

인공신경망 YOLOv5 알고리즘을 이용한 밀싹 흰곰팡이 탐지 서비스 구현

허진경*

Implementation of White Molds Detection Service of Wheat Sprouts using YOLOv5 Algorithm of Artificial Neural Network

JinKyoung Heo*

본 성과물은 산업통상자원부-한국산업기술진흥원-지역혁신클러스터육성(R&D)-국가혁신융복합단지지원 연구사업(과제번호: P0015318)의 지원으로 이루어진 것임

요약

최근 인구감소, 고령화로 인한 농촌 문제를 인공지능 기술을 이용해 해결하기 위해 많은 연구가 진행되고 있다. 가장 대표적인 것이 스마트팜 관련 사업의 추진일 것이다. 또한, 농가소득 증대에 기여할 수 있는 기능성 작물에 관한 관심도 높아지고 있다. 따라서 본 연구는 기능성 작물인 밀싹의 생산성을 향상시키고 상품성을 높일 수 있도록 흰 곰팡이를 조기에 탐지할 수 있는 YOLOv5 기반 모델을 개발하였고 성능평가를 통해 IoU 0.5 이상일 때 mAP 값이 0.714인 결과를 얻었다. YOLOv5 기반 탐지 흰곰팡이 탐지 모델과 MQTT 서비스 그리고 자바 기술을 이용해서 실시간 흰곰팡이 탐지를 위한 시스템을 개발하였다.

Abstract

Many studies are being conducted to solve rural problems due to the decrease in the population of rural areas and aging. The most representative would be the promotion of smart farm-related projects. In addition, interest in functional crops that can contribute to increasing farm income is also increasing. Therefore, in this study developed a YOLOv5 based model that can detect white fungi early to improve the productivity and commerciality of wheat sprouts, which are functional crops. And through the performance evaluation, the result was obtained in which the mAP value was 0.714 when the IoU was 0.5 or higher. We developed a real-time white mold detection system using YOLOv5 based white mold detection model, MQTT service, and Java web application.

Keywords

artificial neural network, object detection, YOLOv5, wheat sprouts white mold detection, realtime streaming

* 연세대학교 산학협력단(공학계열)
- ORCID: <https://orcid.org/0000-0001-7279-0994>

• Received: Sep. 27, 2022, Revised: Oct. 17, 2022, Accepted: Oct. 20, 2022
• Corresponding Author: JinKyoung Heo
Dept. of University Industry Foundation, Yonsei University, Korea
Tel.: +82-2-2123-2147, Email: hjk7902@yonsei.ac.kr

I. 서 론

고령화, 저출산화 심화로 인해 농가 수는 지속해서 감소하고 있으며 감소 폭 또한 매년 커지고 있다. 농촌 인구 감소는 농업 생산성 저하뿐만 아니라 농촌 지역 과소화를 초래해 향후 농촌 인구 유입까지도 저해하는 요소이다.

따라서 생산인력을 대체할 수 있는 무인·자동화 기술 개발과 기능성 작물 재배 등을 통해 농가 경영 안정화와 농업·농촌 소득 증대가 필요하다. 농촌 환경 기계화, 자동화를 통해 정밀농업이 가능할 것이며 노동력 투입을 최소화하여 농업 생산성을 최대화할 수 있을 것으로 예상된다.

본 연구는 기능성 식물 중 밀싹의 생산성 향상을 위해 인공지능을 이용한 흰 곰팡이 탐지 방법을 연구하였다. 밀싹은 향산화 및 생리활성 능력이 뛰어난 이소오리엔틴과 플라본배당체 등의 폴리페놀류가 100g당 최대 약 1,360mg 들어있으며, 운동력 및 지구력 증진 효능이 있는 폴리코사놀 일종의 옥타코사놀은 약 450mg 함유하고 있다. 밀싹의 기능성 물질 함량은 시설 재배할 경우 싹이 나고 9일~12일이 지난 뒤 가장 높은 것으로 나타남에 따라 단시간에 길러 수확할 수 있어 농가소득 증대에 이바지할 수 있는 작물이다.

밀싹은 재배 기간이 9일~12일로 상대적으로 다른 작물에 비해 짧은 기간으로 재배되는데, 새싹 풀에 속하는 밀싹은 습한 환경의 토양에서 재배되기 때문에 곰팡이가 피는 조건과 비슷하며 흙에 있는 흰곰팡이 등 여러 오염요소가 밀싹에 묻어날 수 있어 재배 시 흰 곰팡이 등을 사전에 탐지하여 상품성을 유지하는 것이 필요하다.

