

Hand Bone Image Segmentation (CV 20)

요 약

이번 프로젝트는 Unet, DeepLabv3, MANet, SegFormer, UPerNet 등 다양한 모델과 데이터 증강 기법을 활용해 성능 최적화를 목표로 하였다. Rotate, Contrast, CLAHE, HorizontalFlip, Sharpen, 특정 영역 crop 등의 전처리와 AMP 및 TTA를 적용해 모델 성능과 효율성을 개선하였다. 실험 결과, 데이터 증강과 전처리가 성능 향상에 크게 기여했으며, AMP와 TTA는 학습과 추론 안정성을 높이는 데 효과적임을 확인하였다. 이를 바탕으로 최적화 방안과 향후 연구 방향을 제시하였다.

1. 프로젝트 개요

1.1 연구 배경 및 목적

뼈는 우리 몸의 구조와 기능에 중요한 역할을 하며, 정확한 뼈 분할은 의료 진단과 치료 계획 수립에 필수적이다. 특히 딥러닝 기술을 활용한 뼈 분할은 인공지능 분야에서 중요한 응용으로, 질병 진단, 수술 계획, 의료 장비 제작, 의료 교육 등 다양한 목적으로 활용된다. 뼈의 형태 변형이나 골절 부위를 정확히 파악해 적절한 치료를 시행하거나, 수술에 필요한 정보를 제공하며, 인공 관절 및 임플란트 제작에 활용될 수 있다. 또한 의료 교육에서는 병태와 부상에 대한 이해를 높이고 수술 기술을 연습하는 데 기여한다.

1.2 실험 목표

본 프로젝트의 실험 목표는 semantic segmentation 모델을 활용하여 손 뼈의 구조를 정확히 분리하고 분석하는 것이다. 이를 통해 골절, 변형, 또는 기타 이상을 정밀히 진단하여 적절한 치료 계획을 지원하고, 수술 시뮬레이션 및 의료 장비 제작에도 활용 가능한 데이터를 제공하는 것을 목표로 한다. 또한, 모델의 성능을 최적화하기 위해 데이터 증강, 모델 아키텍처 개선, AMP(Automatic Mixed Precision)와 같은 최신 기법을 적용하여 효율성을 높이고, 다양한 의료 응용 사례에 활용 가능한 모델을 개발하고자 한다.

1.3 팀 소개

수상한 조는 피어세션 전 스트레칭 영상 (10분)을 수행하며, 단순히 리더보드의 점수를 올리는 것을 넘어, 실험 과정에서의 학습과 성장을 최우선으로 합니다. 모든 실험은 [가설 설정 - 문제 정의 - 코드 구현 - 결론 도출]의 체계적인 방식으로 진행하며, 이를 통해 문제 해결 능력을 키우고 새로운 통찰을 얻는 것을 목표로 하고 있습니다.

팀원 모두 열정적이고 적극적인 태도로 협력하며, 아이디어를 공유하고 함께 성장하는 분위기를 조성합니다. 이러한 과정을 통해 팀원들은 각자의 강점을 발휘하고, 더 나은 방향으로 나아가기 위해 끊임없이 도전하고 있습니다. 배움과 성장을 추구하는

팀이라는 점에서 저희 팀은 누구보다 즐겁고 의미 있는 프로젝트를 진행하고 있습니다.

1.4 역할

이름	역할
박수영 (팀장)	일정 관리, streamlit 페이지 구현, 모델 실험 (UNet, UNet++, MANet, SegFormer), Hard voting
김성규 (팀원)	데이터 EDA, 모델 실험(FCN), offline Augmentation
임정아 (팀원)	메타 데이터 EDA, 모델 실험(FCN 크기 비교, PANnet), 이미지 사이즈에 따른 크기 비교, pseudo labeling, soft voting
이승현 (팀원)	베이스라인 코드 변환, 모델 실험(deeplabv3, deeplabv3+, upernet), test time augmentation 실험
김성주 (팀원)	실험 및 협업 환경 구축, 학습 속도 향상, 전처리 증강, 옵티마이저 및 파라미터 탐색, 앙상블
김명철 (팀원)	실험 결과 및 최종 보고서 관리, Data augmentation 실험 수행, SAM 논문 리뷰 진행

2. 관련 연구 및 기법

2.1 모델링 기법

(1) Unet 계열

1) 문제 정의

본 프로젝트 훈련용 데이터의 숫자는 총 800장으로 제한된 상황이며, 겹치는 영역이 많은 손등뼈에 대한 구분이 필요하다. 의료 계열 문제에서 마주할 수 있는 전형적인 상황이다. 위와 유사한 의료 상황에서 사용하기 위해 도입되었고 좋은 성능을 보인 UNet 계열 활용해 Hand bone segmentation의 결과 실험을 진행하였다.

2) 실험 방향

실험은 크게 두 가지 방향으로 진행하였다. 동일한 조건에서 UNet 계열 구조의 변화에 따른 성능 변화를 실험하였다. 또한 backbone 구조 변화에 따른 성능

변화 실험하였다. 성능 변화는 average dice, loss, 시간, 리더보드 결과를 기재하여 비교할 예정이며, 필요할 경우 streamlit으로 시각화 하여 결과 간 비교하였다.

(2) Deeplabv3 계열

1) 문제 정의

손가락 뼈들은 서로 맞닿아 있어 문맥적인 정보를 잘 파악하고, 경계를 찾는 것이 중요하다. deeplabv3 계열 모델은 기본적으로 ASPP를 적용하여 하나의 이미지를 다양한 크기의 feature map에서 바라보며 풍부한 문맥 정보를 추출할 수 있고, encoder-decoder 구조를 사용하여 공간적인 정보를 최대한 유지할 수 있어 적합하다고 생각하였다.

2) 실험 방향

먼저 deeplabv3와 deeplabv3+ 비교하였으며, 모두 backbone 모델로 resnet101 사용하고 공통으로 resize 512를 적용하였다. optimizer로는 adam, learning rate는 0.0001, 100 epoch 적용하였다. 결론적으로 다양한 backbone으로 deeplabv3+를 비교하였다.

(3) MANet

1) 문제 정의

손등에 위치한 뼈들은 겹쳐 있어서 상대적으로 Segmentation 작업을 수행하는데 낮은 성능을 보이고 있다. MA-Net은 Position-wise Attention Block(PAB)과 Multi-scale Fusion Attention Block(MFAB)를 도입해 의료 상황(종양 탐지)에 더 나은 성능을 보였다. 이때 PAB는 픽셀과 전체 관점에서 공간의 의존성을 모델링하며 MFAB는 semantic feature fusion에서 나오는 피쳐 맵 간의 의존성을 탐지한다.

