

딤러닝 기반의 가금류 객체 탐지 알고리즘

명 현 정¹ · 송 주 환^{2*}¹전주대학교 Agro AI학과 석사과정^{2*}전주대학교 인공지능학과 교수

Deep Learning-based Poultry Object Detection Algorithm

Hyun-Jung Myung¹ · Ju-Whan Song^{2*}¹Master's Course, Department of Agro AI, Jeonju University, Jeonju 55069, Korea^{2*}Professor, Department of Artificial Intelligence, Jeonju University, Jeonju 55069, Korea

[요 약]

가금의 평균 체중을 구하기 위해서는 닭이 저울 위로 올라가면 측정된 무게 값 데이터를 쌓아 원시 데이터로 닭의 성장 그래프를 그려 이를 예측한다. 하지만 저울에 닭이 여러 마리가 올라가 무게에 편차가 발생하여 평균 체중 예측에 혼란을 준다. 저울의 고도화와 성능검증, 가금의 평균 체중 예측값을 더 정확하게 교정하기 위해 가금의 개체 수 측정하는 것을 목적으로 객체 인식 모델을 비교하는 실험을 진행한다. 연구 결과 Modified YOLOv5가 98.8%, YOLOv5가 98.5%, YOLOv4가 96%, Mask R-CNN가 90%의 정확도를 보였다. 속도는 Modified YOLOv5가 33분 32초, YOLOv5가 25분 40초, YOLOv4가 66분 67초, Mask-RCNN가 172분 8초가 걸렸다. YOLOv5가 가장 빠른 검출 속력을 보였지만 Modified YOLOv5가 가장 높은 정확도를 보여 닭 개체에 가장 정확한 객체 검출 모델임을 알 수 있었다. 향후 객체 인식 모델을 보완하여 좀 더 정밀한 가금의 개체 수를 측정하고 결과 오차에 대해서 개선할 계획이다.

[Abstract]

In order to obtain the average weight of a poultry, when the chicken goes up on the scale, it accumulates the measured weight value data and draws a growth graph of the chicken with raw data to predict it. However, several chickens rise on the scale, causing variations in weight, which confuses average weight prediction. Experiments are conducted to compare object recognition models with the aim of measuring the number of individuals to accurately correct the scale's advancement and performance verification, and the average weight prediction. As a result of the study, Modified YOLOv5 98.8%, YOLOv5 98.5%, YOLOv4 96%, and Mask R-CNN showed 90%. The speed took 33m 32s Modified YOLOv5, 25m 40s YOLOv5, 66m 67s YOLOv4, and 172m 8s for Mask-RCNN. Although YOLOv5 showed the fastest detection speed, Modified YOLOv5 showed the highest accuracy, indicating that it was the most accurate object detection model for chicken objects. In the future, we plan to supplement the object recognition model to measure the population of more precise poultry and improve it against the result error.

색인어 : 스마트팜, 딤러닝, 객체 검출, 영상 분석, 울로**Keyword** : Smart Farm, Deep Learning, Object Detection, Image Analysis, YOLO<http://dx.doi.org/10.9728/dcs.2022.23.7.1323>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 23 May 2022; Revised 04 June 2022

Accepted 12 July 2022

***Corresponding Author; Ju-Whan Song**

Tel: +82-63-220-2912

E-mail: jwsong@jj.ac.kr

1. 서론

세계적으로 기후 변화와 물 부족, 급속한 도시화로 인한 경작지 감소와 인구 고령화 등으로 예상되는 식량 수급 문제가 예상된다. 이에 대한 해결책으로 스마트팜이 제시되어 스마트팜의 도입이 증가세를 보이고 있다. 스마트팜은 농작물과 가축의 생육 정보와 환경정보 등에 대한 정확한 데이터를 기반으로 언제 어디서나 농작물과 가축의 생육환경을 점검하고, 적기 처방을 함으로써 노동력, 에너지, 양분 등을 종전보다 덜 투입하고도 농축산물의 생산성과 품질 제고가 가능한 농업을 말한다. 이를 구현하기 위해 IoT, 빅데이터, AI, 자동화 시스템 및 로봇 기술들을 시설원예, 축사, 과수원 등에 접목하여 농작물과 가축의 생육환경 유지, 관리를 원격 또는 자동으로 수행할 수 있는 지능화된 농장 형태이다[1]. 전 세계 스마트팜 시장은 연평균 9.8%의 시장 확대로 지속적인 산업 외형 증대를 보이고 있다. 2021년 전 세계 스마트팜 시장 규모는 148억 달러로 추정되며 앞으로 지속해서 증가하여 2025년에는 220억 달러로 연평균 9.8%로 시장이 확대될 전망이다. 각국의 스마트팜 기술혁신과 정책이 뒷받침되어 경쟁력이 향상되고 있는 만큼 스마트팜 시장의 성장세도 가파르게 증가하고 있다[2]. 현재 인공지능을 활용한 가금 중량측정 방법으로는 농촌진흥청이 (주)LG이노텍과 공동연구로 육계의 실시간 영상 이미지를 활용해 육계의 체중을 예측하는 기술이 있다[3]. 육계 체중 예측 기술은 카메라 영상으로 관측한 육계의 크기(면적, pixel)에 대한 정보를 활용해 계군의 평균 체중을 예측할 수 있도록 하는 방식이다. (주)이모션은 육계 농장 내 중량측정 저울을 설치하여 닭이 저울 위로 올라가면 자동으로 저울에 측정된 무게 값이 데이터에 쌓여 닭의 성장 그래프를 통해 가금의 평균 체중을 예측한다. 하지만 다음 그림 1[4]에서 볼 수 있듯이 저울의 무게에 많은 편차가 발생하여 평균 체중 예측에 혼란을 주는 문제가 있다. 우리 연구실에서는 기업과 함께 무게의 정밀도를 높이는 연구를 진행하고 있었다. 문제의 원인을 파악하기 위해 농장에 IP카메라 설치하여 저울에 닭이 여러 마리가 올라가 편차가 발생한 것을 인식하였다. 저울의 고도화와 성능검증, 가금의 평균 체중 예측값을 더 정확하게 교정하기 위해 IP카메라를 설치하여 데이터를 받아 실험을 진행한다. 가금의 개체 수 측정을 자동화시켜 닭이 몇 마리인지 파악해보려 한다. 그래서 본 논문은 인공지능을 통한 객체 탐지를 이용하여 정밀하게 가금의 수를 측정하는 것이 목표이다. 농장에 설치한 IP카메라를 통해 얻은 영상 데이터를 이용하여 닭의 개체 수 측정을 진행한다. 닭의 이미지를 객체 탐지 알고리즘에 적용하여 정확한 개체 수를 측정하는 알고리즘을 찾는다. 이렇게 정확한 육계의 체중을 측정하는 이유는 육계 농가의 매출과 직결된 중요한 문제이기 때문인데 닭의 무게에 따라 경제적 손실이 날수도 이익을 볼 수도 있기 때문이다. 육계 업체는 육계 농가와 출하 무게에 따른 금전적 계약을 맺고 있다.

