

Thesis review study

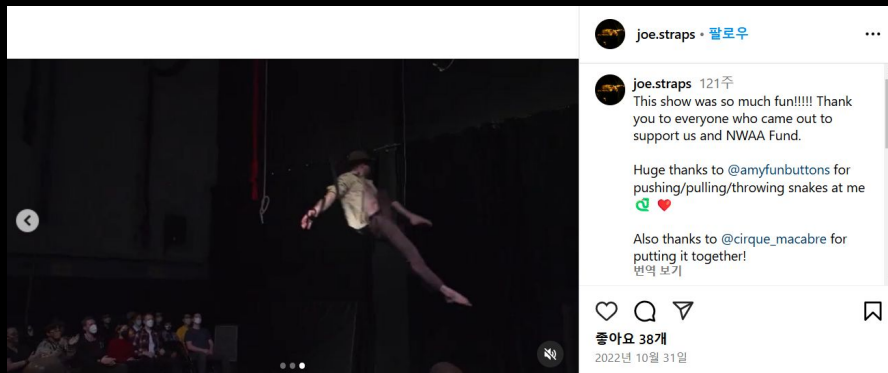
YOLO

조은비, 전민하

“You Only Look Once”

2015년 탄생, 컴퓨터 비전과 객체 탐지 분야의 혁신

창시자는 “You Only Live Once”를 실천중



기존 R-CNN 방식

이미지 내 객체 후보 박스 생성 → 각각의 박스마다 CNN 기반 이미지 분류. 정확도는 높았지만 계산량이 많아서 느리기 때문에 실시간 분석이 어려웠다.

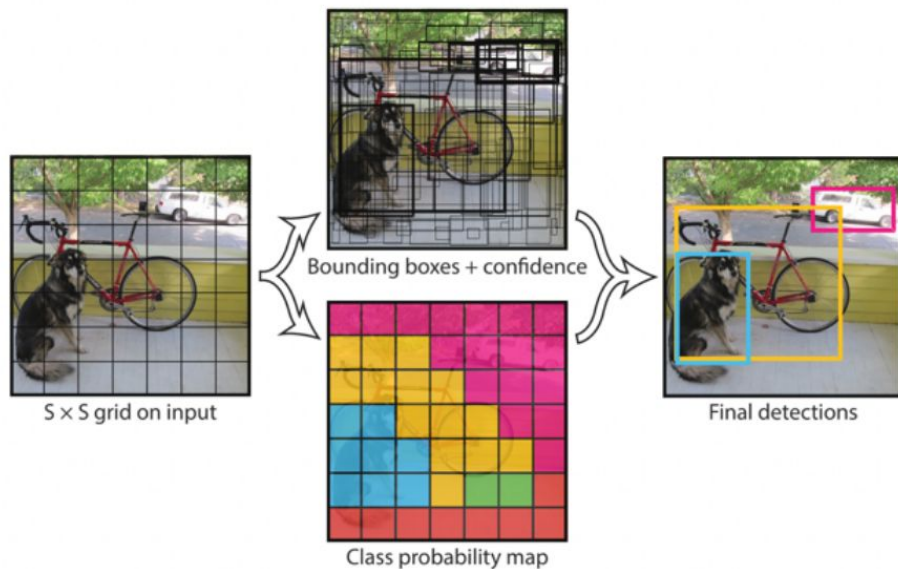
YOLO!

하나의 회귀 문제로 재정의

객체 후보 박스를 생성한 다음에 각각의 박스에 이미지를 분류하던 기존의 처리 과정을 하나의 과정으로 합쳐버림.

이미지를 작은 셀로 나눈 다음에, 각 셀의 후보 박스와 객체 분류 정보를 한번에 출력.

(자세한 과정 설명은 다음 시간에)



