**YOLO (You Only Look Once)**

객체 검출은 분류 뿐 아니라 위치정보까지 판단하는 것이다. YOLO는 이미지를 한 번만 보면 객체를 검출할 수 있다 하여 이름이 지어졌다. 이미지 픽셀로부터 bounding box의 위치, 클래스 확률을 구하기까지의 일련의 절차를 하나의 회귀 문제로 재정의한다. 이런 시스템을 통해 이미지 내에 어떤 물체가 있고 그 물체가 어디에 있는지를 하나의 파이프라인으로 빠르게 구해준다. 객체 검출의 개별 요소를 단일 신경망으로 통합한 모델이다. 입력 이미지를 그리드로 나누고, 각각의 그리드 셀은 bounding box와 그에 대한 점수를 예측한다.

장점:

1. 기존의 복잡한 객체 검출 프로세스를 하나의 회귀문제로 바꾸었기 때문에 복잡한 파이프라인이 필요 없어, 굉장히 빠르다.
2. 예측을 할 때 이미지 전체를 보기 때문에 클래스의 모양에 대한 정보 뿐 아니라 주변 정보까지 학습하여 처리한다.
3. 물체의 일반적인 부분을 학습하기 때문에 자연 이미지를 학습하여 그림 이미지로 테스트할 때 성능이 매우 뛰어나다. 다른 모델에 비해 훈련 단계에서 보지 못한 새로운 이미지에 대해 더 검출 정확도가 높다.

구조

* 컨볼루션 계층은 이미지로부터 특징을 추출하고, 전결합 계층은 클래스 확률과 bounding box의 좌표를 예측한다

훈련단계

* Localization loss의 가중치를 증가시킨다
* 객체가 존재하는 그리드 셀의 confidence loss의 가중치를 증가시킨다
* Bounding box의 너비와 높이에 제곱근을 취한 값을 loss function으로 사용한다
* 과적합을 막기 위해 드롭아웃과 data argumentation을 적용한다

추론단계

* 테스트 이미지로부터 객체를 검출하는 데에 하나의 신경망 계산만 하면 된다
* R-CNN등과 다르게 하나의 신경망 계산만 필요하기 때문에 테스트 단계에서 굉장히 빠르다
* 다중검출의 문제가 생길 수 있는데, 비 최대 억제라는 방법을 통해 개선할 수 있다

한계

* 하나의 그리드 셀마다 오직 하나의 객체만 검출할 수 있다. (공간적 제약) 즉, 하나의 그리드 셀에 두 개 이상의 객체가 붙어있다면 이를 잘 검출하지 못한다
* 데이터로부터 bounding box를 예측하는 것을 학습하기 때문에 훈련단계에서 학습하지 못했던 새로운 종횡비를 마주하면 고전한다
* 큰 상자와 작은 상자의 손실에 대해 동일한 가중치를 둬서 작은 상자의 위치변화에 따른 IOU변화는 큰 상자에 비해 더 심하다 (부정확한 localization)