



커널 아카데미 : AI 부트캠프

Computer Vision Seminar | 2025. 09. 12(금)

목차

- 01. 팀 소개
- 02. 경진대회 수행 절차 및 방법
- 03. 분석 인사이트 및 결과
- 04. 회고

01

팀 소개

팀장/팀원 소개
협업 방식

【김앤장】 CV classification 모델 성능 향상을 위한 방법 마스터



팀장

김수환

이상치탐지/통계
실험환경 구성 및 모델 개발



팀원

김명철

ML 엔지니어/산업공학
모델 개발



팀원

김상윤

ai 엔지니어/컴퓨터학부
모델 개발



팀원

김광묵

웹디자인
모델 개발



팀원

장윤정

개발
모델 개발

경진대회 협업 방식

: Computer Vision [대회] Image Classification

협업 마인드셋 :

CV model task에 대한 전반적인 이해와 성능을 올리기위한 프로세스 마스터를 위한 적극적인 참여

협업 진행 횟수 및 일정 :

1 ~ 2일 간격으로 주기적인 미팅

협업 진행 시 생긴 문제점 :

1. 컴퓨터 비전 모델에 대한 개념이 완벽하게 정립되지 않아서 코드작성에 어려움.
2. 공통의 실험환경에서 실험을 할 수 있도록 Hydra 구성 구조에 대한 설계가 매우 어려웠음.

문제 해결 방법 :

다양한 LLM 모델 활용 및 팀원과의 이슈 공유를 통해 해결

02

경진대회 수행 절차 및 방법

목표 수립
수행 내용 / 수행 결과

경진대회 목표 수립

: Computer Vision [대회] Image Classification

주제

Document Type Classification | 문서 타입 분류

문서는 금융, 보험, 물류, 의료 등 도메인을 가리지 않고 많이 제공됩니다. 이 대회는 다양한 종류의 문서 이미지의 클래스를 예측합니다.

목표

목표

- 1. CV model에 대한 다양한 실험을 통해 성능 개선 방법 및 모델 학습 방법 익히기
- 2. 적극적인 실험을 통해 매일 주어지는 제출횟수 최대한 소비하기
- 3. 뒤처지는 사람 없이 모두 일정 성과를 이룰 수 있도록 협업하기

개요

소개 및 배경 설명

이번 대회는 실제 현업 데이터를 기반으로 17종 문서를 분류하는 과제로, 참가자는 1,570장의 학습 이미지를 활용해 3,140장을 예측하며 다양한 백본 모델과 기법을 적용해 성능을 높이는 경험을 하게 됩니다.

기간

2025. 09. 01 ~ 2025. 09.12

경진대회 수행 내용

: Computer Vision [대회] Image Classification

1

* 개발 환경 구축

1. *Anaconda, miniconda, pyenv*으로 가상환경 구축
2. 패키지 및 프로젝트 관리 *UV*
3. 실험환경구성 *Hydra*

2

* 데이터 분석

1. 이미지 크기 분포 확인
2. 테스트셋 확인
3. 이미지 클래스별 빈도확인

3

* *Feature* 엔지니어링

1. 데이터 전처리
 - 이미지 *Padding, Resolution*
 - 다양한 *Augmentation*
2. *Pretrained model* 활용
 - parameter 54M ~ 304M
3. *TTA, Ensemble method*
 - *tta, CV ensemble, model ensemble, etc ...*

4

* 모델 선택 학습 및 평가

1. *Test set*과 유사하게 증강된 *Validation set* 구축.
2. *StratifiedKFold*를 적용하여 학습.
3. *validation f1-score macro, validation loss*를 통한 성능평가

환경구축

: Computer Vision [대회] Image Classification

1. 배경 및 문제 정의

- 모델 실험에서의 문제점
 1. 너무나도 다양한 하이퍼파라미터(*Learning rate*, *batch size*, *backbone* 등)을 관리하기 어려움
 2. 같은 코드라도 *config* 관리가 안되면 결과 재현이 어려움
 3. 실험 로그 / 결과가 중구난방으로 흩어짐
- Hydra 도입 배경
 1. *Config* 중심의 체계적인 실험 관리
 2. 실험 자동화 및 멀티런 지원
 3. 팀 협업 시 표준화된 구조 제공

2. Hydra

- *YAML* 기반의 계층적 설정관리
- *Command line override* 기능
- 멀티런

3. 프로젝트 내 적용방법

- *Defaults list*로 조합 가능한 구조(*model*, *data*, *optimizer*, *scheduler*, *callback* 등)
- *CV*, *replay*, *img_size* 등 다양한 옵션 등을 *Hydra override*로 제어
- 실험자동화 : *Multi-run*, *WandB* 연동을 통한 실험 *Logging*

환경구축

: Computer Vision [대회] Image Classification

```
configs/
|-- callback
|   |-- early_stopping.yaml
|   |-- lr_monitor.yaml
|-- config.yaml
|-- data
|   |-- data.yaml
|-- model
|   |-- convnext_base.yaml
|   |-- convnext_large_fb_in22k_ft_in1k.yaml
|   |-- convnext_tiny.yaml
|   |-- eff_b0.yaml
|   |-- eff_b1.yaml
|   |-- eff_b2.yaml
|   |-- eff_b3.yaml
|   |-- eff_b4.yaml
|   |-- ghostnet.yaml
|   |-- mobilenetv2.yaml
|   |-- mobilenetv3.yaml
|   |-- resnet34.yaml
|   |-- resnet50.yaml
|-- infer
|   |-- infer.yaml
|-- logger
|   |-- wandb.yaml
|-- optimizer
|   |-- adam.yaml
|   |-- adamw.yaml
|   |-- rmsprop.yaml
|-- scheduler
|   |-- cos_lr.yaml
|   |-- cos_warm.yaml
|   |-- reduce.yaml
|-- train
|   |-- trainer.yaml
```

```
python train.py model=resnet50 data.replay=False optimizer=adamw
```

```
You are using a CUDA device ('NVIDIA GeForce RTX 3090') that has Tensor Cores. To properly utilize them, you should set `torch.set_float32_matmul_precision('medium' | 'high')` which will trade-off precision for performance. For more details, read https://pytorch.org/docs/stable/generated/torch.set_float32_matmul_precision.html#torch.set_float32_matmul_precision
wandb: Tracking run with wandb version 0.21.3
wandb: Run data is saved locally in ./wandb/run-20250912_103607-z8wq8v45
wandb: Run `wandb offline` to turn off syncing.
wandb: Syncing run resnet50-adamw--4lr2[hold_out]
wandb: ★ View project at https://wandb.ai/marscoin8-none/cv_competition
wandb: 🚀 View run at https://wandb.ai/marscoin8-none/cv_competition/runs/z8wq8v45
LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES: [0]
```

