

객체 탐지 기법과 기계학습 라이브러리를 활용한 단감 등급 선별 알고리즘

노승희[†], 강은영^{**}, 박동규^{***}, 강영민^{****}

A Sweet Persimmon Grading Algorithm using Object Detection Techniques and Machine Learning Libraries

SeungHee Roh[†], EunYoung Kang^{**}, DongGyu Park^{***}, Young-Min Kang^{****}

ABSTRACT

A study on agricultural automation became more important. In Korea, sweet persimmon farmers spend a lot of time and effort on classifying profitable persimmons. In this paper, we propose and implement an efficient grading algorithm for persimmons before shipment. We gathered more than 1,750 images of persimmons, and the images were graded and labeled for classifications purpose. Our main algorithm is based on EfficientDet object detection model but we implemented more exquisite method for better classification performance. In order to improve the precision of classification, we adopted a machine learning algorithm, which was proposed by PyCaret machine learning workflow generation library. Finally we acquired an improved classification model with the accuracy score of 81%.

Key words: Persimmon Classification, Convolutional Neural Network Classifier, EfficientDet Object Detection Model, Machine Learning Library

1. 서 론

우리나라의 인구구조는 세계에서 유례를 찾아볼 수 없는 빠른 속도로 고령화되고 있으며, 이러한 인구 고령화는 여러 산업에 영향을 미치고 있다. 특히 노동 가능 인구의 급격한 감소 문제는 타 산업 분야 보다 농업 분야에서 큰 영향을 미치고 있다. 최근 농업 분야의 노동력 부족이라는 문제점을 해결하기 위하여, 이 분야에 데이터 과학 기술을 접목하고 자동화 기계를 도입하고자 하는 연구가 활발하게 이루어

지고 있다. Valeria 등은 딥러닝 기술을 활용한 이미지 분류기법을 사용하여 나뭇잎과 줄기의 이미지만으로 질병의 종류를 파악할 수 있는 모델을 제안하였다[1]. 현재까지의 연구는 토마토와 같이 작물의 표면에 대해 합성곱 신경망과 같은 딥러닝 기법을 적용하는 시도가 일반적이다[2-3]. 최근 실시간으로 과일의 이미지를 수집하여 과일의 수확시기를 예측하기도 하며, 수집한 데이터가 적으면 적대적 생성 모델(GAN)을 사용하여 많은 데이터를 만든 다음 이를 이용하여 분류기의 성능을 높이는 방법도 연구되었

※ Corresponding Author : Young-Min Kang, Address: (48520) Sinseon-ro 428, Nam-gu, Busan, Korea, TEL: +82-51-629-1253, FAX: +82-51-629-1249, E-mail: ymkang@tu.ac.kr

Receipt date : Apr. 27, 2022, Revision date : Jun. 1, 2022
Approval date : Jun. 7, 2022

[†] Dept. of Information and Communications Engineering, Changwon National University (E-mail: sdf8036@naver.com)

^{**} Dept. of System Engineering, Changwon National University (E-mail: kibudy@naver.com)

^{***} Dept. of Information and Communications Engineering, Changwon National University (E-mail: dongu-pak@gmail.com)

^{****} Dept. of Game Engineering, Tongmyong University

※ This work was supported by Tongmyong University Research Grant(2020).

다[4-7]. 이러한 연구들은 단감이나 토마토처럼 매끈한 표면을 가진 과육을 대상으로 진행되는 경우가 일반적이다. 그 밖에도 Kanae 등은 단감 표면의 이미지를 통해 씨앗의 여부 및 개수를 파악하는 딥러닝 모델을 제안하였으며[8], Afonso 등은 과수원의 토마토 인식과 개수를 확인하기 위해 세그멘테이션 모델인 MaskRCNN을 활용한 연구를 수행하였다[9]. 몇몇 농업 분야에서는 첨단 드론과 감시 카메라를 통한 데이터 수집 기법으로 농작물의 생육 정보를 실시간으로 모니터링할 수 있게 되었고, 토양의 양분과 산성도를 파악하여 작물의 수확 시기와 품질을 더욱 정확하게 예측할 수 있게 되었다[10]. 최근에는 농기구에 센서를 부착하고 다양한 자동화 시스템을 도입하는 방법으로 사람이 해야 하는 많은 작업을 기계가 대체하기에 이르렀다[11,12].

컴퓨터 비전 기술은 2010년도 이전까지 객체 탐지나 분류에서 큰 성능 향상을 보여주지 못하며 정체를 겪고 있었다. 그러나 2012년 이미지넷 경진대회에서 딥러닝 기술의 일종인 합성곱 신경망(Convolutional Neural Network, CNN) 구조의 AlexNet이 압도적으로 우승한 이후 인식을 개선의 정체를 돌파하면서 많은 연구와 성과를 거두고 있다. 이미지 처리 분야에서 딥러닝 기술을 활용한 객체 탐지 기술은 최근 여러 산업 분야에서 활발하게 이루어지고 있다. 일반적으로 객체 탐지를 하나의 객체에 적용하는 경우, Fig. 1과 같이 분류 문제와 분류와 영역지정 기술로 세분화할 수 있다. 만일 한 이미지에 여러 객체가 있는 경우 그림의 오른쪽과 같이 객체 탐지와 객체 분할로 문제를 세분화할 수 있다. 객체 탐지는 한 영상에 존재하는 개, 고양이 객체를 영역과 함께 찾는

기술이며, 객체 분할은 객체 탐지에서 더 나아가 개, 고양이 객체와 배경 객체의 영역까지 나누는 기술이 포함된다.

이러한 객체 탐지와 분류는 제조업 분야에서는 불량 제품을 찾는 데에 활용되고 있으며, 생체 인식을 통한 보안 기술, 실시간 비전 기술을 이용한 자율주행 자동차와 같은 분야에서도 활용되고 있으며, 농업 분야에서도 활용이 되고 있다. 농업 분야에서 객체 탐지 기술의 활용이 필요한 이유는 최근 증가하고 있는 비숙련 농업인구와 관련이 깊다. 예를 들어, 농가의 과수 농장에서 병충해가 발생할 경우를 가정해보자. 이 질병이 자주 발생하는 질병이라면 질병에 대처하는 약품 처방을 통해서 빠른 대처가 가능하다. 하지만 그렇지 못한 경우 전문가 자문을 통해 병충해의 종류를 파악하고 대처하게 되는데, 이때 전문가를 섭외하고 파악하기까지 비교적 긴 시간이 소요된다. 만일 이 과정과 단계를 축소하여 빠른 대처를 한다면 농가에서는 병충해의 확산으로 인한 피해를 줄일 수 있으며 이로 인한 생산성의 향상을 기대할 수 있다. 또한 농가에 비숙련 노동력이 투입되어 수확에 참여하는 경우, 양품과 불량품의 선별 교육에 많은 시간이 소요되며, 선별작업의 효율도 좋지 못한 경우가 많다.

노승희 등은 합성곱 신경망 기술을 사용하여 단감을 양품과 불량으로 분류하기 위한 자동화된 알고리즘을 제안하였다. 이 연구 기법은 단감의 앞면, 뒷면, 옆면을 먼저 분류한 후에 각각의 이미지에 기반하여 양품과 불량품으로 분류하는 혼합 2단계 분류 모델을 사용하였다[13]. 이 방법은 양품과 불량품을 구별하는 데에 매우 효과적이기는 하지만 불량 단감의

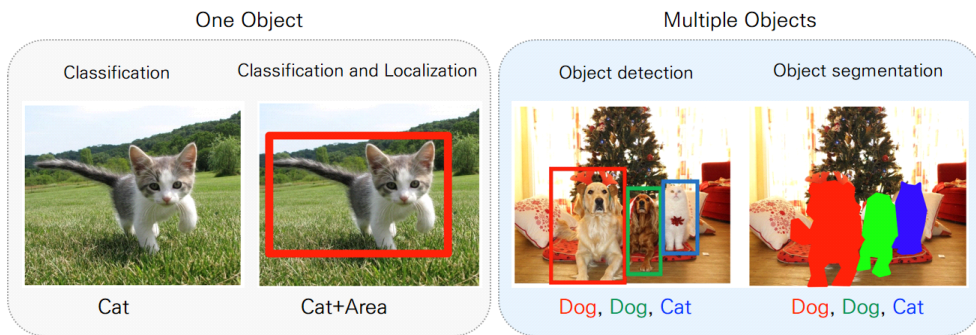


Fig. 1. Difference of object detection of single object and multiple objects in images: left two images show classification and classification + localization of single object in an image, and right two images show the object detection and segmentation for multiple objects in an image.