2) 실험 방향

PAB의 공간별 특징 학습을 통해 손등 뼈의 경계와 구조를 학습하는데 더 나은 성능을 보일 수 있다고 가설을 세웠다. 실험은 인코더별 성능 변화 실험을 진행하였으며 base와 비교를 통해 평균 Dice 값이 높은 것을 리더보드에 제출하였다. base보다 높은 성능을 보인 인코더에 대해서는 더 깊은 형태로 변화하면서 실험하였다.

(4) SegFormer

1) 문제 정의

더 나은 성능 달성을 위해 앙상블을 위한 다양한 특성을 가진 모델을 실험할 필요성이 있다. 기존 CNN 기반의 모델 외에 Transformer 기반의 모델이 없었기 때문에 Transformer 기반의 Segmentation 모델 추가하였다. 제한된 시간이라는 프로젝트 수행 환경을 고려해 ViT에 비해 효율성, 속도를 개선한 SegFormer를 채택하였다.

2) 실험 방향

제한된 의료 데이터셋을 고려했을 때 모델 크기를 키우는 방향보다는 그동안 개선이 이뤄졌던 부분들을 실험하였다. 이미지 사이즈 증가하고 데이터 증강을 넣으면 성능이 개선된다는 실험 결과를 참고하여 resize 512, 1024, 1536과 데이터 증강 추가하였다.

(5) UPerNet

1) 문제 정의

인간의 손에는 다양한 부위가 존재하고, 각 부위에는 뼈나 인대 같은 요소들이 존재하며, 뼈나 인대를 구성하는 성분 또한 차이가 난다. 여태까지 이러한 요소를 독립적으로 탐색했으나, UperNet에서는 동시에 수행하여 각기 다른 손가락 뼈를 효과적으로 구분할 수 있다고 생각하여 해당 모델을 사용하게 되었다. 해당 모델은 Feature Pyramid Network를 기반으로 만들어졌으며, Pyramid Pooling Module을 backbone 마지막 Layer에 연결하였다. 또한, 여러 visual task를 동시에 학습하기 위해, task 별로 conv layer가 포함된 Head를 연결하여 segmentation 수행한다. 해당 task에는 scene understanding, object, part, material, texture recognition task가 포함된다.

2) 실험 방향

먼저 다른 모델과 UperNet 비교하여 성능 개선이 있는지 확인하였다. 모두 backbone 모델로 resnet101 사용하였으며, 공통으로 resize 512를 적용하였다. Optimizer는 adam, learning rate는 0.0001, 100 epoch에 대해 실험을 수행하였다.

2.2 데이터 증강 및 전처리 기법

(1) 데이터 증강 (Data Augmentation)

1) 문제 정의

Hand bone semantic segmentation 작업은 손 뼈의 정확한 분리를 목표로 하지만, 의료 영상 데이터의 제한된 수량과 다양성은 모델의 일반화 성능에 부정적인 영향을 미친다. 특히, 다양한 환자군, 촬영 각도, 해상도, 및 뼈 구조의 차이를 충분히 반영하지 못하면 모델이 실제 환경에서의 성능을 보장하기 어렵다. 이를 해결하기 위해, 본 연구에서는 데이터 증강 기법을 도입하여 데이터의 다양성을 인위적으로 증가시키고 모델의 학습 효율과 일반화 성능을 개선하고자 한다. 회전, 이동, 스케일 조정, 밝기 및 대조 조정 등 다양한 증강 기법을 실험하며, 각각의 기법이 성능에 미치는 영향을 평가하고 최적의 증강 조합을 탐구하는 것을 목표로 한다.

2) 실험 방향

모델 및 백본의 다양성 탐구하기 위해 다양한

모델(예: PAN, FPN, UPerNet, DeepLabv3+)과 백본(ResNet50, ResNet101)을 활용하여 성능을 비교하며, 각 모델의 특성과 데이터와의 적합성을 분석을 목표로 진행하였다. 백본 변경을 통해 모델의 복잡도와 파라미터 수가 성능에 미치는 영향을 평가하였다. 더불어 데이터 증강 기법의 실험하였다. 다양한 증강 기법(예: Contrast, CLAHE, Resize, HorizontalFlip, Rotate)을 적용하여 데이터 다양성을 높이고 모델의 일반화 성능을 향상시켰다. 증강 기법별 성능 변화를 측정하여 최적의 조합을 탐구하였다.

(2) 이미지 크기에 따른 성능 비교

1) 문제 정의

강의에서 Resize Ensemble에서 우리는 객체의 크기에 따라 학습을 달리한다는 것을 배웠다. 따라서 우리는 Resize를 했을 때 어떤 객체를 잘 학습하는지 알아보고, 이에 따라 dice값이 어떻게 변하는 지 알아보고자 하였다.

2) 실험 방향

Base Model은 FCN, Backbone은 Resnet 50, learning rate는 $1e-4$, epoch은 30으로 설정하여 실험을 수행하였다. 다양한 상황에서 Epoch를 돌려본 결과 Epoch가 30일 때 시간 대비 dice 성능이 가장 좋았다. 이를 바탕으로 빠른 실험 결과를 알 수 있을 것이라 예상했다.

(3) 성능이 낮은 손목 부분 crop하여 성능 개선

1) 문제 정의

손목의 겹쳐져 있는 부분의 dice 성능이 상대적으로 낮은 결과를 보이고 있다. 특히 손목의 뼈와 손가락 끝부분들이 상대적으로 다른 뼈에 비해서 크기가 작았다. 이에 손목 뼈 부분만 crop하여 학습시켜 성능 개선하고자 하였다.

2) 실험 방향

먼저 손목 뼈에 집중하기 위해 손목뼈를 제외한 label은 masking을 수행하고 관계성을 위해 잘려진 모든 부분을 labeling을 수행하였다. 기존에 데이터에 추가학습을 수행하였으며 손목 부분을 warm-up 형식으로 먼저 학습 후 전체 이미지 학습하였다. 위 결과와 기존 모델 결과를 앙상블을 수행하였다.

2.3 기타 성능 개선 기술

(1) AMP (Automatic Mixed Precision)

1) 문제 정의

본 프로젝트에서 수행하는 Hand Bone image의 segmentation 작업은 의료 영상 분석에서 중요한 역할을 한다. 그러나 모델의 훈련과 평가에는 높은 계산 비용과 긴 소요 시간이 필요하며, 이는 다양한 모델

아키텍처 및 하이퍼파라미터 설정에 대한 실험을 제한한다. 이에 본 프로젝트에서는 AMP (Automatic Mixed Precision) 기법을 도입하여 GPU 메모리 사용량을 줄이고 훈련 속도를 높이하고자 한다. AMP를 적용했을 때의 메모리 효율성 및 연산 성능을 평가하고, 모델의 성능 저하 없이 훈련 프로세스를 최적화하는 방안을 탐구하는 것이 목표이다.