	Name	Type	Params	Mode
0	accuracy	MulticlassAccuracy	0	train
1	f1_macro	MulticlassF1Score	0	train
2	criterion	CrossEntropyLoss	0	train
3	model	ResNet	23.5 M	train
4	softmax	Softmax	0	train

```
-----
23.5 M Trainable params
0 Non-trainable params
23.5 M Total params
94.171 Total estimated model params size (MB)
221 Modules in train mode
0 Modules in eval mode
Sanity Checking: | 0/? [00:00<?, ?it/s]/d
ata/ephemeral/home/.pyenv/versions/cv_competition/lib/python3.12/site-packages/pytorch_lightning/trainer/connectors/data_connector.py:433:
The 'val_dataloader' does not have many workers which may be a bottleneck. Consider increasing the value of the `num_workers` argument` t
o `num_workers=47` in the `DataLoader` to improve performance.
/data/ephemeral/home/.pyenv/versions/cv_competition/lib/python3.12/site-packages/pytorch_lightning/trainer/connectors/data_connector.py:43
3: The 'train_dataloader' does not have many workers which may be a bottleneck. Consider increasing the value of the `num_workers` argumen
t` to `num_workers=47` in the `DataLoader` to improve performance.
/data/ephemeral/home/.pyenv/versions/cv_competition/lib/python3.12/site-packages/pytorch_lightning/loops/fit_loop.py:310: The number of tr
aining batches (40) is smaller than the logging interval Trainer(log_every_n_steps=50). Set a lower value for log_every_n_steps if you wan
t to see logs for the training epoch.
Epoch 3: 10%|█ | 4/40 [00:01<00:14, 2.55it/s, v_num=8v45, valid_loss=2.200, valid_f1=0.372, train_loss=2.450]
```

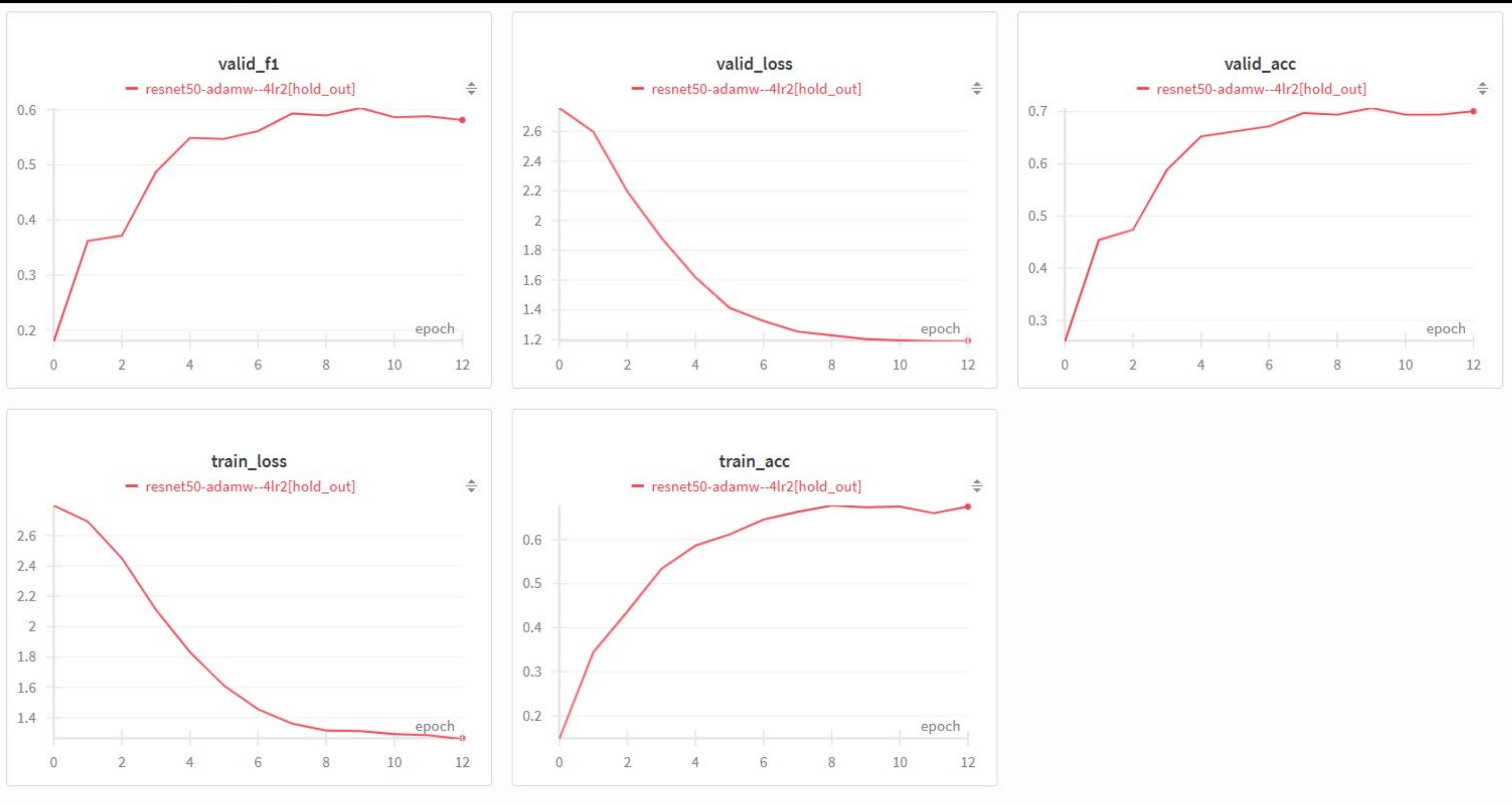

환경구 : Compute

```
You are using a CUDA device ('NVIDIA GeForce RTX 3090') that has Tensor Cores. To properly utilize them, you should set `torch.set_float32_matmul_precision('medium' | 'high')` which will trade-off precision for performance. For more details, read https://pytorch.org/docs/stable/generated/torch.set_float32_matmul_precision.html#torch.set_float32_matmul_precision
wandb: Tracking run with wandb version 0.21.3
wandb: Run data is saved locally in ./wandb/run-20250912_103607-z8wq8v45
wandb: Run `wandb offline` to turn off syncing.
wandb: Syncing run resnet50-adamw--4lr2[hold_out]
wandb: ★ View project at https://wandb.ai/marscoin8-none/cv_competition
wandb: 🚀 View run at https://wandb.ai/marscoin8-none/cv_competition/runs/z8wq8v45
LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES: [0]
```

	Name	Type	Params	Mode
0	accuracy	MulticlassAccuracy	0	train
1	f1_macro	MulticlassF1Score	0	train
2	criterion	CrossEntropyLoss	0	train
3	model	ResNet	23.5 M	train
4	softmax	Softmax	0	train

