



작업내역-세현

☀ Status	시작 전
👤 담당자	
📅 마감일	

프로젝트 내역

발표 문서작성

모델 이미지 전처리 48만장

모델 이미지 라벨링

모델 학습 (yolo8l,yolo8n,yolo5n)

Webapp 내에서 webcam app 제작

학습한 모델 바탕으로 실시간 object detection 가능하게 하려면 비동기 작업 필요

Django channels 적용 웹소켓 적용

Webcam app 사항

- 모델 테이블 작성
- 각 상황에 맞게 뷰 작성
- 비동기 데이터는 컨슈머 작성
 - 도커 aws 관련 협업
 - 서버 작업 파일 관리
 - 및 여러 문제 트러블 슈팅

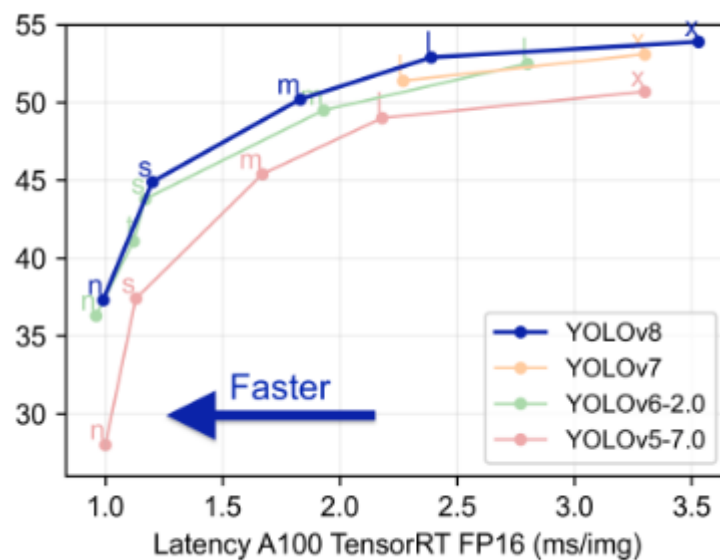
Channels 사용 이유

동영상을 처리할 때 동영상이 다 처리가 끝나야 클라이언트는 응답을 받는다

이에 따라 서비스라면 최소한의 불편을 줄이려면 비동기로 처리하는 부분이 있어야 할 것이다

그렇게 찾아보니 channels에서 async 관련 문서들이 많았고 도움이 되었다

모델간의 속도 차이



Model	size (pixels)	mAP ^{val} 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

.....

Model	size (pixels)	mAP ^{val} 50-95	mAP ^{val} 50	Speed CPU b1 (ms)	Speed V100 b1 (ms)	Speed V100 b32 (ms)	params (M)	FLOPs @640 (B)
YOLOv5n	640	28.0	45.7	45	6.3	0.6	1.9	4.5

yolov5n 이 v100으로 돌려도 저 스피드가 나온다

속도 면에서 v5n>v8n>v8l 였다

아쉬운점

Yolo large 모델은 성능이 좋은데 실시간용으로는 부적합 서버에 올리기에다 부적합 해서 현재는 사용하지 않아서 아쉬웠다 학습 시 가장 시간이 오래 걸렸다

비동기 관련 coroutine이 너무 이해하기 어려웠다. 그냥 동영상 받아서 저장하고 처리하는 기능은 만들었는데 너무 늦어서 코드 꼬일 까봐 적용을 하지 못했다

도커로 nginx를 올리고 daphne를 적용하려고 하는데 막상 이 관련 레퍼런스가 없어서 너무 힘들었다. 정우님이 도커 관련 기본 세팅을 해놓으셔서 몇가지만 수정하면 되긴 했지만 그냥 도커 에 익숙하지 않아 힘들었다.

이메일 관련 aws ses를 사용하려고 했는데 왜 적용이 안되는지 잘 모르겠다. 아쉽다

멘토님한테 밥을 더 얻어 먹지 못해 아쉽다

학교 다닐 때 공부 열심히 할걸 아쉽다

문제 해결

게시판 관련 오류 해결

모델 수정 및 url, view 수정으로 해결했다. 하나 하나씩 더듬어 가며 잘못된 부분을 고쳐 나가서

장고에 대한 이해가 더욱 깊어졌다

웹캠 관련 비동기 처리시 비동기로 작업을 올려서 처리하는 법으로 오류 해결

어디에 찾아봐도 관련 오류를 해결하는 법이 없어서 답답한 상태로 이틀 이상 고민했는데 선생님께서 던져주신 말이 힌트가 되었다. 소켓은 종료가 되어있는데 카메라는 돌아가고 있다는 말을 듣고

이 부분에서 문제가 생겼음을 깨달아서 해결했다

도커 nginx django 서버 연동 오류 해결

도커로 계속 컨테이너를 재생성 하다보면 docker-compose 에서 build로 되어있던 부분의 이미지는

계속해서 쓰레기 값으로 남게된다 이에 따라 한번 지우려다 모든 이미지를 밀어 버린 적도 있고

도커 로그 사용법을 몰라서 계속해서 하나 고치고 서버에 올리고 하는 방법으로 작업도 했었다

도커 내에는 캐쉬가 존재하는데 이 전에 빌드를 하고 수정을 했더라도 캐쉬 값이 남아서 장고를 시작하는데

이상이 없었는데 새로운 경로로 올리니 문제가 생겼다. 이에 따라 장고 기본 설정 불러오는 부분을 옮겼더니

잘 해결되었다

모델 학습 관련 커스텀 데이터 셋

만들 때 시간이 오래 걸려 동생 컴퓨터로 두 대로 나눠서 작업해서 시간 단축을 했다

당시에 학습용 서버는 빌려놓은 상태여서 최대한 빠르게 데이터 셋을 만들어야 했는데

사전에 주어진 이미지 용량이 커서 리사이즈 하는데 시간이 오래걸렸고 라벨링 또한

절대좌표로 주어져 있어서 상대좌표로 변경하는 작업에도 시간이 소요됐다

가치 평가

이 프로젝트를 판매한다면? 1200만원 두당 200씩은 줘야 할거 같다

다들 고생했고 다들 많이 배웠다고 생각한다 그래서 크게 부르지 않았다

느낀점

길게 일할 거면 템포 조절을 잘 해야 할 것 같다. 프로젝트 마지막에 오니 컨디션 관리도 안 되고 너무 힘들다. 다음에 기간이 긴 프로젝트를 한다면 컨디션 관리를 좀 더 잘할 것이다.

막상 로컬로 돌리는 것보다 웹으로 서비스 배포 하고 나니까 기분이 좋다