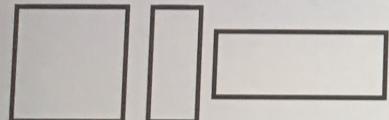


- 많은 object 중 원하는 object를 어떻게 detection 할 것인가?

Sliding Window 방식

Window를 왼쪽 상단에서부터 오른쪽 하단으로 이동시키면서 Object를 Detection하는 방식

다양한 형태의 Window를 각각 sliding 시키는 방식



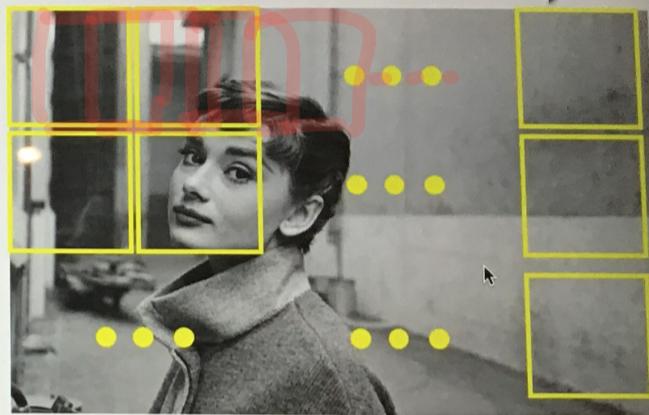
Window를 변경

Window Scale은 고정하고 scale을 변경한 여러 이미지를 사용하는 방식



image를 변경

Window 이동

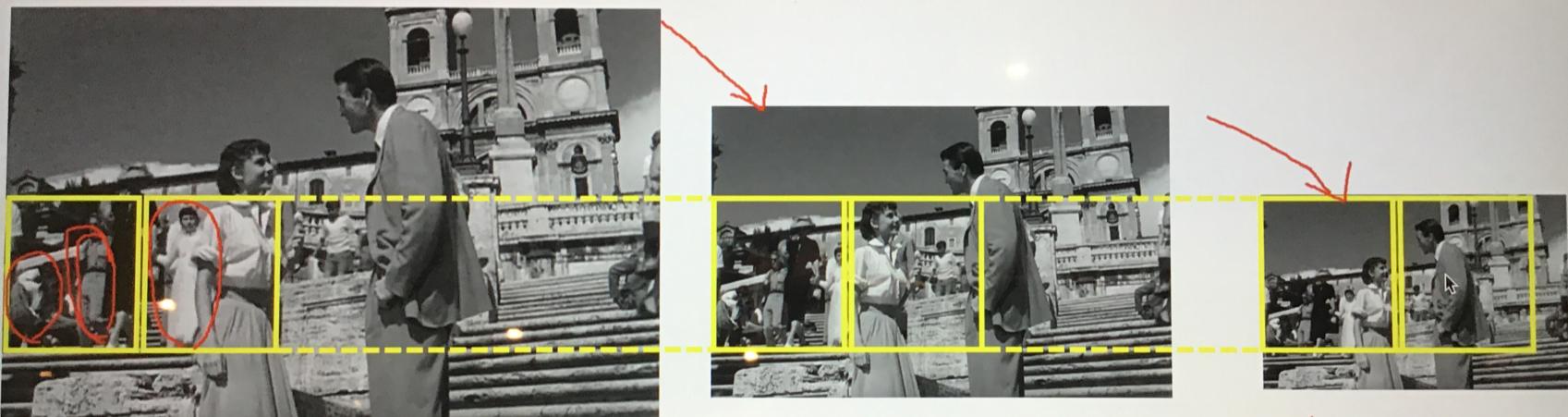


- Object Detection의 초기 기법으로 활용
- 오브젝트 없는 영역도 무조건 슬라이딩 하여야 하며 여러 형태의 Window와 여러 Scale을 가진 이미지를 스캔해서 검출해야 하므로 수행 시간이 오래 걸리고 검출 성능이 상대적으로 낮음
- Region Proposal(영역 추정) 기법의 등장으로 활용도는 떨어졌지만 Object Detection 발전을 위한 기술적 토대 제공