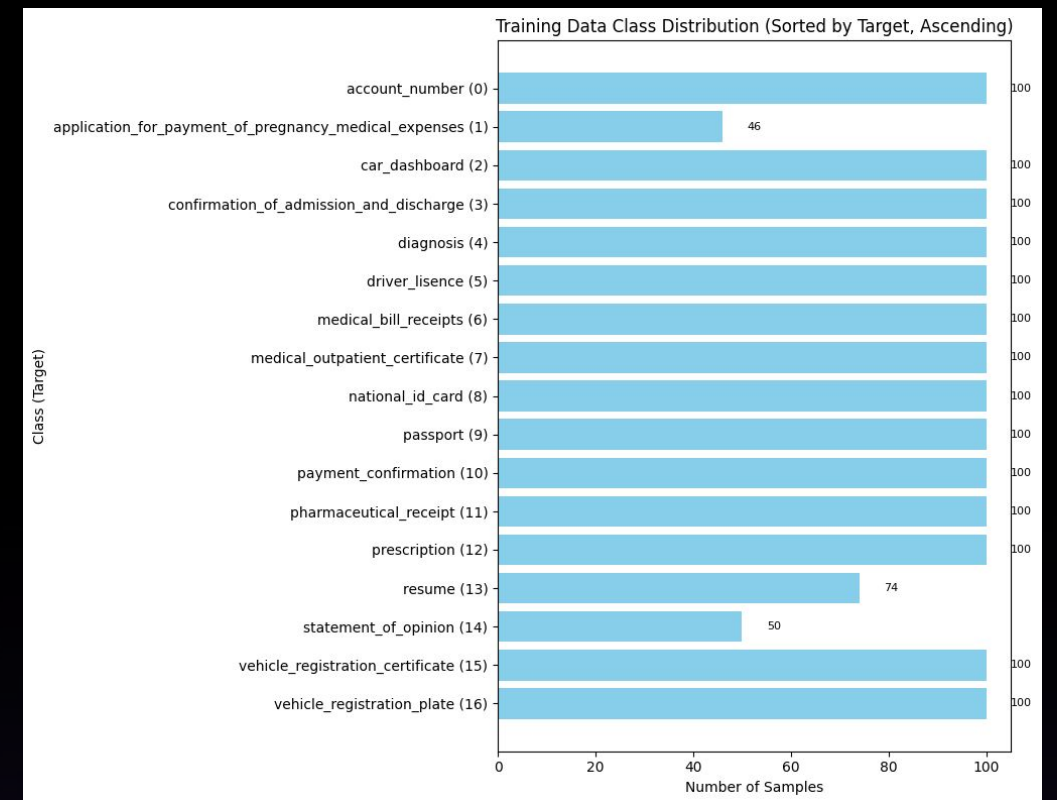
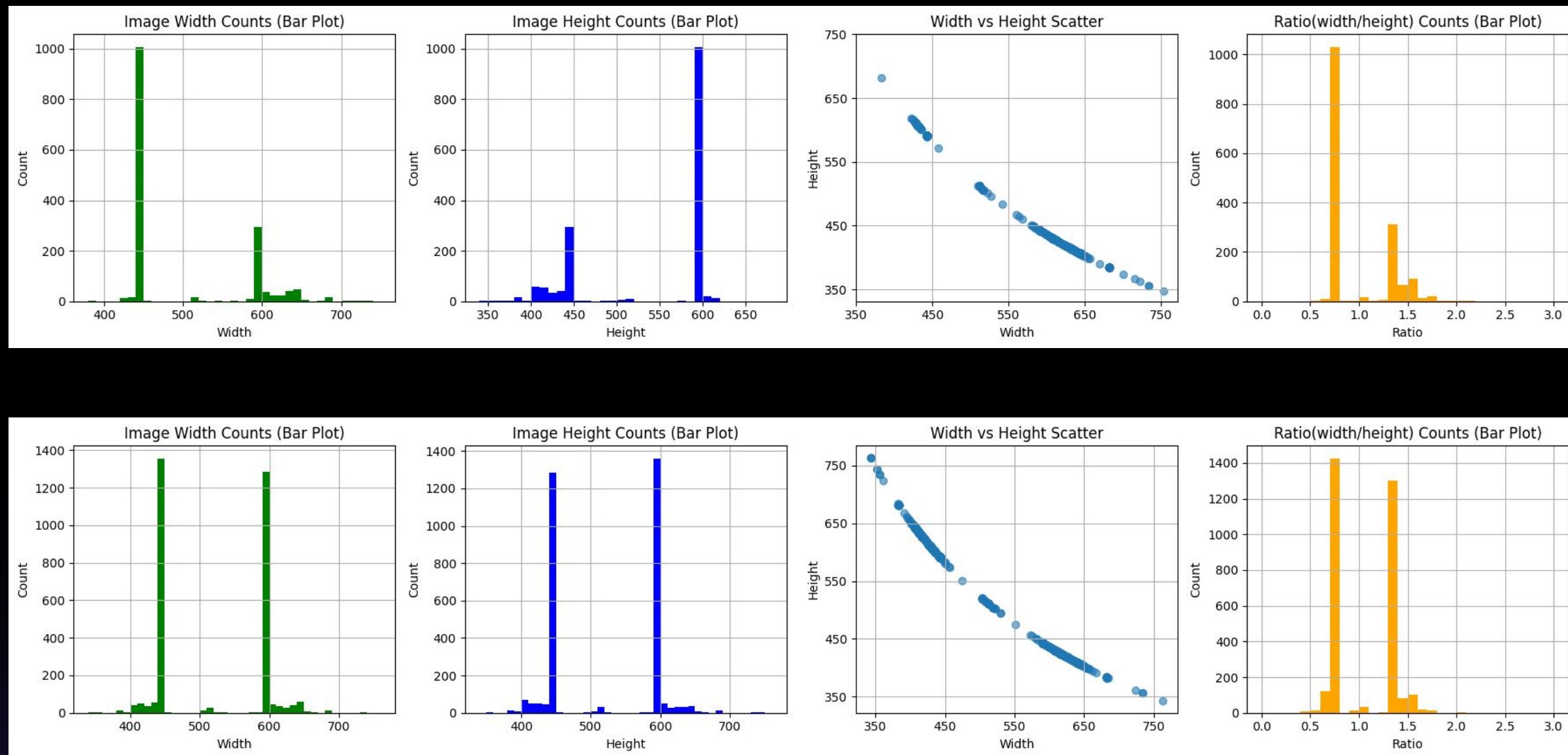
```
-----
23.5 M    Trainable params
0         Non-trainable params
23.5 M    Total params
94.171    Total estimated model params size (MB)
221       Modules in train mode
0         Modules in eval mode
Sanity Checking: | 0/? [00:00<?, ?it/s]/data/ephemeral/home/.pyenv/versions/cv_competition/lib/python3.12/site-packages/pytorch_lightning/trainer/connectors/data_connector.py:433:
  The 'val_dataloader' does not have many workers which may be a bottleneck. Consider increasing the value of the `num_workers` argument` to `num_workers=47` in the `DataLoader` to improve performance.
/data/ephemeral/home/.pyenv/versions/cv_competition/lib/python3.12/site-packages/pytorch_lightning/trainer/connectors/data_connector.py:433:
  The 'train_dataloader' does not have many workers which may be a bottleneck. Consider increasing the value of the `num_workers` argument` to `num_workers=47` in the `DataLoader` to improve performance.
/data/ephemeral/home/.pyenv/versions/cv_competition/lib/python3.12/site-packages/pytorch_lightning/loops/fit_loop.py:310: The number of training batches (40) is smaller than the logging interval Trainer(log_every_n_steps=50). Set a lower value for log_every_n_steps if you want to see logs for the training epoch.
Epoch 3: 10%|██████| 4/40 [00:01<00:14, 2.55it/s, v_num=8v45, valid_loss=2.200, valid_f1=0.372, train_loss=2.450]
```