구체적인 질병을 파악하거나, 불량 요인에 대한 설명 능력이 부족한 한계가 있다. 본 연구에서는 이러한 기존 연구에서 더 나아가 객체 탐지 모델과 머신러닝 분류 모델을 활용하여 단감의 상태를 더욱 구체적으로 진단하는 방법을 제안하고 구현하였다. 단감의 품질을 결정하는 불량 요인은 여러 가지가 있으며 이 요인에 따라 외부의 생김새는 다르게 나타난다. 이러한 생김새를 객체 탐지 기법을 통해서 구분하고 이 데이터를 바탕으로 상품성을 측정하면, 농가에서는 구체적인 질병에 대한 대책을 수립할 수 있어서 생산성 향상을 이룰 수 있을 것이다.

단감 출하 단계의 농가에서는 최종 과정으로 상품의 가격을 책정하기 위한 선별작업을 진행한다. 하지만, 1) 짧은 수확시기에 과일의 상태를 하나씩 진단하기에는 많은 노동력과 시간이 소요된다. 또한 2) 수작업에 의한 진단 시에 발생하는 선별 품질에 대한 명확한 지표가 없어서 작업자의 주관적 판단에 따른 선별작업이 이루어진다는 것이다. 따라서 최종적으로 선별된 과일의 품질에 대한 일관성이 떨어진다는 문제가 있다.

1), 2)와 같은 문제점을 해결하기 위해 본 논문에서는 단감 표면 이미지를 여러 각도에서 촬영한 데이터를 기반으로 등급을 선별하는 알고리즘을 제안한다. 해당 알고리즘은 이미지 분석을 위한 객체 탐지 단계와 단감 등급을 선별하기 위한 머신러닝 분류 단계의 두 단계로 이루어진다.

데이터는 단감의 표면에 대한 정보를 모두 담기 위해 하나의 단감으로부터 5장의 이미지를 촬영하였다. 촬영한 단감의 표면은 윗면, 아랫면, 120도 각도의 옆면 사진 3장으로 구성되어 있다. 이렇게 촬영한 5장의 이미지는 단감의 특징을 대변하는 전체 면의 데이터로 볼 수 있으며, 촬영한 5장의 이미지는 객체 탐지 모델을 통해 각종 병충해와 스크래치 여부 및 불량 위치를 판단한다. 그리고 스크래치와 불량에 해당하는 객체 데이터를 수치화하여 테이블로 재정의 한다. 만들어진 테이블은 머신러닝 분류 모델의 입력값으로 들어가고 추론을 통해 최종적으로 단감의 품질 등급이 선별된다. 이렇게 2단계의 과정으로 단감의 품질 등급을 선별하는 알고리즘을 통해 단감의 등급 선별 자동화가 가능해짐은 물론이고, 선행 연구 [13]에서 어려웠던 세 가지 문제를 개선할 수 있었다. 1) 불량 요인별 불량 영역 개수와 면적 등의 정량적

정보 추출이 가능해졌다. 2) 불량요의 종류와 크기에 따른 가중치 부여가 가능해졌다. 3) 불량 객체 종류의 범위가 넓어졌다. 그리고, 이를 통해 선행 연구에 대비해 더욱 높은 성능을 갖게 되었고, 양품/불량 2단계 분류에서 B품을 추가한 3단계 분류가 가능해졌다.

2. 관련 연구

현재까지도 농업 분야에서는 수확한 과일을 분류하거나 과일의 병충해를 인식하는데 숙련된 인력의 수작업에만 의존하는 경우가 많다. 최근 딥러닝에 기반한 비전 기술의 비약적인 발전으로 이러한 문제를 개선하기 위해 머신러닝과 컴퓨터 비전을 활용한 연구가 활발히 진행되고 있다. 일반적으로 작물이 병충해에 노출되면 열매나 잎에서 반점이나 변색과 같은 증상이 나타난다. 이러한 병충해 발생 유무를 식별하거나 과실의 결점을 인식하기 위해 합성곱 신경망을 이용한 분류 자동화에 대한 연구가 이루어져왔다. Ferentions 등은 과수원에서 촬영된 이미지와 잎만 따로 수집하여 촬영한 이미지를 학습하였을 때를 비교하여 어느 데이터가 분류에 더 효율적인가에 대한 연구를 수행하였다[14]. 이 연구에서는 작물의 잎에 대한 이미지 데이터를 획득한 후, 합성곱 신경망 기법을 이용하여 이를 58개의 클래스로 분류하였고, VGGNet, GoogleNet, AlexNet 등 이미지 분류 모델 5개를 사용하여 성능이 가장 나은 2개의 모델을 선정하였다. 그 이후 검증용 데이터를 통한 예측도를 비교하여 식물 이미지 분류에 대한 효율성을 입증하였다.

농작물의 경우 공산품과는 달리 시간에 따라 그 품질이 달라지는 경우가 많다. 따라서 시기에 따른 농작물의 품질 특성을 평가하는 것 역시 매우 중요한 일이다. 특히, 온실 속에서 재배되는 농작물의 경우 적절한 수확시기를 파악하기 위해서는 시기별 농작물 품질 평가가 매우 중요하다. 온실 농작물 중에서도 토마토와 같은 후숙 과일은 수확시기에 따른 품질 특성 평가가 더욱 중요하다. 고팡은 등은 대규모의 토마토 영상 데이터셋을 수집하여 토마토 개체의 속도 분류를 실시간으로 수행하는 딥러닝 모델을 제안하였다[15]. 이 논문에서 제안한 모델은 객체 탐지 분야에서 빠른 속도와 높은 정확도를 보이는 CNN 기반의 단일 스테이지 검출기(Single Stage Detector)를 활용하였다. Tian 등은 과수원 수확 시기에 발생

하는 사과와 병변을 인식하기 위해 CycleGAN과 YOLOv3-Dense를 활용하여 연구를 수행하였다[16]. 이와 같이 농업 분야에서 단감, 사과, 감귤, 토마토와 같은 다양한 농작물의 상태를 머신러닝과 딥러닝 기술을 이용하여 진단하기 위한 연구들이 활발하게 이루어지고 있다.

2.1 EfficientDet 객체 탐지 모델

본 논문에서는 단감 등급 선별 알고리즘의 한 부분으로서 단감 이미지에서 불량 영역을 검출하는 방법을 사용하는데, 객체 탐지 기법으로는 EfficientDet 모델을 사용하였다. EfficientDet은 모델 사이즈를 단계별로 변경시키는 것이 가능하며, 각 단계별 모델의 성능을 최대화하는 효율성에 초점을 맞춘 객체 탐지 모델로 구글 브레인 팀에서 개발하였다. 구글 브레인 연구팀은 객체 탐지 모델을 다양한 제약 조건의 환경에서 구축이 가능하도록 두 가지를 개선하였다. 두 가지 개선 사항은 1) 기존 FPN(Feature Pyramid Networks) 구조를 개선한 BiFPN(Weighted Bi-directional FPN)을 사용한 것이고, 2) 컴파운드 스케일링(Compound Scaling) 기법을 모델에서 적용할 수 있는 모든 곳에 적용한 모델 스케일링(Model Scaling) 기법을 사용한 것이다[17]. 연구팀은 이 두 가지 새로운 기법을 사용하여 여러 단계의 모델을 만들었는데, 이로 인하여 임베디드 시스템부터 대용량 서버까지 다양한 제약조건을 가진 환경에서 이 모델을 사용할 수 있다. 모델의 크기가 작으면 적은 리소스를 가진 컴퓨팅 환경에서 사용할 수 있기는 하지만 정확도가 떨어질 수 있으며, 모델의 크기가 클 경우 정확도는 높아질 수 있지만 컴퓨팅 자원이 많이 필요하다. 본 논문에서는 이러한 정확도/효율성 트레이드오프(trade off)를 고려하여 입력 이미지의 크기에 적합한 모델을 선정하였다. 또한 EfficientDet은 YOLOv3, RetinaNet, AmoebaNet+NAS-FPN+AA 등의 여러 객체 탐지 모델보다 정확도와 수행 속도에서 높은 성능을 보여주고 있다[18].