따라서 본 연구는 CNN 기반의 YOLO v5 알고리즘을 이용하여 밀싹의 흰 곰팡이를 탐지하는 모델을 개발하고 시스템을 구축하였다.

II. 관련 연구

2.1 CNN ResNet 모델

[1]연구에서는 표고버섯에 발생한 푸른곰팡이를

탐지하기 위해 CNN 모델 중 심층 신경망 형태인 ResNet을 활용하여 병충해 감염 여부를 판단하는 연구를 하였다. 표고버섯 병충해 이미지 데이터를 분석하기 위한 모델은 56-layer로 구성하여 분석하였다. 분석 모델의 원본 데이터 테스트는 Train 65% Test 72%, Augmentation 데이터 테스트에서 Train 87%, Test 81%가 측정되었고 전처리된 데이터를 활용하여 데이터의 절대적인 양에 따른 모델의 정확도 비교평가를 진행하였다, 그 결과 학습데이터가 많을수록 분석 모델의 정확도가 향상되는 것을 확인하였다.

2.2 Faster R-CNN과 CNN의 결합

[2]에서는 MobileNet v1, Inception v2, ResNet50 및 ResNet101등의 CNN 특징추출자들과 Faster R-CNN 모델을 결합한 딥러닝 알고리즘을 사용하여 토마토 질병과 해충을 감지하고 식별하는 방법을 제안하였다. Inception v2와 Resnet50의 수렴 속도는 중급이었고 Resnet101이 가장 빠르게 수렴하는 것으로 나타났다. 세균성궤양병을 제외한 모든 질병에 대해 MobileNet v1의 검출률이 매우 낮았으며 ResNet50의 탐지율은 MobileNet v1보다 높지만 다른 알고리즘보다 낮았다. Inception v2의 탐지율은 전반적으로 높았으며 ResNet101의 탐지율은 모든 모델 중에서 가장 높은 것으로 확인했다.

2.3 분할 이미지를 활용한 R-CNN 기반 모델

[3] 연구에서는 딥러닝 알고리즘을 이용하여 딸기 영상 데이터의 병충해 존재 여부를 자동으로 검출할 수 있는 서비스 모델을 제안하였다. 모델은 CNN 기반 YOLO를 선정하여 기존의 R-CNN 기반 모델의 느린 학습속도와 추론속도를 개선하였다. 또한, 일반적인 학습데이터 셋을 학습한 경우보다 병충해에 특화된 분할 이미지 데이터셋을 학습했을 때 병충해 검출률은 81.35%에서 91.93%로 병충해 검출 신뢰도는 73.35%에서 83.41%로 향상되는 것을 확인하였다.

III. 데이터셋 준비

3.1 데이터 수집 모듈

제안하는 밀싹의 흰곰팡이 이미지 탐색 모델을 학습시키기 위해 밀싹 재배기에서 촬영한 이미지를 수집하기 위해 덤스카메라 1대와 IP 카메라 2대로 구성하였으며, 이 중 IP 카메라 1대는 실시간 흰곰팡이 탐지 서비스를 위해 사용했다. 그림 1은 밀싹 재배기와 흰곰팡이 이미지 수집을 위한 카메라이다.

데이터는 밀싹 재배기 안에 설치된 카메라 2대를 이용하여 5분마다 촬영을 하고, 촬영과 동시에 내부 서버에 직접 저장하는 방식으로 수집한다. 사선 방향은 덤스 카메라를 이용하고, 수직 방향은 IP 카메라를 이용하여 촬영했다.

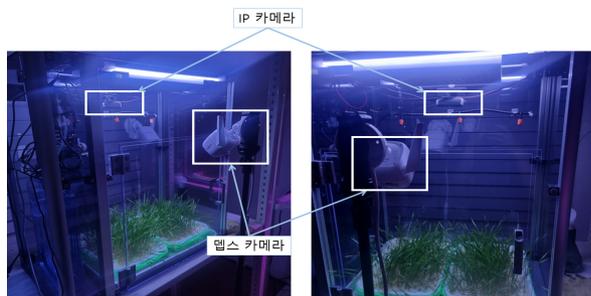


그림 1. 밀싹 재배기와 카메라
Fig. 1. Wheat sprouts incubator and camera

3.2 데이터셋 구성

표 1은 인공적인 식물재배실에서 밀싹을 재배하면서 2021년 9월 17일부터 2021년 10월 5일까지 19일간 흰곰팡이 피해 여부와 관계없이 수집한 15,761개 파일에 대한 정보이다.

