

[LG U+ Why Not SW Camp 1기]

IPTV 홈쇼핑 채널 추천 서비스

모델 정의서

팀명 : 배할머니네 손주들

팀장 : 박시원

팀원 : 이상민 이지선 임세연 전서영

목차

1 모델 개요 및 선정 이유

1.1 사람 객체 탐지 모델 (YOLOv8s)

1.2 의류 객체 탐지 모델 (YOLOv8s)

1.3 벡터화 모델 (Resnet50)

2 데이터셋 구성 및 전처리

2.1 COCO 데이터셋

2.2 K-Fashion 데이터셋

2.3 방송 화면 캡처 데이터셋

3 모델 구조 및 기술 사용

3.1 YOLOv8

3.2 Resnet50

1. 모델 개요 및 선정 이유

1.1. 사람 객체 탐지 모델

모델 개요

사용 모델	COCO 데이터 Pretrained Yolov8s 모델
사용 목적	실시간 방송 화면에서 사람 객체를 탐지하여 이후 데이터 처리에 활용
활용 방식	탐지된 사람 객체 이미지를 분리하여 의류 객체 탐지 모델의 입력 데이터로 제공
성능과 효율성	YOLO 모델은 높은 정확도와 빠른 처리 속도를 제공하여 실시간 처리에 적합
경량화	모델 경량화를 통해 낮은 연산 자원으로도 높은 성능 발휘
유연성	다양한 데이터셋과 환경에 적응 가능하며, 방송 화면과 같은 비정형 데이터에 강점

모델 선정 이유

비교 항목	YOLOv8s (small)	YOLOv8m (medium)	YOLOx
성능(mAP@50)	COCO 데이터셋 기준 약 44.9%	COCO 데이터셋 기준 약 50.2%	COCO 데이터셋 기준 약 32.8%
처리 속도 (FPS)	약 140 FPS	약 80FPS	약 150FPS
모델 크기	약 11MB	약 25MB	약 5MB
연산량 (GFLOPs)	약 8.7 GFLOPs	약 15.5 GFLOPs	약 6.5GFLOPs
메모리 요구량 (GPU)	약 1GB	약 2GB	약 500MB
적합성 (실시간 서비스)	실시간 서비스에 최적화	실시간 서비스에서 사용 가능하지만 제약 있음	실시간 서비스에 매우 적합, 경량화된 구조
사용 목적	경량화된 모델로 빠르고 실시간 처리가 필요한 환경	성능과 처리 속도 간 균형이 필요한 환경	자원 제약 환경에서 정확도가 중요하지 않은 실시간 객체 탐지

결론

사람 객체 탐지를 위한 최종 모델로 YOLOv8s (small) 모델 선정

- 높은 정확도(44.9%)와 빠른 처리 속도(약 140FPS)로 실시간 서비스에 적합
- 경량화된 구조(11MB)로 자원 효율성이 우수
- 실시간 방송 화면에서 사람 객체 탐지 요구 사항을 충족함.

- YOLOv8m은 정확도는 높으나 속도가 낮아 실시간 처리에 제약이 있으며, YOLOX는 속도가 빠르지만 정확도가 낮아 부적합함.

1.2. 의류 객체 탐지 모델

모델 개요

사용 모델	의류 데이터 학습 Yolov8s 모델
사용 목적	방송 화면 및 홈쇼핑 이미지에서 의류 객체를 탐지하고 카테고리 분류를 수행
활용 방식	의류 객체 이미지를 분리하여 벡터화 모델의 입력 데이터로 제공
성능과 효율성	COCO Pretrained Yolov8s 모델을 의류 데이터로 추가학습하여, 다양한 의류 카테고리의 정확한 탐지 및 빠른 처리 속도로 실시간 분석 가능
경량화	모델 경량화를 통해 낮은 연산 자원으로도 높은 성능 발휘
유연성	의료 데이터셋 기반으로 커스터마이징하여, 홈쇼핑 및 방송 데이터 분석에 효과적

모델 선정 이유

비교 항목	YOLOv8s (small)	YOLOv8m (medium)	YOLOx_tiny
성능(mAP@50)	약 91.8%	약 90.8%	약 91.1%
처리 속도 (FPS)	약 344 FPS	약 162FPS	약 438 FPS
모델 크기	약 21MB	약 50MB	약 10MB
연산량 (GFLOPs)	약 8.7 GFLOPs	약 15.5 GFLOPs	약 6.5GFLOPs
메모리 요구량 (GPU)	약 1GB	약 2GB	약 500MB
적합성 (실시간 서비스)	실시간 서비스에 최적화	실시간 서비스에서 사용 가능하지만 제약 있음	실시간 서비스에 매우 적합, 경량화된 구조
사용 목적	경량화된 모델로 빠르고 실시간 처리가 필요한 환경	성능과 처리 속도 간 균형이 필요한 환경	자원 제약 환경에서 정확도가 중요하지 않은 실시간 객체 탐지