2.1.1 FPN 모델

EfficientDet에서 사용한 FPN은 딥러닝 모델의 아키텍처에서 계층별로 특징 맵을 생성하고 하향식(top-down)으로 특징을 결합하는 방식이다. 이는 각 단계에 적용된 컨볼루션 필터가 반대방향으로 흐르

게 하여 작은 해상도나 큰 해상도에서 얻을 수 있는 특징을 섞을 경우보다 정교하게 객체를 예측할 수 있다는 가정에서 나온 구조이다. 하지만, 이 방식은 단방향으로만 움직이는 특징으로 인하여 특징 추출에 제한적일 수 밖에 없다. PANet(Path Aggregation Network)은 이러한 문제점을 개선한 구조로 FPN에서 특징맵을 한번 더 생성한 후 상향식(bottom-up)으로 특징을 추출하여 성능을 높인 방식이다. 그 외에도 AutoML을 사용하여 특징맵 구조를 자동화한 NAS-FPN 등 다양한 FPN 방식의 모델이 존재한다.

2.1.2 BiFPN 모델의 필요성

구글 브레인 연구팀은 FPN을 채택한 모델 구조의 문제점으로 서로 다른 입력 특징들은 해상도에 따른 구분없이 단순히 더하는 방식에 문제를 제기하였다. 그리고, 출력 특징에 부여되는 가중치를 다르게 가져가야 함을 주장하며, 간단하지만 효과적인 BiFPN(Weighted Bi-directional FPN) 구조를 제안하였다. 이 구조를 사용하면 서로 다른 입력 특징들의 중요성을 학습을 통해 배울 수 있으며 이로 인하여 객체 탐지 성능을 크게 향상시킬 수 있었다[18].

2.1.3 모델 스케일링의 장점

모델 스케일링이란 EfficientNet에서 제안한 컴파운드 스케일링 기법을 통해 모델의 크기와 연산량을 결정하는 요소들인 입력 이미지 크기, 깊이, 높이를 동시에 고려하여 효율적으로 모델의 크기를 조절하는 방법을 의미한다. EfficientDet 모델인 이 방법을 통해 높은 성능을 달성할 수 있었다. 이러한 아이디어를 객체 탐지에도 적용할 수 있으며 백본(Backbone), 특징 네트워크, 박스/클래스 예측 네트워크 등 적용 가능한 모든 곳에 적용한 것이 모델 스케일링이다.

2.2 PyCaret 머신러닝 워크플로우 라이브러리

PyCaret은 적은 코드로 머신러닝 워크플로우를 자동화하는 오픈 소스 라이브러리이다[19]. 머신러닝 워크플로우란 1) 작업 정의, 2) 모델 개발, 3) 모델 도입까지의 모든 과정에 대한 흐름을 뜻한다. 1) 작업 정의 과정은 머신러닝의 문제를 구성하는 단계로서 특정 비즈니스에서 문제를 해결하기 위한 솔루션을 정의하고 데이터를 파악하는 단계이다. 2) 모델 개발

은 머신러닝 모델을 통해 데이터를 처리할 수 있도록 준비하고 모델 및 인프라를 구축하는 단계이다. 3) 마지막으로 모델 도입에서 프로그램 운영을 위해 모델을 배치하고 유지하는 단계로서 차세대 모델을 위해 추가적으로 데이터 수집을 하게 된다.

이와 같이 머신러닝 워크플로우 자동화 소프트웨어는 머신러닝 모델 개발시 많은 시간이 소요되었던 코딩, 전처리, 모델 선택, 파라미터 튜닝 작업을 Auto-ML을 통해 자동화해주는 이점이 있다. 이 때문에 머신러닝 작업을 쉽게 할 수 있으며 작업 생산성도 높일 수 있다. 이 라이브러리는 인기있는 머신러닝 라이브러리인 scikit-learn 패키지를 기반으로 하고 있으며 분류, 회귀, 군집화, 이상치 탐지 등 다양한 모델을 지원한다. 본 논문에서는 PyCaret과 훈련 데이터셋을 통해 분류 모델을 비교하고 가장 성능이 좋은 모델 2개를 채택하여 단감 등급 선별 알고리즘에 적용하였다.

3. 단감 등급 선별을 위한 시스템

Fig. 2는 본 논문에서 구축한 단감 등급 선별을 위한 전체 시스템으로 데이터 수집부터 모델 훈련 및 알고리즘 구축까지 모든 과정을 나타낸 것이다. 그림과 같이 연구를 위하여 수집한 데이터는 농가에서 수확한 단감이 대상이며, 이 데이터의 구축은 활

영을 위한 환경의 구축한 스튜디오에서 진행되었다.

그림에서 ①은 수확한 단감을 스튜디오 환경에서 촬영하는 과정으로 하나의 단감에 대하여 윗면 1장, 아랫면 1장, 옆면 3장의 사진을 촬영하였다. ②는 훈련용 데이터셋을 통하여 학습을 진행하는 과정으로 객체 탐지 모델과 머신러닝 분류 모델의 학습의 두 단계 과정을 진행한다. 먼저 객체 탐지 모델의 학습을 위하여 단감의 표면에 대한 불량 객체 영역 지정과 레이블링을 수행하였다. 다음으로 레이블링된 객체 데이터를 머신러닝 분류 모델을 위하여 수치화하는 과정을 한 번 더 수행한 다음 머신러닝의 입력을 위하여 테이블 형태의 데이터로 만든다. 이렇게 각 모델을 위한 데이터와 레이블링이 완료되면 모델별로 학습을 진행한다. 객체 탐지 모델은 EfficientDet을 사용하였으며, EfficientDet의 여러 버전별로 학습을 수행하여 가장 성능이 좋은 모델을 선정하였다. 머신러닝 분류 모델은 PyCaret을 이용하였다. 이 머신러닝 워크플로우를 통하여 다양한 머신러닝 분류 모델을 학습하여 가장 성능이 좋은 모델 2개를 선정하였다. 그림의 ③은 검증용 데이터셋을 통하여 분류를 진행하는 과정이다. 이 과정에서는 ②에서 만들어진 객체 탐지 모델과 머신러닝 분류 모델 중에서 성능이 가장 좋은 모델을 사용하여 단감 등급 선별을 수행하였다. 검증을 위해서는 객체 탐지를 통하여 불량 객체를 탐지하고 이 데이터를 세 가지 등급 중 하나로 분류하였다. 분류에 사용된 알고리즘은 ②에서 만들어진 성능이 우수한 머신러닝 분류 모델을 사용하였다.

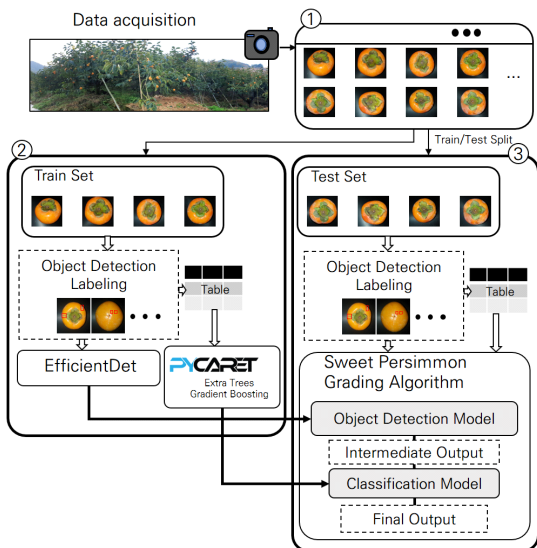


Fig. 2. Detailed process of the proposed sweet persimmon grading system.

3.1 단감 이미지 데이터 수집 방법

본 논문에서 사용된 이미지는 수확한 단감을 단감 농가의 숙련된 전문가와 연구진들이 직접 촬영하여 데이터셋을 구축하였다. 단감의 등급 선별은 단감 농가의 숙련된 전문가가 육안 분류를 통해서 진행하였으며, 불량 요인 역시 전문가의 분류로 진행되었고 사진 촬영은 본 연구의 연구진들이 진행하였다. 단감 출하를 위한 육안 분류에서 전문 작업자는 단감의 여러 면의 상태와 색상을 보는데, 단감의 표면은 구형이므로 표면 전체를 사진 한 장에 담기는 어려운 일이다. 더불어, 단감을 선별하기 위해서는 특정 단면만 가지고는 불량인지 판별할 수 없다. 이러한 문제를 해결하기 위하여 본 논문에서는 단감의 표면

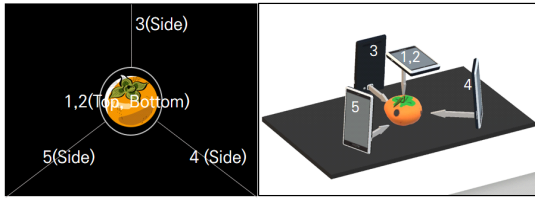


Fig. 3. An environment for sweet persimmon image acquisition: 1, 2 image shows a top and bottom view, 3, 4, 5 show a three cut side images for panoramic view.

