

Custom CNN과 사전학습 모델의 미세조정을 통한 식물 질병 분류 성능 비교

최혜진¹, 홍헬렌²

¹서울여자대학교 소프트웨어융합학과 학부생

²서울여자대학교 소프트웨어융합학과 교수

hyejin1379@naver.com, hlhong@swu.ac.kr

Comparing a Custom CNN against Fine-tuned Pre-trained Models for Plant Disease Classification

Hyejin Choi¹, Helen Hong²

¹Dept. of Software Convergence, Seoul-Women's University

²Dept. of Software Convergence, Seoul-Women's University

요 약

본 연구는 New Plant Diseases Dataset 기반 38개 클래스의 잎 이미지로 구성된 데이터셋(학습 70,295장, 검증 17,572장)을 사용해 커스텀 CNN과 사전학습 VGG16/ResNet50/EfficientNet-B0의 미세조정(fine-tuning)을 공정 비교했다. Accuracy, Macro-Precision, Macro-Recall, Macro-F1을 성능평가표를 통해 비교 제시한다. 전반적으로 사전학습+미세조정 모델이 기준선 대비 성능 우위를 보였으며, EfficientNet-B0는 경량 대비 최고 성능으로 성능-효율 균형이 가장 우수했다.

1. 서론

농업 현장에서 병해의 조기·정확 진단은 수확량과 약제 사용 최적화에 핵심이다. 딥러닝, 특히 CNN은 병변의 시각적 패턴을 자동 학습해 높은 분류 성능을 보여 왔으나, 유사 증상 간 시각적 근접성, 데이터 불균형, 실환경 변동성은 여전히 과제다. 본 연구의 목적은 커스텀 CNN과 사전학습 백본을 동일 파이프라인에서 비교하고, 혼동행렬 기반 취약 클래스를 진단해 간결한 개선 로드맵을 제안하는 데 있다.

2. 제안 방법

2.1 데이터셋

데이터셋은 Kaggle의 New Plant Diseases Dataset을 기반으로 하며, 38개 클래스로 분류된다[1]. 학습데이터는 70,295장, 검증 데이터는 17,572장이다. 입력크기는 224x224로 정하고, RGB로 전처리한다. 이미지를 반전, 회전, 밝기·대비 변화를 통해 데이터 증강을 적용한다.

2.2 CustomCNN 구조

Custom CNN은 그림 1과 같이 5개의 합성곱 블록(필터

수:32→64→128→256→512)으로 구성되며, 각 블록 뒤에 Dropout(p=0.25)을 적용한다. 분류기는 Dense(1500)-Softmax(38)로 설계한다. 분류기 입력 직전에는 추가로 Dropout(p=0.4)을 적용했으며, 전체 매개변수의 수는 약 7.84 million이다.

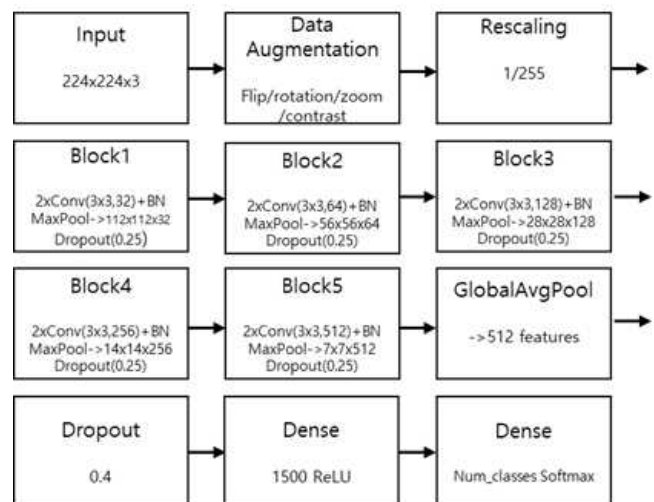


그림 1. Custom CNN 구조

비교 실험을 위해 VGG16, ResNet50, EfficientNet-B0을 사용하였으며, 모두 ImageNet 사전학습 가중치로 초기화하고, 최상위 분류기는 제외하였다[2-4]. 사전학습 백본 출력 상단에 Global Average Pooling을 적용한다. 학습은 2단계 전이학습 절차로 수행하였다. 1단계에서는 백본을 고정(freeze)한 채 분류 헤드만 학습하였고, 2단계에서는 상단 합성곱 블록을 선택적으로 해제(unfreeze)하여 낮은 학습률로 미세조정(fine-tuning)하였다. 평가는 Top-1 정확도(Accuracy), Macro-Precision, Macro-Recall, Macro-F1(클래스별 F1의 산술평균) 지표로 보고한다.

