

재구조화된 RetinaNet을 활용한 객체 탐지에 관한 연구

김준영[†], 정세훈^{††}, 심춘보^{†††}

A Study on Object Detection using Restructured RetinaNet

Jun Yeong Kim[†], Se Hoon Jung^{††}, Chun Bo Sim^{†††}

ABSTRACT

Searching for portable baggage through the system before boarding an airplane at an airport is important because it prevents many risks. In addition to these dangerous items, personal and confidential information leaks are occurring at airports through data storage devices. In the airport search system, there is a need for a system that searches not only dangerous items but also devices that can leak data. In this paper, we proposed a model that searches for a data storage device by improving the existing model. A comparative evaluation was conducted using existing algorithms. As a result, it was confirmed that the performance of the proposed model is 74 in the training data and 46.73 in the test data, which is superior to the existing model.

Key words: Object Detection, Airport, Data Storage Device, FPN

1. 서 론

소셜미디어 서비스(SNS)가 발전함에 따라 다양한 형태의 데이터들이 수집되고 있다. 하루 동안에도 수 많은 데이터가 수집되며 이를 활용할 수 있는 빅데이터 기술들에 대한 필요성과 중요성이 중요한 시대이다. 이러한 빅데이터를 활용하기 위해 딥러닝을 활용한 연구들이 진행되고 있다. 컴퓨터 하드웨어의 발전으로 인해 복잡한 연산에 소요되는 시간이 줄어들면서 빅데이터와 딥러닝을 활용한 다양한 연구가

확대되고 있는 실정이다. 딥러닝은 다양한 분야의 문제들을 해결하기 위해 사용되고 있으며, 컴퓨터 비전 분야에서는 인간이 보고 인식하는 것처럼 시각 요소를 분석하여 컴퓨터가 대신 분석할 수 있도록 하는 딥러닝에 관한 연구가 활발하게 진행되고 있다[1-3].

컴퓨터 하드웨어가 발전해 연산 능력이 향상되어 컨볼루션 신경망(이하, CNN) 기반의 딥러닝 모델들에 관한 연구가 증가하고 있고 딥러닝을 활용한 객체 탐지 기술들이 연구되고 있다[4-6]. 객체탐지 기술은 영상 내의 다양한 객체를 구분하여 탐지하는 기술로

* Corresponding Author : Chun Bo Sim, Address: (57922) 255, Jungang-ro, Suncheon-si, Jeollanam-do, Republic of Korea, TEL : +82-61-750-3830, FAX : +82-61-750-3834, E-mail : cbsim@scnu.ac.kr
Receipt date : Nov. 13, 2020, Revision date : Nov. 24, 2020, Approval date : Nov. 25, 2020

[†] Suncheon National University, Dept of Multimedia engineering
(E-mail : kjoone3k@naver.com)

^{††} Andong National University, School of Creative Convergence
(E-mail : jungsh@anu.ac.kr)

^{†††} Suncheon National University, Dept of Multimedia engineering

* This paper has been written with the support of the Agricultural Science and Technology Research Project (No. PJ015151032020) of the National Agricultural Science Institute of the Rural Development Administration and this research was supported by the MSIT(Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program(IITP-2020-0-01489) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation)

컴퓨터가 스스로 객체를 인식하고, 신호를 인식하는 기술이다. 객체탐지 알고리즘은 작물 병해충 탐지, 작물 과실 탐지, 드론을 활용한 객체탐지, 자율주행 자동차를 위한 객체탐지, 불법 주정차 탐지, 이미지 검색, 글자 탐지, 얼굴 인식 등 다양한 분야에 활용이 가능하다.