중 윗면 1장, 아랫면 1장, 옆면 3장의 촬영을 진행하였다. 이를 통해 대부분의 표면을 사진에 담을 수 있었다. Fig. 3은 촬영 환경을 그림으로 표현한 것으로 검은색 무반사 판재 위에서 촬영을 진행하였으며, 단감의 위치와 촬영 방향을 구별하기 위해 번호와 선으로 표시해두었다. 1번 영역은 단감이 꼭지가 위로 향하는 상태로 촬영하였고(윗면), 2번은 1번과 같은 위치에서 단감의 침이 위로 향하는 상태에서 촬영하였으며(아랫면), 2, 3, 4번 위치에서 단감을 2번과 같은 상태로 두고 옆면을 촬영하였다. 3, 4, 5번 위치의 이미지는 120도 회전된 3개의 이미지이므로 이 5장의 이미지는 객체의 전체 이미지의 특징을 포함한다.

3.2 단감의 등급과 관련 요인

단감의 표준 등급은 정품, 파지(불량) 두 가지로 나누어지지만, 많은 단감 재배 농가에서는 수익을 높이기 위하여 중간 등급으로 B품을 추가하여 분류한다. 정품 등급은 상품성이 높고 흠이 없는 등급이며, B품은 약간의 불량을 포함하고 있으나 상품화가 가능한 등급, 파지 등급은 상품성이 없어 시장 출하가 불가능한 등급이다. 상품의 가격은 출하량과 출하 시기, 수요 등 여러 요인에 의해 결정되기 때문에 상품

의 품질이 곧바로 가격으로 결정되지는 않는다. 이러한 상품의 등급은 각 농가마다 농민들만의 고유한 기준에 의해서 선정되는 경우가 많다. 하지만 대부분의 농가에서 공통적으로 활용하는 대표적인 불량 요인이 존재하며, 이는 다음과 같다. Fig. 4는 단감 등급을 결정하는 불량 요인의 사진 촬영 결과이다.

(a) 탄저병(Anthrax) : 탄저균의 감염으로 발생하는 대표적인 병해이다. 탄저병이 과일에 발병할 경우 과일의 표면에 검은색의 작은 반점이 생기고, 이것이 점차 확대되어 암갈색의 타원형 병반이 되어 과일이 움푹하게 들어가게 된다.

(b) 각지벌레(Scale Insects) : 각지벌레는 식물에 기생하는 벌레로서 크기가 1~3 mm에 불과한 작은 개체이지만 하얀색을 띠고 있어서 식물에 붙어있으면 쉽게 육안으로 판별할 수 있다. 각지벌레를 제때 제거하지 않을 경우 식물이 말라죽을 수 있으며 그을 음병을 발생시킬 수도 있다.

(c) 노린재(Stink Bug) : 노린재는 과즙을 먹고 사는 곤충으로 입 모양이 과일을 찢어서 즙을 빨아들이기에 알맞은 형태를 띠고 있다. 노린재에 의해 흡즙된 과일은 흡즙 부위가 오목하게 들어가 갈색을 띠며 스펀지 상의 찰과상을 받은 것처럼 나타난다.

(d) 상처(Wounds) : 과일의 상처는 보통 작업자의 실수나 바람, 태풍 등의 자연현상으로 발생한다. 구체적으로는 수확하는 과정에서 과일끼리 충돌하거나 단감의 표면이 다른 단감의 침에 찢리는 경우, 운반 중 떨어뜨리는 경우 등의 요인이 있다. 과일의 상처가 난 부위와 그 주위는 시간이 지나면 검게 변할 수 있다.

이 외에도 검은 반점, 꼭지 들림 등과 같은 다양한 요인들이 존재한다. 이러한 요인들은 복합적으로 작

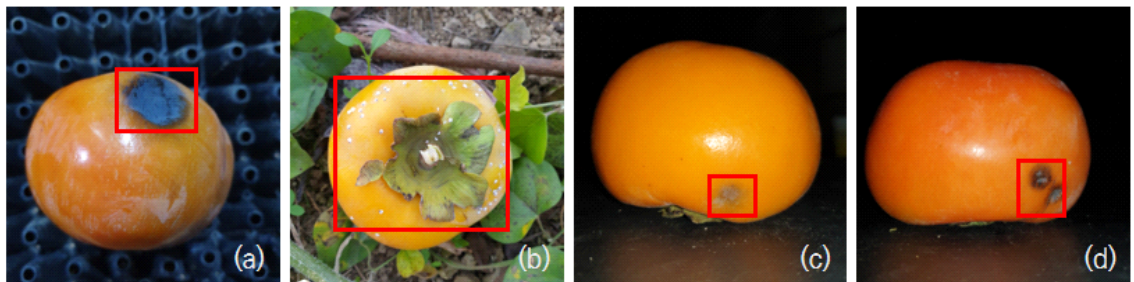


Fig. 4. Defected persimmon image data with different factors. (a) Anthrax, (b) Scale Insects, (c) Stink Bug, and (d) Wounds.

용하여 단감의 등급을 결정하게 된다. 위의 불량 요인이 하나도 없거나, 단순한 작은 반점, 작은 스크래치가 있는 경우 정품으로 취급할 수 있다. 반면, 탄저병의 경우 단감의 당도를 떨어뜨리며, 상처가 생겼을 경우 시간이 지나면 주변이 검게 변색되어 외관상 크게 문제가 없을지라도 불량으로 분류하는 경우가 많다.

3.3 레이블링과 데이터 증식

Fig. 5는 단감에 대한 레이블링 작업을 나타내는 도형이다. 레이블링 작업은 단감 재배와 수확 관련 분야의 전문성이 있는 재배 농가의 숙련가가 참여하였다. 이 작업은 먼저 하나의 단감에 다섯 면의 이미지를 촬영한 후, 이 이미지를 바탕으로 객체 탐지를 위한 영역 레이블링과 분류를 위한 레이블링을 동시에 수행한다. 객체 탐지 이후에 세분화된 분류를 수행하기 때문에 전문가가 이미지 분석 후에도 단감을 육안 분석하는 방법을 사용하였다. 이런 과정을 통하여 모든 단감에 정품, B 등급, 파지의 세 등급을 부여하였다.

이렇게 등급이 부여된 단감에 대하여 양품 단감을 제외한 단감 이미지 집합에서 5장의 각 면 이미지를 분석하였다. 이와 같이 분류된 불량 단감에 대하여 1 단계 객체 탐지 기법에 활용될 수 있도록 불량 객체를 하나하나 레이블링하고, 해당 면적을 다시 한번 수치화하였다. 불량 객체의 레이블은 불량 객체가 탄저병, 깍지벌레, 노린재, 상처 등 어떤 카테고리에 속하는가에 대한 레이블링이다. 이러한 데이터는 2 단계 과정인 머신러닝의 입력 값에 활용될 수 있도록 학습을 진행하였다. Fig. 6은 불량 객체를 수치화하는 과정에 대한 예제로 5장의 단감 이미지에서 검은 반점 2개와 상처 1개로 총 3개의 불량 객체가 존재한다. 여기서 각 객체마다 가로와 세로의 픽셀값을 곱하여 면적을 계산한 후 종류별로 합산한다. 검은 반점의 경우 1,350, 1,200의 면적값을 얻었기에 합산하

여 총 2,550이 누적되었고, 상처의 경우 하나의 객체에서 300의 면적 값을 구할 수 있다. 이미지에 나타나지 않은 객체는 0으로 처리하고, 전문가가 부여한 등급을 부여하였다.