표 1. 데이터셋
Table 1. Dataset

Method		CCTV filming			
File type		JPG			
Resolution		640 x 480px			
Place	A	Camera angle	Front	File no.	4,055
			Side	File no.	3,910
	B	Camera angle	Front	File no.	3,890
			Side	File no.	3,906

15,761개의 이미지는 19일간 5분 단위로 촬영된 식물 이미지였기 때문에 사진의 변화가 크지 않았기 때문에 전체를 다 사용할 수는 없었으며, 데이터 전처리를 통해 어두운 환경에서 촬영된 사진과 흰곰팡이가 없는 정상 밀싹 사진 등을 제외하고 자연광 환경에서 흰 곰팡이가 명확히 보이는 이미지만 학습용 데이터로 선정하였다. 그 결과 최종적으로 123개만 인공지능 학습데이터로 사용하였다.

따라서 원본 이미지 데이터의 한계점이 존재하였다. 첫째, 학습에 사용 가능한 이미지 개수가 123개로 학습용으로 비교적 적었고 둘째, 640x480화소로서 파일의 크기가 0.1MB가 되지 않는 CCTV 이미지로 저화질이였다. 셋째, 어노테이션 레이블을 1개만 설정하고 모델링 진행 시 mAP(mean Average Precision)가 0.65 이상 향상되지 않는 문제가 발생했다.

또한, OpenCV를 이용하여 화질 개선 작업이나 이미지 증식 작업할 경우 mAP가 0.4 이하로 더욱 감소하는 결과가 발생하였다. mAP를 개선하기 위해 어노테이션 방법을 바꿔보았다.

3.3 데이터 구성 한계 극복을 위한 어노테이션

데이터셋 구성의 한계점을 극복하려는 방안으로 다음과 같은 방법을 사용했다.

첫째, 자연광이 충분히 들고 명확하게 곰팡이가 보이는 이미지만을 선별했으며 둘째, 곰팡이가 흙과 명확하게 구분되도록 좁게 레이블링 작업을 수행했다. 셋째, 흙과 명확하게 구분되는 선명한 곰팡이는 'mold' 비교적 선명하지 않은 곰팡이는 'blur'로 2가지 클래스로 나누어 레이블을 구분하였다.

어노테이션 도구로 'ROBOFLOW'를 사용하였으며 학습비율은 학습용 데이터셋 70%, 검증용 데이터셋 30%로 구성하였다.

어노테이션 작업은 이미지에서 곰팡이 위치를 찾아서 경계 상자(Bounding box)로 레이블링하고 각각의 레이블 명은 흰곰팡이는 mold, 곰팡이 의심 또는 선명하지 않은 곰팡이는 blur로 정했다.

IV. 객체탐지 모델

4.1 이미지 탐지 학습 모듈

본 연구의 시스템 구현을 위해서 Faster R-CNN, EfficientNet, YOLO v3, YOLO v5 모델을 비교 테스트하였고 이중 neural network(CNN) 기반 최신 알고리즘인 YOLOv5를 사용했다. YOLOv5는 이전의 YOLO 시리즈와는 다른 backbone을 사용하여 FPS와 mAP면에서 뛰어난 성능을 보인다.

Faster R-CNN는 이미지 객체탐지에 있어서 매우 성능이 좋은 모델로 잘 알려져 있으며 테스트 결과 최대 mAP 0.99가 도출되기도 했다. 그러나 1회 모델링 실행 속도가 느려 시의성이 중요한 곰팡이 탐지에 적절하지 않다고 판단하였다.

EfficientNet 역시 이미지 분류 작업에서 기존보다 훨씬 적은 파라미터로 좋은 성능을 내는 모델로 알려져 있으나 테스트 결과 정확도(Accuracy)가 1.0으로 과소적합 문제가 발생했고 보유한 데이터셋의 정확도 탐지에 알맞지 않은 모델로 판단했다.