객체탐지 알고리즘은 크게 1단계(One-Stage) 기반의 객체탐지 알고리즘과 2단계(Two-Stage) 기반의 객체탐지 알고리즘으로 나눌 수 있다. 객체탐지 알고리즘은 이미지 속에서 객체를 분리하는 것과 분리된 객체를 분류하는 두 가지 과정이 있다. 1단계 기반의 객체탐지 알고리즘은 두 가지 과정을 동시에 처리할 수 있는 알고리즘이고 2단계 기반의 객체탐지 알고리즘은 두 가지 과정을 순차적으로 처리하는 방식이다. 일반적으로 1단계 기반의 객체탐지 알고리즘은 처리 속도가 빠르지만 정확도가 낮다는 단점이 있고 2단계 기반의 객체탐지 알고리즘은 처리 속도는 느리지만 정확도가 높다는 단점이 있다. 이러한 장점 및 단점을 포함한 객체탐지 기술은 다양한 분야에서 활용되고 있다.

대표적인 객체탐지 기술의 예는 공항 검색대의 엑스선(이하, X-Ray) 시스템이다. 국내 또는 국외 여행을 할 때 비행기를 타고 여행할 경우 비행기에 탑승하기 전 보안 검색대를 통과하게 된다. 보안 검색대에서는 X-Ray를 이용하여 휴대용 수화물의 객체탐지 기능을 포함한 감시 기술을 활용한 조사를 진행한다. 항공 보안법에 따라 휴대가 불가능한 품목들을 탐지하고 제거한다. 1986년 김포공항 국제선 청사 폭탄테러 이후 테러 사건이 발생하지 않고 있으나, 테러에 대한 여러 위협 상황을 항상 존재하고 있다. 테러 위협에서 안전하고 편안한 여행과 항공기의 안전 운항을 위하여 항공기 탑승 전에 보안검색을 실시한다. 현재 공항의 검색대에서는 항공 보안법에 따라 위해 물품들을 정하고 검색시스템을 구축하여 휴대용 수화물들을 검색하고 탐지해낸다. 보안검색에 사용되는 장비로는 문형 금속탐지 장비, 휴대용 금속탐지 장비, 엑스선 검색 장비, 폭발물 탐지 장비, 폭발물 흔적 탐지 장비가 있다[7-9].

이러한 폭발 탐지뿐만 아니라 데이터 저장장치들을 통해 개인 정보나 회사의 기밀정보들이 해외로 유출되는 사건들도 발생하고 있다. 공항의 보안검색에서는 위험 물품뿐만 아니라 이러한 개인 정보 및

기밀정보들의 해외 유출을 예방하기 위해 데이터 저장장치들을 탐지하고 확인할 필요가 있다. 현재 데이터 저장장치들에 대한 검출 시스템이 없기 때문에 개인 정보 및 기밀정보가 해외로 유출되는 경우가 발생한다.

본 논문에서는 기존 RetinaNet 알고리즘을 재구조화하여 객체탐지 알고리즘의 성능 향상에 관한 연구를 진행하였다. 제안하는 객체탐지 알고리즘의 성능평가를 위해 공항에서 보안 검색대를 통과할 때 얻을 수 있는 X-Ray 영상을 활용하여 데이터 저장장치 객체를 탐지하여 개인 정보 및 기밀정보의 해외 유출을 방지하고자 한다. 객체탐지 알고리즘은 다양한 분야에 실시간으로 활용할 수 있게 1단계 기반의 객체탐지 알고리즘을 개선하여 빠르고 정확한 성능을 가질 수 있도록 연구한다.

본 논문의 구성은 다음과 같다. 2장에서는 객체탐지 알고리즘과 관련된 연구를 조사하고, 3장에서는 제안하는 객체탐지 알고리즘에 대하여 설명하고 성능평가에 활용되는 데이터에 대해 설명한다. 4장에서는 제안한 알고리즘으로 실험한 결과를 기존의 알고리즘들과 비교하여 성능을 평가하며, 5장에서는 결론 및 향후 연구 방향에 대해 설명한다.