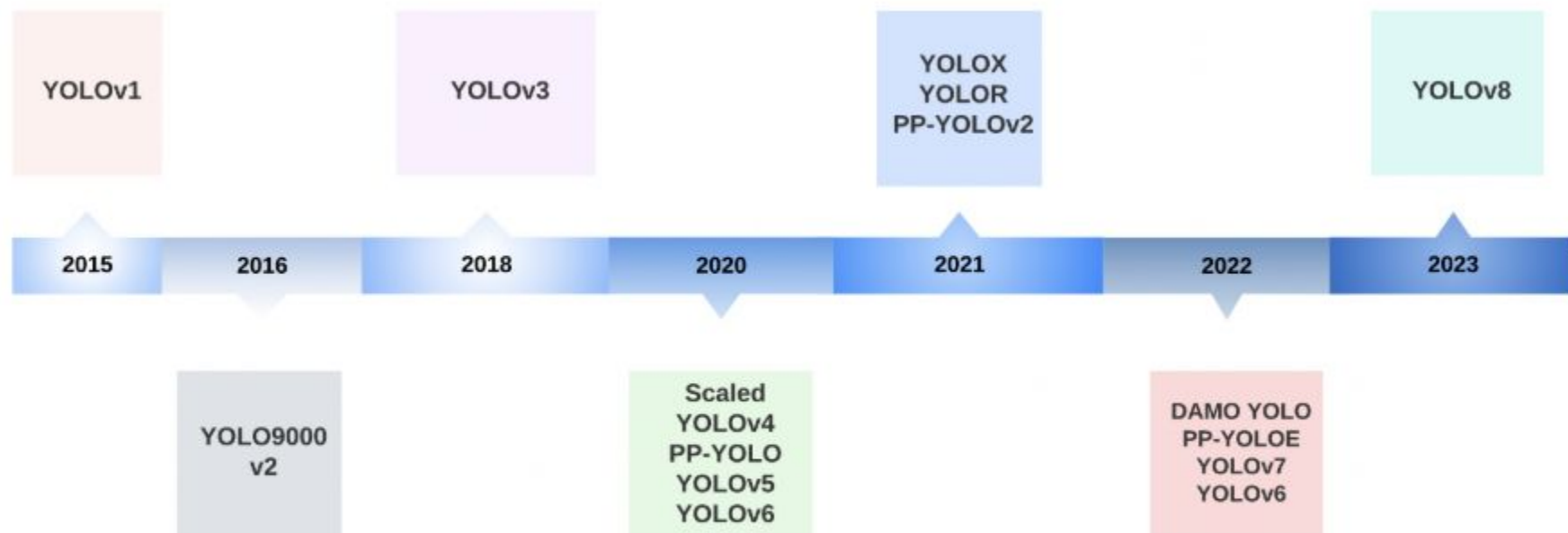


Figure 1: A timeline of YOLO versions.

YOLO v1

- 2016년
- 여러 물체가 겹쳐 있을 경우 정확도 낮아짐
- 새로운 형태의 바운딩 박스는 예측 어려움
- 그리드 당 2개의 바운딩 박스 좌표 예측(좌표를 직접 예측)

YOLO v2

- 2017년 발표
- 기존의 프레임워크 Darknet을 개선한 Darknet19를 사용했고, 파라미터가 감소해서 속도가 개선됨
- 그리드 당 5개의 Anchor 박스를 찾음(미리 지정된 형태를 가진 경계 박스. 사용자가 개수와 형태를 임의로 지정할 수 있음)

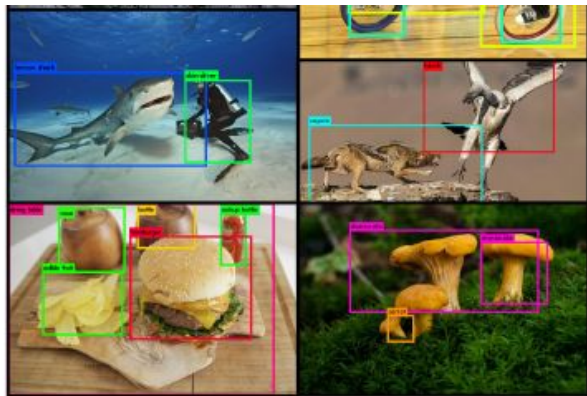


Figure 1: YOLO9000. YOLO9000 can detect a wide variety of object classes in real-time.

YOLO v3

- 2018년
- Darknet-19 → Darknet-53
- Loss function을 softmax에서 binary cross entropy로 변경해서 multi label classification이 가능하게 됨.

YOLO v4

- 2020년
- Darknet53 → CSP(cross-stage partial connections: 중복 기울기 정보를 없앴)
- 성능과 정확도 향상, 속도 감소

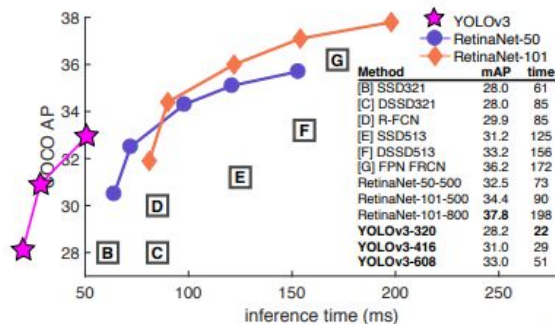


Figure 1. We adapt this figure from the Focal Loss paper [9]. YOLOv3 runs significantly faster than other detection methods with comparable performance. Times from either an M40 or Titan X, they are basically the same GPU.