본 논문에 수집한 이미지 데이터는 총 1,750장이지만 객체 탐지 모델과 머신러닝 분류 모델에 맞추어 데이터를 재분류하였다. 먼저, 1 단계의 객체 탐지 모델을 위한 데이터는 전체 데이터 1,750장에서 데이터 정제를 통해 1,424장을 추출하였다. 추출한 데이터에서 훈련 데이터를 1,124장(78.9%)으로 하였고 300장(21.1%)을 테스트 데이터로 활용하였다. 또한, 성능을 향상하기 위해 훈련용 데이터는 데이터 증식을 통해 원래 데이터의 4배 크기로 만들었다. 객체 탐지 모델에서 검출한 데이터는 불량 객체의 위치와 크기만을 가지고 있기 때문에 정품, B 등급, 파지 분류로 바로 사용할 수 없다. 따라서 2 단계인 머신러닝을 통한 분류를 수행하였다. 머신러닝 분류 모델에 사용한 데이터는 1,750장의 원시 이미지 데이터로부터 5 장의 이미지를 묶는 작업부터 진행하였다. 이 단계는 Fig. 6과 같이 5장의 이미지를 하나의 그룹으로 묶어서 350개의 테이블 데이터를 생성하였다. 그리고, 생성된 350개의 데이터 중에서 훈련과 테스트 용으로 부적합한 88개를 제거한 262개의 데이터를 추출하였다. 이렇게 추출된 데이터 262개 중에서 훈련 데이터 202개, 테스트 데이터 60개로 나누어 실험을 진행하였다.

스튜디오에서 촬영한 최초의 단감 이미지는 3,048 × 3,048 크기 픽셀인데, 이 이미지들은 처리속도 향상을 위하여 512*512 크기 픽셀로 변경하는 전처리를 하였다. 본 연구의 객체 탐지 모델에서 사용한 데이터는 학습 효율을 높이기 위하여 데이터 증식을 수행하였다. 이 방법은 각 이미지와 객체의 위치 각도를 90도 회전하는 작업을 3번 반복하였다. 이를 통해 원본 데이터와 90도, 180도, 270도로 회전된 이미지가 생성되어 1,124장의 이미지를 4,492장으로 증식하였다.

4. 단감 등급 선별 알고리즘

Fig. 7은 단감 품질 선별 알고리즘을 도식화한 것이다. 알고리즘에는 객체 탐지 모델과 머신러닝 분류 모델 2가지를 사용하였다. 객체 탐지 모델은 환경에 따라 효율적으로 모델 구축이 가능한 EfficientDet을 채택하였고, 분류 모델은 PyCaret을 통해 다양한 머

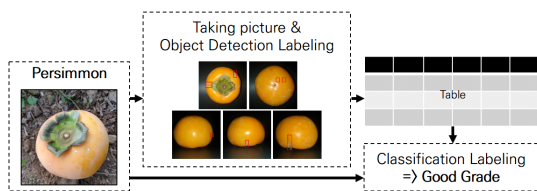


Fig. 5. Schematic of persimmon labeling method.

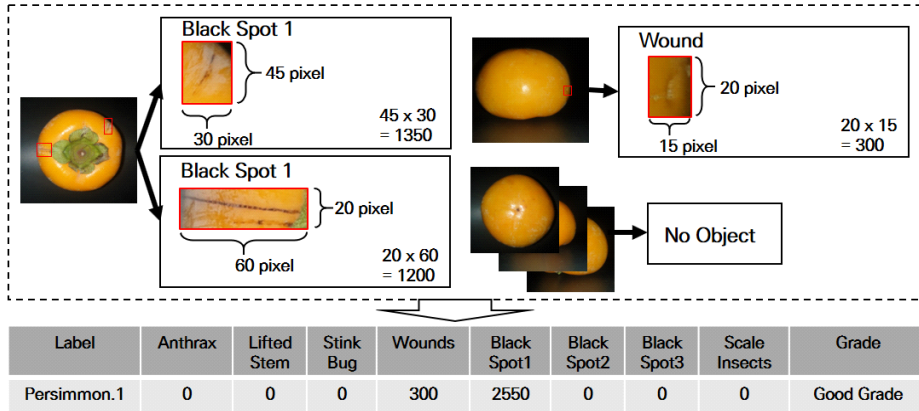


Fig. 6. Schematic result of EfficientDet object quantification method: the table shows a label, defect factors, and grade.

신러닝 모델을 테스트하여 가장 성능이 좋은 모델 2개를 채택한다는 것을 설명하였다. 이 절에서는 이를 이용하여 분류를 수행하는 알고리즘을 설명한다.

알고리즘의 순서는 다음과 같다. 1) 객체 탐지 모델에서 단감의 이미지 5장을 입력값으로 받은 후 객체 탐지를 통해 불량 요인 객체 영역과 개수를 반환 받는다. 반환받은 객체들은 Fig. 7과 같이 중간 결과물이 되는데 각 요인들은 요인의 이름과 개수, 픽셀의 면적 값 형태가 된다. 이 데이터는 다음 단계의 입력을 위하여 테이블에 저장된다. 이렇게 새로 생성된 데이터 테이블은 2) 머신러닝 분류 모델의 입력값으로 사용되고 모델은 해당 데이터를 통한 추론으로 최종 등급을 부여하게 된다.

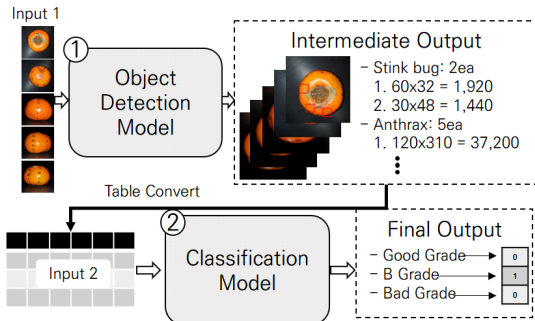


Fig. 7. Persimmon classification algorithm.

5. 실험 결과

본 논문에서는 단감 분류 알고리즘의 세부 모델로서 EfficientDet과 PyCaret 분류 모델을 생성하기 위

해 훈련 데이터셋을 사용하였고, 두 모델을 결합한 단감 분류 알고리즘을 검증하기 위해 테스트 데이터셋을 사용하였다. 학습에 사용된 컴퓨팅 환경은 구글에서 제공되는 클라우드 기반의 주피터 노트북 개발 환경으로 Colab Pro를 사용하였으며, GPU는 NVIDIA Tesla P100, RAM은 12GB로 설정된 상태에서 모델 훈련을 진행하였다.

5.1 EfficientDet 버전별 모델링과 그 결과

Table 1은 EfficientDet의 버전별 모델의 결과표로서 IoU(Intersection over Union) 지표를 사용하여 모델의 성능을 평가하였다. IoU란 두 객체 면적의 합집합 대비 교집합의 크기를 계산하여 두 객체가 얼마나 일치하는지를 나타내는 값으로 식 (1)과 같다. 식 (1)의 B_{gt} (Ground-truth bounding box)는 레이블링된 경계 상자의 면적을 의미하고, B_p (Predicted bounding box)는 모델이 추론한 결과의 경계 상자 면적을 의미한다. 일반적으로 객체 탐지 분야에서는 모델이 인식한 객체의 위치와 실제 객체의 위치를 IoU를 통해 비교하여 성능평가 기준으로 사용된다. 이 성능평가 수치가 0에 가깝다는 것은 두 물체는 겹치지 않는다고 볼 수 있으며, 1에 가깝다는 것은 두 물체는 많이 겹친다고 판단할 수 있다. Table 1의 첫 번째 행에서는 IoU 값이 0.50:0.95로 나타나 있는데 이는 0.5에서 0.95까지 0.05의 간격으로 mAP (mean Average Precision) 값을 구한 것이다. 그리고 표의 0.50과 0.75는 특정 구간에서의 결과 값을 의미한다.

Table 1. Experimental results of mean Average Precision(mAP) depending on EfficientDet object detection model versions B0, B1, B2, B3, and B4.