: Comp



EDA

: Computer Vision [대회] Image Classification



- 이미지의 width, height, ratio 확인, 학습에 사용할 이미지 사이즈 결정

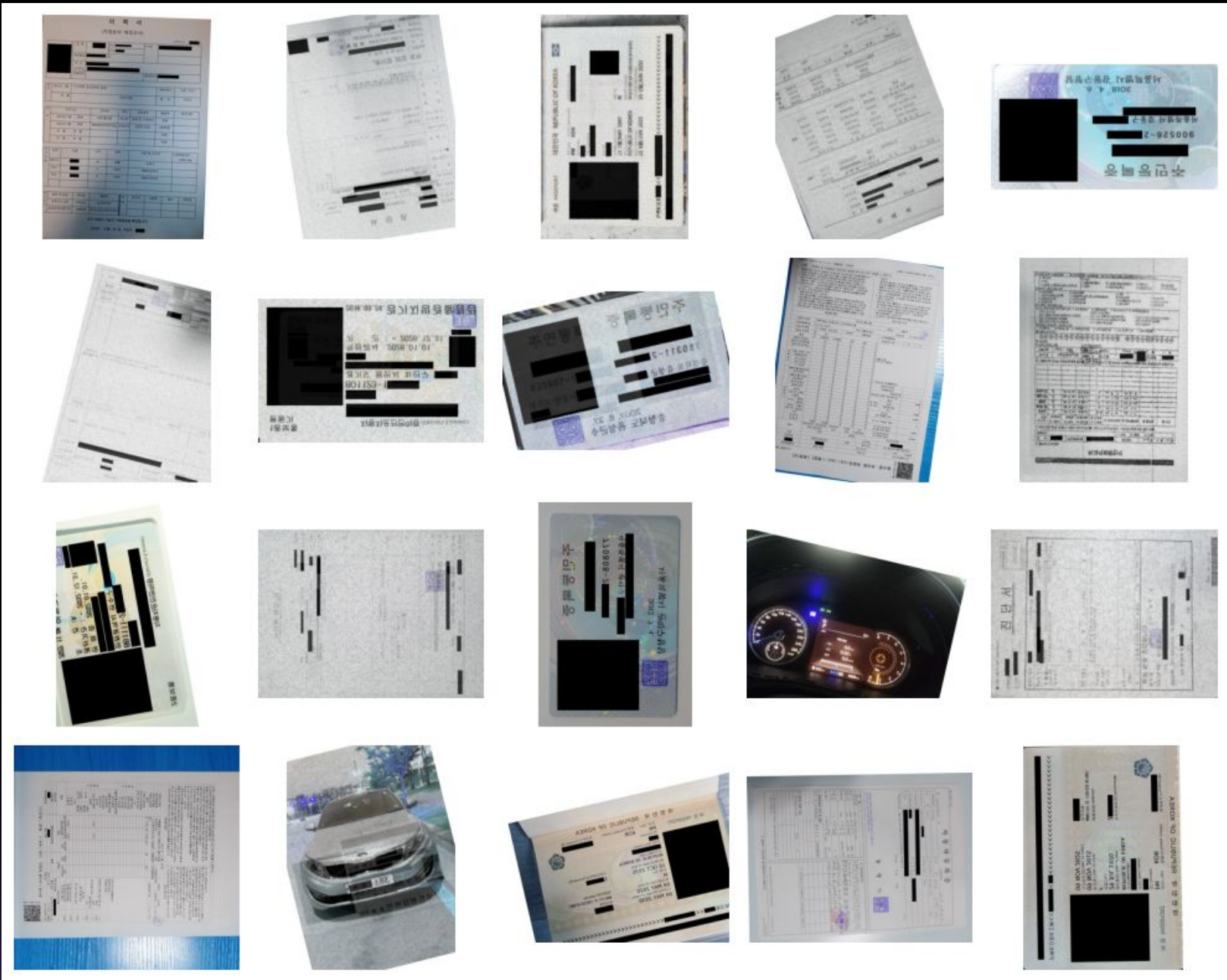
- 학습 데이터 각 클래스 별 이미지 분포 확인

EDA

: Computer Vision [대회] Image Classification



- train



- test

Augmentation

: Computer Vision [대회] Image Classification

image shape : (591, 443, 3)

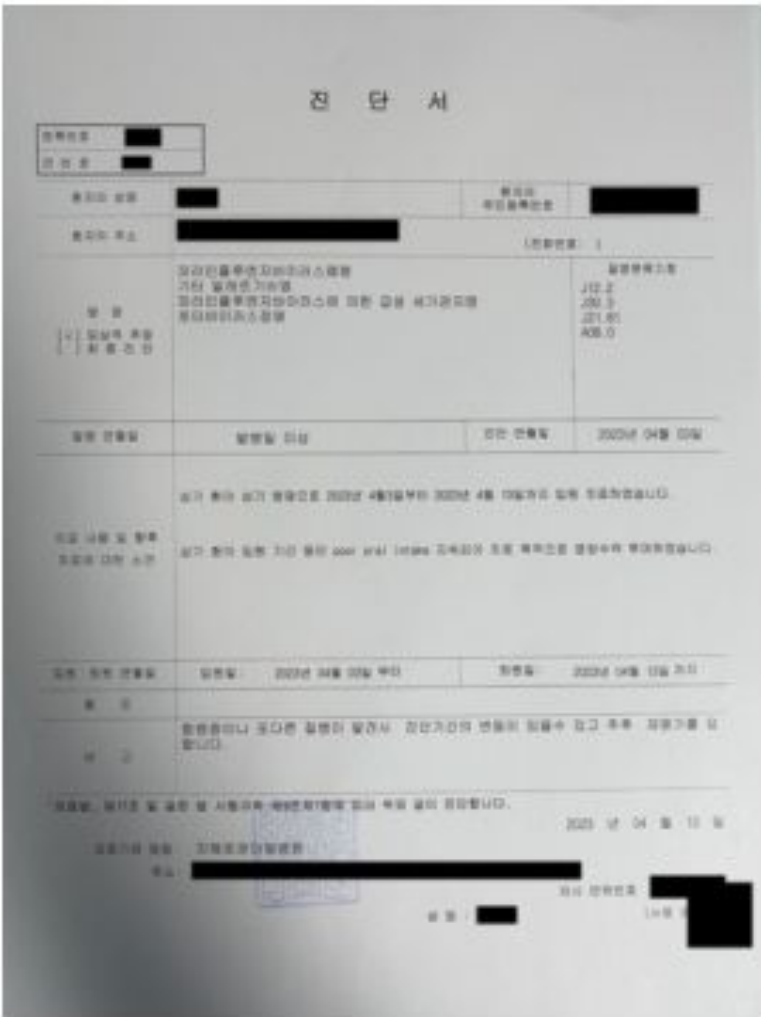


image shape : (150, 150, 3)



image shape : (150, 112, 3)

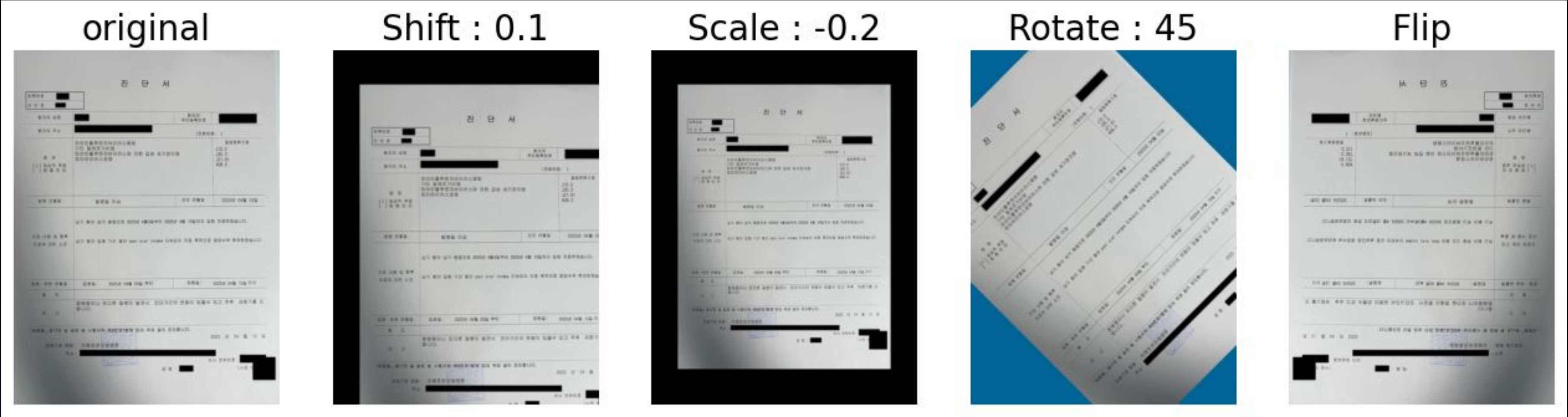


image shape : (150, 150, 3)