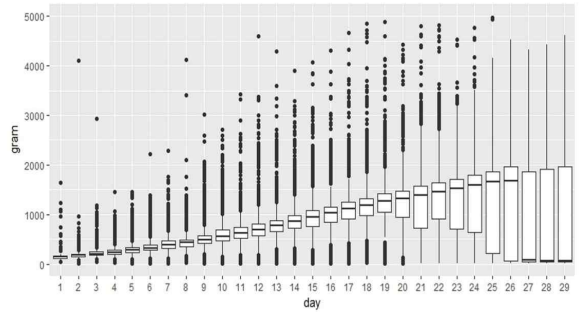


그림 1. 닭의 무게 원시 데이터 그래프
Fig. 1. Chicken Weight Raw Data Graph

출하 시 육계 업체에서 요구한 닭의 무게가 오차범위(50g) 이하로 관리되어 있어 만족하는 조건에 출하하면 인센티브, 조건을 벗어나면 페널티가 부과된다. 그래서 농장주는 이익을 얻기 위해 가금의 정확한 무게를 파악하길 원한다. 또한 하루에 일일이 수백 마리의 닭의 무게를 측정해야 하므로 소요되는 인력과 시간, 가금류의 스트레스가 증가하여 성장과 품질 저하의 문제 및 조류 인플루엔자의 전염 확산 가능성에 대한 문제들을 해결하여 노동력 절감, 품질 향상 등 다양한 긍정적인 효과를 제공한다. 실시간으로 중량이 낮거나 높으면 문제가 생긴 것으로 판단할 수 있어 닭이 건강 상태를 파악하는 역할도 가능하여 생육 관제를 중량으로 할 수 있다. 본 연구의 구성은 다음과 같다. 2장에서는 관련된 선행 연구를 정리하고, 3장에서는 가금의 탐지를 위해 본 연구에서 진행될 시스템 설계와 구성을 설명한다. 4장에서 실험 환경과 3개의 알고리즘을 적용하고 실험 결과를 활용하여 가금 개체 검출 시스템에 가장 정확한 객체 검출 알고리즘을 설명한다. 마지막으로 5장에서 결론과 추후 연구에 대해 제시한다.

II. 관련 연구

2-1 객체 검출

객체 검출(Object Detection)은 컴퓨터 비전과 이미지 처리와 관련된 컴퓨터 기술로서, 디지털 이미지와 비디오로 특정한 계열의 시맨틱 객체 인스턴스(예: 인간, 건물, 자동차)를 감지하는 일을 다룬다[5].

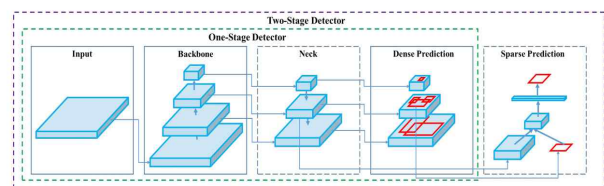


그림 2. 객체 검출
Fig. 2. Object Detector

객체 검출(Object Detection)은 분류(Classification)와 지역화(Localization) 과정을 거쳐 물체를 인식하는데 여기서 분류는 바운딩 박스(Bounding Box) 안의 물체가 무엇인지를 판단하고 지역화는 물체의 정확한 위치를 예측하는 것이다. 객체 검출의 과정에서 분류와 지역화 작업을 어떻게 처리하느냐에 따라 다음의 그림 2와 같이 두 가지 방식으로 나뉜다. One-stage Detector는 분류와 지역화 작업을 한 번에 처리하며 여기에는 YOLO[6], SSD[7], RetinaNet[8] 등이 속한다. Two-stage Detector는 분류와 지역화 작업을 나눠서 처리하며 여기에는 R-FCN[9], Faster R-CNN[10], Mask R-CNN[11] 등을 포함한다. One-stage Detector는 검출 속력이 빠르지만, 속도와 비교해 정확도가 떨어지고 Two-stage Detector는 속도는 느리지만 정확도가 높다.