	IoU	B0	B1	B2	B3	B4
mean Average Precision	0.50:0.95	0.322	0.358	0.422	0.455	0.392
	0.50	0.624	0.661	0.775	0.813	0.715
	0.75	0.301	0.346	0.412	0.459	0.392

$$IoU = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \quad (1)$$

EfficientDet의 백본 네트워크는 B0부터 B7까지 있으나 B5 이상의 경우 동일한 GPU 환경에서 B4 버전의 두 배 이상의 많은 계산 시간이 소요되어 정확도를 높이는 시간대비 컴퓨팅 자원을 지나치게 많이 소모하는 한계가 있었다. 본 논문에서는 정확도/효율성 트레이드오프를 고려하여 객체 탐지 모델의 다섯 가지 백본(Backbone) 네트워크 버전을 학습하였으며, 이 중에서 가장 성능이 우수한 모델을 찾기 위한 실험을 진행하였다. 다섯 가지 백본 네트워크는 B0에서 B4까지 버전이다. 이 백본 네트워크는 버전이 높아질수록 각 층의 레이어 평균 채널 수가 커지며 층의 깊이 또한 깊어지게 된다. 그리고, 층의 깊이와 채널 수에 따라 BiFPN과 막스/클래스 예측 네트워크의 레이어의 크기도 커지게 된다. 이 때문에 B0 네트워크가 채널이 가장 작은 크기의 모델이며 숫자가 커질수록 모델의 채널 크기가 커진다. 버전이 높아질수록 백본 네트워크 입력 데이터의 크기가 커지는 것이 일반적이지만, 본 논문에서는 입력 데이터의 크기를 512×512로 모든 모델에 동일하게 적용하였

다. 결과는 B0의 경우 IU 0.50:0.95 구간에서 32.2%, 0.50 임계구간에서는 62.4%, 0.75 임계구간에서는 30.1%의 mAP를 보여 모든 구간에서 가장 낮은 성능을 보였으나, B3의 경우 IoU 0.50:0.95 구간에서 45.5%, 0.50에서 81.3%, 0.75에서 45.9%의 성능을 보여줌으로 버전이 높아질수록 대체로 좋은 성능이 나타나는 것을 볼 수 있다. 하지만 B4에서는 B2, B3 모델보다 성능이 낮아지는 것을 확인할 수 있었다.

5.2 PyCaret을 이용한 머신러닝 등급 선별 결과

본 논문에서는 수치화된 데이터 테이블을 통한 등급 선별 모델을 찾기 위해 PyCaret을 사용하여 분류가 가능한 대부분의 머신러닝 모델을 테스트하였다. PyCaret은 Gradient Boosting Classifier, Extra Trees Classifier, Logistic Regression, Random Forest Classifier, K Neighbors Classifier, Naive Bayes, Decision Tree Classifier, Ada Boost Classifier, SVM-Linear Kernel, Linear Discriminant Analysis의 10가지 주요 머신러닝 모델을 모두 적용하고 그 성능을 보여준다. Table 2는 PyCaret의 결과 테이블에서 성능을 나타내는 지표로 관련 설명은 아래와 같다.

Table 2. Experimental results of PyCaret workflow model: PyCaret workflow generates 10 different machine learning models and measured performance data of the models.

Model	Accuracy	AUC	Recall	Precision	F1	Kappa	MCC	TT(Sec)
Gradient BoostingClassifier	0.9188	0.9738	0.8361	0.9120	0.9084	0.8236	0.8331	0.230
Extra Trees Classifier	0.9188	0.9923	0.8572	0.9206	0.9136	0.8403	0.8472	0.472
Logistic Regression	0.9062	0.9046	0.8270	0.8986	0.8905	0.8071	0.8210	1.225
Random Forest Classifier	0.9062	0.9866	0.8431	0.9207	0.9047	0.8132	0.8194	0.684
K Neighbors Classifier	0.8875	0.9654	0.7828	0.8752	0.8709	0.7588	0.7767	0.142
Naive Bayes	0.8875	0.9381	0.8100	0.9173	0.8867	0.7826	0.7944	0.027
Decision Tree Classifier	0.8875	0.9102	0.8014	0.8831	0.8787	0.7739	0.7841	0.027
Ada Boost Classifier	0.8750	0.9001	0.7817	0.8659	0.8637	0.7536	0.7629	0.191
SVM-Linear Kernel	0.8312	0.000	0.6378	0.7629	0.7869	0.6657	0.6931	0.081
Linear Discriminant	0.8312	0.9543	0.6361	0.7563	0.7809	0.6029	0.6447	0.020

- 정확도(Accuracy): 전체 데이터 수 중 예측 결과와 실제값이 동일한 데이터가 차지하는 비율을 의미한다.
- AUC(Area Under the ROC Curve): ROC 커브에서 전체 면적 대비 선 그래프가 차지하는 면적 비율로서 0에서 1의 값을 가지며, 1에 가까울수록 좋은 성능을 나타낸다.
- 재현율(Recall): 실제값이 양성인 것 중에서 모델이 양성으로 판단한 비율을 의미한다.
- 정밀도(Precision): 모델이 양성이라고 분류한 것 중에서 실제값이 양성인 비율이다.
- F1 점수(F1 Score): 정밀도와 재현율의 조화평균으로 주로 분류 클래스 간의 데이터가 불균형이 심할 때 사용한다.
- 카파 통계량(Kappa): 실측값과 예측값의 측정 범주에 대한 일치도를 측정하는 방법으로 0에서 1의 값을 가지며, 1에 가까운 수일수록 측정이 잘된 것으로 판단할 수 있다.
- 매튜 상관계수(Matthews Correlation Coefficient, MCC): 실측값과 예측값 사이에 얼마나 상관관계를 가지는지 측정하는 척도로서 -1에서 1 사이의 값을 가지며 1에 가까울수록 두 값이 비슷하다고 판단할 수 있다.

본 논문에서는 좋은 모델을 선정하기 위해 정확도(Accuracy), AUC, 정밀도(Precision)를 지표로 활용하였다. 본 연구에서 준비한 훈련용 데이터에 대한 정확도를 기준으로 살펴보았을 때, Gradient Boosting Classifier와 Extra Trees Classifier가 91.88%로 동일하게 나타났다. 이 두 모델은 다른 모델이 비하여 정확도 척도에서 가장 높은 성능을 보여주었다. 하지만, F1 점수와 AUC의 경우 Extra Trees Classifier는 91.36%, 99.23%로 가장 높은 성능을 보여주었고, Gradient Boosting Classifier는 90.84%, 97.38%로 두 번째로 높은 성능을 보여주었다.

5.3 단감 분류 알고리즘 결과

Table 3은 EfficientDet과 머신러닝 분류기를 결합한 단감 분류 알고리즘을 테스트 데이터셋을 통해 추론한 결과값이다. 테스트는 모델의 성능이 가장 좋은 EfficientDet-B3 모델과 Gradient Boosting/Extra Trees를 결합한 알고리즘으로 검증하였다. 정밀도의 경우 두 모델이 83%로 같은 결과를 보여주었으나, 정확도와 재현율, F1 점수는 Gradient Boosting을 결합한 알고리즘이 조금 더 높은 성능을 보여주었다.

실험 결과 정확도 척도에서 훈련 데이터셋은 91%를 나타내었으나 테스트 데이터셋은 81% 수준의 결과를 반환하였다. 이렇게 차이를 보이는 이유는 훈련 데이터셋에 사용한 모델은 단일 모델로서 작업자가 직접 영역을 지정하여 레이블링한 데이터를 통한 결과값이지만, 테스트 데이터셋에 대한 처리에서는 EfficientDet 모델의 추론으로 얻은 결과에 대해 분류를 적용하였기 때문에 분류기에 제공되는 입력 단계에서 발생하는 오차가 추가적으로 존재하기 때문이다.

Fig. 8은 분류 그룹이 두 개인 경우와 세 개인 경우의 F1 점수를 시물레이션을 통해 비교하고 있다. 모두 500번의 예측 에폭(epoch)을 진행하였고, 매 에폭마다 100번의 예측을 실시하였다. 시작 단계에서는 무작위 예측을 실시하여 2개 그룹 분류를 기준으로 50%의 정확도에서 시작해 100번째 에폭에서는 완전한 예측 성능, 즉 100% 정확도를 갖도록 하였다. 그래프의 가장 위에 나타나는 점선으로 표시된 꺾인 직선이 예측기의 정확도 변화를 보이고 있다. 무작위 예측 단계에서 두 개 그룹 분류기의 F1 점수는 0.50, 세 개 그룹 분류기의 F1 점수는 0.33 정도이다. 100번째 에폭까지 예측의 정확도를 100%로 높이고, 이후 500 에폭까지 40,000번의 예측은 오류 없이 정확하게만 이루어지는 완벽한 분류기를 시물레이션했을 때 F1이 어떻게 변화하는지 보이고 있다. 두 개 그룹 분류기의 F1 점수는 점선으로 표시된 곡선이고, 세 개 그룹 분류기는 가장 아래 나타난 곡선과 같이 F1

Table 3. Experimental results of persimmon grading algorithm: B3 backbone network for EfficientDet and Gradient Boosting/Extra Trees classifiers.