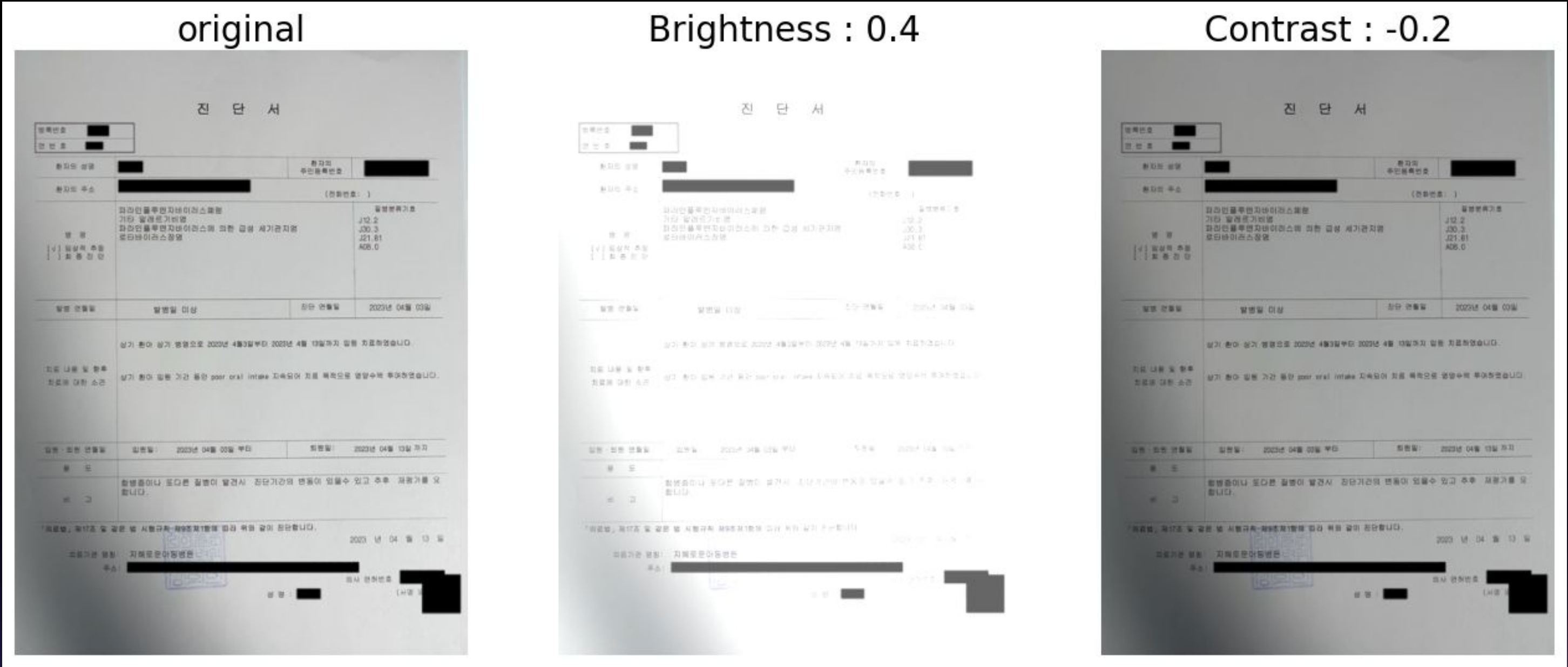
Augmentation

: Computer Vision [대회] Image Classification



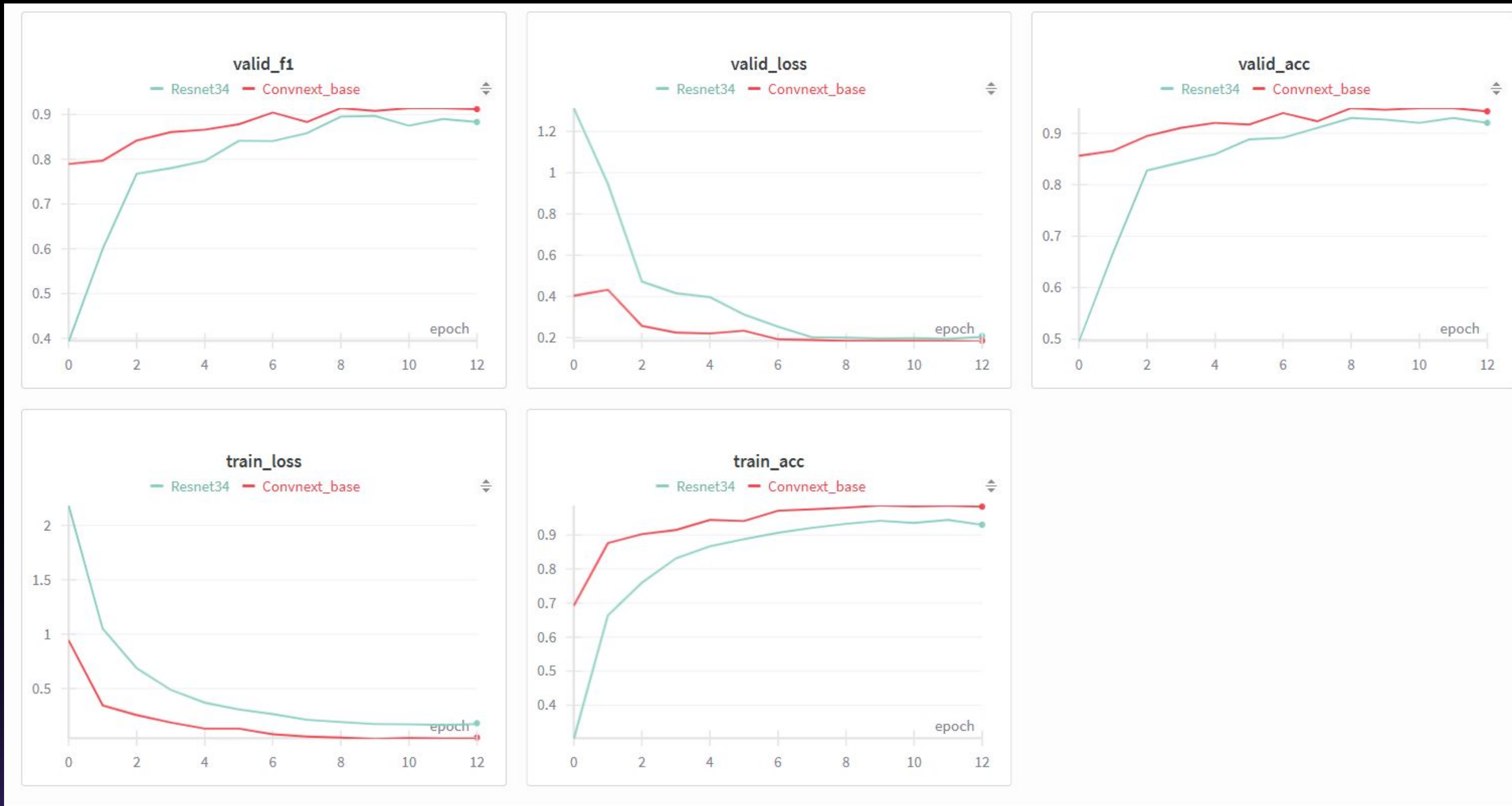
Augmentation

: Computer Vision [대회] Image Classification



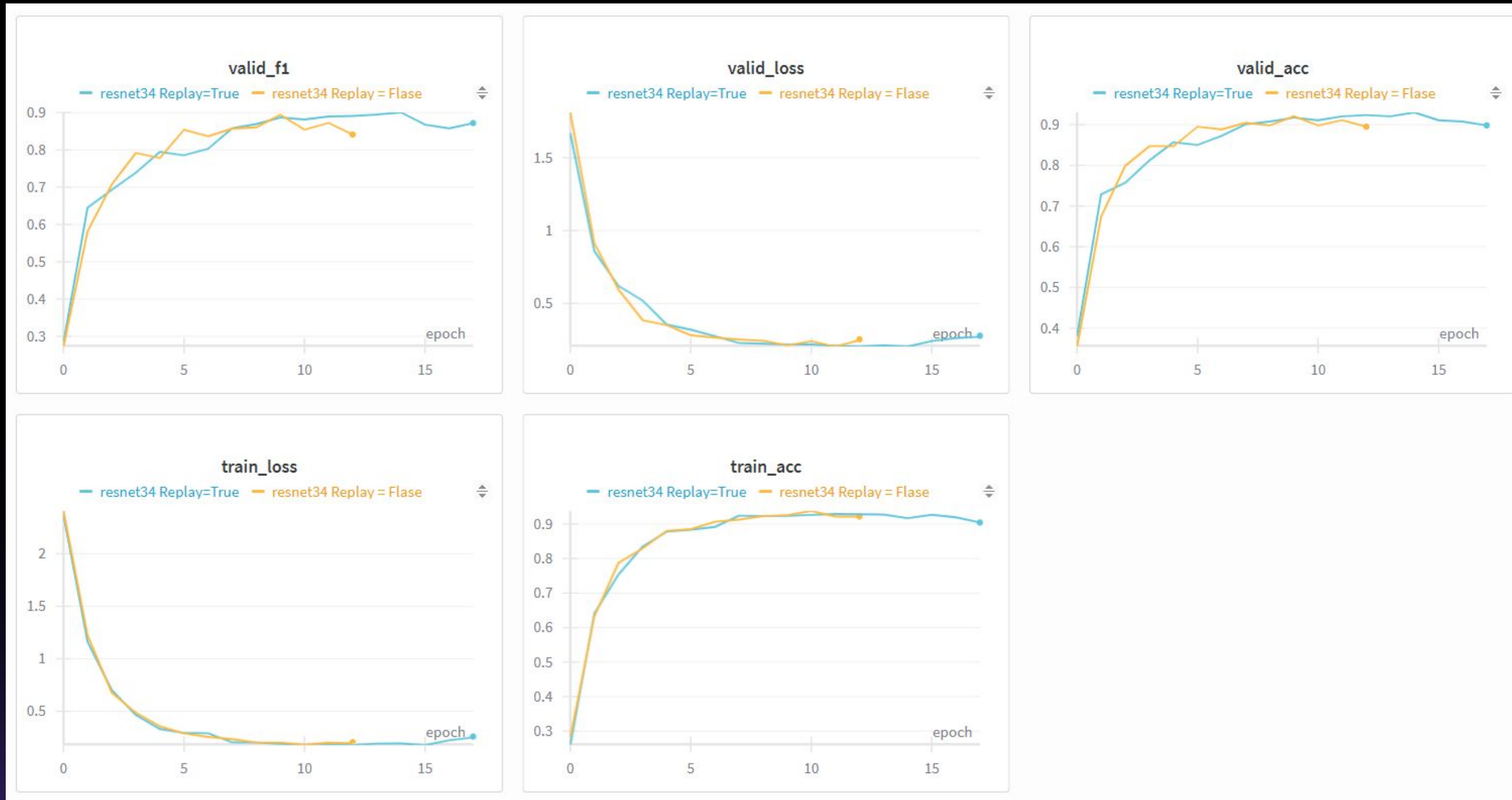
경진대회 수행 결과

: Computer Vision [대회] Image Classification



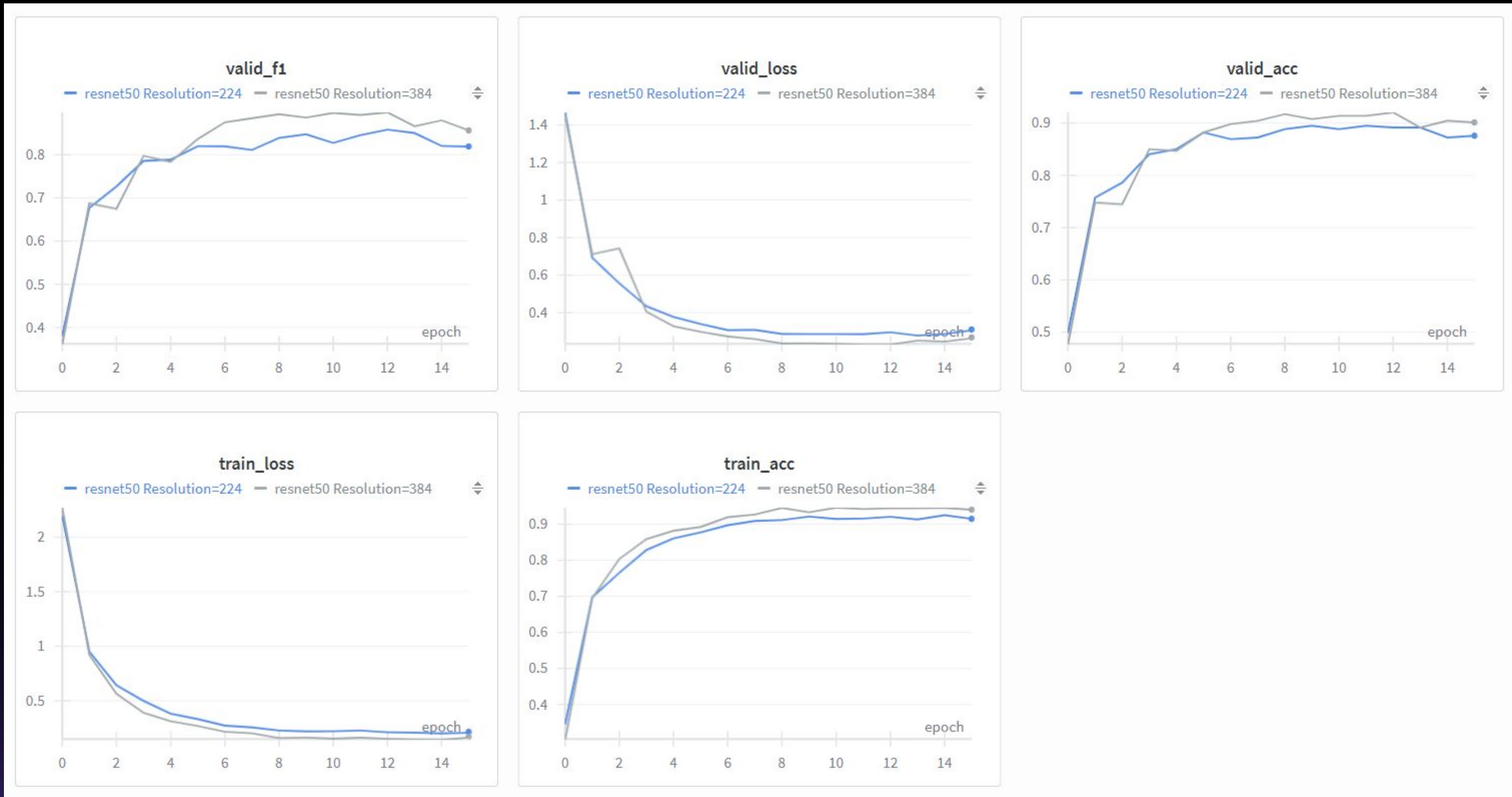
경진대회 수행 결과

: Computer Vision [대회] Image Classification



경진대회 수행 결과

: Computer Vision [대회] Image Classification



03

최종 결과

문제 발생 배경
해결 방법 및 결과
최종 모형 구성 설명

경진대회 인사이트 공유

: Computer Vision [대회] Image Classification

01. 문제 발생 배경

1. *Test set*에 여러 *augmentation*이 적용.