2) 실험 방향

실험 조건으로 backbone은 resnet50, Resize(256, 256), num_epochs는 20, learning rate는 0.0001, BCEWithLogitsLoss와 Adam을 사용하였다. original train time과 use_amp time을 비교하여 성능과 실험 시간의 개선점을 파악한다.

(2) TTA (Test Time Augmentation)

1) 문제 정의

TTA(Test Time Augmentation)란 model을 테스트할 때 augmentation을 적용하는 방법으로, 원본으로부터 변형된 여러가지 augmentation에 대해 평가를 실시하고, 최종 분류 값이 무엇인지 예측한다. model에 한 가지 이미지를 주는 것보다 여러 가지 변형된 이미지를 사용하여 평가를 하게 되면 예측과 GT에 대한 오차 값은 작아지고, 더욱 정확한 예측이 가능해지므로, 해당 프로젝트에 유용하다고 생각하였다.

2) 실험 방향

같은 모델에 대하여 TTA 적용 유무에 따른 수치 비교하였다. ttach 라이브러리 사용하여 각 증강 기법 적용 후 merge할 때 mean 방식으로 진행하였다. HorizontalFlip와 Multiply 적용에 대한 실험을 수행하였다.

3. 프로젝트 수행 및 방법

3.1 EDA (데이터셋 분석)

(1) 픽셀 겹침 분석

라벨 데이터는 (클래스, 이미지 w, 이미지 h) 형태로 구성되어 있으며, 각 클래스에 해당하는 영역은 1로, 그렇지 않은 영역은 0으로 나타난다. 이러한 구조로 인해 한 픽셀에 여러 클래스가 매핑되는 경우가 발생한다. 이를 시각화한 결과, 손등 부분의 픽셀에서 클래스 간 겹침이 많이 발생하는 것을 확인할 수 있었다. 특히 Pisiform와 Triquetrum, Trapezoid와 Trapezium 등 특정 클래스 간에 겹침이 두드러지게 나타났다. 이러한 겹침은 학습 과정에서 모델 성능 저하의 원인이 될 가능성이 있으므로, 이를 해결하기 위한 방안을 모색할 필요가 있다.

(2) 이상치 분석

매니큐어 유무에 따른 이상치를 분석하기 위해 두

이미지를 비교한 결과, 매니큐어를 바른 경우와 바르지 않은 경우 모두 육안으로 보기에 큰 차이가 없었다. 성별, 나이, 키, 몸무게 등 기본적인 조건이 유사한 두 사례를 비교했을 때도 매니큐어가 모델에 영향을 미칠 가능성은 낮아 보였다. 따라서 매니큐어와 관련된 변수에 대한 추가적인 이상치 분석은 필요하지 않다고 판단되었다.

(3) 메타데이터 분석

성별과 나이에 따른 키와 체중 분포를 분석한 결과, 여성이 남성보다 평균적으로 체중과 키가 모두 낮은 경향을 보였다. 특히 50대 여성의 키 분포가 다양하게 나타났으나, 평균 키와 표준편차에는 큰 영향을 미치지 않았다. 이러한 메타데이터의 분포 차이는 성별 및 나이 데이터를 활용한 모델링 과정에서 중요한 정보로 활용될 수 있을 것으로 보인다.

3.2 실험 환경 및 하드웨어 구성

(1) 서버

- CPU: Intel® Xeon® Gold 5120
- GPU: V100 32GB 1EA

(2) 협업 및 의사소통

- 데일리 스크럼에 실험하고 있는 프로젝트 진행 상황 공유, 피어세션 시간에는 수행한 내용에 대한 보고서 작성 및 공유
- Notion에 실험 결과 및 보고서에 대한 내용 정리
- 효율적인 서버 관리를 위해 시간표를 통해 공유.

3.3 실험 변수 및 평가 기준

실험은 다양한 데이터 전처리, 증강 기법, 모델 아키텍처, 그리고 백본의 성능을 비교하기 위해 설계되었다. 평가지표로는 주로 평균 Dice 점수(Avg Dice)를 사용했으며, 학습 손실(Train loss), 검증 손실(Valid loss), 그리고 리더보드 점수(Leaderboard score)를 보조 지표로 활용했다.

평균 Dice 점수를 기반으로 최적의 데이터 증강 기법, 이미지 크기, 모델 계열 및 백본을 도출했다. 이러한 평가 방식은 모델 성능의 세부적인 비교와 성능 향상 요소를 명확히 식별하는 데 중점을 두고 진행하였다.

4. 실험 결과

4.1 데이터 전처리 및 증강 (실험)

(1) Albumentations

Augmentation (epoch 50)	backbone	Avg Dice
Default	resnet 50	0.9266
Rotate	resnet 50	0.9269
RandomBrightnessContrast	resnet 50	0.9256

Contrast	resnet 50	0.9272
CLAHE	resnet 50	0.9274
HorizontalFlip	resnet 50	0.9270
Sharpen	resnet 50	0.9273

(2) image resize

size (epoch 30)	backbone	Avg Dice
256 x 256	resnet 50	0.8431
512 x 512	resnet 50	0.8575
1024 x 1024	resnet 50	0.9644

4.2 모델 학습 과정 (실험)

(1) Unet 계열

backbone	Avg Dice	Train loss	Valid loss	Leaderboard
UNet (a)	0.95041	0.00033	0.01789	0.9474
UNet++ (b)	0.95347	0.00023	0.02526	0.9538
resnet50 (b)	0.95347	0.00023	0.02526	0.9538
resnet101	0.95347	0.00023	0.02526	0.9517
efficient-b5 (c)	0.95290	0.00035	0.021821	0.9511
efficient-b6	0.95168	0.00037	0.033961	-
resnest50d	0.95356	0.00018	0.039582	-
gernet_l (d)	0.95283	0.00025	0.038	0.9513