결론

의류 객체 탐지를 위한 최종 모델로 의류 데이터 커스텀 YOLOv8s (small) 모델 선정

- COCO Pretrained 모델을 의류 데이터셋으로 추가 학습하여 다양한 의류 카테고리 정확히 탐지
- 경량화된 구조(약 21MB)와 빠른 처리 속도(약 344 FPS)로 실시간 방송 및 홈쇼핑 데이터 분석

에 적합

- YOLOv8m은 정확도와 속도 간 균형이 있지만 실시간 처리에는 제약이 있으며, YOLOX tiny는 처리 속도는 빠르지만 메모리 제한 환경에서 성능 저하 가능성이 존재함.

1.3. 벡터화 모델

모델 개요

사용 모델	ResNet50
사용 목적	1. 크롤링한 홈쇼핑 이미지를 벡터화하여 데이터 베이스 구축 2. 탐지된 의류 객체 이미지를 벡터화하여 유사도 검색 입력 데이터로 제공
활용 방식	벡터화된 데이터를 기반으로 ANNOY 알고리즘을 통해 유사도 검색을 수행
성능과 효율성	ImageNet으로 학습된 ResNet50은 Top-1 정확도 약 76%, 빠른 처리 속도 제공
경량화	Fully Connected Layer 제거로 모델 경량화, 실시간 처리에 적합
유연성	다양한 데이터셋 및 이미지 카테고리에 유연하게 적용 가능

모델 선정 이유

	Resnet50	ResNet50-IBN-A ¹	ViT(VisionTransformer) ²
성능	ImageNet 기준 Top-1 정확도 76.2%, Top-5 정확도 93.1% ³	정확도는 ResNet50 대비 약 1.5% 향상	대규모 데이터셋에서 뛰어난 성능 제공, Top-1 정확도 약 85%
처리 속도 및 실시간성	단일 이미지 처리 속도 약 6ms	추가 연산으로 인해 평균 처리 속도 약 9ms 소요	높은 연산량으로 평균 처리 속도 약 15ms 소요
자원 요구량	경량화된 구조로 GPU 메모리 사용량 약 500MB	Instance Normalization으로 추가 메모리 소모	Transformer 기반으로 메모리 사용량 약 1GB
적합성	실시간 IPTV 환경에서	실시간 서비스에는 다소	실시간 서비스 환경에

¹ Pan, X., Luo, P., Shi, J., & Tang, X. (2018). Two at once: Enhancing learning and generalization capacities via ibn-net. In Proceedings of the european conference on computer vision (ECCV) (pp. 464-479).

² Dosovitskiy, A. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

³ "Image Classification on ImageNet", Papers with Code, <https://paperswithcode.com/sota/image-classification-on-imagenet>

	최적화	부적합	부적합
--	-----	-----	-----

결론

이미지 벡터화를 위한 최종 모델로 ResNet50 모델 선정

- ImageNet 학습 기반 성능(Top-1 정확도 76.2%)과 빠른 처리 속도(6ms)를 제공
- 경량화된 구조로 실시간 처리에 적합
- ResNet50-IBN-A는 정확도가 더 높지만 추가 메모리 소모와 처리 속도 저하 발생하며, ViT는 높은 정확도를 제공하나 연산량과 메모리 요구량이 커 실시간 서비스에는 부적합

2. 데이터셋 구성 및 전처리

2.1. COCO 데이터셋

데이터셋 구성

데이터셋명	COCO(Common Objects in Context) 데이터셋
개요	- 일반적인 객체 탐지를 위한 대규모 데이터셋 - 약 33만 개 이미지, 150만 개 객체 주석 포함
출처	Ultralytics github https://github.com/ultralytics/ultralytics/blob/main/ultralytics/cfg/datasets/coco.yaml
사용 목적	- 모델의 기본 객체 탐지 학습 - 사람 객체 탐지 성능 검증
데이터 구조	- JSON 포맷 - 이미지 경로 및 Bounding Box, Class 정보 포함
주요 클래스	- 사람, 자동차, 동물 등 80개 클래스 - 사람은 class 0으로 분류되며, 해당 객체만 탐지하여 사용함
특징	- 다양한 객체와 환경에 대한 풍부한 데이터 제공

데이터 전처리

COCO 데이터셋은 YOLOv8s 모델의 사전학습 시 사용되었으며, 추가적인 전처리 및 사용은 하지 않는다.