2. *Augmentation* 적용 방식을 온라인으로 할 지, 오프라인으로 할 지에대한 고민이 존재.
3. *Validation set*에 *online augmentation*을 적용할 경우, 매번 달라지는 변형을 어떻게 처리할 지에 대한 고민이 존재.
4. 학습을 진행하면서, 특정 클래스를 잘 구분하지 못하는 현상이 발생.

경진대회 인사이트 공유

: Computer Vision [대회] Image Classification

02. 해결방법

1. 강한 증강을 **train set**에 적용하여 학습시키고, 여러 증강이 적용된 **test set**에 맞게 **validation set** 또한 여러 증강을 적용
2. 온라인 증강과 오프라인 증강의 장단점을 팀원들과 논의한 결과, 온라인 증강을 적용.
3. **ReplayCompose**를 적용하여 동일한 증강이 적용되도록 하는 것과 **epoch** 수를 늘려 여러 증강을 많이 학습시켜 **robust**하게 만드는 것을 고려.
4. 더 큰 **backbone**과 고해상도 이미지를 쓰면 표현력이 좋아져 클래스 구분을 명확히 할 수 있다고 생각.

03. 결과

해결방법들을 도입하면서 모델의 성능이 상당히 개선되었고, 많은 모델의 **F1-SCORE**가 **0.9** 이상을 달성.

Train은 온라인 증강, **Validation**은 **ReplayCompose**, 필요 시 **epoch** 확장으로 강건성을 확보하는 것이 합리적이라는 인사이트 얻음.

경진대회 인사이트 공유

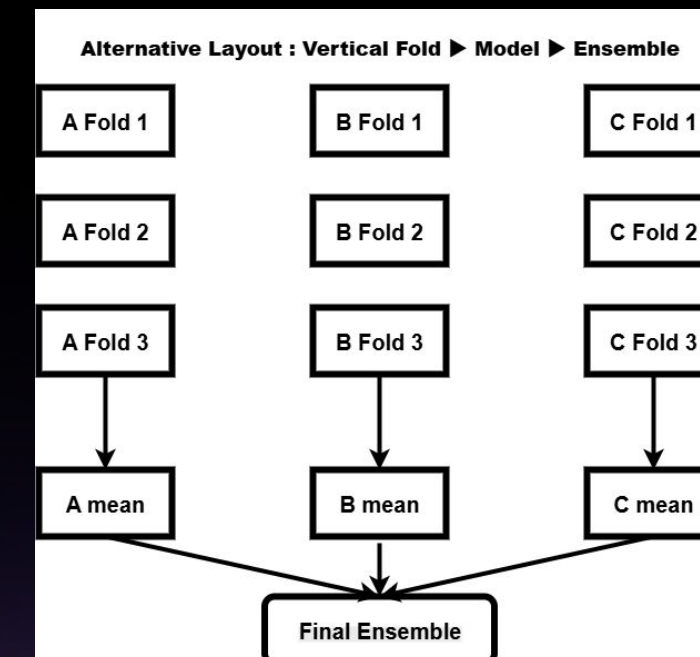
: Computer Vision [대회] Image Classification

04. 모델 구성

1. `convnextv2_large.fcmae_ft_in22k_in1k_384`,
`efficientnetv2_rw_m.agc_in1k`,
`deit3_large_path16_384.fb_in22k_ft_in1k`,
`swinv2_base_window12to24_192to384.ms_in22kft_in1k`,
`regnety_320.swag_ft_in1k`
5개의 모델을 *StratifiedKFold*를 이용해 학습.

2. Fold마다 *softmax*로 로짓 값을 변환후, 각 *Fold*의 확률을 평균하여 모델의 평균을 계산.

3. 각 모델의 평균을 모델별 가중치와 곱해 가중 평균을 구하고,
*argmax*를 취해 최종 평균을 산출.