2. 관련 연구

일반적인 객체탐지 알고리즘은 두 단계의 프로세스에 걸쳐 객체탐지를 진행한다. 첫 번째 프로세스에서는 영상으로부터 객체의 후보 영역들을 선택하고 그 후 각 후보 영역으로부터 특징을 찾아낸다. 두 번째 프로세스에서 찾아낸 특징들을 분류기를 이용하여 객체 후보 영역의 다중 클래스 분류를 진행한다. 객체탐지 방법에 따라 경계 라벨링의 회귀를 통한 지역화를 수행한다. 딥러닝 기반 객체 검출 알고리즘은 구조적 특성과 방식에 따라 1단계 기반의 객체탐지 알고리즘과 2단계 기반의 객체탐지 알고리즘으로 나눌 수 있다.

2.1 1단계 기반의 객체탐지 알고리즘

1단계 기반의 객체탐지 알고리즘은 객체 후보를 찾는 단계 없이 미리 정의된 라벨링으로부터 분류와 경계 라벨링 회귀를 수행하는 딥러닝 모델로 객체 검출을 실시한다. 대표적으로 YOLO(You Only Lool

Once)와 SSD(Single Shot multibox Detector), Retina Net이 있다.

YOLO는 Faster R-CNN과 달리 객체 인식 문제를 회귀분석 문제로 접근한 방식이다. 이미지 내의 경계 박스(Bounding Box)와 클래스 확률(Class Probability)를 회귀 문제로 간주하여 이미지를 한 번 탐색하는 것으로 객체의 클래스와 위치를 찾아낸다. 입력 이미지를 $S \times S$ 크기의 그리드로 나누고 각각의 그리드 셀들은 여러 경계박스로 구분한다. 경계박스 내부에 객체가 존재할 확률값을 의미하는 신뢰도 점수(Confidence Score), 객체가 어떤 클래스에 포함되는지에 대한 조건부 확률값(Conditional Class Probability)를 갖는다. 이렇게 YOLO는 하나의 CNN을 통해 경계박스에 대한 클래스 확률을 계산한다[10].

SSD는 후보 영역을 생성하기 위해 영역 제안 네트워크(Region Proposal Network, 이하 RPN)를 훈련시키지 않고 여러 크기의 특징맵(Feature Map)을 사용하여 객체를 인식하는 방식이다. 모델로부터 얻은 특징맵은 계층이 진행됨에 따라 크기가 줄어들게 되며 다양한 크기의 객체를 검출할 수 있게 된다. 서로 다른 크기의 특징맵에서 다양한 크기의 객체를 검출함으로써 성능을 개선을 제시한다[11].

RetinaNet은 1단계 객체탐지 알고리즘이 속도만 빠르지만 R-CNN 계열의 2단계 객체 탐지 알고리즘보다 정확도가 낮아 이 문제를 해결하기 위한 연구를 진행하였다. 이 문제는 극단적인 클래스 불균형 문제 때문에 발생했다는 것을 발견했고 클래스 불균형 문제를 해결하기 위해 크로스 엔트로피 손실함수를 수정한 Focal Loss를 제안하였다. Focal Loss는 잘 분류된 예제들은 작은 가중치를 부여하여 분류하기 어려운 예제들에 학습을 집중하여 클래스 불균형 문제를 해결한다[12].

2.2 2단계 기반의 객체탐지 알고리즘

2단계 기반의 객체탐지 알고리즘은 객체를 포함할 가능성이 높은 영역을 선택적 탐색과 같은 컴퓨터 비전 기술을 활용하는 방식이다. 그리고 딥러닝 기반의 영역 제안 네트워크를 통해 순차적으로 수행한 후 선택된 후보 영역들에서 분류한다. 이러한 과정으로 인해 2단계 기반의 객체탐지 알고리즘은 높은 정확성을 가진다. 대표적으로 R-CNN, Fast R-CNN과 Faster R-CNN이 있다.