2-2 YOLO(You Only Look Once)

YOLO는 객체 검출(Object Detection)을 위한 딥러닝 기반의 지도학습인 알고리즘 중 하나이다. You Only Look Once의 줄임말로 말 그대로 단 한 번만 본다는 의미를 지니며 이미지를 단일 회귀로 물체의 분류와 위치를 판단한다. YOLO는 CNN 구조를 기반으로 하며 네트워크 구조(network architecture)는 GoogLeNet 모델을 기반으로 하며 24개의 Convolutional layers와 2개의 Fully Connected layers로 구성되어 있다. 다음의 그림 3은 YOLO의 구조를 나타낸 것이다. YOLO로 이미지를 처리하는 방법은 입력 이미지의 크기를 조정하고, 이미지에서 컨볼루션 네트워크를 실행하고, 모델의 신뢰도에 따라 결과 탐지를 임계값화 한다. 바운딩 박스는 x, y, w, h 및 Confidence Score까지 5개의 요소로 구성된다. (x, y)는 그리드 셀의 경계를 기준으로 바운딩 박스의 중심 좌표이다. (w, h)는 바운딩 박스의 너비와 높이를 뜻한다. Confidence Score 은 예측한 바운딩 박스와 모든 정답 바운딩 박스 간의 IOU를 나타낸다. 각 그리드 셀은 또한 C 조건부 클래스 확률, $\Pr(\text{Class}|\text{Object})$ 을 예측한다. 이러한 확률은 객체가 포함된 그리드 셀에서 조건화된다. 바운딩 박스 B의 수와 관계없이 그리드 셀당 클래스 확률의 한 세트만 예측한다. 테스트 시간에는 각 상자에 대한 클래스별 신뢰 점수를 제공하는 조건부 클래스 확률을 곱한다. Confidence Score은 (1)의 식으로 나타낸다.

$$\Pr(\text{Class}_i|\text{Object}) * \Pr(\text{Object}) * IOU_{pred}^{truth} = \Pr(\text{Class}_i) * IOU_{pred}^{truth} \quad (1)$$

이 점수는 해당 클래스가 상자에 나타날 확률과 예측 상자가 개체에 얼마나 잘 적합한지 인코딩한다. YOLO의 시스템은 탐지를 회귀 문제로 모델링 한다. 이미지를 $S \times S$ 그리드로 나누고 각 그리드 셀에 대해 B 경계 상자, 해당 상자에 대한 신뢰도 및 C 클래스 확률을 예측한다. 이러한 예측은 $S \times S \times (B \leq 5 + C)$ 텐서로 인코딩된다[6]. 현재까지 YOLO는 5가지 버전이 있으며 각각 버전에 따라 차이가 있다.

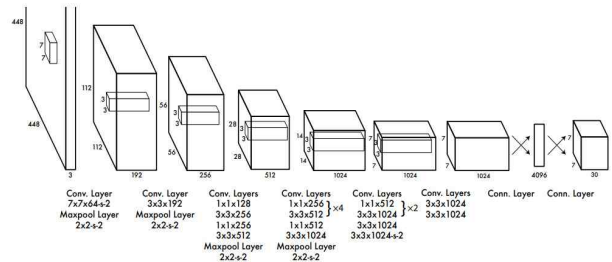


그림 3. YOLO 구조
Fig. 3. YOLO Structure

YOLOv1은 검출 속도가 빠르지만, 정확도가 낮다는 문제가 있었고 YOLOv2[12]는 Darknet19를 백본 아키텍처로 사용하여 검출 속도와 성능이 개선되었다. YOLOv3[13]은 Darknet53을 기반으로 하여 빠른 검출 속도를 가졌지만, 정확도가 낮았고 YOLOv4[14]는 CSPNet 기반으로 설계하여 전 버전보다 AP, FPS가 최적화되었다. YOLOv5는 전 버전인 YOLOv4와 비교하여 CPS-Darknet을 백본으로 사용하기 때문에 성능은 비슷하지만 낮은 용량을 가지고 있고 Pytorch로 구현하여 빠른 속도를 가지고 있다. YOLOv5는 가지 종류의 백본을 가지고 있는데 YOLOv5-s, m, l, x이며 차례대로 작고 가벼운 백본부터 가장 크고 무거운 백본 순이다. 본 논문에서는 YOLO의 다섯 번째 버전인 YOLOv5를 이용하여 연구를 진행한다. 속도가 빠르고 실시간 검출 가능하며 정확도가 많이 개선되어 성능 또한 준수하다.

2-3 가금 분석

닭 도체의 중량규격은 닭 도체의 무게에 따라 해당 호수와 병행하여 닭 도체의 중량규격 구분기준 및 호수별 중량 범위는 특대 (15호 이상) 1,451g 이상, 대 (13~14호) 1,251g 이상 ~ 1,450g 이하, 중 (10~12호) 951g 이상 ~ 1,250g 이하, 중소 (7~9호) 651g 이상 ~ 950g 이하, 소 (6호 이하) 451g 이상 ~ 650g 이하의 5개 규격으로 구분한다[15]. 이처럼 양계 농가에서는 가금의 호수에 따라 최적의 출하 시기를 맞춰야 하는데 무게에 따라 사육하는 기간의 편차가 크지 않아 가금의 무게를 수시로 측정하는 작업을 수행해야만 하며 출하 시에는 하루에 수백 마리의 닭을 저울에 올려 일일이 측정해야 한다. 하지만 복잡한 방역 문제로 다른 사람의 도움을 받기 힘들어 시간이 많이 소요되고 가금류의 스트레스가 증가하여 성장과 품질 저하의 문제를 일으키며 꾸준히 발생하는 조류 인플루엔자의 전염을 확산시킬 가능성이 높아지는 이유가 되기도 한다. 또한 이렇게 측정된 체중은 예상한 평균 체중의 정확도가 떨어져 오차가 발생한다. 현재 하림 동우팜투데이, 마니커, 체리부로, 한강식품과 같은 육계 업체들은 패널티와 인센티브를 통해 육계 무게를 관리한다. 농가와 업체 간의 표준계약서에서 육계 수요처들이 출하 전과 출하할 때 무게가 오차범위(50g) 이하로 관리하게