(2) deeplabv3 계열

backbone	Avg Dice	Train loss	Valid loss	Leaderboard
resnet101	0.9420	0.0005	0.0244	0.9450
resnet101	0.9496	0.0005	0.0204	0.9446
resnet50	0.9503	0.0005	0.0205	0.9450
resnet101	0.9496	0.0005	0.0204	0.9446
resnext10_32x8d	0.9489	0.0004	0.0213	0.9470
efficientnet-b8	0.9515	0.0005	0.0191	0.9489
regnetx_320	0.9484	0.0003	0.0220	0.9467
xception71	0.9512	0.0004	0.0196	0.9470
Efficientnet_cc_b1_8e	0.9472	0.0005	0.0209	0.9421

(3) MANet

backbone	Avg Dice	Train loss	Valid loss	Leaderboard
(Base)	0.9101	0.0008	0.1287	-
ResNet50	0.8989	0.0012	0.1385	-
Res2Net50	0.8957	0.0012	0.1460	-

_26w_4s				
ResNest50 d_4s2x40d	0.9026	0.0009	0.1361	–

(4) SegFormer

backbone	Avg Dice	Train loss	Valid loss	Leaderboard
mit-b0 (512)	0.9610	0.0005	–	0.9573
mit-b0 (1024)	0.9667	0.0005	–	0.9646
mit-b0 (1536)	0.9011	0.0005	–	–

(5) UperNet

Backbone (resize)	Avg Dice	Train loss	Valid loss	Leaderboard
resnet101 (1024)	0.9649	0.0003	0.0209	0.9649
mit_b5 (1024)	0.9653	0.0004	0.0198	0.9649
Efficientnet v2_xl (1024)	0.9656	0.0003	0.0218	0.9613
efficientnet-b8 (1024)	0.9640	0.0006	0.0410	0.9558
densenet161 (1024)	0.9662	0.0003	0.0229	0.9640
swin transformer (512)	0.95012	0.0003	0.0330	–
swin transformer (1024 + Augmentation)	0.9731	0.0006	0.0293	0.9700

5. 최종 실험 결과 및 분석

5.1 모델별 성능 비교

모델 성능을 비교하기 위해 UNet, DeepLabV3, MANet, SegFormer, UperNet 계열 모델을 대상으로 실험을 진행하였다. UNet 계열에서는 UNet++, ResNet50, ResNet101 등의 백본을 사용한 모델들이 평균 Dice 점수 0.95347로 가장 높은 성능을 기록하였다. DeepLabV3 계열에서는 EfficientNet-b8 백본을 사용한 모델이 평균 Dice 점수 0.9515로 가장 높은 성능을 보였다. MANet 계열은 다른 모델에 비해 낮은 성능을 기록했으며, 평균 Dice 점수가 최대 0.9101에 그쳤다. SegFormer 계열에서는 mit-b0 백본을 사용한 모델이 평균 Dice 점수 0.9667로 높은 성능을 기록하였다.

UperNet 계열에서는 DenseNet161 백본이 평균 Dice 점수 0.9662를 기록했으며, Swin Transformer를 활용해 1024 크기 입력과 데이터 증강을 적용했을 때 평균 Dice 점수가 0.9731로 가장 높은 성능을 달성하였다.

모델 외에 추가적으로 Optimizer에 대한 실험을 진행하였다. 실험을 진행한 Optimizer는 Adam, AdamW, Lion, RMSProp을 선택하여 보다 빠른 실험 진행을 위해 30 epochs만큼 학습하였다. 그 결과, Adam과 AdamW는 유사한 학습속도를 보이고 Lion과 RMSProp은 전자보다는 빠른 학습속도를 보임을 확인하였다.

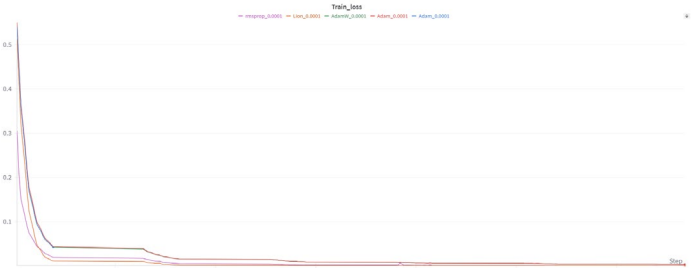


그림 1 Optimizer별 Train Loss 추이

5.2 데이터 증강 및 전처리 기법 효과 분석

(1) 데이터 증강 (Data Augmentation)

데이터 증강의 효과를 확인하기 위해 Albumentations 라이브러리를 활용한 다양한 증강 기법을 실험하였다. 기본 설정과 비교했을 때, CLAHE와 Sharpen 증강을 적용한 경우 각각 평균 Dice 점수가 0.9274, 0.9273으로 가장 높은 성능을 기록하였다.

(2) 이미지 크기에 따른 성능 비교

입력 이미지 크기 변화에 따른 성능 비교를 위해 256, 512, 1024로 다르게 하여 FCN 모델의 ResNet 50으로 30 epochs만큼 실험을 진행하였다. 이미지 크기를 조정하는 실험에서는 256x256 크기에서 평균 Dice 점수 0.8431, 512x512 크기에서 0.8575로 성능이 증가하였으며, 1024x1024 크기에서는 0.9644로 가장 높은 성능을 달성하였다. 특히 UperNet 모델에서 1024 크기의 입력과 데이터 증강을 결합했을 때 성능이 대폭 향상되었으며, 평균 Dice 점수가 0.9731로 나타나 데이터 증강과 입력 크기의 중요함을 확인하였다.

(3) 성능이 낮은 손목 부분 crop하여 성능 개선

이미지의 손목 부분에 대해 crop한 데이터를 추가로 학습을 진행한 결과, 기존보다 이른 시간에 학습이 진행됨을 확인하였고, 성능 지표 역시 향상되었다. 이를 통해 추가적인 데이터를 제공함으로써 모델의 성능을 향상시킬 수 있음을 확인하였다.

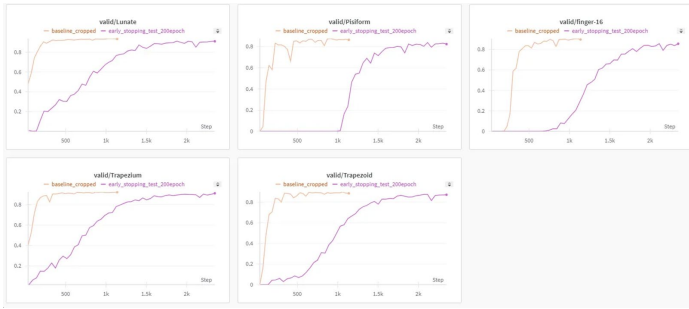


그림 2 Crop 데이터에 대한 성능 향상 효과

5.3 Ensemble model

(1) Hard Voting

모델 성능 개선을 위해 Hard Voting 기반의 앙상블 실험을 진행하였다. Hard Voting은 각 클래스 픽셀별로 여러 모델의 예측 결과를 종합한 후, 설정된 threshold를 초과하는 경우에만 결과에 반영하는 방식이다. 이를 통해 서로 다른 모델 간 예측 결과를 보완하고 정밀도를 높이는 것이 목표였다.