2.2. K-Fashion 데이터셋

데이터셋 구성

데이터셋명	K-Fashion 데이터셋
개요	- 패션 영역과 속성, 스타일 정보를 인식 및 도출할 수 있도록 구축된 학습용 이미지 데이터셋 - 총 120만건 분량의 K-Fashion 이미지 데이터셋 구축
출처	AI Hub K-Fashion 이미지 https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=51
사용 목적	- 모델의 기본 객체 탐지 학습 - 사람 객체 탐지 성능 검증
데이터 구조	- JSON 포맷

	- 대분류 10가지, 세부속성 186가지, 스타일 23가지 레이블링 정보
주요 클래스	<ul style="list-style-type: none"> - 대분류 : 상의 카테고리 / 하의 카테고리 / 아우터 카테고리 / 컬러 / 디테일 / 프린트 / 소재 / 상의 기장 / 하의 기장 / 아우터 기장 / 원피스 기장 / 소매기장 / 넥라인 / 칼라 / 상의,아우터,원피스 / 하의 - 스타일 : 클래식 / 매니시 / 페미닌 / 히피 / 모던 / 컨트리 / 젠더리스 / 스포티 / 레트로 / 밀리터리 / 프레피 / 톰보이 / 로맨틱 / 웨스턴 / 소피스트케이티드 / 리조트 / 키치,키덜트 / 스트리트 / 섹시 / 오리엔탈 / 아방가르드 / 힙합 / 펑크
특징	- 대분류에 상의 / 하의 / 아우터 / 원피스가 포함되어 있어, 해당 정보를 이용하여 4가지 카테고리로 분류 가능

데이터 전처리

카테고리 매핑 및 YOLO 형식 변환	<ul style="list-style-type: none"> - 상의 : 0, 하의 : 1, 아우터 : 2, 원피스 : 3 으로 카테고리를 매핑하여 YOLO 형식의 주석 파일(annotation file)을 생성 - JSON 데이터에서 제공된 Bounding Box 좌표를 YOLO 형식(x_center, y_center, width, height)으로 변환 - 변환된 데이터는 각 이미지에 대해 YOLO 형식의 텍스트 파일로 저장
스타일 필터링	<ul style="list-style-type: none"> - 스타일 정보를 기준으로 특정 스타일만 선별 : 기타, 레트, 로맨틱, 리조트, 매니시, 모던, 소피스트케이티드, 스포티, 젠더리스, 힙합 - 각 스타일별로 1000장씩 랜덤 추출하여 데이터 균형을 맞춤
카테고리별 데이터 수집	<ul style="list-style-type: none"> - 상의, 하의, 아우터, 원피스의 각 카테고리별로 2000장 이상의 데이터를 확보 - 다양한 스타일이 골고루 포함되도록 스타일별로 랜덤 샘플링
학습 데이터셋 구성	<ul style="list-style-type: none"> - 학습(train), 검증(val), 테스트(test) 세트를 구성: 각 카테고리별로 2200개(train), 300개(val), 300개(test)의 데이터를 중복 없이 랜덤 추출 - 샘플링은 각 카테고리 및 스타일이 균등한 비율로 유지되도록 설정
데이터 증강	- 데이터 다양성을 높이기 위해 Augmentation 적용

2.3. 방송 화면 캡처 데이터셋

데이터셋 구성

데이터셋명	방송 화면 캡처 데이터셋
-------	---------------

개요	<ul style="list-style-type: none"> - 나는술로, 나혼자산다, 드라마 등 방송 화면을 직접 캡처하여 수집한 데이터셋 - 라벨링 도구(labelImg)를 사용해 각 객체에 대한 Bounding Box와 Class 정보를 수동으로 생성
출처	Youtube 화면 캡처하여 직접 수집
사용 목적	<ul style="list-style-type: none"> - YOLOv8 모델의 의류 객체 탐지 성능 개선. - 실제 방송 환경에서 발생하는 다양한 조건(조명, 앵글, 배경)을 반영하여 학습 데이터 다양성 확보.
데이터 구조	<ul style="list-style-type: none"> - txt 포맷 - class 정보, YOLO 형식 좌표 데이터 (x_center, y_center, width, height). - 원본 이미지와 주석 파일 포함
주요 클래스	- 상의 : 0, 하의 : 1, 아우터 : 2, 원피스 : 3
특징	<ul style="list-style-type: none"> - 실제 방송 화면을 기반으로 수집되어 다양한 조명 조건, 촬영 각도, 복잡한 배경을 포함. - 다른 데이터셋에서 다루지 않는 실제 환경의 객체 탐지 성능을 보완. - 방송 화면의 특성상, 동작 중인 객체(사람)와 정적인 배경이 혼합되어 있어 학습 데이터의 복잡성 증가.