되어있는데 만족하는 조건에 출하 시 인센티브가 주어지고 조건을 벗어나면 패널티가 부과된다. 이외에도 산란계 농가 또한 산란을 최적화를 위한 체중조절이 필수적인데 체중이 미달이거나 초과일 경우 산란율과 달걀 중량의 감소 하며 사료 효율이 떨어진다. 따라서 정확한 가금의 체중 측정은 농가의 매출과 직결된 중요한 문제이다. 이러한 문제로 인해 가금의 정밀한 생체중량 자동측정의 필요성이 제기되었다. 하지만 닭을 저울에 일일이 올려 무게를 확인하기에는 큰 노동력 손실과 정확도가 떨어진다. 이를 위해 계사 안에 중량측정기를 설치하여 닭이 스스로 중량측정기에 올라가면 자동으로 측정된 값을 서버에 저장하는 방식으로 무게를 측정한다. 그렇지만 중량기 위에 닭이 한 마리만 올라가지 않고 여러 마리가 올라갈 경우 많은 편차가 발생하여 평균 체중 예측에 혼란을 줄 수 있다. 그래서 저울의 고도화와 성능검증, 가금의 평균 체중 예측값을 더 정확하게 교정하기 위해 IP카메라를 설치하여 데이터를 받아 실험을 진행한다. 가금의 객체를 탐지하고 인식하여 개체 수를 정밀하게 파악하고 YOLO를 사용하여 가금의 객체 수를 탐지하는 연구를 하고자 한다.

III. 시스템 설계 및 구성

3-1 시스템 설계

육계 농가의 IP카메라를 통해 닭의 영상을 촬영하여 서버에 저장되면 서버에 저장된 이미지를 효율적으로 처리하기 위해 데이터 가공을 거쳐 다양한 시간대로 이미지를 무작위 선정한다. 선정된 이미지들은 라벨링 작업을 통해 위치 정보를 얻어 개체의 종류와 위치를 예측할 수 있도록 한다. 총 이미지 1,000장에서 학습 데이터 80% 테스트 데이터 15% 검증 데이터 5%로 구성하여

데이터셋을 구축한다. YOLOv5 알고리즘을 이용하여 객체 탐지를 하기 위해 준비한 데이터셋을 학습시킨다. YOLO에서 Convolution Layer와 Max-Pooling을 통해 이미지 내의 검출할 가금의 특징을 찾아내는 Feature Map을 추출한다. 이렇게 나온 결과는 개체를 분류하는데 사용한다. Fully Connected Layer에서는 클래스 확률과 Bounding Box의 좌표를 예측한다. 최적의 바운딩 박스를 찾는 비최대억제(Non-maximum Suppression)를 통해 검출기(Object Detector)가 예측한 많은 박스 중 detect 확률을 통해 가장 정확한 위치와 크기의 바운딩 박스 하나를 예측한다. 이렇게 훈련된 데이터를 통해 검증 데이터의 개체 분류와 위치를 예측해 객체를 검출할 수 있다. 다음 그림 4는 가금류의 검출을 목표로 한 전체 시스템 설계이다.

3-2 데이터셋 구성

이 연구에서 사용하기 위해 육계 농장의 IP카메라에서 영상 데이터를 활용한다. 영상 데이터는 IP카메라가 2880x1620 해상도, 24fps로 생성하며 1프스 기간 동안 IP 카메라 3대에서 수집한 영상의 데이터 규모이다. 여기서 프스는 육계의 입추에서 출하까지 1주기이며 약 32일이다. 1프스 동안의 영상 3세트를 확보하였으며 각 영상을 1일 구간으로 나누어 사용한다. 카메라 1개의 영상 중 5일령에 해당하는 하루 데이터를 사용하며 영상에서 약 60초 간격으로 사진을 추출하여 1,566장의 이미지를 얻었고, 1분 간격으로 사진을 생성하여 1일 2,880장씩 30일이면 86,000장의 이미지가 생성된다. 용량은 약 4TB 정도로 중량측정기를 통한 무게 정보와 IP카메라 영상 정보가 서버에 저장된다. 각각 따로 저장된 영상과 무게를 영상에서 생성한 프레임 이미지의 이름에 무게를 동기화시켜 저장한다. 이렇게 촬영된 영상을 통해 서버에 저장된 이미지 데이터를 얻어 데이터 중 일부를 선정하여 데이터셋을 구성한다. 선정된 데이터는 1프스 기간 중 시간대를 다양하게 1,000장을 무작위 선정한다. 객체 검출 전 makesense.ai 사이트에서 선정한 이미지 데이터의 라벨링 작업을 수행하여 객체의 위치 데이터를 얻는다[16]. 본 논문은 선정된 1,000장의 이미지를 학습 데이터 80%, 테스트 데이터 15%, 검증 데이터 5%로 구성하여 데이터셋을 구축한다. 클래스는 개수는 1개이며 클래스의 이름은 chicken으로 설정한다.