파이썬 딥러닝 컴퓨터 비전

이미지 Scale 조정에 따른 여러 크기의 Object Detection

⇒ Sliding Window 예시

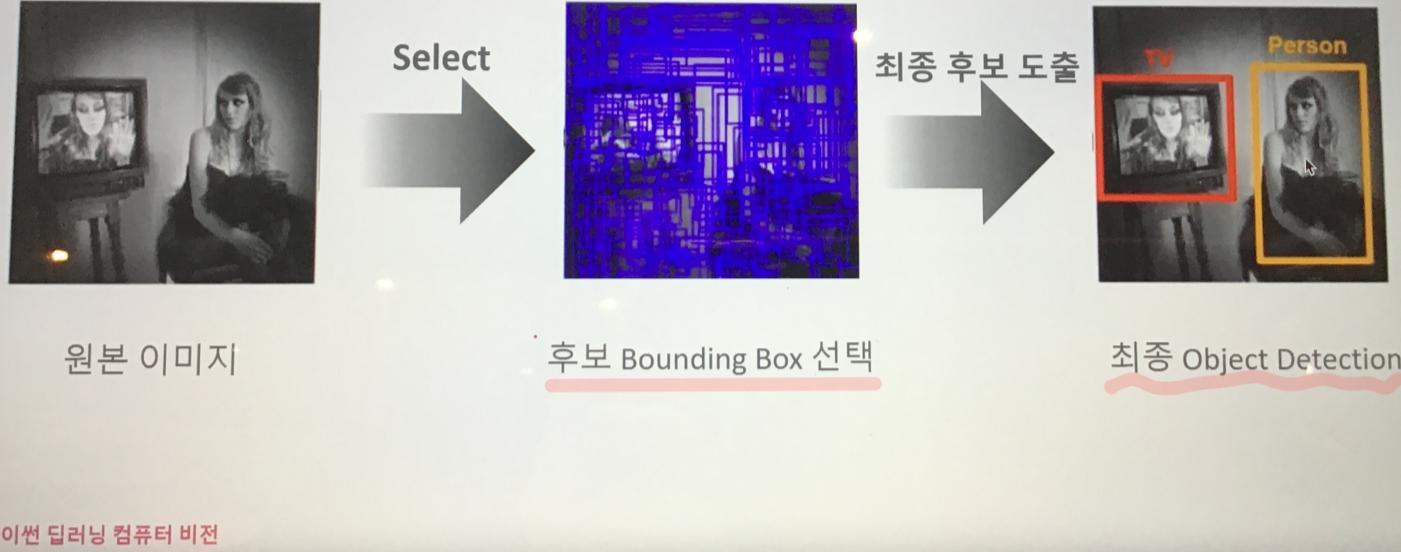


파이썬 딥러닝 컴퓨터 비전

- Sliding Window 가 아닌 region proposal 방식.

Region Proposal(영역 추정) 방식

"Object가 있을 만한 후보 영역을 찾자"



↑ Sliding Window 방식은
object가 있는지 없는지
모르기 때문에, window를
sliding하여 다 찾아줘야 함
+ object의 크기도 모르므로,
image scaling으로 탐색!

↓
전체를 다 훑지 않고,
있을 만한 곳을 먼저
찾아! (Region proposal)

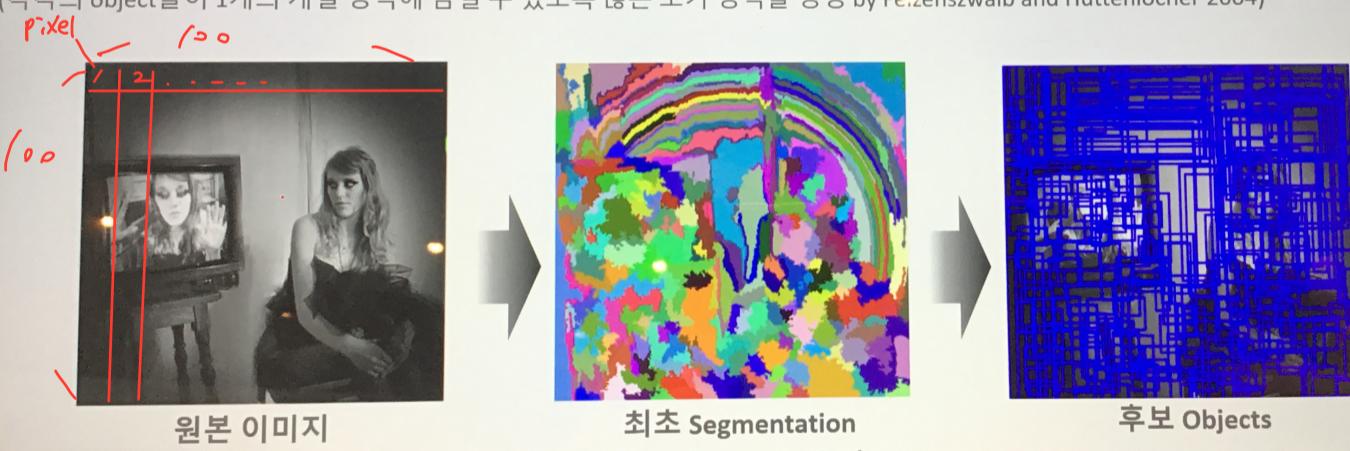
Selective Search – Region Proposal의 대표 방법