R-CNN은 CNN 구조를 사용하여 객체가 포함된 영역을 찾아 분류하기 위한 알고리즘이다. 객체가 포함된 영역을 영역 제안(Region Proposal)이라고 정의한다. 영역 제안은 선별적 검색(Selective Search) 알고리즘을 사용하여 검색 및 분류할 수 있으며, 선별적 검색 알고리즘은 입력 이미지에 대해 2,000개 정도의 다른 영역을 생성하고 생성된 영역 중 객체가 포함될 확률이 가장 높은 영역을 검색하는 알고리즘이다. 이러한 목표를 이루기 위해 R-CNN은 영역 제안들을 동일한 크기로 변환하고 CNN을 통해 입력 이미지의 특징들을 추출한다. 추출한 특징들은 서포트 벡터 머신(Support Vector Machine, 이하 SVM)을 이용하여 분류하는 과정을 거친다. 최종적으로 회귀 학습을 통해 객체 경계박스 위치를 보정하여 정확한 물체의 위치를 찾는다. 하지만 R-CNN은 이미지를 분류하고 영역 제안을 검출해내는데 복잡한 과정을 거쳐 느린 속도를 가졌다[13].

Fast R-CNN은 R-CNN의 느린 속도를 개선하기 위해 제안된 모델로 R-CNN과 동일하게 영역 제안을 찾기 위해 선별적 검색을 사용한다. 그러나 R-CNN과 다르게 생성된 2,000개의 영역 제안을 이미지로부터 분류를 수행하지 않고 입력 영상 전체를 CNN에 통과시켜 특징맵을 추출한 후 선별적 검색을 통해 영역 제안을 검출한다. 또한 분류기의 손실값과 경계박스의 손실값을 계산하는 중간 계산 과정을 단순화하여 훈련 단계를 간소화시키며 속도를 개선한다. 분류기로는 SVM 대신 소프트맥스(Softmax) 함수를 사용하여 성능을 높였다. Fast R-CNN의 프로세스를 단순화시켜 R-CNN에 비해 100배 이상 빠른 검출 속도를 보인다[14].

Faster R-CNN의 경우 기존 R-CNN과 Fast R-CNN의 문제점이었던 느린 속도 문제를 보완한 알고리즘이다. Fast R-CNN의 영역 제안의 병목현상을 개선하기 위해 선별적 검색 알고리즘을 삭제하고 RPN을 활용하였다. RPN은 특징맵을 입력으로 받아 객체의 위치를 추정하여 영역 제안을 출력해주는 네트워크로 객체 제안과 객체 점수(Objectness Score)를 출력한다. 다양한 크기의 객체를 검출하기 위해 앵커를 도입하여 여러 크기의 객체를 탐지한다. 슬라이딩 윈도우가 특징맵을 통과할 때 각 슬라이드마다 경계박스 후보와 분류를 위한 점수 및 회귀를 위한 좌표를 생성한다. Faster R-CNN은 기존 R-CNN,