YOLOv3는 특징 추출을 수행하기 위해 Darknet-19와 잔상(Residual) 네트워크를 결합한 새로운 네트워크를 사용한다. v3에서 사용하는 컨볼루션 계층(Convolution layer)은 연속적인 3×3과 1×1 구조이며 53개의 컨볼루션 계층을 사용하기 때문에 darknet-53라 칭한다. Darknet-53은 초당 높은 부동소수점 연산을 수행하며, GPU 활용성이 좋아 성능 효율성이 크게 향상되었지만 그만큼 가중치가 커졌기 때문에 속도 즉, FPS 관점에서는 성능이 저감되었으나 여전히 다른 객체 검출 알고리즘보다는 상대적으로 빠른 속도로 객체 검출을 할 수 있다 [3][4][5].

YOLOv5는 이전의 YOLO 시리즈와는 다른 백본을 사용하여 FPS와 mAP면에서 뛰어난 성능을 보인다. 가중치별로 지금까지 나온 YOLOv5의 모델들은 YOLOv5 s, m, l, x, s6, m6, l6, x6로 총 8개이다. 모델별로 s부터 x6까지 점진적으로 정확도가 향상되는 반면 속도는 느려진다. 이들은 각각 depth multiple과 width multiple의 차이로 발생하게 된다. 또한, 이전의 시리즈와 다른 백본의 아키텍처를 사용하는데 CSPNet을 이용하여 학습능력을 강화하는 것과 동시에 정확도를 유지하며 경량화를 가능하게 구성되었다. 기본적으로 본래 파라미터 수가 감소하

면 정확도도 감소하게 되지만 병목현상을 방지하여 단축 연결(Short connection)로 덧셈 연산이 필요 없게 되며 결과적으로 연산량이 적어짐에도 정확도는 향상되는 효과가 있다[4][6].

따라서 더 알맞은 정확도를 도출할 수 있고 결과 처리가 가장 빠른 YOLO v5를 채택하였다. YOLO v5의 파라미터 img_size는 600, batch_size는 32, epochs는 300, weigth_size는 1로 설정하였다.

4.2 모델 평가 지표

학습한 이미지의 객체 검출의 정확도를 파악하기 위해 IoU(Intersection over Union)를 평가지표로 사용하였다. IoU는 알고리즘이 설정한 바운딩 박스와 사용자가 설정한 바운딩 박스의 중첩된 면적을 합집합에 면적으로 나뉜 비율로 IoU 값은 1에 가까울수록 명확하게 검출된 것이며 0.5 이상이면 제대로 검출됐다고 판단한다.

샘플 IoU 점수(Sample IoU scores)는 그림에서 보는 것처럼 0.5만 되어도 3분의 2 이상이 겹쳐지기 때문에 제대로 검출됐다고 판단하는데, 일반적인 IoU를 계산하는 방법은 두 상자가 겹치지 않으면 어느 정도의 오차로 교집합이 생기지 않은 것인지 알 수 없어 경사 소실(Gradient vanishing) 문제가 발생한다.

경사 소실문제를 해결하려는 방법으로 GIoU (Generalized IoU)는 두 경계 상자 A와 B를 모두 포함하는 최소 크기의 상자(C)를 활용하는데, 기존 IoU에서 C 상자 중 A와 B 모두 겹치지 않는 영역의 비율을 뺀 값이 GIoU가 된다. 물론 GIoU는 클수록 좋다. 실제 객체 탐지에서는 손실함수로 1-GIoU를 사용하며 손실 값의 범위는 0~2이다[7].

GIoU는 경사 소실 문제는 개선했지만, 수렴 속도가 느리고 부정확하게 상자를 예측하는 문제점이 있어 이를 해결하기 위해 DIoU(Distance IoU)는 IoU와 함께 중심 좌표를 활용한다. 경계 상자에 대한 좋은 손실함수는 중첩되는 영역(Overlap area), 중심점과 거리(Central point distance) 그리고 비율(Aspect ratio) 이 세 요소를 고려하는 것이다[8]. 본 연구에서는 IoU의 계산에 DIoU를 적용했다.

본 연구는 합성곱 신경망(CNN) 모델에서 탐지된 객체의 정확도 평가지표로 mAP를 사용했다. 그림 2처럼 mAP는 Precision-Recall Curve를 통해 도출하며 평가한다. 본 연구 모델에서 객체탐지의 정확도를 담보하기 위해서 [4][6]을 참고하여 mAP 지표 평균값은 0.7 이상으로 설정하였다.