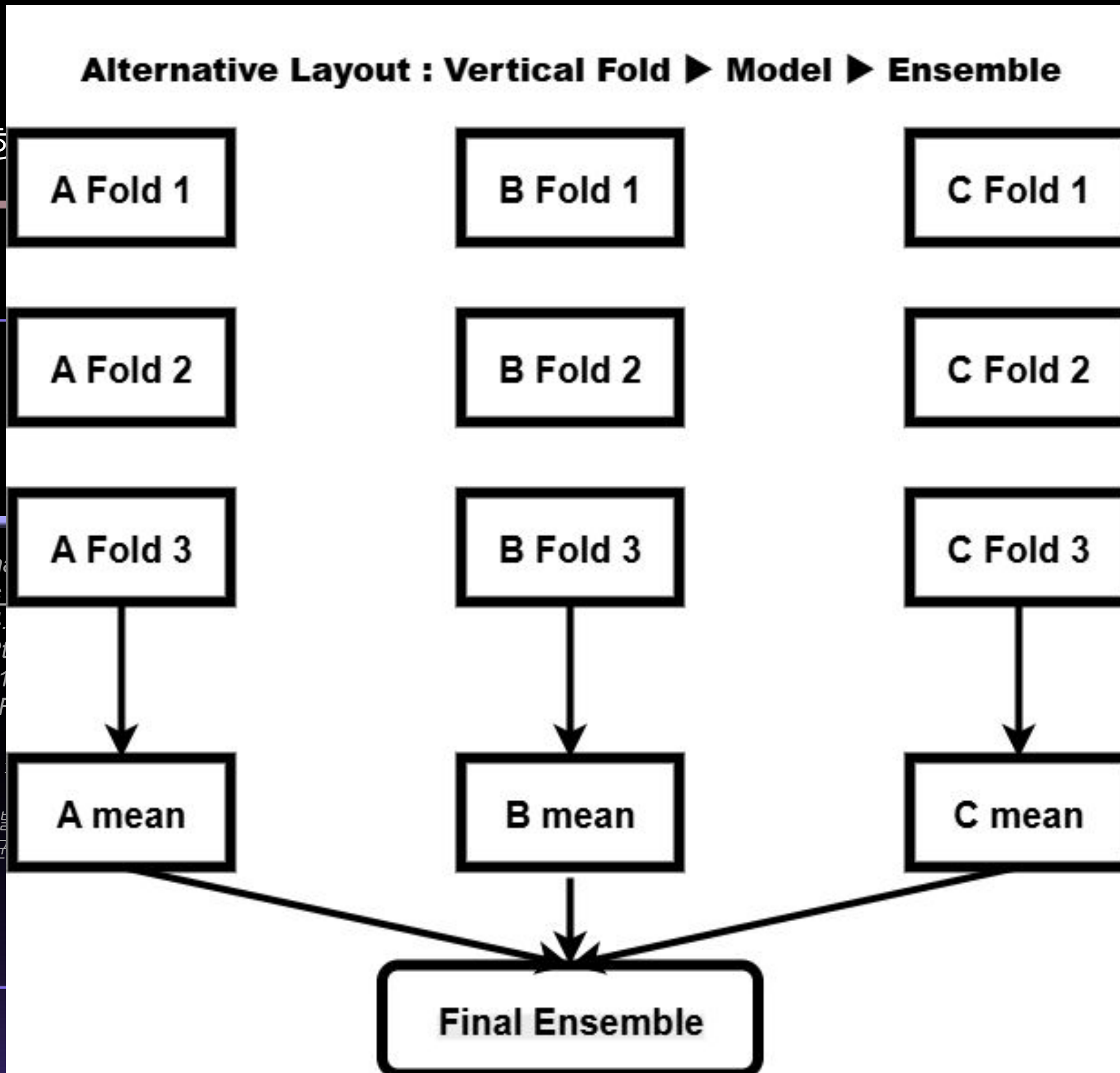


04. 모델 구성

1. convnextv2_large.fcma
efficientnetv2_rw_m.agc
deit3_large_path16_384.
swinv2_base_window12t
regnety_320.swag_ft_in1
5개의 모델을 StratifiedKF

2. Fold마다 softmax로 로

3. 각 모델의 평균을 모델별
argmax를 취해 최종 평균



04

회고

우리 팀의 목표 달성도
느낀점 및 향후 계획

경진대회 회고

: Computer Vision [대회] Image Classification

Point 1

우리 팀의 처음 목표에서 어디까지 도달했는가

* 처음에 목표했던 CV model에 대한 전반적인 이해수준이 높아졌고, 성과로서 최종순위 4위 달성

Point 2

우리 팀이 잘했던 점

- * 문제 해결을 위한 밀도있는 유기적 회의 진행
- * 적극적으로 팀에 기여를 하기 위한 방법 모색
- * 문제 공유와 해결방안 모색

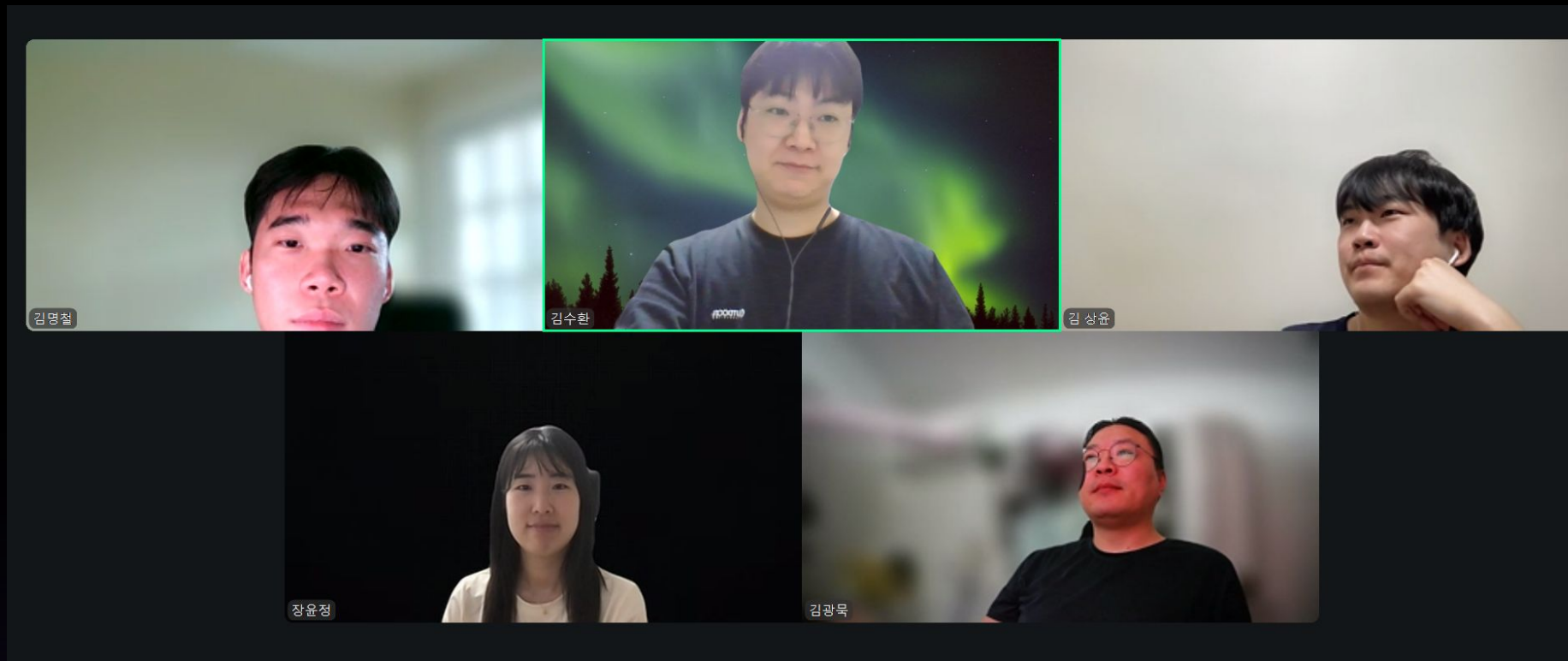
Point 3

협업하면서 아쉬웠던 점

- * 향후 계획 : 작업 환경 구성을 좀 더 빠르게 했으면...
공동작업 부분에서 좀 아쉬웠다

경진대회 진행 소감

: Computer Vision [대회] Image Classification



- * 김수환** 실험환경 구성이 늦게 이루어져 협업이 잘 이루어지지 않았던게 많이 아쉬웠다. `code` 작성에 어려움을 겪는 인원들도 쉽게 실험을 할 수 있도록 빠르게 환경이 구성이 되었다면 매우 많은 실험과 피드백이 이루어지지 않았을까라는 아쉬움이 남는다.
- * 김명철** 협업의 중요성을 깨달았다. 여러 실험을 많이 했지만 그것이 팀원분들에게 제대로 공유되지 못했었다. 다음 경진대회는 Hydra와 WANDB를 적극적으로 이용해서, 팀원 모두와 같이 대회를 즐겼으면 좋겠다.
- * 김상운** 처음 접하는 딥러닝이기에 어렵기는 했지만 두 번째 경진 대회라 처음 진행할 때 보다는 여러 요소들의 변화에 따른 점수 증가를 재밌게 즐겼던 것 같습니다. 다음 경진대회는 좀 더 좋은 환경 구축을 통해 팀원들과의 더 많은 소통을 통해 더 좋은 결과를 내고 싶습니다.
- * 김광묵** 이번 경진대회는 포기하고 싶은 지점이었던 것은 명백하다. '하이드라'라는 새로운 개념 덕분에 회의중에 나온 방향성과 개선방안에 도움이 되지는 못했지만 메모 후 혼자 적용, 테스트 해보는 수준이었다. 흥미를 잃지 않게 편안하게 진행해준 팀원들에게 감사의 마음을 전한다.
- * 장윤정** 여러가지 가설 수립 및 실험을 진행해보지는 못했지만, 파이프라인을 구축해볼 수 있어서 좋았습니다. 다음 경진대회 때는 기본 환경을 빠르게 구축하고 여러 실험을 해보고 싶습니다.

Life-Changing Education

감사합니다.