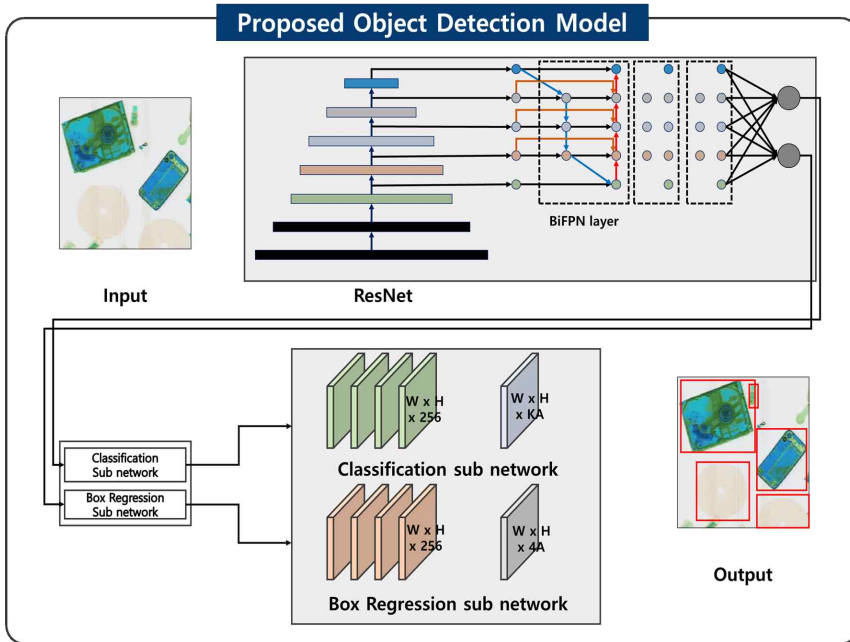


Fig. 1. Proposed Object Detection Model Configuration.

Fast R-CNN 대비 정확도 및 속도 측면에서 우수한 성능을 보였지만, 실시간으로 사용하지 못한다는 문제점이 존재한다[15].

3. 제안한 객체탐지 알고리즘

3.1 제안한 알고리즘의 개요

본 논문에서는 기존 RetinaNet 알고리즘을 재구조화한 객체탐지 알고리즘을 제안한다. Fig. 1은 제안하는 객체탐지 알고리즘의 전체적인 구성도를 제시한다.

제안하는 객체탐지 알고리즘은 RetinaNet 모델의 성능을 향상시키기 위해 FPN(Feature Pyramid Network)을 재구조화하여 활용한다. 기존 FPN 구조는 CNN을 지나면서 크기를 증대시키고 특징맵을 생성 및 결합하여 객체탐지를 수행한다. 그리고 피라미드 층에서는 객체의 해상도가 낮아지기 때문에 특징맵을 활용한다. 특징맵을 활용하기 위하여 최근접 이웃 업샘플링(Nearest Neighbor Upsampling, 이하 NNU) 기법을 적용하고 객체가 포함된 해상도의 크기를 맞춰 준 후 특징맵을 통합한다. 제시한 알고리즘을 활용하여 상위 피라미드 층의 추상화된 정보와 하위 피라미드 층의 크기가 작은 객체들에 대한 정보

를 모두 활용하면서 객체탐지를 진행할 수 있다. 기존 FPN 방식에서는 피라미드 층의 특징맵들은 NNU 기법을 활용한다. 객체가 포함된 해상도의 크기를 맞춰 특징맵을 결합하지만 입력 특징들의 해상도가 다르기 때문에 각각의 특징맵들은 출력에 기여하는 정도가 다를 수 있다, 또한 각각의 특징맵들을 통합할 때 가중치를 부여하여 성능을 향상시킬 수 있도록 설계한다.

Fig. 2는 BiFPN의 계층 구조도이다. BiFPN 구조는 기존 FPN 구조인 Top-down 방식에서 Top-down과 Bottom-up 방식을 추가하고 다른 피라미드 층에 가중치를 부여하여 피라미드 층을 합치게 한다. 동일한 층에서 갈색 선과 같이 엣지를 추가하여 여러

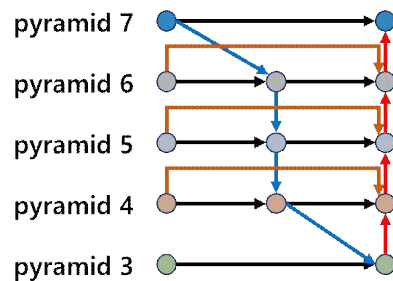


Fig. 2. BiFPN Layer Configuration.

특징들이 합쳐질 수 있도록 구성된 구조이다. 이러한 구조를 여러 번 반복하고 사용하여 개선된 구조의 특징을 결합할 수 있다.

기존 FPN 구조에서 활성화 함수로 ReLU(Rectified Linear Unit)를 적용하였다. ReLU는 시그모이드(Sigmoid) 함수의 단점인 그라디언트 소실 문제와 속도 문제를 해결하였다. 그러나 ReLU가 음수들을 모두 0으로 처리하기 때문에 한번 음수가 나오게 되면 더 이상 해당 노드는 학습되지 않는 단점이 있다. 이렇게 ReLU가 학습 중에 오류가 발생할 수 있다.