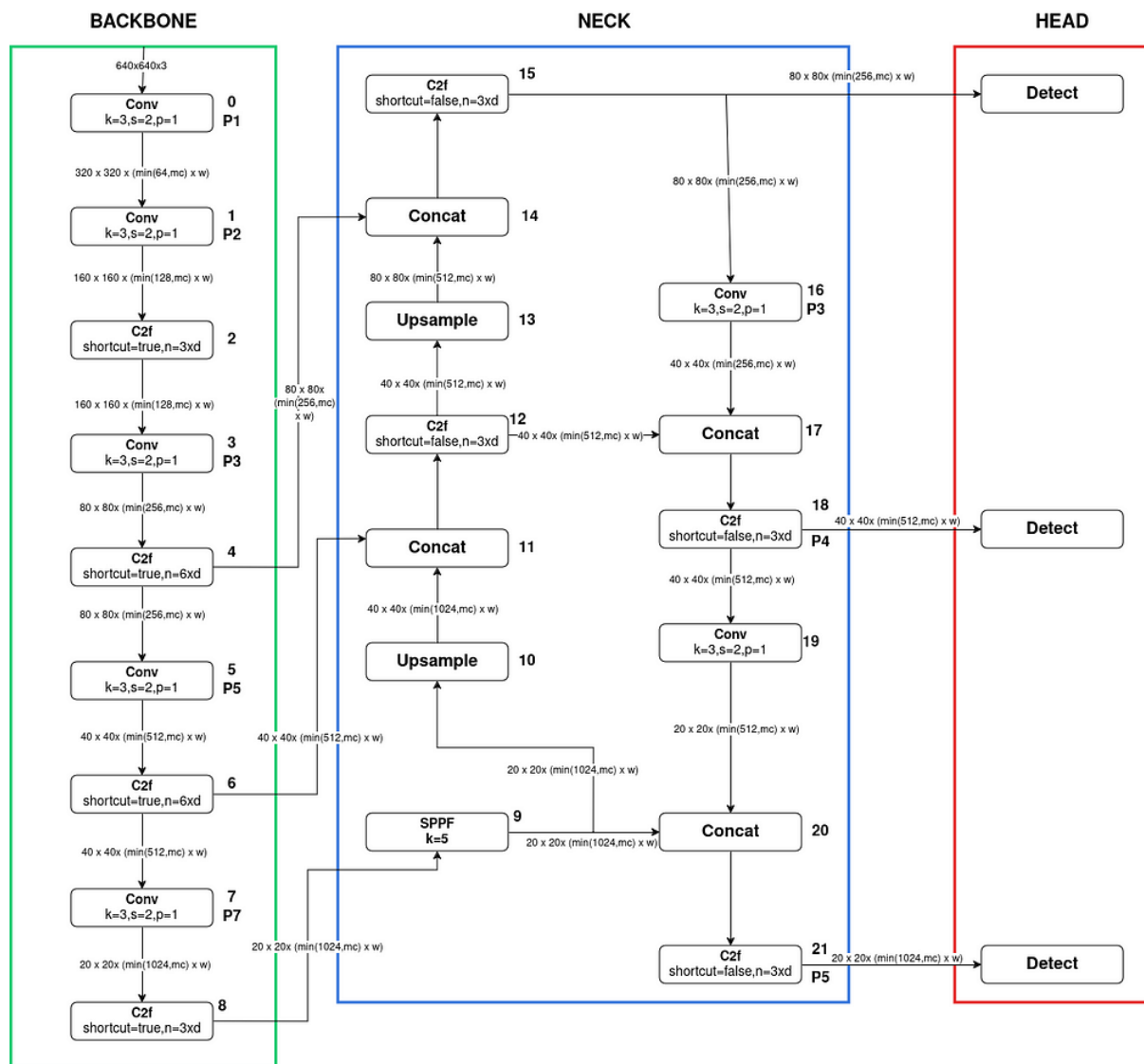
데이터 전처리

카테고리 매핑 및 YOLO 형식 변환	<ul style="list-style-type: none"> - 상의 : 0, 하의 : 1, 아우터 : 2, 원피스 : 3 으로 카테고리를 매핑하여 YOLO 형식의 주석 파일(annotation file)을 생성 - labelImg 프로그램을 사용해 의류 객체에 박스를 생성하고 주석 파일 생성
카테고리별 데이터 수집	- 상의, 하의, 아우터, 원피스의 각 카테고리별로 100장 이상의 데이터를 확보
학습 데이터셋 구성	- 학습(train), 검증(val), 테스트(test) 세트를 구성: 각 카테고리별로 8:1:1의 비율로 train:val:test 데이터셋으로 구분
데이터 증강	- 데이터 다양성을 높이기 위해 Augmentation 적용

3. 모델 구조 및 기술 사용

3.1. YOLOv8

네트워크 구조



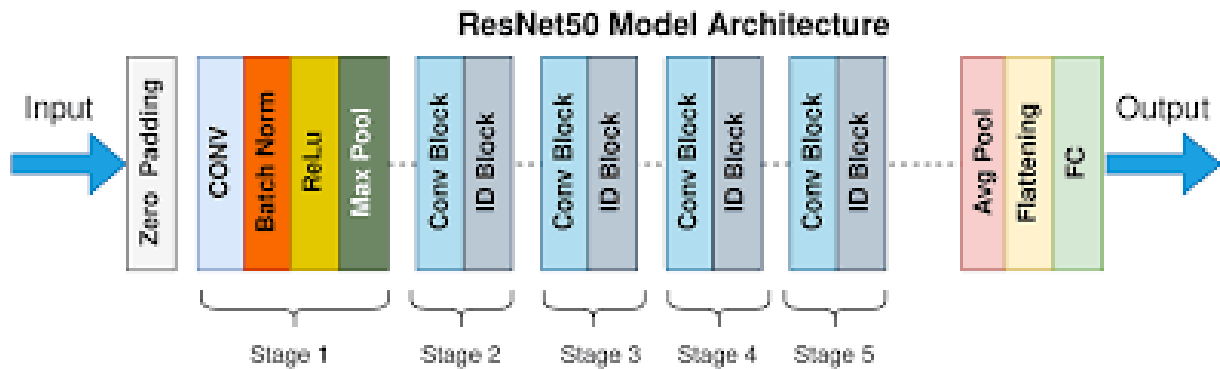
기술적 사양

항목	YOLOv8s
입력 크기	640x640 픽셀
활성화 함수	SiLU (Sigmoid Linear Unit)
Optimizer	SGD (Stochastic Gradient Descent), 대안으로 Adam 사용 가능.
Loss 함수	<ul style="list-style-type: none"> - 객체 분류 손실 : BCE (Binary Cross-Entropy) Loss - 바운딩 박스 회귀 손실 : CloU (Complete Intersection over Union) Loss