첫 번째 실험에서는 SegFormer, UNet++, UperNet, DeepLabV3+ 등의 모델을 조합하여 Hard Voting 앙상블을 진행하였다. 결과적으로 입력 데이터의 최고 성능(0.9649)보다 각각 0.0039(0.9688) 및 0.0043(0.9692)만큼 향상된 성능을 기록하였다. threshold 값을 높였을 경우 결과 이미지의 테두리가 더 부드럽게 나타나며 정밀도가 향상됨을 시각적으로 확인하였다.

두 번째 실험에서는 리더보드에서 높은 성능을 기록한 UperNet ResNet101과 UperNet mit b5를 포함하여 서로 다른 세 가지 모델과 조합을 설계하여 실험하였다. (아래 그림3 참고) threshold 값에 따라 성능 차이를 비교한 결과, 낮은 threshold 값(1)에서는 더 높은 Dice 점수를 기록(0.9676)했으며, threshold를 2로 높였을 때는 정밀도가 증가하면서 0.9689의 성능을 달성하였다. 그러나 기존 UperNet 최고 성능 모델(0.9700)과의 앙상블에서는 threshold가 낮은 경우 잘못된 예측이 포함되어 오히려 성능이 하락하는 결과(0.9696)를 보였다.

Hard Voting 앙상블은 threshold 조정에 따라 성능과 정밀도가 변화하며, 상황에 따라 적절한 threshold 설정이 필요하다. 특히, 개별 모델 간 성능 차이가 크면 threshold 설정이 성능 향상에 중요한 역할을 한다는 점을 확인하였다.

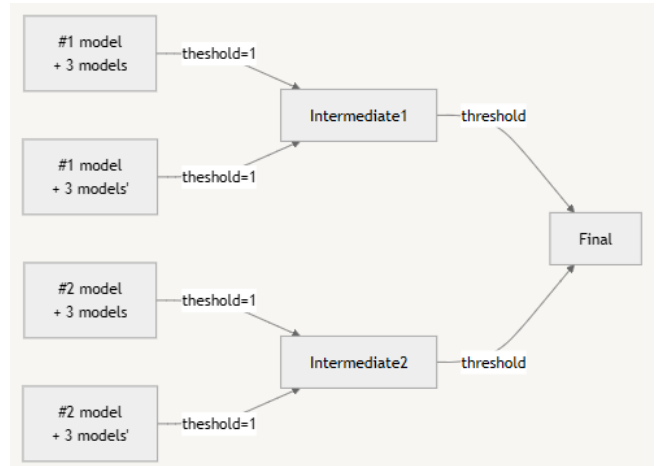


그림 3 Hard Voting 실험2 설계

(2) Soft Voting

모델 성능을 개선하기 위해 Soft Voting을 적용하여 앙상블 실험을 진행하였다. Soft Voting은 여러 모델의 예측 값을 결합해 평균을 내고, 이를 기반으로 threshold를 적용해 최종 예측을 생성하는 방식이다. 실험에서는 서로 다른 모델을 조합한 경우와 동일 모델에서 백본만 변경한 경우의 성능을 비교하였다.

첫 번째 실험에서는 UNet++, UperNet(mit_b5), FCN(resnet50) 세 모델을 조합하여 Soft Voting을 수행하였다. Threshold를 0.2로 설정했을 때는 Dice 점수가 0.8733으로 나타났고, 0.5로 높였을 때는 0.9570의 성능을 기록하였다. 이는 threshold 값이 결과의 정밀도에 중요한 영향을 미친다는 점을 보여준다.

두 번째 실험에서는 동일한 모델(UperNet)에 백본을 변경하여 조합한 경우를 평가하였다. UperNet(mit_b5), UperNet(efficientnetv2_xl), UperNet(efficientnet-b8)을 조합한 결과, Dice 점수 0.9677로 더 높은 성능을 달성하였다. 이는 동일한 모델 구조에서도 백본 조합이 성능 향상에 크게 기여할 수 있음을 확인시켜준다.

결론적으로 Soft Voting은 여러 모델 간 예측 값을 보완하며 성능을 향상시키는 데 효과적이었다. threshold 값 조정과 백본 선택이 최종 성능에 중요한 요소임을 확인하였으며, 특히 동일한 모델 내에서 백본 조합이 가장 높은 성능을 기록하였다.

6. 결론

6.1 프로젝트 결과 요약

이번 프로젝트는 Hand Bone Image Segmentation의 성능을 최적화하기 위해 다양한 모델(Unet, DeepLabV3, MANet, SegFormer, UPerNet)과 데이터 증강 기법을

활용하였다. Albumentations를 통해 다양한 증강 기법(Rotate, Contrast, CLAHE, HorizontalFlip, Sharpen 등)을 적용하여 데이터 다양성을 높였으며, 이미지 크기를 조정(256, 512, 1024)해 최적의 입력 크기를 도출하였다.

모델 성능 비교에서는 SegFormer와 UPerNet 계열이 가장 높은 성능을 기록하였으며, 특히 UPerNet(DenseNet161)과 Swin Transformer를 활용해 1024 크기 입력 및 데이터 증강을 적용한 결과, 평균 Dice 점수 0.9731로 최고의 성능을 달성하였다. 또한 AMP(Automatic Mixed Precision)와 TTA(Test Time Augmentation) 기법을 활용해 학습 및 추론 효율성을 높였다.

앙상블 실험에서는 Hard Voting과 Soft Voting을 활용하여 모델의 성능을 종합적으로 평가하였다. Hard Voting은 threshold 값 조정을 통해 예측 결과의 정밀도를 향상시켰으며, Soft Voting은 동일 모델에서 백본 조합을 통해 성능을 효과적으로 보완하였다. 두 방법 모두 threshold 조정이 최적의 결과를 도출하는 데 중요한 요소임을 확인하였다.

6.2 한계점 및 향후 연구 방향

(1) 한계점

1) 데이터 양의 제한성

본 프로젝트에서 사용된 데이터셋은 800장으로 제한적이었다. 이는 모델의 일반화 성능에 영향을 미칠 가능성이 있으며, 다양한 환자군과 상황을 반영하기 어렵다.