본 연구에서는 FPN 구조에 활성화 함수인 SELU를 사용하였다. 수식 1은 SELU의 수식이다. x 가 0보다 클 경우는 RELU와 동일한 형태이지만 x 가 0 이하일 경우 지수 선형 형태의 함수를 보인다. SELU는 ReLU와 동일하게 그라디언트 소실 문제가 없고 빠른 속도로 학습할 수 있다. SELU는 ReLU와 달리 음수가 나올 때 0으로 처리가 되지 않기 때문에 학습 중에 죽는 경우가 없다. 식의 α 값을 조절함으로써 활성화 함수의 분산이 일정하게 나와 성능이 좋다.

$$selu(x) = \lambda \begin{cases} x & \text{if } x > 0 \\ \alpha e^x - \alpha & \text{if } x \leq 0 \end{cases} \quad (1)$$

분류(Classification) 서브 네트워크는 BiFPN 구조에서 특징맵을 입력으로 받아 3×3 컨볼루션 레이어(Convolutional layer)를 256개의 필터로 컨볼루션 연산을 4회 진행한다. $k \times A$ 개(k : object 개수, A : 앵커 개수)의 필터로 컨볼루션 연산을 수행한 후 예측을 위해 시그모이드 함수를 적용한다. 박스 회귀(Box Regression) 서브 네트워크는 분류 서브 네트워크와 동일한 구조로 각각의 위치에서 전체 앵커 박스와 Ground-truth object간의 좌표에 대한 오프셋을 예측한다.

박스 회귀 서브 네트워크에서는 Smooth L1 손실 함수를 적용하여 오차 값이 낮을 경우 객체가 포함된 영역으로 판단하고 손실 값을 빠른 속도로 줄인다. 수식 2는 Smooth L1 손실함수의 수식이다.

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (2)$$

분류 서브 네트워크에서는 Focal loss 함수를 사용하여 클래스 불균형 문제를 해결한다. Focal loss 함수는 잘 분류된 예제에 대한 중요도는 분류하기 어려운 예제의 학습에 집중할 수 있도록 만들어준다. 수식 3은 Focal loss 함수의 수식이다. 수식의 γ 값이

Table 1. Development Environment

	Details
OS	Window 10
CPU	Intel Core i7-9700K
RAM	32GB
GPU	Geforce GTX 2060 SUPER
Language	python
IDE	Pycharm Community 2018.3.7.
Library	Tensorflow, Keras, etc.

커질수록 잘 분류된 예제에 대한 중요도가 낮다는 것을 의미한다. 수식의 γ 값이 0일 경우 크로스 엔트로피 손실함수와 동일한 식을 가지게 된다.

$$FL(p_t) = -(1 - p_t)^\gamma \log(p_t) \quad (3)$$

3.2 시스템 구현 환경

본 논문에서 제안하는 알고리즘의 구현 환경은 Table 1의 개발 환경에 따른다.

3.3 학습 및 테스트 데이터

본 논문에서는 공항 검색대에서 촬영된 X-Ray 영상을 활용하여 학습 및 테스트를 진행한다. 데이터 저장장치 이미지는 총 1,293개의 이미지로 SDCARD, USB, SSD, PHONE, CD, HDD와 ETC 등 7가지 객체로 라벨링을 진행한다. 전체 데이터 저장장치 이미지는 총 15,208개(SDCARD 2087개, USB 3947개, SSD 646개, PHONE 2087개, CD 934개, HDD 4543개와 ETC 573개)의 객체가 존재한다. Fig. 3은 데이터 저장장치 이미지 및 객체 라벨링 파일의 일부이다.

