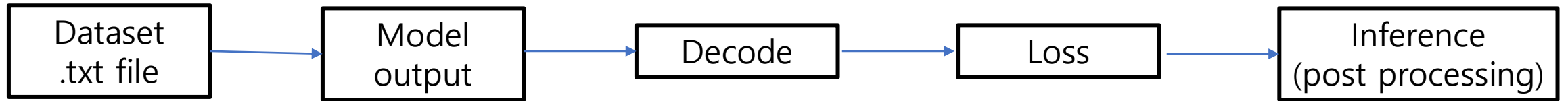


Yolo v4 변경 순서



Dataset

Txt 파일 변경

기존 YOLOv4 txt : 파일명.jpg x1,y1,x2,y2,C

변경 YOLOv4 txt : 파일명.jpg x1,y1,x2,y2,C,junction1 x, junction1 y ° ° °



Txt 파일을 추가한 만큼 encoding (dataset.py) 변경

Model output

3개의 Head부분 변경

기존 YOLOv4 :

$? \times 13 \times 13 \times 3 \times (\text{Class 개수} + 5)$, $? \times 26 \times 26 \times 3 \times (\text{Class 개수} + 5)$, $? \times 52 \times 52 \times 3 \times (\text{Class 개수} + 5)$



변경 YOLOv4 :

$? \times 13 \times 13 \times 3 \times (\text{Class 개수} + 5 + ??)$, $? \times 26 \times 26 \times 3 \times (\text{Class 개수} + 5 + ??)$, $? \times 52 \times 52 \times 3 \times (\text{Class 개수} + 5 + ??)$

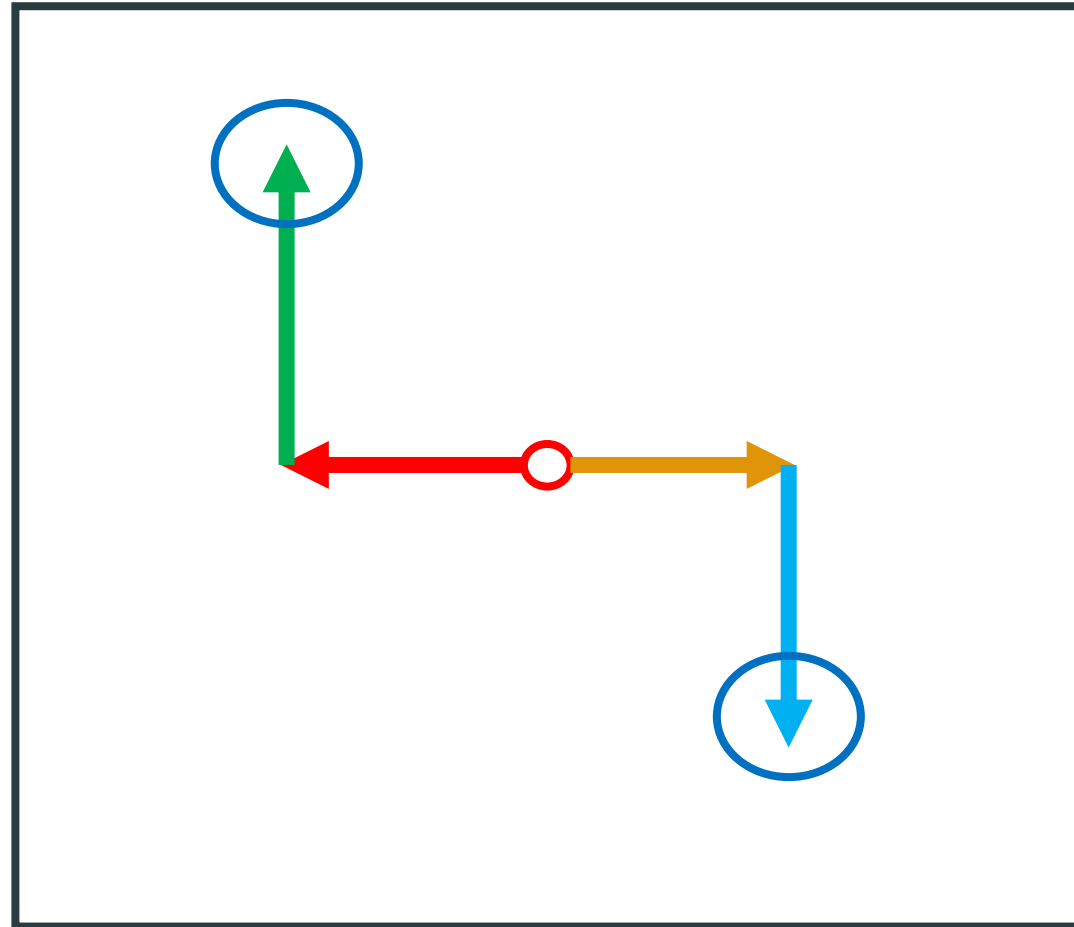
Decode

Junction 점 표현

여러가지 방법 존재, 검출한 셀의 좌상단으로부터 표현, 셀의 중점으로부터 표현 등등

Tanh 함수 사용, model output -1~1 사이로 정규화한 것을 어떤 Scale(정하기 나름)을 곱해서 어떤 점(셀의 좌상단, 중점 등등)으로부터 표현

