YOLO

You Only Look Once: Unified, Real-Time Object Detection

▼ 1. 어떤 문제를 풀고자 했는가? (Abstract)

- object detection (이미지내 multiple object 검출)에 새로운 접근방식 적용
- 기존의 multi-task 문제를 하나의 regression 문제로 재정의
- 이미지 전체에 대해 하나의 신경망, 한번의 계산만으로 bounding box와 클래스 확률을 예측

▼ 2. 어떤 동기/상황/문제점에서 이 연구가 시작되었는가? (Introduction)

💡 - object detection : 분류뿐만 아니라 위치 정보도 판단해야함!
기존) DPM, R-CNN : 복잡함, 처리 속도 느림, 최적화 어렵
→ **YOLO 등장!** : You Only Look Once 한번만 보면 객체 검출가능

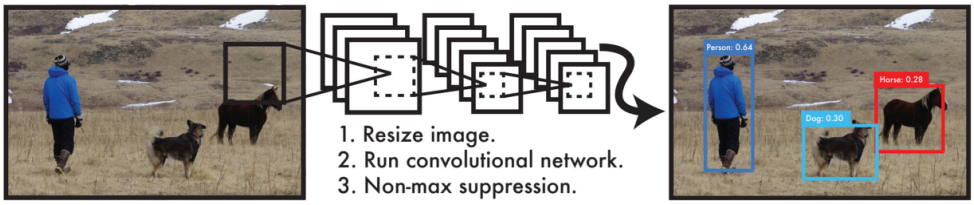


Figure 1: The YOLO Detection System. Processing images with YOLO is simple and straightforward. Our system (1) resizes the input image to 448 × 448, (2) runs a single convolutional network on the image, and (3) thresholds the resulting detections by the model's confidence.

- 객체 검출을 하나의 회귀 문제로 보고 절차 개선
- (1) 하나의 convolution network가 여러 bounding box와 그 bounding box의 클래스 확률을 동시에 계산
- (2) 이미지 전체를 학습해 바로 검출 성능을 최적화

YOLO의 장점

1. 굉장히 빠름 - 복잡했던 객체 검출 프로세스를 하나의 회귀 문제로 바꿈!
2. 훈련과 테스트 단계에서 이미지 전체를 바라봄! 주변정보까지 처리!
3. 물체의 일반적인 부분을 학습, 훈련단계에서 보지 못한 새로운 이미지에 대해 더 robust!

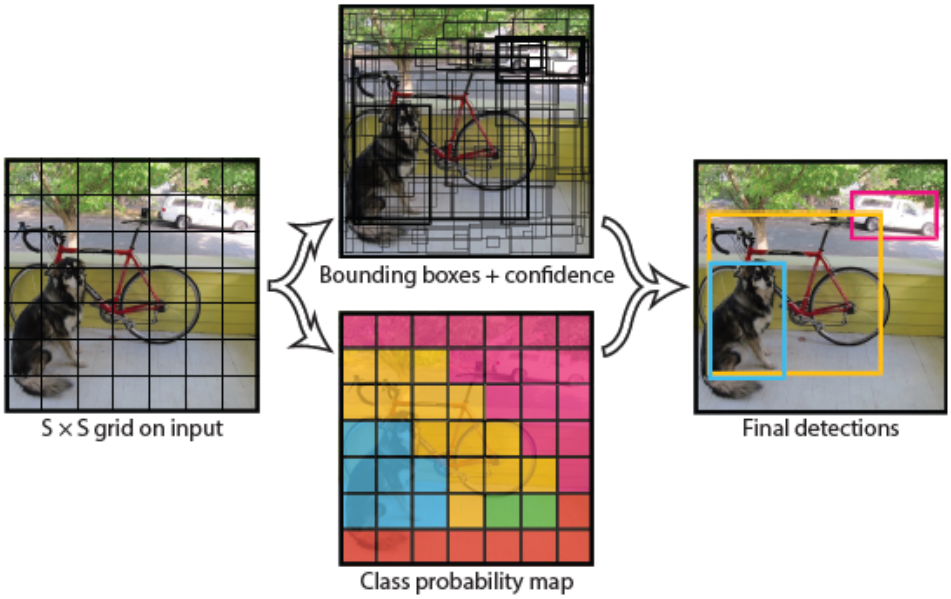
YOLO의 단점

- 최신 객체 검출 모델에 비해 정확도 떨어짐
- 특히 작은 물체에 대한 검출 정확도 ↓
- 속도와 정확성은 trade- off 관계

▼ 3. 이 연구의 접근 방법은 무엇인가?(Method)