3.4 객체탐지 모델의 성능평가 방법

본 논문에서 제안하는 객체탐지 모델의 성능평가 방법으로는 mAP(mean Average Precision)를 사용하여 Faster R-CNN, YOLO, SSD, RetinaNet과 비교평가를 진행한다. mAP를 구하기 전 객체탐지 모델의 Precision 값과 Recall 값을 구한다. Precision 값은 모든 검출 결과 중 옳게 검출한 비율을 뜻하는 정밀도를 말하며, Recall 값은 검출해야 하는 객체 중에서 제대로 검출된 것의 비율을 의미하는 재현율을 뜻한다. Precision과 Recall은 각각 식 (4), 식 (5)와 같다.

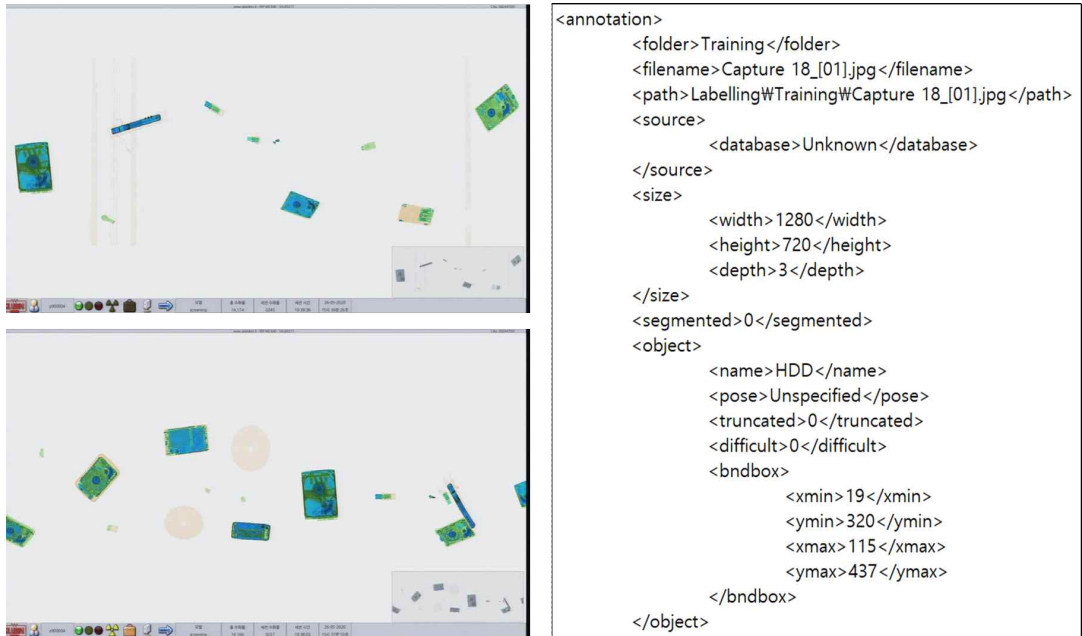


Fig. 3. Train & Test Data Images and Annotations.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (4)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ -ative} \quad (5)$$

각 클래스의 Precision 값과 Recall 값을 구한 후 Precision-Recall 그래프를 그려 그래프의 아래쪽 면적을 모두 더하면 각 클래스의 AP값을 구할 수 있다. 각 클래스의 값을 모두 더한 후 클래스 개수로 나눠 최종 mAP 값을 계산한 후 비교평가를 진행한다.

4. 실험 및 성능평가

본 논문에서는 제안하는 객체탐지 모델과 Faster R-CNN, YOLO, SSD, RetinaNet을 이용하여 데이터 저장장치의 객체탐지를 수행한다. 비교평가를 위해 Table 2와 같이 전체 이미지 1,293장은 학습 데이터 970장, 테스트 데이터 323장으로 분류한다. 학습 데이터에는 12,488개의 객체가 존재하며, 테스트 데

Table 2. Number of Images and Objects

	Number of images	Number of objects
Total	1,293	15,208
train	970	12,488
test	323	2,720

이터에는 2,720개의 객체가 존재한다. 학습 데이터와 테스트 데이터에 존재하는 각 클래스의 객체 수는 Table 3과 같이 나누어 진행하였다.