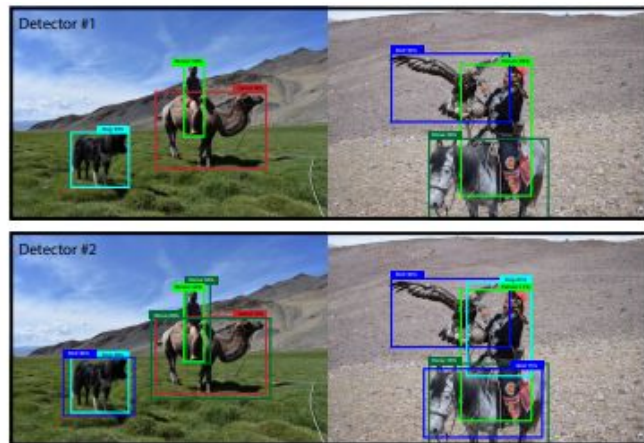


Figure 5. These two hypothetical detectors are perfect according to mAP over these two images. They are both perfect. Totally equal.

- 모두가 당연하게 Caffe 프레임워크로 R-CNN을 쓸 때,
갑자기 본인이 만든 다크넷(마법진이 로고)
프레임워크로 YOLO를 만들었음
- 조랑말을 이력서에 계속 사용해왔음



Joseph Redmon
@pjreddie

I stopped doing CV research because I saw the impact my work was having. I loved the work but the military applications and privacy concerns eventually became impossible to ignore.

[게시물 번역하기](#)

- YOLO v3 논문에서 “사람들이 CV를 얼룩말 몇 마리 있는지 관찰하기 같은 행복한 일에만 쓰면 좋겠는데” “CV 연구를 편당하는 군대는 새로운 기술로 사람을 죽이거나 하지는 않았.. 오 잠시만” 이런 글을 적어두더니,
- 사생활 침해와 군사적으로 악용될 가능성을 우려하면서, 컴퓨터 비전 연구 중단을 발표(2020년 2월)하고 갑자기 서커스하러 다님.
- 이후에도 개발은 계속됐다. 그동안 Darknet과 YOLO를 개발/유지보수 해오던 개발자 Alexey Bochkovskiy가 2020년 4월 YOLO v4를 발표. 사람들이 술렁거렸지만 창시자가 “상관없다!”고 공표. Alexey Bochkovskiy는 “나는 그의 개발 AI 봇”이라고 자처하면서 평화롭게 창시자 없는 YOLO 개발이 이어짐..

I've worked with.



home darknet coq tactics publications projects résumé



Darknet: Open Source Neural Networks in C

prisingly difficult.” [18] If humans have a hard time telling the difference, how much does it matter?

But maybe a better question is: “What are we going to do with these detectors now that we have them?” A lot of the people doing this research are at Google and Facebook. I guess at least we know the technology is in good hands and definitely won’t be used to harvest your personal information and sell it to.... wait, you’re saying that’s exactly what it will be used for?? Oh.

Well the other people heavily funding vision research are the military and they’ve never done anything horrible like killing lots of people with new technology oh wait....!

I have a lot of hope that most of the people using computer vision are just doing happy, good stuff with it, like counting the number of zebras in a national park [13], or tracking their cat as it wanders around their house [19]. But computer vision is already being put to questionable use and as researchers we have a responsibility to at least consider the harm our work might be doing and think of ways to mitigate it. We owe the world that much.

In closing, do not @ me. (Because I finally quit Twitter).

YOLO v5

- 2020년 6월
- YOLO 시리즈 중 가장 논란이 많았던 버전으로, YOLO의 원저자인 Alexey가 아닌 Glenn Jocher가 있는 Ultralytics에서 개발.
 - 이후로 모든 버전에서 모두 저자가 바뀜.
- PyTorch 기반으로 구현되었으며, 이전 버전보다 사용성과 접근성이 향상됨.
- 다양한 크기의 pre-trained 모델 제공(n/s/m/l/x 모델).
 - 목적에 맞게 속도-정확도 균형을 선택
- 다양한 입력 포맷과 플랫폼을 지원 (웹캠 스트리밍, 모바일 등).
- Mosaic 방식의 **데이터 증강 기법** 도입.
- 논문 X, DarkNet 프레임워크를 사용하지 않은 YOLO의 첫번째 사례
 - DarkNet: C언어 기반
 - Pytorch: Python 기반

YOLO v7

- 2022년 7월
- Yolo4와 저자가 동일함
- 다양한 최적화 기법을 통해 실시간 객체 탐지 성능을 극대화
- 추론 비용을 증가시키지 않으면서도 훈련 효율과 최종 정확도를 향상시킴

YOLO v6

- v7 보다 2개월 늦게 발표
- Meituan, 중국의 메이투안에서 개발
- 여러 가지 최적화 기술을 통해 모델 속도와 정확도를 개선.
- 특히 실시간 처리 성능에 강점을 둠.
- 출판되었을 때 가장 성능이 좋은 모델이었음.