1. **Unified Detection**

YOLO는 객체 검출의 개별요소를 **단일 신경망으로 통합**
→ 높은 정확도를 유지하면서 end-to-end training과 real-time speeds를 가능하게 함!
- 입력이미지를 S x S 그리드로 나눔
- grid cell 안에 object가 있으면 grid cell은 object를 탐지
- grid cell은 B개의 bounding box와 각 box의 신뢰도(confidence score) 예측



$$\Pr(Object) * IOU_{pred}^{truth}$$

- IOU : 객체의 실제 bounding box와 예측 bounding box의 합집합 면적 대비 교집합 면적의 비율
- 그리드 셀에 아무 객체 없으면 Pr(Object) = 0,
  이상적인 경우 Pr(Object) = 1
- 각각의 bounding box는 5개의 예측치 **x,y,w,h,confidence**로 구성
  - grid cell 중 object의 중앙과 가까운 cell이 object detection을 하게되며, 각각의 grid cell은 class의 확률인 C를 예측

$$C(conditional\ class\ probabilities) = \Pr(Class_i|Object)$$

: B가 background가 아닌 object를 포함하는 경우의 각 class 별 확률

- 테스트 단계에서는 C와 개별 bounding box의 confidence score를 곱해줌

$$
\begin{aligned}
&class\ specific\ confidence\ score \\
&= \Pr(Class_i|Object) * \Pr(Object) * IOU_{pred}^{truth} \\
&= \Pr(Class_i) * IOU_{pred}^{truth}
\end{aligned}
$$

= bounding box에 특정 클래스 객체가 나타날 확률과 예측된 bounding box가 그 클래스 객체에 얼마나 잘 들어맞는지!

2. **Network Design**

💡 하나의 CNN 구조로 디자인

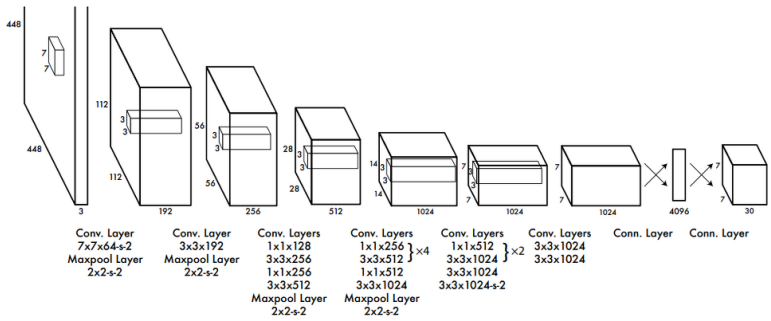


Figure 3: The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection.

- Conv layer - Fully connected layer
  Conv layer: 이미지로부터 특징 추출
  Fully connected layer : 클래스 확률과 bounding box의 좌표 예측
- GoogLeNet의 신경망 구조 따옴
  - 24개의 Conv layer, 2개의 fully connected layer
  - GoogLeNet의 인셉션 구조 대신 1x1 축소 계층과 3x3 컨볼루션 계층의 결합 이용

3. **Training**
4. **Inference(추론)**

💡 - 추론 단계에서도 test 이미지로부터 객체 검출할때 **하나의 신경망 계산**만 하면 됨!
- test 단계에서 특히 빠름! 하나의 신경망 계산만 해주면 되기 때문
- but, **다중검출** 문제점 존재!
: 객체의 크기가 작거나 객체가 경계에 인접하는 경우, 하나의 객체를 여러 그리드 셀이 동시에 검출할 수 있다
→ non - maximal suppression이라는 방법을 통해 개선

5. **Limitations of YOLO**

💡 - **Spatial Constraints(공간적 제약) 문제**
: 하나의 그리드 셀은 오직 하나의 객체만 검출하므로 하나의 그리드 셀에 두 개 이상의 객체가 붙어있다면 이를 잘 검출하지 못하는 문제
- 새로운 종횡비(aspect ratio)를 마주하면 고전
- 큰 bounding box와 작은 bounding loss에 대해 동일한 가중치 적용 → 성능에 문제를 줌! 부정확한 localization 문제

▼ 4. 실험은 어떻게 이루어졌는가? (Experiments)

💡 1. Comparision to Other Real-Time Systems
- 정확도는 Fast-R-CNN이나 Faster R-CNN, VGG-16 등이 더 높지만, FPS가 낮아 실시간 검출 모델로 적용 불가!
- 적당히 정확도가 높고, 속도도 빠른 모델은 YOLO 계열!

2. VOC 2007 Error Analysis
   - Fast R-CNN과 비교!
   - Fast R-CNN은 YOLO 보다 localization error가 작지만, background error(배경에 아무 물체 없는데 물체가 있다고 판단) 가 큼!

3. Combining Fast R-CNN and YOLO

- YOLO가 Fast R-CNN에 비해 back ground error가 훨씬 작으므로 둘이 결합하면 굉장히 높은 성능을 얻을 수 있지 않을까?
- R-CNN이 예측한 bounding box와 YOLO가 예측한 bounding box가 유사한지 체크!
- Fast R-CNN과 YOLO를 결합한 모델은 독립적으로 돌려 앙상블하는 방식이므로 YOLO보다 느림. but Fast R-CNN보다는 빠르므로, Fast R-CNN 보다는 결합 모델 사용이 낫다

4. VOC 2012 Results
   - 속도 측면에선 YOLO가 빠르고, 정확도 측면에선 Fast R-CNN과 결합한 모델이 good

5. Generalizability: Person Detection in Artwork
   - 실제 이미지 데이터는 훈련 데이터셋과 테스트 데이터셋의 분포가 다를 수 있음
   - YOLO는 훈련 단계에서 접하지 못한 새로운 이미지도 잘 검출

▼ 5. 결론 및 요약 (Conclusion)

💡 - YOLO는 단순하면서도 빠르고 정확
- 훈련단계에서 보지 못한 새로운 이미지에 대해서도 강건!