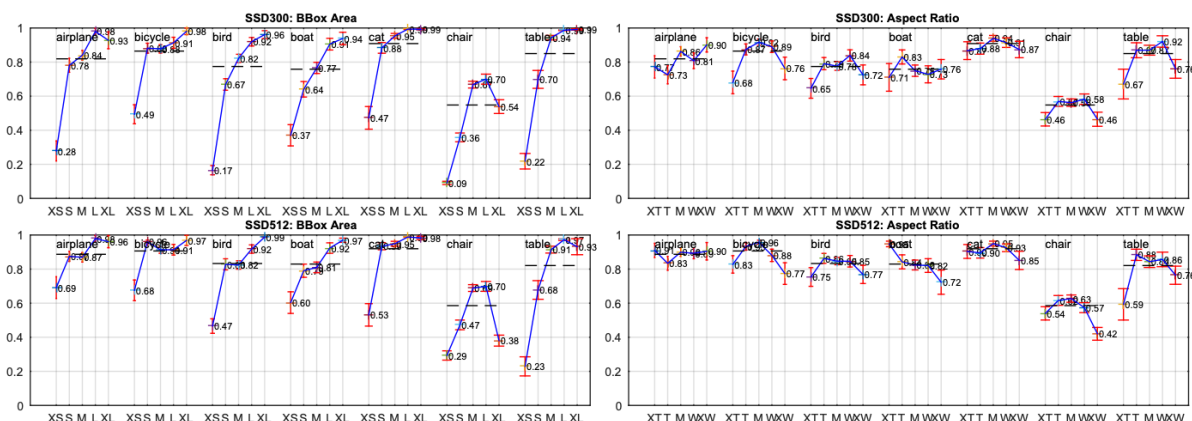
## Hard negative mining

**Default box**를 매칭하고 나면, 전체 **default box**의 개수는 상당히 많지만, 경계 박스와 매칭된 **default box**는 적다.

모든 negative 훈련 샘플을 사용하는 대신에, 신뢰도 손실 점수를 기반으로 negative sample을 제외합니다. 그리하여, 최종적으로 negative sample과 positive sample의 비율이 3:1이 됩니다.

## Experiments Results

Method	data	mAP	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
Fast [6]	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
Fast [6]	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
Faster [2]	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
Faster [2]	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
Faster [2]	07+12+COCO	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
SSD300	07	68.0	73.4	77.5	64.1	59.0	38.9	75.2	80.8	78.5	46.0	67.8	69.2	76.6	82.1	77.0	72.5	41.2	64.2	69.1	78.0	68.5
SSD300	07+12	74.3	75.5	80.2	72.3	66.3	47.6	83.0	84.2	86.1	54.7	78.3	73.9	84.5	85.3	82.6	76.2	48.6	73.9	76.0	83.4	74.0
SSD300	07+12+COCO	79.6	80.9	86.3	79.0	<b>76.2</b>	57.6	87.3	88.2	88.6	60.5	85.4	<b>76.7</b>	<b>87.5</b>	<b>89.2</b>	84.5	81.4	55.0	81.9	<b>81.5</b>	85.9	78.9
SSD512	07	71.6	75.1	81.4	69.8	60.8	46.3	82.6	84.7	84.1	48.5	75.0	67.4	82.3	83.9	79.4	76.6	44.9	69.9	69.1	78.1	71.8
SSD512	07+12	76.8	82.4	84.7	78.4	73.8	53.2	86.2	87.5	86.0	57.8	83.1	70.2	84.9	85.2	83.9	79.7	50.3	77.9	73.9	82.5	75.3
SSD512	07+12+COCO	<b>81.6</b>	<b>86.6</b>	<b>88.3</b>	<b>82.4</b>	76.0	<b>66.3</b>	<b>88.6</b>	<b>88.9</b>	<b>89.1</b>	<b>65.1</b>	<b>88.4</b>	73.6	86.5	88.9	<b>85.3</b>	<b>84.6</b>	<b>59.1</b>	<b>85.0</b>	80.4	<b>87.4</b>	<b>81.2</b>



## Conclusion


Multiple Categorie를 위해서 매우 빠른 Single-shot object detection을 도입하였습니다. 우리의 모델의 핵심은 Multiple feature maps을 다루기 위해, output box에 접근하는 것이다.

feature Map의 모든 Pixel마다 크기와 가로가 다양한 default box를 활용하는 것이다. 이러한 특징 때문에 다양한 크기를 갖는 객체를 효율적으로 탐지할 수 있습니다.

## Reference

### SSD: Single Shot MultiBox Detector

We present a method for detecting objects in images using a single deep neural network. Our approach, named SSD, discretizes the output space of bounding boxes into a set of

 <https://arxiv.org/abs/1512.02325>



<https://bkshin.tistory.com/entry/%EB%85%BC%EB%AC%B8-%EB%A6%AC%EB%B7%B0-SSDSingle-Shot-MultiBox-Detector-%ED%86%BA%EC%95%84%EB%B3%B4%EA%B8%B0>