2) 손목 뼈와 손가락 끝 부분의 낮은 성능

손목 및 손가락 끝부분과 같이 크기가 작은 부위에서 Segmentation 성능이 상대적으로 낮게 나타났다. 이는 뼈 구조의 복잡성과 겹침으로 인해 발생한 문제이다.

3) 최적의 증강 조합 탐색 미흡

Albumentations를 활용한 데이터 증강 실험에서 다양한 기법의 개별 효과를 확인했지만, 증강 기법들의 조합에 대한 심층적인 탐색이 부족하였다.

(2) 향후 연구 방향

1) 데이터셋 확대 및 다양화

실제 의료 현장에서의 성능을 개선하기 위해 더 많은 데이터를 확보하고, 다양한 촬영 각도와 해상도를 포함한 데이터셋으로 실험을 확장해야 한다.

2) 세부 부위 성능 개선

손목 뼈와 손가락 끝부분의 Segmentation 성능을 높이기 위해 해당 부위를 특화한 학습 방법(예: Region-

specific Training)이나 추가적인 전처리 기법을 도입할 필요가 있다.

3) 최적화된 증강 기법 조합 탐구

데이터 증강 기법들의 조합에 대한 더 많은 실험을 수행하여 최적의 조합을 탐색하고, 모델의 일반화 성능을 더욱 향상시킬 수 있도록 한다.

4) 더 다양한 모델 및 백본 실험

Transformer 기반 모델 외에도 새로운 구조나 하이브리드 모델을 실험하고, 다양한 백본 조합을 통해 모델 성능의 한계를 뛰어넘는 방법을 모색한다.

5) 실제 의료 환경에서의 테스트

연구 결과를 실제 의료 환경에서 검증하고, 모델의 실질적인 응용 가능성을 평가하여 연구 결과를 현실에 적용할 수 있도록 한다.

6.3 자체 평가 의견

(1) 박수영

1) 나는 내 학습목표 달성을 위해 무엇을 어떻게 했는가?

모델 실험 결과를 조회하기 위해 streamlit 라이브러리를 사용해 시각화 웹페이지 구현했고, Segmentation Models Pytorch 라이브러리 활용을 위한 코드 기능 추가를 통해 팀이 모두 사용할 수 있도록 했습니다.

2) 나는 어떤 방식으로 모델을 개선했는가?

UNet, UNet++, SegFormer에 대한 실험 수행을 통해 최적의 조합을 찾으려고 시도했고, hard voting 방식의 앙상블 코드를 만들어서 최종 성능 개선을 위한 실험을 진행했습니다.

3) 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떤 깨달음을 얻었는가?

모델 실험을 하면서 단일모델로 dice 0.964를 기록하고, hard voting를 수행했을 때는 기존 모델들의 약 0.004 정도 향상되는 결과를 달성했습니다. 다만, 무조건적인 앙상블이 항상 결과가 좋았던 것은 아니었기 때문에 이에 대한 충분한 실험이 필요하다는 교훈을 얻었습니다.

4) 전과 비교하여 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

지난 객체탐지에서는 사용하지 않았던 OpenMMLab의 라이브러리를 이번에 사용하면서 단일 모델로는 팀에서 높은 성능을 기록할 수 있었습니다.

5) 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

여러 실험의 결과를 종합해 가장 좋은 성능을 낼

것으로 기대되는 모델을 훈련하는데 예상보다 많은 시간이 소요되거나 새롭게 마주한 오류로 인해 충분히 학습을 하지 못했습니다.

6) 한계와 교훈을 바탕으로 다음 프로젝트에서 시도해볼 것은 무엇인가?

훈련 시간이 기하급수적으로 늘거나 갑작스레 발생할 수 있는 오류 가능성을 고려해 이번 프로젝트 마일스톤보다 여유있도록 일정 관리를 해볼 계획입니다.

(2) 김성규

1) 나는 내 학습목표 달성을 위해 무엇을 어떻게 했는가?

이번 목표는 가설설정을 설정하고 실험하기와 적극적으로 협업하기였습니다. EDA를 통해 집중해야하는 부분이 무엇인지 찾아내고, 부족한부분을 채우기 위해 가설을 설정하였으며, 실험을 통해 성능향상에 기여하였습니다. 팀에 도움이 될 수 있는 방법을 계속해서 고민하고 제안하였습니다.

2) 나는 어떤 방식으로 모델을 개선했는가?

offline-Augmentation을 진행하였습니다. 성능이 낮은 손목부분과 손가락 끝부분 등을 crop하여 새로운 dataset을 추가하였고, 손목부분만 확대한 이미지, 손가락부분만 확대한 이미지 등을 추가하여 성능향상에 기여하였습니다.

3) 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떤 깨달음을 얻었는가?

EDA를 통해 작은 객체 탐지에 집중해야한다는 사실을 알았고, 이에 집중하려고 하였으며, 작은 부분을 crop하여 이미지를 사용하였고 이를 통해 예측을 잘 수행하지 못하였던 label들의 성능을 0.82~0.91에서 0.88~0.94로 향상시켰습니다. 작은 객체탐지를 위해 이미지 크기가 중요하다는 것을 깨달았습니다.

4) 전과 비교하여 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

모델 학습을 효율적으로 하기위해 early stopping을 구현해보았습니다. 동일한 조건에서 효율적인 실험을 할 수 있었습니다. Inference한 결과들을 시각화하며 진행해보았습니다. 제대로 예측하지 못하고, 학습이 부족한 부분이 무엇인지 확인할 수 있었습니다.

5) 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

코딩실력에 한계를 느꼈습니다. 한가지 실험을 하기 위해 다양한 시행착오를 겪으며 너무 많은 시간이 소모되었습니다. 이미 학습이 완료된 checkpoints에 추가로 학습하여 fine-tuning 하는 방법을 사용했으면 조금 더 성능이 좋아지거나, 학습에 소요되는 시간을 많이 줄일 수 있지 않을까 하는 아쉬움이 많이

남습니다. SAM 모델을 활용해보고자 했는데 시도해보지 못하였습니다. EDA를 통해 알아낸 사실들을 너무 쉽게 지나친것 같습니다. Validation data를 선정할 때 합리적으로 선정하지 못하였습니다.

6) 한계와 교훈을 바탕으로 다음 프로젝트에서 시도해볼 것은 무엇인가?

한가지 실험에 너무 매몰되지 않고, 시야를 넓게 보려고 노력할 것입니다. EDA를 진행하는것도 중요하지만 EDA의 결과를 꼼꼼하게 분석하고 진행방향을 설정할 것입니다. 또한 이를 바탕으로 Validation dataset을 선정하도록 하겠습니다. 다양한 모델을 우선적으로 학습시킨 후 baseline을 만들고 다양한 시도를 해보고 싶습니다. SOTA 모델들을 사용하여 최신 모델들을 직접 다뤄보고 싶습니다.