Decode



Loss

예측한 Junction 점의 좌표와 GT junction 점의 좌표의 점과 점사이의 거리,
주차 구역의 방향이라면 예측한 방향 벡터와 GT 방향 벡터가 이루는 각도 등등

Loss가 0에 가까울 수록 좋은 방향으로 loss 설정

Inference
(post processing)

추가한 만큼 post processing을 변경해주고, inference할 때 image에 예측한 정보들 시각화

Quantitative Evaluation (정량 평가)

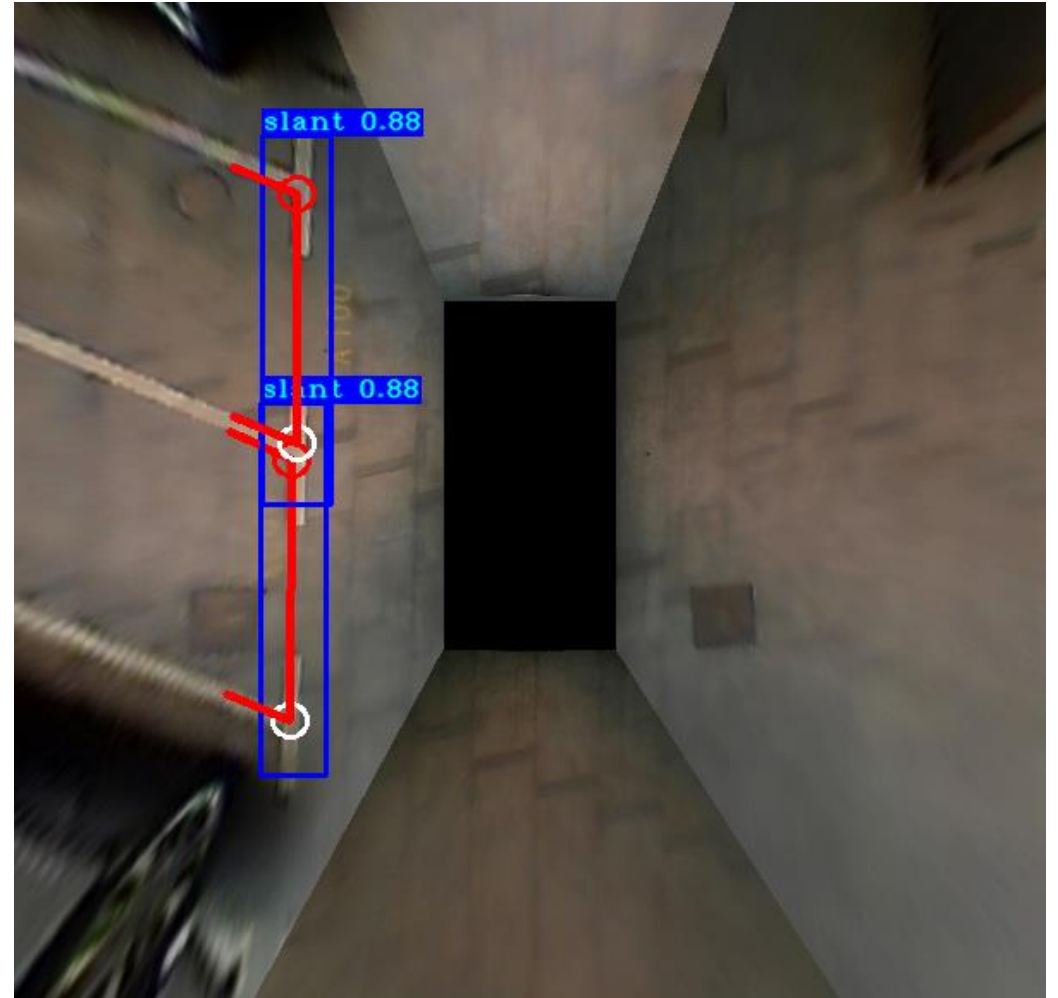
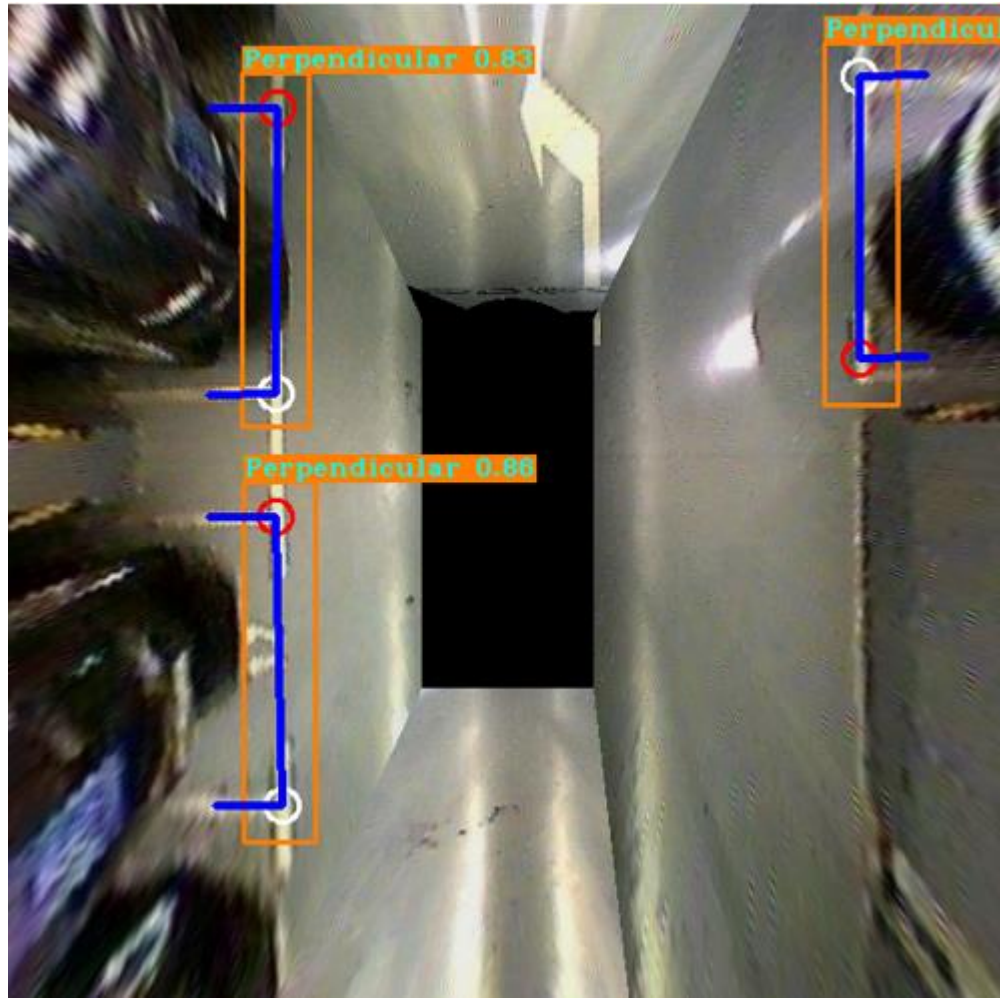
mAP, recall, precision, classification rate (Accuracy)

