구강 이미지 데이터 데모 결과

<데모 테스트한 모델 링크>

AI-Hub

<구강 이미지 데이터 데모 확인 내용>

구강 이미지 데이터의 활용 AI 모델 및 코드 항목에서 하위 항목 AI 모델 다운로드 진행 후 3가지 경우에 대한 사전 학습모델과 학습모델데이터가 존재

- 1. 치아 번호 생성 및 분류 모델 목표 : 치아 번호 생성 및 분류,
- 2. 치석 및 보철물 탐지 모델 목표 : 치석 및 보철물 탐지,
- 3. 치주질환(충치, 치주염, 치은염) 탐지 모델 목표 : 치주질환 탐지

로 구성되어있고 각각 아래의 사전 모델을 이용해 데모 테스트를 진행함

disease_model_saved_weight.pt

hygiene_model_saved_weight.pt

tooth_number_saved_weight.pt

<데모 모델 정보>

░ 치아 번호 생성 및 분류 모델 정보

☑ 모델 개요

항목	내용	
모델 아키텍처	FPN (Feature Pyramid Network)	
Encoder (백본)	EfficientNet-B7	
사전 학습 가중치	ImageNet	
입력 크기 (학습)	(224, 224, 3)	

입력 크기 (테스트)	(112, 112, 3) → 후처리시 (448, 448) 으로리사이즈	
출력 클래스 수	337	
활성화 함수 (출력)	None (후처리로 argmax 사용)	
손실 함수	CrossEntropyLoss (학습시 사용) mIoU 기반 정량 평가 (테스트시)	
최적화 기법	AdamW + OneCycleLR 스케줄러	
후처리 방식식	np.argmax(output, axis=0) → 각 픽셀에서 가장 높 은 클래스 선택	
시각화 결과 저장	test_results 폴더에 저장 (입력, 정답 마스크, 예측 마스크 포함)	
성능 평가 지표	mIoU 계산	

🌎 클래스 구성 (총 33개)

클래스 번호	라벨 이름	설명
0	B&G	Background & Gums (배경/잇몸)
1~8	11~18	우측 상악 치아
9~16	21~28	좌측 상악 치아
17~24	31~38	좌측 하악 치아
25~32	41~48	우측 하악 치아

💡 라벨 번호는 FDI 치아 번호 체계 기반으로 구성되어 있음.

목적

구강 이미지 내에서 각 치아의 위치와 번호를 분할 및 시각화하는 목적의 세그멘테이션 모델

📌 주요 특이사항

- 테스트 이미지 해상도는 112x112 이지만, 시각화를 위해 448x448 으로 업샘플링
- 예측 결과는 컬러맵과 범례(legend)를 포함해 시각화하여 저장
- 클래스 수가 많기 때문에 고정된 색상 범례와 mapping dictionary(dict_map)를 사용

₩ 치석 및 보철물 탐지 모델 정보

📌 모델 기본 정보

항목	내용
모델 아키텍처	Unet++
백본(Encoder)	EfficientNet-B7
사전 학습 가중치	ImageNet
입력 크기	(224, 224, 3)
출력 클래스 수	107
손실 함수	CrossEntropyLoss (학습 시) mIoU 기반 정량 평가 (테스 트 시)
후처리 방식	np.argmax(output, axis=0) → 픽셀마다 가장 높은 확률의 클래스 선택
시각화 결과 저장	test_results 폴더에 저장 (입력, 정답 마스크, 예측 마스크 포함)
성능 평가 지표	mIoU 계산

라벨 번호	클래스명	설명
0	background	배경 (치아나 병변 외 영역)
1	am	Amalgam – 은-주석 합 금 충전물
2	cecr	Composite Resin - 복 합 레진 충전물
3	gcr	Glass-ionomer Cement – 유리이온머 시멘트 충전물
4	mcr	Metal-Ceramic Restoration – 금속-도 재 복합 보철물
5	ortho	Orthodontic Appliance – 교정 장치 (브라켓 등)
6	tar1	치석 1단계 - 경미한 침 착
7	tar2	치석 2단계 - 중간 침착
8	tar3	치석 3단계 – 심한 침착
9	zircr	Zirconia Ceramic Restoration – 지르코니 아 보철물

📔 추론 결과

테스트 이미지에 대해 예측 마스크 및 정답 마스크 화 클래스별 색상 및 레이블 범례 포함된 이미지 저장 예측 결과당 개별 mIoU 점수 출력

🦷 치주질환(충치, 치주염, 치은염) 탐지 모델 정보

🧠 모델 기본 정보

항목	내용	
모델 아키텍처	Unet++	
Encoder (백본)	EfficientNet-B7	
사전 학습 가중치	ImageNet	
입력 크기	(224, 224, 3)	
출력 클래스 수	10개 (다중 클래스 세분화)	
손실 함수	CrossEntropyLoss (학습시)	
	mIoU 기반 정량 평가 (테스트 시)	
후처리 방식	np.argmax(output, axis=0) → 픽셀별로 가장 높은 확률의 클래스 선택	
시각화 결과 저장	test_results 폴더에 저장 (입력, 정답 마스크, 예측 마스크 포함)	
성능 평가 지표	mIoU 계산	