EfficientDet	Classifier	Accuracy	Precision	Recall	F1 Score
B3	Gradient Boosting Classifier	0.81	0.83	0.82	0.82
	Extra Trees Classifier	0.80	0.83	0.80	0.81

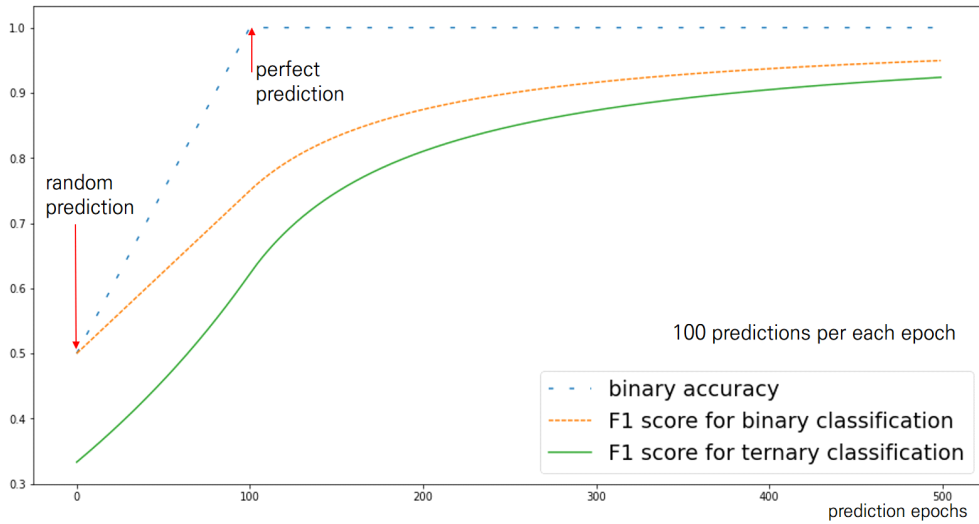


Fig. 8. F1 score simulation for binary and ternary classifications.

점수가 변화하다. 세 개 그룹 분류기의 F1이 0.9까지 수렴하는 데에는 훨씬 긴 시간이 필요하다는 것을 확인할 수 있다. 선행 연구가 단감을 양품과 불량품으로만 단순 비교한 것에 비해 세 단계의 분류를 하는 제안된 기법의 F1 점수가 0.82 수준에 도달한 것은 높은 수준의 정확도로 예측을 수행하는 것임을 확인할 수 있다.

본 연구는 기존 연구의 모델에서 발생했던 문제점을 보완한 새로운 모델을 구축함으로써 다음과 같은 개선점을 찾을 수 있었다.

1. 기존 연구에서는 이미지 전체를 보고 품질을 구별하는 방법을 사용하였기 때문에 어떠한 불량 요인이 상품 등급에 영향을 미쳤는지에 대한 정량적 파악이 어려웠다. 본 연구에서는 객체 탐지 기법을 통해 단감 이미지 내에 있는 불량 객체의 종류와 수, 면적과 같은 상세한 정보와 함께 정량적 파악이 가능해졌다.

2. 단감의 사진 이미지와 객체 탐지 기법을 사용하였기 때문에 구체적인 불량의 종류와 크기에 따른 가중치 부여가 가능해졌다. 가령, 탄저병의 경우 가로와 세로가 특정한 픽셀 크기 이상으로 인식되면 불량으로 간주한다. 하지만, 검은 반점의 경우 동일한 크기의 객체가 인식되더라도 B품으로 분류될 수 있다. 이처럼 불량 요인에 따라 부여되는 가중치를 다르게 적용할 수 있어 이전 기법보다 세부적인 분류

가 가능하다. 즉, 작은 불량 객체지만 등급에 큰 영향을 미치는 요인이나, 반대로 큰 객체지만 등급에 영향이 적은 요인들을 세부적으로 분석할 수 있게 되었다.

3. 불량 객체에는 여러 종류가 있는데 본 연구에서는 이전보다 범위를 더 넓혀서 분류 모델을 구축할 수 있게 되었다. 이전의 많은 연구에서는 깍지벌레, 노린재, 검은 반점과 같은 여러 가지 불량 요소들을 인식하고 세부적인 등급을 부여하는 것이 어려웠다. 더불어 탄저병과 상처 같은 요인들도 객체의 크기와 개수가 적으면 불량을 정품으로 오분류하는 경우가 많았다. 하지만, 객체 탐지를 통해 작은 불량 객체 인식이 가능해지고, 객체 종류의 범위를 넓힐 수 있었다. 이러한 상세한 객체 탐지를 통해 전문가 수준의 과실 질병 탐지와 예방이 가능해졌다.

4. 제안된 기법은 3단계 분류를 통해 최상급은 아니지만 판매는 가능한 수준의 상품을 따로 분류해 낼 수 있다. 이는 소득을 창출할 수 있는 상품을 불량으로 분류해 출하지 못 하여 발생하는 손실을 최소화할 수 있다. 또한 상품을 판매하고자 하는 대상이나 특정한 기준을 요구하는 구매처의 제약조건에 따라 불량 제품의 범위를 조절할 수 있는 유연한 분류기를 구현할 수 있다.

Table 4는 제안된 기법이 기존 선행 연구에서 제안된 기법에 비해 어떠한 장점을 갖는지 정리하여 보이고 있다. 위에서 언급한 요인들로 인해 이전 연구에 비하여 더 정밀한 과실 표면 분석이 가능해졌으

Table 4. Advantage of the proposed method compared to previous work.

	Proposed method	Previous method
Advantages by object detection model		
Defect factor identification	○	×
Defect quantification	○	×
Specific defect product elimination	○	×
Advantages by multi-class classification		
# of classification groups	3	2
Product loss by mis-classification	Minimized	Accuracy dependent
Overall advantages		
Specifiable defect identification based merchandising targets		

며, 이를 통하여 과실의 등급을 불량, 정품 2단계에서 B품을 추가한 3 단계로 판별이 가능하게 되었다. 그리고, 등급의 세분화는 농가의 소득을 증가시킬 수 있는 직접적인 요인이 될 것으로 기대된다.

6. 결론 및 향후 연구과제

농업 분야는 작업량과 인력이 수확시기에 집중되기 때문에 타 산업과 비교하여 연간 노동량의 불균형이 심한 편이다. 이로 인해, 많은 과수 농가에서는 수확시기마다 인력 부족으로 인한 어려움을 겪고 있다. 이러한 인력 부족 문제를 극복하기 위해 농업 분야에서 과실 분류 자동화 연구가 지속적으로 이루어지고 있다. 본 논문에서는 EfficientDet과 머신러닝 분류 모델을 결합하여 단감의 등급을 선별하는 알고리즘을 제안하였으며 기존의 2 단계 등급 선별보다 더 세분화된 3 단계 등급 선별 기법을 제안하고 구현하였다. 실험을 위하여 총 1,750장의 이미지 데이터를 수집한 후 두 모델에 맞게 전처리 작업과 전문가의 레이블링을 먼저 수행하여 진행하였다. 이러한 자동 분류 기기는 농가의 수작업 분류 작업을 덜어주기 위한 용도로 활용될 수 있을 것이다. 더불어 이러한 분류 기기의 데이터가 매년 누적된다면 시계열 분석을 통해 다년간 질병의 추세를 파악하는 것이 가능해질 것이다. 다년간의 시계열 데이터가 축적된다면 이를 바탕으로 과일에 발생하는 질병의 추세를 파악할 수 있어 추세에 따른 적절한 대응이 가능해질 것이며 나아가 수확량 예측까지 가능해질 것이다. 기계학습 기반의 단감 등급 분류 시스템은 농가에 다음과 같은 실질적 기여를 제공할 수 있다.

1. 연간 인력 수요와 공급의 불일치에 의해 발생하는 수확시기 인력난에 의한 원활한 상품 출하 지연 방지
2. 불량 요인 데이터의 축적을 통해 병충해 발생에 대한 과학적 시계열 분석이 가능하며 농가 소득 저하의 원인이 되는 불량 발생 요인에 대한 사전 대응 가능

실험은 불량 요인이 되는 객체 탐지를 위한 모델로 EfficientDet을 사용하였고, 백본 네트워크는 버전 B0에서 B4까지 성능을 테스트하였다. 테스트 결과 백본 네트워크 버전 B3이 IoU 0.50:0.95구간에서 45.5%로서 가장 높은 성능을 보여주었다. 다음 단계인 머신러닝 기법을 통한 분류를 위하여 머신러닝 워크플로우 라이브러리인 PyCaret을 사용하였다. 이를 이용하여 10여 종류의 다양한 분류 모델을 비교하고 그 성능을 측정하였다. PyCaret에서 사용한 여러 분류 모델 중에서 Gradient Boosting Classifier와 Extra Trees Classifier의 정확도가 91.88%로 가장 높은 성능을 보여주었다. 그리고, 두 모델을 결합한 단감 등급 선별 알고리즘의 최종적인 정확도는 81% 수준의 성능을 보여주었다.