3-3 YOLO 훈련 및 Detection

YOLO는 보통 객체 검출기의 기본 구조인 Head와 Backbone으로 구성되어 있다. Head는 Backbone에서 나온 추출한 결과를 바탕으로 물체의 위치를 찾고 Backbone은 통해 이미지의 Feature Map을 추출하여 분류를 할 수 있다. 본 연구에 사용되는 YOLOv5-s의 백본을 조금 수정하여 객체를 검출하는데 yaml 파일의 Depth Multiple와 Width Multiple 값을 조금 더 높인다. YOLOv5의 s, m, l, x 4개의 백본 중 s

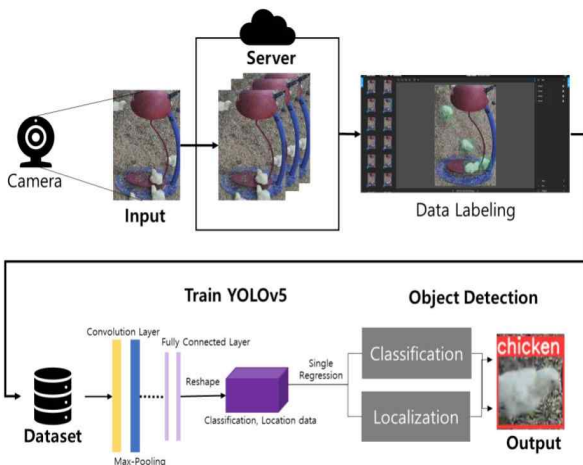


그림 4. 가금 검출 시스템 설계
Fig. 4. Poultry Detection System Design

가 Depth Multiple, Width Multiple 값이 0.33, 0.50으로 가장 작다. Depth Multiple 값이 클수록 더 깊은 모델이 되고 Width Multiple 값이 클수록 레이어 필터 수가 증가한다. 기존 YOLOv5 모델이 학습한 결과도 비교 모델 중 가장 높은 정확도가 나왔지만 기존 모델이 학습한 결과보다 정확도가 조금 더 증가 하는 것을 볼 수 있었다. 하지만 Depth Multiple과 Width Multiple 값을 너무 크게 하면 크고 무거워지기 때문에 속도가 느려지고 손실이 발생할 수 있어 YOLOv5-m 보다는 작은 값으로 설정했다. Depth Multiple과 Width Multiple 값을 수정한 YOLOv5 모델을 사용하여 데이터셋을 딥러닝 기반의 합성곱 신경망 객체 검출 알고리즘에 적용하여 객체 인식을 수행할 수 있도록 학습한다. 24개의 Convolutional layers와 Max-Pooling이 쌍으로 이루어져 있는 층과 Fully Connected layers를 통과하면 2차원 배열 형태의 이미지를 1차원 배열로 평탄화하고 다시 1차원 배열에서 3차원의 데이터로 바꾼다. 데이터의 클래스 이름과 위치를 이용하여 단일 회귀하여 분류와 물체의 위치를 예측하는 작업을 단일 회귀한다. 물체를 분류하여 클래스 이름과 물체의 위치를 찾는다. 이렇게 훈련된 데이터를 통해 검증 데이터의 개체 분류와 위치를 예측해 객체를 검출할 수 있다.

IV. 구현 및 결과

4-1 실험환경

본 연구는 양계에 설치된 IP카메라로부터 영상 이미지를 받아 데이터를 구축하였다. IP카메라는 2880x1620 해상도와 24fps를 제공한다. 중량을 측정하는 데에 사용한 스마트 저울은 LO-1000 모델을 사용한다. 연구에 사용된 장비들은 다음 표 1과 같다. OS는 Windows 10 환경이며 실험은 google colab에서 작업했고, Python 3.7.13 버전을 기반으로 하였다. GPU를 이용하기 위해 CRUD 버전 11.1을 활용 하였고 파이토치 1.10.0 버전을 사용한다. 객체 검출을 위해 다양한 객체 검출 모델 중 YOLO의 5번째 모델인 YOLOv5를 사용한다. 라벨링 작업은 makesense.ai 사이트에서 수행하였다.

표 1. 실험환경

Table 1. Experiment Environment

| Equipment | Model |
|--------------------|--|
| Processor | Intel(R) Core(TM) i5-8250U CPU @ 1.60Hz 1.80GHz |
| RAM | 8GB |
| OS | Windows 10 |
| IP Camera | 2880x1620 Resolution, 24fps |
| Smart Weight Scale | LO-1000 |

4-2 평가지표

객체 검출 알고리즘 성능을 mAP(mean Average Precision)로 평가한다. mAP를 구하기 위해서는 Precision-Recall 곡선과 Average Precision(AP)가 필요하다. 우선 Precision과 Recall을 알아보기 위해서는 TP, FP, FN, TN에 대해 먼저 파악하여야 하는데, 이해를 돕기 위하여 표 2로 정리했다. 각각 TP는 옳은 검출, FP는 검출되어야 할 것이 검출되지 않음, FP 틀린 검출, TN 검출되지 말아야 할 것이 검출되지 않았다는 의미이다. Precision은 정밀도이며 모델이 옳다고 검출한 결과 중에서 실제로 옳은 검출을 한 비율이다. 그러니까 답이라고 예측한 결과에서 실제로 답인 경우의 비율이다. Precision은 다음의 식(2)으로 나타낼 수 있다.

$$Precision = \frac{TP}{TP + FP} = \frac{TP}{AllDetection} \quad (2)$$

Recall은 재현율이며 실제로 옳은 값에서 모델이 옳다고 검출한 비율이다. 그러니까 실제 정답의 답 수 중에서 모델이 답이라고 예측한 비율이다. Recall은 다음의 식(3)으로 나타낼 수 있다.

$$Recall = \frac{TP}{TP + FN} = \frac{TP}{AllGround Truth} \quad (3)$$

Precision-Recall 곡선은 x값을 Recall, y값을 Precision으로 그래프로 나타낸 것인데 Precision와 Recall 두 메트릭의 균형을 캡슐화하며 최대한의 효과를 볼 수 있고 알고리즘의 전반적인 성능을 확인할 수 있다. PR 곡선은 recall 값에 따라 변화하는 Precision 값을 볼 수 있다. 하지만 Precision-Recall 곡선 그래프로는 어느 한 값으로 알고리즘의 성능을 판단하기에는 불가능하다. 모델의 정밀도는 높지만, 재현율이 낮으면 객체를 옳다고 분류할 때 모델이 정확하지만 옳은 예측을 한 결과 중 일부만 분류할 수도 있다. 따라서 답이라고 분류할 때 정확하지만 답인데도 답이 아니라 판단할 가능성이 크다. 모델의 재현율은 높지만, 정밀도가 낮으면 모델은 대부분의 옳은 예측을 한 객체들을 잘 분류하지만 많은 옳지 않은 객체를 옳다고 분류한다.