	- 객체성 손실 : BCE Loss
Backbone	CSPDarknet53 기반 수정 : 경량화, Residual Block 개선, 얇은 계층 구조
Neck	PANet (Path Aggregation Network) : Low-level Feature 강화 및 Path Aggregation
Head	<ul style="list-style-type: none"> - 앵커 프리(Anchor-free) 구조 : Anchor 설정 제거로 간결한 학습, 다양한 크기의 객체 탐지 효과적 - 객체성(Objectness), 분류(Classification), 회귀(Regression) 작업을 개별적으로 수행.

3.2. Resnet50

네트워크 구조



기술적 사양

항목	ResNet50
입력 크기	224x224 픽셀
활성화 함수	ReLU (Rectified Linear Unit)
Optimizer	SGE (Stochastic Gradient Descent)
Loss 함수	벡터화 목적에서는 Loss 함수가 사용되지 않음. 대신, 학습된 모델의 FC 레이어 출력을 특징 벡터로 추출하여 사용
Backbone	Residual Block과 Bottleneck Block으로 구성. 네트워크에서 추출된 특징을 FC 레이어로 전달하여 최종 벡터화 작업 수행
Head	FC 레이어를 활용해 이미지의 특징을 벡터 형태로 변환. Global Average Pooling 레이어 출력 후 FC 레이어를 사용하여 고정 길이의 벡터 생성하여, 이미지 유사도 계산 및 검색 시스템에서 입력 데이터로 활용.