3. 실험 및 결과분석

표 1은 동일 파이프라인에서의 성능 비교를 요약한다. 사전학습+미세조정 모델들이 기준선(Custom CNN) 대비 Accuracy 및 Macro-F1에서 전반적으로 우수했으며, 그중 EfficientNetB0-FT가 성능-효율 균형에서 가장 두드러졌다. ResNet50-FT는 안정적인 수렴을 보였고, VGG16-FT는 정확도는 높으나 연산·파라미터 비용이 상대적으로 컸다. Custom CNN은 절대 성능은 낮지만 경량·고속 추론이라는 실무적 이점이 있다.

훈련 곡선 특성을 보면, 기준선 CNN은 초기 급상승 후 검증 정확도가 95% 내외로 수렴했고, 중반부에 검증 손실의 일시적 재상승으로 경미한 과적합 신호가 있었다. 이후 학습률 감소·정규화·증강에 의해 큰 진동 없이 안정화되었다. 사전학습 모델들은 헤드 학습만으로도 높은 기준선을 형성했으며, 상단 블록을 부분 해제한 미세조정 구간에서 지표가 추가 개선되었다.

클래스 수준 오류 분석(혼동행렬)에 따르면, 전반적으로 정확한 분류가 우세하나 증상 유사 클래스에서 오분류가 잔존했다. 사과(Apple): Apple_scab ↔ Black_rot, Cedar_apple_rust ↔ Apple_healthy; 토마토: Leaf_Mold ↔ Septoria_leaf_spot; 피망: healthy ↔ Bacterial_spot 등이 대표적이다. 이는 시각적 유사성·환경 변동·데이터 분포 등이 복합 작용한 결과로, 증강 강화·Focal Loss·경량 어텐션·신뢰도 임계 기반 재분류·양상블의 단계적 적용을 통해 개선할 수 있다. 특히 Macro-Recall/Precision 향상에 직접적 효과가 기대된다.

표1. 모델별 성능 평가 결과

Model	Accuracy(%)	Macro-F1(%)	Macro-Precision(%)	Macro-Recall(%)
VGG16-FT	98.60	98.50	98.30	98.40
ResNet50-FT	98.90	98.80	98.60	98.70
EfficientNetB0-FT	99.20	99.10	99.00	99.10
Custom CNN	95.33	95.30	95.10	95.00

4. 개선안

개선 방향은 다음과 같다. 먼저 회전·스케일 변환·컷아웃·컬러-지터의 강도를 상향하고, 필요 시 클래스 가중 증강을 도입한다. 이어서 Focal Loss($y > 1$)를 적용하여 어려운 샘플의 기여도를 확대한다. 또한 백본 상단에 SE, CBAM, Spatial Attention과 같은 경량 블록을 추가한다. 마지막으로 신뢰도 임계값 이하 샘플에 대해 재분류를 수행하고, ResNet·EfficientNet의 보완적 양상블로 예측 분산을 감소시킨다.

5. 결론

본 연구는 식물 질병 진단을 위한 딥러닝 모델의 성능을 체계적으로 비교·분석하고, 강건성과 신뢰성을 향상시키기 위한 개선 방법을 제시하였다. 실험 결과, 사전학습 모델을 미세조정하는 전이학습 방식이 기준선 대비 일관된 성능 우위를 보였으며, 특히 EfficientNet-B0은 가장 높은 정확도와 연산 효율성을 동시에 달성하여 실무 적용성이 높은 경량 모델로 확인되었다. 또한 혼동행렬 분석을 통해 오분류가 빈번한 취약 클래스를 식별하였다.

향후 연구에서는 다양한 배경·조명·촬영 조건을 포함하는 실제 농업 환경 데이터에 대한 외부 검증을 수행하여 모델의 일반화 성능을 평가하고자 한다.

감사의 글

본 연구는 서울여자대학교 SW중심대학추진사업단의 지원의 연구결과로 수행되었음(2025).

References

- [1] “New Plant Diseases Dataset”, Kaggle, 2025년 9월 20일 검색, <https://www.kaggle.com/datasets/vipooooool/new-plant-diseases-dataset>
- [2] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” in Proc. ICLR, 2015. (arXiv:1409.1556)
- [3] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in Proc. CVPR, 2016, pp. 770 - 778. doi:10.1109/CVPR.2016.90
- [4] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” in Proc. ICML, PMLR vol. 97, 2019, pp. 6105 - 6114.