YOLO v8

- 2023년 1월, 논문 X
- API 사용성과 확장성이 향상되어 다양한 실시간 비전 애플리케이션에 활용될 수 있음
- Anchor-Free Detection: 앵커박스를 사용하지 않고 객체의 중심을 직접 예측, NMS 속도 향상.
- 다양한 모델 지원
 - YOLOv8-seg (인스턴스 세분화)
 - YOLOv8-pose (포즈 추정)
 - YOLOv8-obb (객체 탐지)
 - YOLOv8-cls (분류) 모델.
- 통합 프레임워크: 객체 감지, 인스턴스 세분화, 이미지 분류 모델을 위한 새로운 리포지토리 출시.

YOLO v6

- v7 보다 2개월 늦게 발표
- Meituan, 중국의 메이투안에서 개발
- 여러 가지 최적화 기술을 통해 모델 속도와 정확도를 개선.
- 특히 실시간 처리 성능에 강점을 둠.
- 출판되었을 때 가장 성능이 좋은 모델이었음.

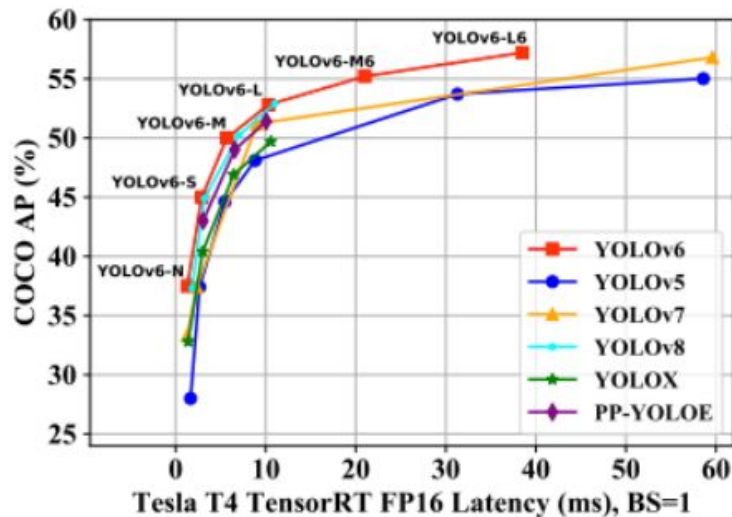
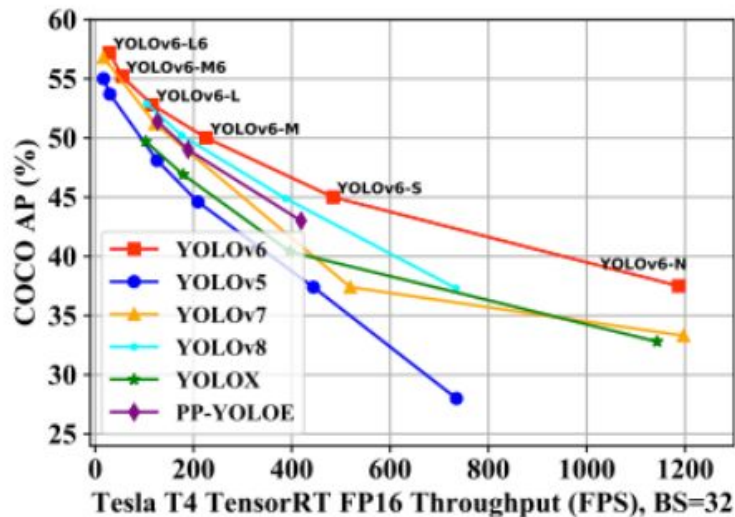


Figure 1: Comparison of state-of-the-art efficient object detectors. Both latency and throughput (at a batch size of 32) are given for a handy reference. All models are test with TensorRT 7.

YOLO v9

- 2024년 2월
- 성능 개선 및 정보 병목 현상 완화.
- YOLOv7과 저자가 동일함 -> v7의 개선버전!
 - PGI(Programmable Gradient Information) 도입
 - 정보 병목 현상을 해결
 - 심층 네트워크 계층에서 필수 데이터를 보존하고, 정확한 모델 업데이트를 촉진
 - GELAN: 파라미터 활용도와 계산 효율성을 극대화

YOLO v10

- 2024년 7월
- 기존 YOLO 모델에서 중요했던 NMS(Non-Maximum Suppression)를 제거하고, 모델 구조 전반을 효율화하여 속도와 정확도를 개선.

YOLO R

YOLO X

YOLO NAS

- 작은 객체를 효율적으로 탐지