표 2. 혼돈 행렬

Table 2. Confusion Matrix

| Ground truth | Predict Result | |
|--------------|------------------------|------------------------|
| | Positive | negative |
| Positive | TP (true positive) | FN (false negative) |
| Negative | FP (false positive) | TN (true negative) |

따라서 답이라고 잘 예측해서 답이 아닌 경우에도 답이라고 할 가능성이 크다. 그래서 두 값을 종합하여 알고리즘을 평가하기 위해 AP(Average Precision)을 사용해야 한다. AP는 알고리즘의 성능을 하나의 값으로 표현하며 Precision-Recall 그래프 아래의 면적이다. mAP(mean Average Precision)는 각각의 클래스에 대한 AP의 평균이다. 다음 식(4)는 mean Average Precision를 구하는 식으로 Object Detection의 평가지표가 된다.

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \tag{4}$$

4-4 연구 결과

1,000장의 이미지 데이터셋을 확보하여 학습, 검증, 시험으로 분류하여 실험을 진행한다. 학습 데이터 80% 테스트 데이터 15% 검증 데이터 5%로 구성되어 있다. 다음 표 3에 데이터셋 구성이 정리되어 있다. 시험용 데이터는 학습 후 모델 평가에 사용한다. Batch size, Epoch, 이미지 크기는 각 모델에 가장 적합한 값으로 설정하여 가장 정확도가 높게 나온 경우로 평가한다. 성능 평가를 비교할 모델은 YOLOv5를 수정한 Modified YOLOv5, YOLOv5, YOLOv4, Mask R-CNN 네 객체 탐지 모델이다. 가장 최근에 나온 YOLO 모델이며 속도가 빠른 YOLOv5, 성능이 YOLOv5와 비슷하며 FPS 성능이 준수한 YOLOv4를 1-stage Detector 중 비교 모델로 선정하였다. 2-stage Detector 중에서는 Mask R-CNN이 최신의 모델이며 정확도가 준수하여 선정하게 되었다. 따라서 세 모델이 비교 모델로서 적합하다고 판단하여 정확도 및 속도를 비교한다. Precision, Recall, mAP, Time 을 성능검증 지표로 사용한다. 각각 네 모델은 mAP가 Modified YOLOv5는 0.988, YOLOv5는 0.985, YOLOv4가 0.96, MasK R-CNN이 0.9로 Modified YOLOv5가 가장 높았고, 그다음 YOLOv5, YOLOv4, Mask R-CNN 순이었다. Precision는 Modified YOLOv5가 0.98, YOLOv5가 0.939, YOLOv4가 0.79, Mask R-CNN이 0.667로 Modified YOLOv5가 가장 Precision 값이 높았다. Recall은 Modified YOLOv5가 0.953 YOLOv5가 0.977, YOLOv4가 0.97, Mask R-CNN이 0.667로 YOLOv5가 Recall 값이 가장 높았다. 속도는 25분 40초로 YOLOv5가 빠른 속도를 보여 가장 빠른 시간 내에 처리 할 수 있었고 Modified YOLOv5는 33분 32초, YOLOv4는 66분 67초, Mask-RCNN은 172분 8초가 걸렸다.

표 3. 데이터셋 구성
Table 3. Dataset Configuration

| Data | Training | Test | Validation |
|-------|----------|------|------------|
| Count | 800 | 150 | 50 |

표 4. 알고리즘 성능 비교 결과

Table 4. Algorithm Performance Comparison Results

| | Time(ms) | Precision | Recall | mAP |
|-----------------|----------|-----------|--------|-------|
| Modified YOLOv5 | 33m32s | 0.98 | 0.953 | 0.988 |
| YOLOv5 | 25m40s | 0.939 | 0.977 | 0.985 |
| YOLOv4 | 66m67s | 0.79 | 0.97 | 0.96 |
| Mask R-CNN | 172m8s | 0.667 | 0.667 | 0.9 |

다음 표 4를 통해 정리된 알고리즘 성능 비교 결과를 볼 수 있다. 실험 결과로 미루어 보아 YOLOv5가 가끔의 객체 검출 알고리즘 중 가장 적합한 것을 알 수 있다. YOLOv5가 압도적으로 빠른 속력과 가장 높은 Recall 값을 가지고 Modified YOLOv5가 가장 높은 Precision, mAP 결과값을 가졌다. 성능 지표 결과에서 기존의 YOLOv5가 Modified YOLOv5보다 7분 58초 느리지만, 정확도가 가장 높게 나와 가장 뛰어난 성능을 보였다고 할 수 있다. 이로써 Modified YOLOv5가 답 개체에 가장 정확한 객체 검출 모델임을 알 수 있었다. 다음 그림 5는 Modified YOLOv5의 mAP 그래프이다. 그림 6은 Modified YOLOv5의 Precision, Recall 그래프이다. 훈련을 거듭할수록 값이 점점 증가해 1에 수렴하여 높은 정확도를 보여준다. 그림 7은 Modified YOLOv5의 손실 그래프이며 훈련 횟수가 증가함에 따라 loss 값이 점점 0에 수렴하고 시스템 정확성과 신뢰성이 기대한 수준에 도달해 훈련이 완료된 것을 알 수 있다. 그림 8은 YOLOv5, YOLOv4, MASK R-CNN 순서대로 각각 세 알고리즘에서 가끔을 검출한 결과 이미지이다. 비슷한 이미지에서 YOLOv4와 MASK R-CNN은 탐지하지 못한 객체가 있는 것을 볼 수 있다. 그림 9는 Modify YOLOv5에서 검출한 이미지로 물체가 겹쳐있거나 가려져 있어도 이미지 내 객체가 모두 검출되었다.