빠른 Detection과 높은 Recall 예측 성능을 동시에 만족하는 알고리즘

컬러, 무늬(Texture), 크기(Size), 형태(Shape)에 따라 유사한 Region을 계층적 그룹핑 방법으로 계산

Selective Search는 최초에는 Pixel Intensity기반한 graph-based segment 기법에 따라 Over Segmentation을 수행

(각각의 object들이 1개의 개별 영역에 담길 수 있도록 많은 초기 영역을 생성 by Felzenszwalb and Huttenlocher 2004)



파이썬 딥러닝 컴퓨터 비전

MacBook Pro

Selective Search의 수행 프로세스

- 개별 Segment된 모든 부분들을 Bounding box로 만들어서 Region Proposal 리스트로 추가
- 컬러, 무늬(Texture), 크기(Size), 형태(Shape)에 따라 유사도가 비슷한 Segment들을 그룹핑함.
- 다시 1번 Step Region Proposal 리스트 추가, 2번 Step 유사도가 비슷한 Segment들을 그룹핑을 계속 반복하면서 Region Proposal을 수행



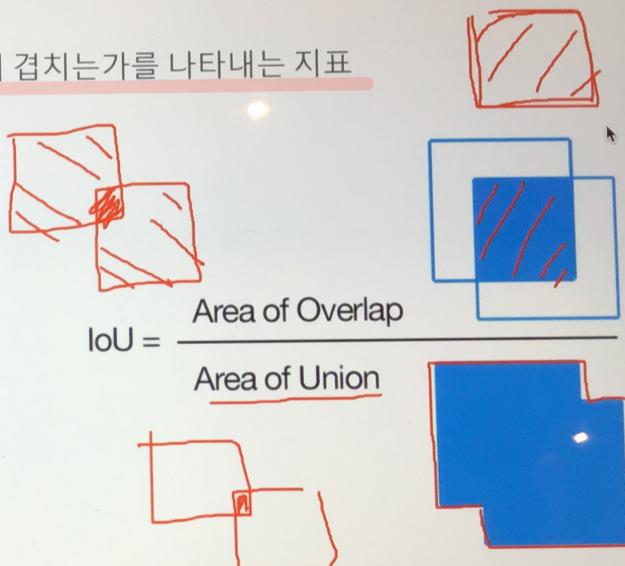
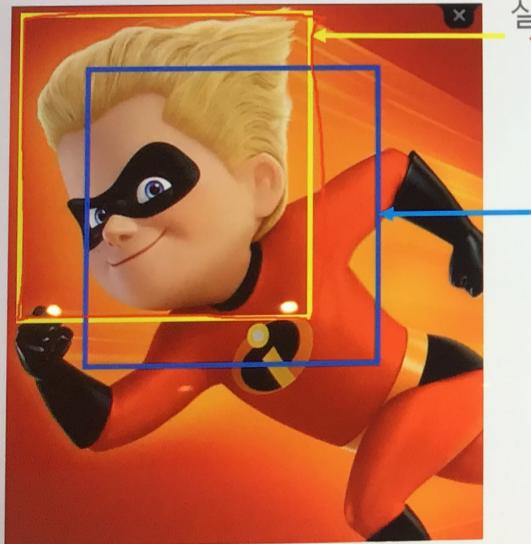
파이썬 딥러닝 컴퓨터 비전

MacBook Pro

Object Detection 성능 평가 Metric - IoU

IoU: Intersection over Union

모델이 예측한 결과와 실측(Ground Truth) Box가 얼마나 정확하게 겹치는가를 나타내는 지표



IoU는 개별 Box가 서로 겹치는 영역/ 전체 Box의 합집합 영역

파이썬 딥러닝 컴퓨터 비전

MacBook Pro