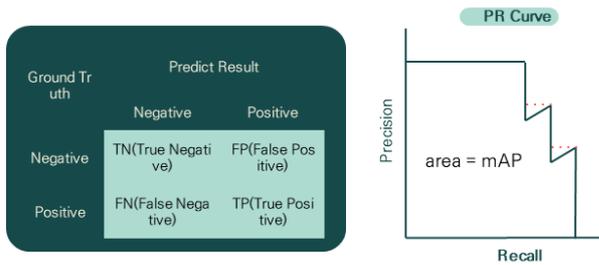


그림 2. mAP(클래스별 평균 정밀도의 평균)
Fig. 2. mAP(mean Average Precision)

4.3 모델 검증

본 연구의 모델 성능평가를 위해 IoU가 0.5 이상인 이미지 데이터셋에 대한 객체의 정확도를 실험하였다. 실험 환경은 표 2와 같다.

표 2. 시험 환경

Table 2. Experimental environment

HW/SW	Environment
CPU	Inter core TM i5 1135G7
GPU	NVidia Geforce RTX 2080 Super
Ram	16GB
OS	64bit OS
Algorithm	YOLO v5
Evaluation criteria	IoU >= 50%, mAP >= 70%

실험 결과, IoU 0.5 이상일 때 mAP 값은 0.714을 보여서 기준치 이상의 성능을 유지했다. 이 시스템은 정상적인 밀착을 찾는 것이 목적이 아니고 흰곰팡이를 찾는 것이 목적이므로 곰팡이(Mold)를 탐지하는 AP를 평가 항목으로 사용하였다. 그림 3은 mold의 mAP가 0.714임을 보여준다.

그림 4는 본 연구 모델에서 다른 학습(Train) 데이터와 검증(Val) 데이터의 손실(Loss)값과 정밀도(Precision) 그리고 재현율(recall)의 변화를 그래프로 나타낸 것이다. 박스 손실(Box_loss) 값은 학습이 진행되는 동안 우하향으로 진행되고, 정밀도와 재현율은 우상향으로 진행되는 것으로 보아 학습이 정상적으로 진행되고 있음을 보여준다.

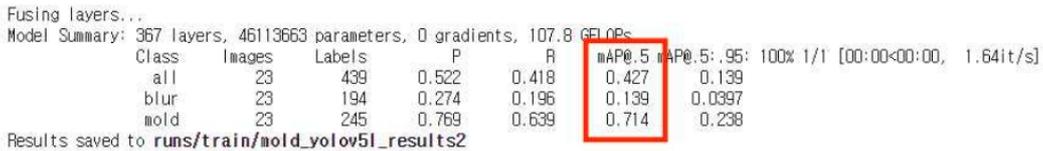


그림 3. 모델 실험 결과(mAP)
Fig. 3. Model experimental result(mAP)