(3) 임정아

1) 나는 내 학습목표 달성을 위해 무엇을 어떻게 했는가?

나는 데이터를 보며 어떤 데이터들이 있는지 파악했다. 주로 메타 데이터를 중심으로 파악했으며, 이를 통해 새로운 관점을 만들 수 있는지 고민했다. 겹치는 부분이 많아 이를 보기 위해 노력했다. 이미지 사이즈가 커질수록 정보가 많아져 학습이 잘 될 것이라고 가설했고, 실험을 진행했다. Image Segmentation 분야에서 최신 모델이 무엇이 있는지 파악하고 실험하려 했다. pseudo labeling을 통해 Label Noise를 제거하려고 노력했다. Soft Voting을 진행했다.

2) 나는 어떤 방식으로 모델을 개선했는가?

Soft Voting을 통한 모델 개선, 이미지 크기 변화를 통한 모델 개선, 다른 모델 시도(PAN net)

3) 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떤 깨달음을 얻었는가?

EDA를 통해 데이터의 단순 분석에서 끝나는 것이 아니라, 이를 통해 어떤 방향으로 발전시켜야 하는지 insight를 얻어야 한다. 대회에선 최대한 많은 제출 기회를 사용하자.

모델을 위한 실험을 하는 도중에도 다양하게 모델을 실험해 볼 수 있었는데, 내 실험에만 너무 집중했다. → 다음번엔 가볍게 돌릴 수 있는 모델을 실험하면서 내가 실험할 모델에 대한 전처리를 끝내자.

모델의 크기를 줄이는 것은 좋은 시도이나, 모델의 크기를 줄였을 때 변하는 것도 당연히 존재하기 마련이다 → 이를 고려해서 실험할 수 있는 건 빨리 하고, 그렇지 않은 것은 하지 말자.

4) 전과 비교하여 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

EDA를 진행하지 못했던 점이 아쉬웠는데, 이 부분에

대한 insight를 얻고 갔다. Streamlit의 적극적인 활용 → 어떤 부분이 예측되지 않는지 눈으로 확인할 수 있었다. (Thanks to 수영이형) 크기 비교 실험

5) 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

이미지 크기에 대해 조사했다고 하면서, 2048에 대해선 train 해보지 않은 점. 왜 이미지 크기는 1/2가 되어야 했는지 명확한 근거가 없는 점 (binary하게 단순하게 생각함) 기존에 학습한 모델을 재 사용해서 epoch를 늘릴 수 없었던 점. 서버가 터졌을 때 model을 복구할 수 있는 수단을 마련해두지 않은 점

6) 한계와 교훈을 바탕으로 다음 프로젝트에서 시도해볼 것은 무엇인가?

모델 경량화는 빠르게 시도. 서버가 터졌을 때 model 복구를 원활하게 하기 위해 반드시 모든 모델을 local에 저장할 것. (이를 빠르게 할 방법도 찾아볼 것) 가설을 미리 적어두고 실험할 것. (머리 속에 있는 내용들을 실제로 적어두면 어떤 내용을 실험할 지 보임) EDA를 통해 모델을 개선할 점을 빠르게 파악할 것.

(4) 이승현

1) 나는 내 학습목표 달성을 위해 무엇을 어떻게 했는가?

aistage에서 제공한 베이스라인 코드를 파이썬 코드로 변환하여 yaml 확장자의 config 파일을 통해 모델 실험을 편하게 할 수 있도록 하였으며, torchvision에 있는 모델 외에도 huggingface에 있는 모델을 사용할 수 있도록 코드를 수정하였다. 또한, inference를 진행할 때 test time augmentation을 적용하여 손 뼈의 각 부분이 잘 구분될 수 있도록 노력하였다.

2) 나는 어떤 방식으로 모델을 개선했는가?

해당 프로젝트에서 deeplabv3, deeplabv3+, upernet 모델을 실험했는데, 특징 추출을 잘 할 수 있다고 보여지는 encoder 모델을 선정하고 교체하면서 성능 향상이 이뤄지는지 확인해보았다. 기본적으로 파라미터 사이즈가 큰 모델을 선정하여 각 손 뼈들의 특징을 잘 추출할 수 있도록 하였다. 또한, test time augmentation을 사용해 여러 번 변형된 이미지에서 inference를 진행하고, 예측 값을 평균하거나 최댓값만 가져와 사용해서 더욱 향상된 예측을 할 수 있도록 하였다.

3) 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떤 깨달음을 얻었는가?

deeplabv3와 deeplabv3+로 실험한 결과는 비슷했으며, 리더보드 상 0.94 ~ 0.95의 dice 값 분포를 보여줬다. upernet의 경우 0.96 이상의 dice 값을 보여줬다. 리더보드에 최종 제출한 모델 또한 swin

transformer를 encoder로 사용한 upernet이었으며 0.97의 dice 값을 얻었다. 이러한 실험을 통해서 대표적인 segmentation 모델인 deeplabv3 계열 모델이 생각보다 점수가 낮게 나와서 아쉬웠으며, 다른 모델로 많이 실험하면 좋지 않았을까하는 아쉬움이 든다.

4) 전과 비교하여 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

hugging face에서 모델을 가져와 사용한 것이 새로웠다. 기존에는 한정된 모델을 가지고 실험을 했었기 때문에 hugging face에 있는 모델을 사용한다는 생각을 하지 못했었다. 그러나 이번에는 많은 모델을 실험할 필요성을 느꼈기에 hugging face 모델을 사용할 수 있도록 코드를 수정하고 실험을 진행해보았다. 또한 test time augmentation을 적용하여 inference 시 더욱 정확한 예측이 가능할 수 있도록 하였다.

5) 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

모델 하나 실험하는데 시간을 너무 많이 사용하여 실험한 모델보다도 좋은 성능을 지닌 모델을 실험하지 못한 것이 아쉬웠다. 그래서 프로젝트가 끝나는 시기와 와도 큰 성능 향상을 하지 못했으며, 시간 분배를 하지 못해 여태까지 실험했던 것들을 다 적용하지 못한 것에 대해 슬프다.

6) 한계와 교훈을 바탕으로 다음 프로젝트에서 시도해볼 것은 무엇인가?

시간 분배를 잘 해서 프로젝트 기간 안에 모든 실험을 적용할 수 있도록 할 것이며, 하나의 모델에만 집착하지 않고 성능이 잘 나오지 않다면 바로 포기하는 결단력을 가져서 다양한 모델을 실험할 수 있도록 노력할 것이다.