🥎 클래스 구성 (총 10개)

클래스 번호	클래스 이름	설명 (추정)
0	background	배경
1	C1	충치 단계 1
2	C2	충치 단계 2
3	C3	충치 단계 3
4	GI1	치은염 1단계
5	GI2	치은염 2단계
6	GI3	치은염 3단계

7	PD1	치주염 1단계
8	PD2	치주염 2단계
9	PD3	치주염 3단계

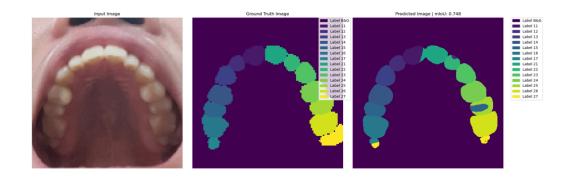
 \wp 클래스명은 코드 내 $dict_map1$ 을 기반으로 해석한 것으로, 실제 라벨링 정의 문서 기준으로 수정될 수 있음

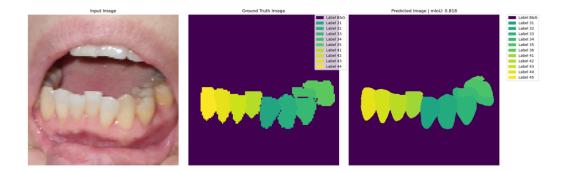
🮨 시각화

- 원본 이미지 + 예측 마스크를 2개 subplot으로 저장
- matplotlib.cm.viridis 컬러맵기반시각화
- 각 클래스는 색상 및 범례(legend)로 표시됨

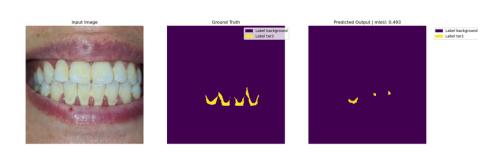
<데모 이미지>

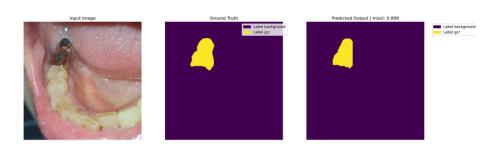
치아 번호 생성 및 분류 모델 - 목표 : 치아 번호 생성 및 분류에 대한 데모 결과 이미지





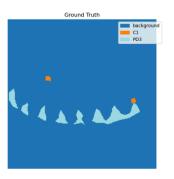
치석 및 보철물 탐지 모델 - 목표 : 치석 및 보철물 탐지에 대한 데모 결과 이미지

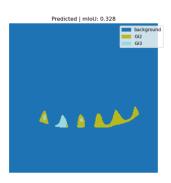




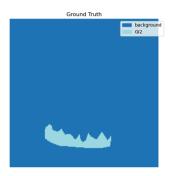
치주질환(충치, 치주염, 치은염) 탐지 모델 - 목표 : 치주질환 탐지에 대한 데모 결과 이미지

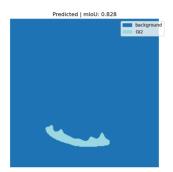












☑ 작업 중 겪은 주요 문제 요약

📁 1. 경로 처리 문제

- 문제점: 상대 경로와 절대 경로를 혼용하여 파일 불러오기 에러 발생
- 조치: 모든 경로를 절대 경로로 통일하여 해결
- 추후 계획: 실제 운영 시에는 다시 상대 경로로 변경 예정

🧠 2. 클래스 정의 혼동

- 문제점: dict_map 에 대한 설명이 없어서 클래스 번호와 의미 파악에 어려움
- 조치: 외부 검색 및 코드 분석을 통해 클래스 의미를 확인하고 정의 정리

🎨 3. 시각화 저장 관련 문제

- 문제점: 기존 코드에서 plt.savefig() 를 통한 시각화 저장이 작동하지 않음
- 조치:
 - a. 먼저 시각화 결과를 화면에 출력하여 정상 작동 여부 확인
 - b. 이후 저장 코드로 수정 및 재확인하여 문제 해결

남아 있는 문제점

4. Ground Truth vs Prediction 차이

- 관찰: 시각화된 결과에서 Ground Truth와 Predicted 마스크 간 차이가 다수 발견됨
- 원인 추정:
 - 전체 데이터가 아닌 **일부데이터**만으로 학습 및 테스트 진행
 - 。이에 따라 예측 정확도(mIoU 등)가 낮을 가능성 있음

📌 추가 확인한 사항

- 기존에 제공된 이미지/마스크 외에 새로운 이미지를 추가 적용
- 해당 이미지에 대해 모델의 예측(Prediction) 및 범례 시각화가 정상 작동함을 확인