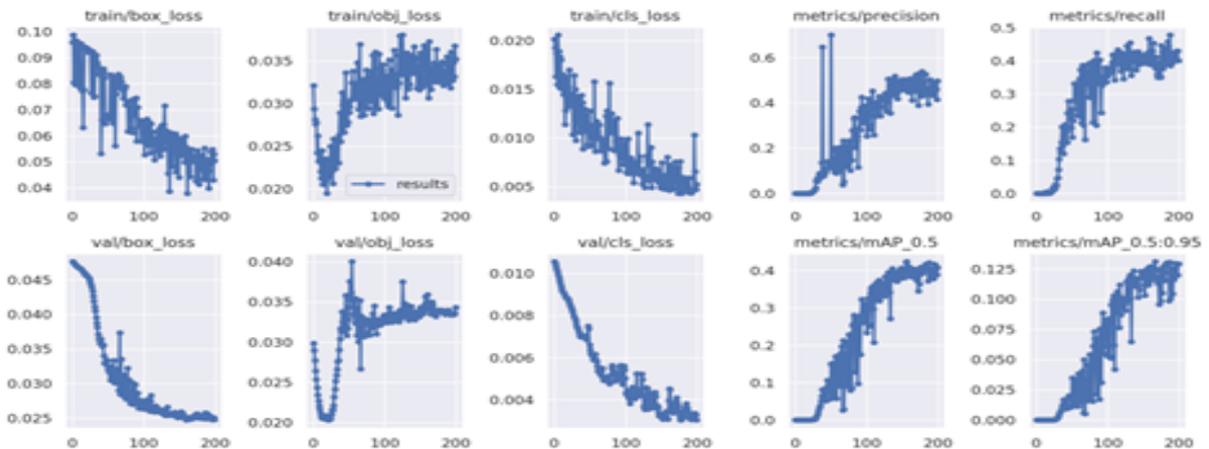


그림 4. 학습 시 손실과 mAP
Fig. 4. Training loss and mAP

V. 시스템 구성

5.1 실시간 흰곰팡이 탐지시스템

밀짚 재배기의 카메라를 통해 RTSP 프로토콜을 이용하여 실시간으로 AI 서버에 이미지가 전송되고 AI 서버는 흰곰팡이 탐지 모델에 이미지를 입력하여 흰곰팡이 객체를 탐지하여 경계상자를 표시한다.

AI 서버는 경계상자가 표시된 이미지를 MQTT 서버를 통해 자바 기반 웹 애플리케이션 서버 (WAS)에 전송하고 WAS는 이를 웹서비스를 통해 실시간으로 사용자의 화면에 보여준다.

이 시스템은 밀짚의 흰곰팡이를 24시간 탐지할 수 있으므로 흰곰팡이 발생 초기에 알려 확산을 최소화할 수 있다.

5.2 개발환경 및 실행환경

초당 30프레임 이상 처리하기 위해서는 고성능 GPU가 필요하다. 표 3는 AI 서버의 실행환경이다.

표 3. AI 서버 실행환경

Table 3. AI server runtime environment

AI server runtime environment	
CPU	Intel(R) Core(TM) i9-10900X CPU 3.70GHz
Memory	128GB
GPU	GeForce RTX 3090 Ti * 2
Storage	SSD 1TB
OS	linux Ubuntu 18.04.5 LTS

YOLOv5를 이용한 흰곰팡이 탐지 모델을 실시간 서비스 시스템에 적용하기 위해 그림 5의 시스템을 구성하였다.

표 4는 개발에 사용한 언어 및 도구들이다. 실시간 흰곰팡이를 탐지하는 서비스는 파이썬 기반 텐서플로우 라이브러리를 사용했으며, 웹 서비스를 제공하기 위해 자바를 사용하였다.

표 4. 개발 언어 및 도구

Table 4. Development language and tools

Language and tools		
Language	Python, JAVA, HTML, Javascript, CSS	
Dev. tools	Java	Java SE 1.8 전자정부프레임워크 3.9
	Python	Anaconda 2021.05 & 2021.11 PyCharm community edition 2021
	WAS, MQ	Apache-tomcat-9 ActiveMQ 5.16.3
Library	Python	OpenCV 4.5.4, YOLOv5, Tensorflow 2.7 paho-mqtt 1.6
	JavaScript	JQuery 3.6.0 paho-mqtt javascript client 1.0.3

그림 6은 실시간 흰곰팡이 탐지 서비스 실행 화면이다. 흰곰팡이가 피고 인식이 되면 흰곰팡이로 인식한 영역에 경계 상자와 흰곰팡이일 확률을 출력한다.

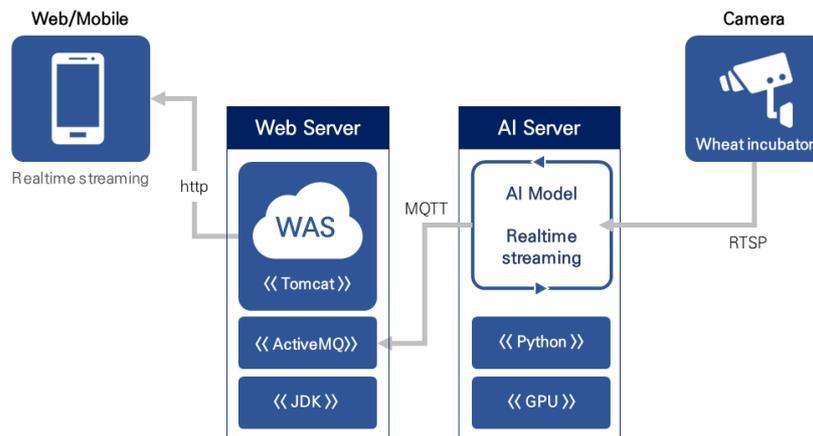


그림 5. 시스템 구성도

Fig. 5. System architecture



그림 6. 실시간 흰곰팡이 탐지 서비스
Fig. 6. Realtime white molds detection service

VI. 결론 및 향후 과제

본 연구는 기능성 식물 중 밀싹의 생산성 향상을 위해 인공지능을 이용한 흰 곰팡이 탐지 방법을 연구하고 시스템을 구축하였다. 탐지 모델로는 Faster R-CNN, EfficientNet, YOLO v3, YOLO v5 모델을 비교 테스트하였고 YOLOv5를 시스템의 모델 알고리즘으로 선정하였다.