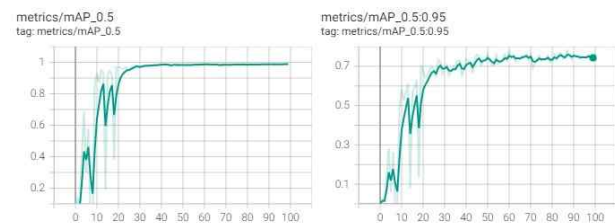


그림 5. YOLOv5 mAP 그래프
Fig. 5. YOLOv5 mAP Graph

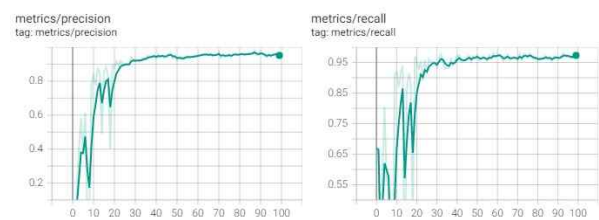


그림 6. YOLOv5 Precision Recall 그래프
Fig. 6. YOLOv5 Precision Recall Graph

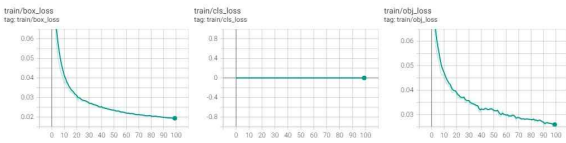


그림 7. YOLOv5 손실 그래프
Fig. 7. YOLOv5 Loss Graph

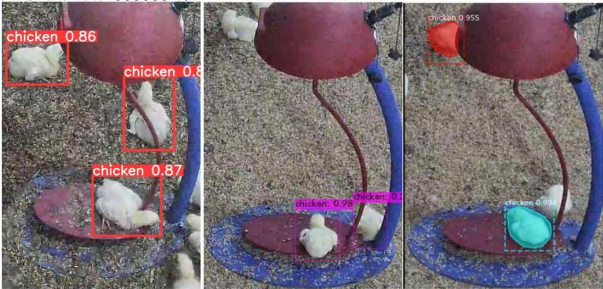


그림 8. YOLOv5, YOLOv4, MASK R-CNN 검출 이미지
Fig. 8. YOLOv5, YOLOv4, MASK R-CNN Detection images

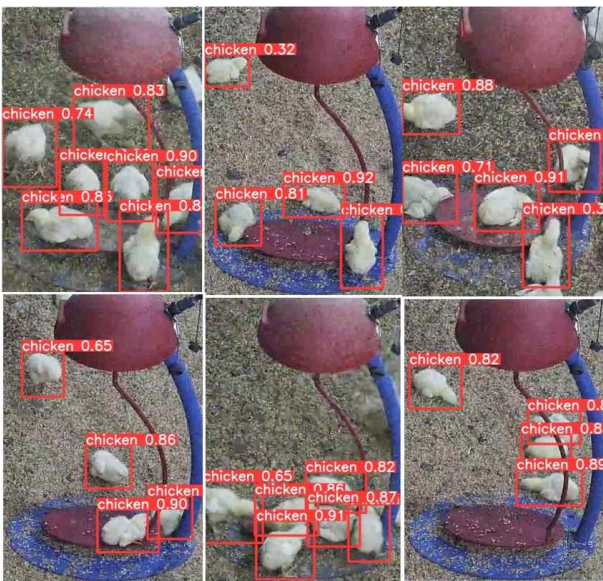


그림 9. Modify YOLOv5 검출 이미지
Fig. 9. Modify YOLOv5 Detection images

V. 결론 및 추후 연구

양계 농가는 육계 업체와 출하 무게에 따른 금전적 계약을 맺고 있는데 무게의 오차범위에 따른 인센티브와 패널티를 부과하여 정확한 가금의 중량측정이 농가의 금전적인 부분에 큰 영향을 미친다. 그렇지만 수백 마리의 닭을 일일이 저울에 올려 측정하는 일은 많은 인력과 시간이 소요되고 닭의 스트레스 증가로 품질 저하의 문제와 조류 인플루엔자의 전염 확산 가능성을 높이는 요인이 된다. 하지만 닭을 저울에 일일이 올려 무게를 확인하기에는 큰 노동력 손실과 정확도가 떨어진다.