이와 같은 2단계의 단감 등급 선별 알고리즘은 객체 탐지 분야와 머신러닝 분야가 가진 개별적인 분류 기법으로 인한 한계점을 극복할 수 있었다. 객체 탐지 모델의 경우 이미지 내에서 불량 객체를 정밀하게 파악할 수 있지만, 다섯 장의 단감 이미지를 전부 파악하여 하나의 등급을 부여하는 것은 어렵다. 반대로 머신러닝 분류 모델의 경우 이미지의 다양한 불량 요인을 분석해 내는 일에는 어려움이 있다. 본 논문

에서 구현한 방법은 객체 탐지 모델과 머신러닝 분류 모델을 결합하여 불량 요인에 따라 불량 객체를 탐지하고 이를 정량화함으로써 단감의 세부적인 등급 선별이 가능해졌다. 더불어, 선행 연구에서 어려웠던 1) 불량 객체의 종류별 개수 및 면적의 정량적 파악, 2) 불량 종류와 크기에 따른 가중치 부여, 3) 불량 객체 종류의 범위 확대까지 수행할 수 있었다. 무엇보다 선행 연구에서 양품과 불량이라는 이진 분류만 가능했던 한계를 극복하고 양품, B품, 파지의 3단계 선별이 가능해졌기 때문에 농가 소득 향상에 기여할 수 있을 것으로 예상된다.

본 논문의 실험 결과 EfficientDet을 활용한 객체 탐지 기법은 조도와 색상의 영향에 매우 민감한 홍시와 같은 객체 판별은 좋은 성능을 보여주지 못하는 한계가 있었다. 이에 따라 조도와 색상에 민감한 과일 객체의 판별을 위한 보다 더 심화된 연구가 이루어져야 할 것이다. 향후에 본 연구의 결과를 임베디드 시스템과 자동화된 기기에 적용한다면 더욱 고도화된 분류 기계의 제작도 필요할 것으로 보인다.

REFERENCE

- [1] M. Valeria, E.G. Carlos, A.Z. Laura, M.C. Jose, G. Jorge, and G. Hamurabi, et al., "Comparison of Convolutional Neural Network Architectures for Classification of Tomato Plant Diseases," *Applied Sciences*, Vol. 10, Issue 4, ID 1245, 2020.
- [2] H. Ham, D. Kim, J. Chae, S. Lee, Y. Kim, and H.U. Cho, et al., "A Study of Tomato Disease Classification System Based on Deep Learning," *The Transactions of the Korean Institute of Electrical Engineers*, Vol. 69, No. 2, pp. 349-355, 2020.
- [3] G. Wang, Y. Sun, and J. Wang, "Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning," *Journal of Computational Intelligence and Neuroscience*, Vol. 2017, Article ID 2917536, 2017.
- [4] S.P. Mohanty, D.P. Hughes, and M. Salathe, "Using Deep Learning for Image-Based Plant Disease Detection," *Journal of Frontier in Plant Science*, Vol. 7, No. 1419, pp. 1-7, 2016.
- [5] L.G. Nachtigall, R.M. Araujo, and G.R. Nachtigall, "Classification of Apple Tree Disorders Using Convolutional Neural Networks," *Proceedings of the 2016 IEEE 28th International Conference on Tools with Artificial Intelligence*, pp. 472-476, 2016.
- [6] J. Park, H. Kim, and K. Kim, "Assessing Impact of DCGAN Image Data Augmentation for CNN based Tomato Disease Classification," *Journal of Digital Contents Society*, Vol. 21, No. 5, pp. 959-967, 2020.
- [7] K. Masuda, M. Suzuki, K. Baba, K. Takeshita, T. Suzuki, and M. Sugiura, et al., "Noninvasive Diagnosis of Seedless Fruit Deep Learning in Persimmon," *The Horticulture Journal* 90, Vol. 2, pp. 172-180, 2021.
- [8] M. Kanae, S. Maria, B. Kohei, T. Kouki, S. Tetsuya, and S. Maya, et al., "Noninvasive Diagnosis of Seedless Fruit Using Deep Learning in Persimmon," *The Horticulture Journal Preview*, Vol. 90, No. 2, pp. 172-180, 2021.
- [9] M. Afonso, H. Fonteijn, F.S. Fiorentin, D. Lensink, M. Mooij, and N. Faber, et al., "Tomato Fruit Detection and Counting in Greenhouses Using Deep Learning," *Frontiers in Plant Science*, Vol. 11, Article 571299, 2020.
- [10] J.W. Oh, H.K. Kim, and I.T. Kim, "Design and Implementation of Fruit harvest time Predicting System based on Machine Learning," *Smart Media Journal*, Vol. 8, No. 1, pp. 74-81, 2019.
- [11] J.S. Kim and E.K. Kim, "Agricultural Autonomous Robots System for Automatic Transfer of Agricultural Harvests," *Journal of the KIECS*, Vol. 16, No. 4, pp. 749-754, 2021.
- [12] S.H. Oh, S.W. Yang, H.C. Kim, D.H. Kim, and Y.H. Doh, "Development of Automated Quantitative Spray Control System for High Quality Crop Cultivation," *The Journal of The Institute of Internet, Broadcasting and Com-*

munication, Vol. 17, No. 3, pp. 267-274, 2017.

[13] S.H. Roh and D.G. Park, "Sweet Persimmons Classification based on a Mixed Two-Step Synthetic Neural Network," *Journal of Korea Multimedia Society*, Vol. 24, No. 10, pp. 1358-1368, 2021.

[14] K.P. Ferentions, "Deep Learning Models for Plant Disease Detection and Diagnosis," *Computers and Electronics in Agriculture*, Vol. 145, pp. 311-318, 2018.

[15] K.E. Ko, S. Yang, and I.H. Jang, "Real-Time Tomato Ripeness Classification System based on Deep Learning Model for Object Detection," *Journal of Institute of Control Robotics and Systems*, Vol. 24, No. 11, pp. 999-1004, 2018.

[16] Y. Tian, G. Yang, Z. Wang, E. Li, and Z. Liang, "Detection of Apple Lesions in Orchards Based on Deep Learning Methods of CycleGAN and YOLOV3-Dense," *Journal of Sensors*, Vol. 2019, Article ID 7630926, 2019.

[17] P. Dollar, M. Singh, and R. Girshick, "Fast and Accurate Model Scaling," *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 924-932, 2021.

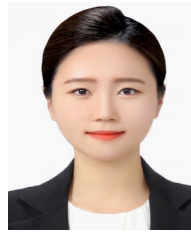
[18] M. Tan, R. Pang, and Q.V. Le, "EfficientDet: Scalable and Efficient Object Detection," *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10778-10787, 2020.

[19] PyCaret Low-code Machine Learning, <https://pycaret.org/> (accessed April 21, 2022).



노 승 희

2019년 국립창원대학교 산업시스템 공학과(이학사)
 2020년~현재 국립창원대학교 정보통신공학과(공학석사과정)
 관심분야: 딥러닝, 강화학습, 임베디드 시스템, 스마트 그리드



강 은 영

2021년 국립창원대학교 산업시스템 공학과(이학사)
 관심분야: 데이터 분석, 딥러닝, 헬스케어



박 동 규

1993년 부산대학교 전자계산학과(이학사)
 1996년 부산대학교 전자계산학과(이학석사)
 1999년 부산대학교 전자계산학과(이학박사)

2000년~2002년 영산대학교 멀티미디어 공학과 전임강사
 2002년~현재 창원대학교 정보통신공학과 교수
 관심분야: 모바일 시스템 및 서비스, 바이오 헬스케어, 머신러닝



강 영 민

1996년 부산대학교 전자계산학과(이학사)
 1999년 부산대학교 전자계산학과(이학석사)
 2003년 부산대학교 전자계산학과(이학박사)

2003년~2005년 한국전자통신연구원 연구원
 2005년~현재 동명대학교 게임공학과 교수
 관심분야: 컴퓨터 그래픽스, 게임 프로그래밍, 머신러닝