모델 검증을 위해 IoU가 0.5 이상인 이미지 데이터셋에 대한 객체의 정확도를 실험하여 IoU 0.5 이상일 때 mAP 값이 0.714인 결과를 얻었고, 이 모델을 실시간 곰팡이 탐지시스템에 적용했고, 이를 통해 인공지능 모델이 농작물의 피해 여부를 실시간으로 확인하는 시스템을 구축하는데 사용할 수 있음을 확인했다.

곰팡이는 다른 병충해에 비해 피해 면적에 대한 정량적인 크기 기준이 없다. 그래서 어노테이션 크기의 영역을 어느 정도 크게 설정하느냐에 따라 학습 후 탐색하는 범위에도 영향을 준다. 이 논문에서는 곰팡이가 핀 영역을 모두 경계상자로 어노테이션 하였기 때문에 어노테이션한 경계상자의 크기가 다양하다.

향후 곰팡이병의 어노테이션 크기에 대한 기준을 정하고 같은 크기로 어노테이션하여 더 높은 정확도를 갖는 시스템이 필요하다.

본 연구를 통해 약용작물인 밀싹 재배에 있어 흰 곰팡이를 초기부터 탐지하여 진단 및 예방에 이바지할 수 있어 밀싹의 생산량 증대 효과를 가져올 수 있을 것으로 기대된다. 향후, 농가에서 쉽게 사용할 수 있도록 모바일 앱 서비스 시스템으로 제공할 계획이다.

References

- [1] KyeongHo Jo, SeHoon Jung, and ChunBo Sim, "A Study of Shiitake Disease and Pest Image Analysis based on Deep Learning", Journal of Korea Multimedia Society Vol. 23, No. 1, pp. 50-57, Jan. 2020. <https://doi.org/10.9717/kmms.2020.23.1.050>.
- [2] Na Myung Hwan, Cho Wanhyun, and Kim Sangkyoon, "A Construction of Web Application Platform for Detection and Identification of Various Diseases in Tomato Plants Using a Deep Learning Algorithm", Journal of Korean Society for Quality Management, Vol. 48, No. 4, pp. 581-596, Dec. 2020. <https://doi.org/10.7469/JKSQM.2020.48.4.581>.
- [3] J. S. Lee, S. K. Lee, D. W. Kim, S. J. Hong, and S. I. Yang, "Trends on Object Detection Techniques Based on Deep Learning", Electronics and Telecommunications Trends, Vol. 33, No 4, Aug. 2018. <https://doi.org/10.22648/ETRI.2018.J.330403>.
- [4] Ik-Su Kim, Moon Gu Lee, and Yongho Jeon, "Comparative Analysis of Defect Detection Using YOLO of Deep Learning", Journal of the Korean Society of Manufacturing Technology Engineers, Vol. 30, No. 6 pp. 514-519, Dec. 2021. <https://dx.doi.org/10.7735/ksmte.2021.30.6.513>.
- [5] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement", CVPR, ArXiv:1804.02767, Apr. 2018. <https://doi.org/10.48550/arXiv.1804.02767>.
- [6] Glenn-jocher, 2020, viewed 17 November 2021, YOLOv5, <https://github.com/ultralytics/yolov5> [Accessed: Jun. 28, 2022]
- [7] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese, "Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression", CVPR, arXiv:1902.09630v2, pp. 658-666, Apr. 2019.

[8] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression", CVPR, arXiv:1911.08287v1, Vol. 34, No. 7, Nov. 2019. <https://doi.org/10.1609/aaai.v34i07.6999>.

저자소개

허진경 (JinKyoung Heo)



2004년 2월 : 조선대학교

전산통계학과(공학박사)

2022년 9월 ~ 현재 : 연세대학교

산학협력단(공학연구원) 연구교수

관심분야 : 빅데이터, 인공지능,

인공지능경망, 영상처리, 객체탐지