이를 위해 육계 농장 내 중량측정 저울을 설치하여 닭이 저울 위로 올라가면 자동으로 저울에 측정된 무게 값이 데이터에 쌓여 닭의 성장 그래프를 통해 가금의 평균 체중을 예측한다. 그렇지만 중량기 위에 닭이 한 마리만 올라가지 않고 여러 마리가 올라갈 시 많은 편차가 발생하여 평균 체중 예측에 혼란을 줄 수 있다. 그래서 저울의 고도화와 성능검증, 가금의 평균 체중 예측값을 더 정확하게 교정하기 위해 IP카메라를 설치하여 데이터를 받아 실험을 진행한다. 가금의 객체를 탐지하고 인식하여 개체 수를 정밀하게 파악하는 것이 목적으로 가금 중량측정 업무 지원을 위한 시간 단축과 최소 인력으로 자동화된 가금 개체 검출방식을 비교 분석을 시행한다. 본 연구에서는 가금의 객체 인식을 위하여 딥러닝 기반의 합성곱 신경망 객체 검출 알고리즘에서 1-stage Detector 객체 검출방식인 YOLOv5, YOLOv4 모델과 2-stage Detector 객체 검출방식인 MASK R-CNN 3가지 알고리즘을 선정하였다. YOLOv5는 모델에 수정을 거쳐 Modified YOLOv5도 비교 모델에 포함한다. 양계 농가에 IP카메라를 설치하여 가금 모니터링 자료를 수집하였으며 가금에 대한 총 1,000장의 학습 데이터를 구축하였다. 이 중 학습 데이터 구축 비율을 학습 데이터 80%, 테스트 데이터 15%, 검증 데이터 5%로 나누고 딥러닝 알고리즘 학습을 각각 수행하였으며 테스트 데이터셋을 통해 그 결과를 도출하였다. 연구 결과 Modified YOLOv5가 33분 32초, YOLOv5가 25분 40초, YOLOv4가 66분 67초, Mask R-CNN이 172분 8초가 걸렸으며 Time 부분에서 압도적인 차이를 보였다. Modified YOLOv5는 98.8%, YOLOv5 모델은 98.5%의 정확도를 보였으며 YOLOv4 모델은 96%, MASK R-CNN 모델은 90%의 정확도를 보였다. 이러한 연구 결과로 Modified YOLOv5가 닭 개체에 가장 정확한 객체 검출 모델임을 알 수 있었다. 향후 객체 인식 모델을 보완하여 좀 더 정밀한 가금의 개체 수를 측정하고 결과 오차에 대해서 개선할 계획이며 이 연구에 이어 동영상에 적용하여 닭의 개체 수를 자동으로 찾는 부분도 연구 진행 중이다.

참고문헌

- [1] Science and Technology Job Promotion Agency. Smart Farm Technology and Market Trend Report [Internet]. Available: https://www.compa.re.kr/cop/bbs/selectBoardArticle.do?bbid=BBSMSTR_00000000042&nttid=1980&bbsTyCode=BBST01&bbsAttrbCode=BBSA03&authFlag=&pageIndex=1&searchCnd=&searchWrd=&goMenuNo=3010000
- [2] S.Y.Kim, Science & ICT Policy and Technology Trend, Ministry of Science and ICT, No. 200, October 2021
- [3] RURAL DEVELOPMENT ADMINISTRATION. See the chicken weight! Development of broiler weight prediction

technology [Internet]. Available:

https://www.rda.go.kr/board/board.do?mode=view&prgId=day_farmprmninfoEntry&dataNo=100000764525#url

[4] C.Y. Wang, Y.J. Chen, and C.F. Chien, "Industry 3.5 to empower smart production for poultry farming and an empirical study for broiler live weight prediction." *Computers & Industrial Engineering*, Vol. 151, January 2021. <https://doi.org/10.1016/j.cie.2020.106931>

[5] S. Dasiopoulou, V. Mezaris, I. Kompatsiaris, V. K. Papastathis, and M. G. Strintzis, "Knowledge-Assisted Semantic Video Object Detection," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 15, No. 10, pp. 1210-1224, October 2005. <https://doi.org/10.1109/TCSVT.2005.854238>

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779-788, May 2016. <https://doi.org/10.1109/CVPR.2016.91>

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg. SSD: Single shot multibox detector. in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 21-37, 2016.

[8] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Vol.42, No. 2, pp. 2980-2988, July 2018. <https://doi.org/10.1109/TPAMI.2018.2858826>

[9] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Advances in Neural Information Processing Systems(NIPS)*, pp. 379-387, 2016. <https://doi.org/10.48550/arXiv.1605.06409>

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems (NIPS)*, pp. 91-99, 2015. <https://dl.acm.org/doi/10.5555/2969239.2969250>

[11] K. He, G. Gkioxari, P. Dollar, and R. Girshick, Mask R-CNN, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2961-2969, 2017. <https://doi.org/10.1109/ICCV.2017.322>

[12] J. Redmon and A. Farhadi, YOLO9000: better, faster, stronger, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7263-7271, 2017. <https://doi.org/10.1109/CVPR.2017.690>

[13] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018. <https://doi.org/10.48550/arXiv.1804.02767>

[14] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv preprint arXiv:2004.10934*, 2020. <https://doi.org/10.48550/arXiv.2004.10934>

[15] Korea Institute for Animal Products Quality Evaluation. Chicken grading standards [Internet]. Available: <https://www.koreascience.or.kr/article/JAKO200272540190214.page>

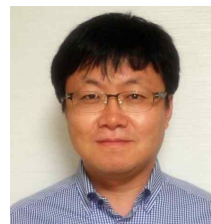
[16] Labeling for Object Detection [Internet]. Available: <https://www.makesense.ai/>



명현정 (Hyun-Jung Myung)

2021년: 전주대학교 스마트미디어학과 (학사졸업)

2021년 : 전주대학교 스마트미디어학과
2021년~현재 : 전주대학교 일반대학원 Agro AI학과 재학 중
※관심분야 : 인공지능, 스마트팜



송주환 (Ju-Whan Song)

1997년 : 전북대학교 대학원 (이학석사)
2003년 : 전북대학교 대학원 (이학박사 - 컴퓨터그래픽스)

2006년~현재 : 전주대학교 인공지능학과 교수
2019년~현재 : 전주대학교 일반대학원 Agro AI학과 교수
※관심분야 : 인공지능, 스마트팜, 영상처리, 사물인터넷 (IoT), 디지털콘텐츠, 모바일 프로그래밍