기존의 TP(Yolov4에서 쓰인 것은, 예측한 바운딩 박스와 GT 바운딩 박스가 이루는 IoU가 일정 이상일 때 TP)를 parking slot에 맞게 재정의

Ex) GT junction과 예측한 junction의 사이 거리가 S pixel 이하일 때 TP

Ex2) GT junction과 예측한 junction의 사이 거리가 S pixel 이하이고, 예측한 방향벡터와 GT 방향벡터가 이루는 사잇각이 θ 도 이하일 때 TP

최종 inference



주차구역에 대한 바운딩 박스를 검출 + 주차구역의 Type 검출 +
junction 2개 검출 + 주차구역의 occupancy 검출 + 주차구역의 방향 검출

Quantitative Evaluation
(정량 평가) 예시

Type Classification Confusion Matrix

	True Class			
		Perpendicular	Parallel	Slant
Predicted class	Perpendicular	909	0	0
	Parallel	0	827	0
	Slant	0	1	43

Perpendicular Type Classification Accuracy : 1.0000

Parallel Type Classification Accuracy : 0.9994

Slant Type Classification Accuracy : 0.9994

모든 GT 박스 개수 : 2168 개

Threshold : **12** pixel & **10** degree를 만족하는 TP 박스 개수 : 1780 개