(5) 김성주

1) 나는 내 학습목표 달성을 위해 무엇을 어떻게 했는가?

문제와 가설을 세우고 실험하려고 노력했다. 달라진 팀에 맞추어 협업과 실험환경을 구축했고 지속적으로 프로젝트 환경을 개선하려고 노력했다. 메모하고 계획하는 습관을 세우려고 노력했다.

2) 나는 어떤 방식으로 모델을 개선했는가?

전처리 방식들을 실험해보며 데이터 센트릭한 관점으로 모델을 발전시키려고 했다. 실험에 걸리는 시간이 너무 컸기 때문에 이를 줄이고자 모델의 파라미터 자료형 크기를 조절하는 AMP를 적용하여 실험 시간을 유의미하게 줄일 수 있었다. 옵티마이저의 종류와 파라미터를 탐색하며 최종적으로 모델의 성능을 높이기 위해 노력했다.

3) 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떤 깨달음을 얻었는가?

완성된 환경과 코드는 없다는 것을 느꼈다. 실험환경, 협업공간, 프로젝트 코드 등에서 끊임없이 개선할 점과 수정해야 할 점들이 보였고 내가 수행하고 기여하며 발전시킬 수 있는 것들이 많았기에 이를 수행하는 즐거움을 느낄 수 있었다.

4) 전과 비교하여 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

실험시간을 줄이고 코드의 오류를 찾기 위해 학습프로세스를 라인 바이 라인으로 분석하며 부스트캠프 강의내용을 확실하게 이해하며 깊게 배울 수 있었다. 추가적으로 코드의 치명적인 오류를 최종 실험 전에 고칠 수 있었고 이에 보람을 느꼈다. 협업환경과 실험환경이 새로운 팀으로 변경되면서 상호간의 작업 환경이나 결과들을 더 쉽고 빠르게 공유하고 확인할 수 있어 좋았고 보람있었다.

5) 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

전체 프로젝트 기간을 더 꼼꼼하게 구분하고 계획하지 못했던 것이 아쉽다. 더 많은 실험과 가설, 논문 공부를 병행하지 못한 거 같아 아쉽다. 항상 그렇듯이 마지막에는 시간이 너무 부족하고 아깝게 느껴졌다.

6) 한계와 교훈을 바탕으로 다음 프로젝트에서 시도해볼 것은 무엇인가?

프로젝트 코드와 협업 환경을 더 체계화해서 불필요한 시간 소모를 더 줄이기 위해 초반에 더 많은 시간을 소모해야겠다. 또한 전체적인 계획을 세우고 세분화된 계획 수립에 더 시간을 들여 실험과 공부를 병행할 수 있도록 노력할 것이다. 또한 내 생각을 더 남기고 공유하기 위해 오피시디언을 더 많이 키려고 노력할 것이다.

(6) 김명철

1) 나는 내 학습목표 달성을 위해 무엇을 어떻게 했는가?

노선을 적극적으로 활용하여 실험 결과와 진행 상황을 체계적으로 정리하고 공유하였다. [가설 설정 - 문제 정의 - 코드 구현 - 결론 도출]의 구조를 바탕으로 실험 과정을 기록하며 팀원들과의 협업을 원활히 진행하였다. 실험의 전반적인 관리와 보고서 초안을 작성하며, 전체적인 방향성을 잡는 데 기여하였다.

2) 나는 어떤 방식으로 모델을 개선했는가?

Albumentations의 demo 기능을 활용하여 다양한 증강 기법의 시각적 결과를 사전에 판단하였다. 의료 영상 데이터 특성상 뼈의 경계와 대비가 분명하게

나타나는 경우 좋은 결과를 보이는 것을 발견하였고, 이러한 인사이트를 바탕으로 Contrast와 CLAHE 같은 증강 기법을 우선적으로 실험에 적용하였다. 이를 통해 데이터 증강이 모델 성능에 미치는 긍정적 영향을 확인하였다.

3) 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떤 깨달음을 얻었는가?

의료 영상 데이터의 특성에 적합한 증강 기법 선택이 성능 향상의 중요한 요인임을 깨닫게 해주었다. 또한, 노선을 기반으로 한 협업 방식 덕분에 다양한 실험 보고서가 체계적으로 정리되었으며, 각 팀원이 진행한 실험 결과를 빠르게 공유하고 피드백할 수 있는 환경을 마련하였다. 이러한 협업 방식은 실험 효율성을 극대화하고 새로운 시도를 장려하는 데 효과적임을 느꼈다.

4) 전과 비교하여 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

리더보드 순위에 집착하기보다 실험 보고서 작성에 집중하여 프로젝트를 진행하였다. 이는 결과 중심보다는 과정 중심의 연구자 마인드를 갖추게 해주었다. 특히, 의료 영상 데이터를 개별적으로 분석하고 각 데이터의 특성에 맞는 접근법을 시도하며 실험의 질을 높였다. 이러한 연구 방식은 단순한 결과 도출을 넘어 실험의 의미와 방향성을 더 깊이 이해하게 해주었고, 향후 연구에서도 큰 도움이 될 경험이었다.

5) 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

SAM 모델을 활용해보고자 했으나, 시간과 자원 부족으로 인해 활용하지 못한 점이 아쉬웠다. 또한, 팀원들이 도출한 EDA 결과를 데이터 증강이나 모델 실험에 더 적극적으로 반영하지 못했다. 예를 들어, 손목의 평균적인 회전 각도를 기반으로 Rotate 증강을 적용하는 방식처럼 데이터를 보다 세밀하게 활용하는 실험을 시도하지 못한 점이 아쉬움으로 남는다. 이러한 제한은 실험의 다양성과 정밀도를 높이는 데 있어 중요한 교훈을 남겼다.

6) 한계와 교훈을 바탕으로 다음 프로젝트에서 시도해볼 것은 무엇인가?

다음 프로젝트에서는 EDA 결과를 바탕으로 프로젝트 방향성을 팀원들과 더 깊게 논의할 예정이다. 특히, 데이터 특성을 세밀히 분석하고 이를 증강 기법이나 모델 실험에 반영할 수 있도록 협업의 질을 높이고자 한다. 또한, 노선을 통한 협업 방식을 유지하면서도 주기적인 실험 보고서 미팅을 통해 진행 상황과 개선 방향을 논의하고 실험 과정을 더 철저히 관리할 계획이다. 이러한 방식을 통해 프로젝트의 완성도를 더욱 높이고자 한다.