Table 4는 제안하는 객체탐지 모델을 활용하여 데이터 저장장치 이미지를 학습 데이터와 테스트 데이터를 구분하여 mAP 측정 지수를 비교한 결과표이다. Faster R-CNN의 경우 학습 데이터에서 69.09%, 테스트 데이터에서 39.07%, YOLOv2의 경우 학습 데이터에서 56.37%, 테스트 데이터에서 31.80%, SSD의 경우 학습 데이터에서 58.24%, 테스트 데이터에서 32.00%, RetinaNet의 경우 학습 데이터에서 73.37%, 테스트 데이터에서 44.87%의 mAP 지수가 측정되었다. 제안하는 객체 탐지 모델은 학습 데이터

Table 3. Number of Objects by Class

Object name	Train Data	Test Data	Total Data
SDCARD	1,686	401	2,087
USB	3,191	756	3,947
SSD	526	120	646
PHONE	1,723	364	2,087
CD	770	164	934
HDD	3,744	799	4,543
ETC	457	116	573

Table 4. Performance Evaluation

Model	train/test	Average Precision							mAP
		SDCARD	USB	SSD	PHONE	CD	HDD	ETC	
Faster R-CNN	train	52.8	56.6	73.6	76.2	95.5	95.6	33.3	69.09
	test	39.4	29.6	35.6	44.1	49.6	59.7	15.5	39.07
YOLOv2	train	51.5	39.9	52.4	60.5	87.6	81.2	21.5	56.37
	test	28.6	21.4	32.5	38.7	42.6	52.3	6.5	31.80
SSD	train	53.9	37.8	56.5	61.8	91.5	83.6	22.6	58.24
	test	28.3	23.4	31.8	37.6	43.4	53.6	6.1	32.00
RetinaNet	train	73.2	54.6	76.1	84.3	98.3	94.3	32.8	73.37
	test	42.6	30.8	50.1	56	60.4	59.8	14.4	44.87
Study of proposed	train	71.5	56.8	75.5	83.2	98.4	97.3	35.3	74.00
	test	43.4	32.4	50.4	58.9	63.4	61.1	17.5	46.73

에서 74.00%, 테스트 데이터에서 46.73%의 mAP가 측정되었으며, 기존 모델과 비교하여 높은 인식률과 실시간 객체 측정이 가능하다는 결과를 보였다.

Fig. 4와 Fig. 5는 Faster R-CNN, YOLOv2, SSD, RetinaNet과 제안하는 객체탐지 모델의 비교평가에 대한 결과 그래프이다. 제안하는 모델의 mAP는 RetinaNet에 비해 비교적 높게 확인되었다. 이러한

주요 요인으로는 RetinaNet과는 다르게 FPN의 특징맵을 합쳐줄 때 각각 층의 특징맵에 가중치를 주어 해상도 별로 최종 결과에 영향을 끼치는 정도를 다르게 하여 성능을 향상시킬 수 있는 것으로 확인하였다. 그리고 제안하는 객체탐지 모델은 다른 모델에 비해 FPS(Frame Per Second)가 비교적 높게 측정되어 실시간성을 가진 객체탐지 모델로 평가된다.

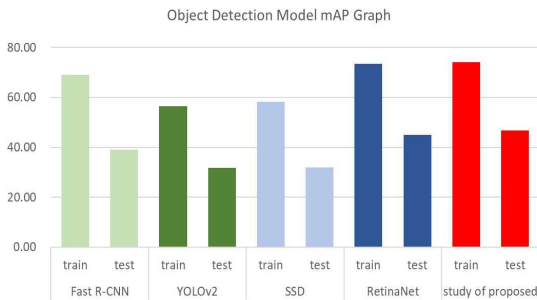


Fig. 4. Object Detection Model mAP Graph.

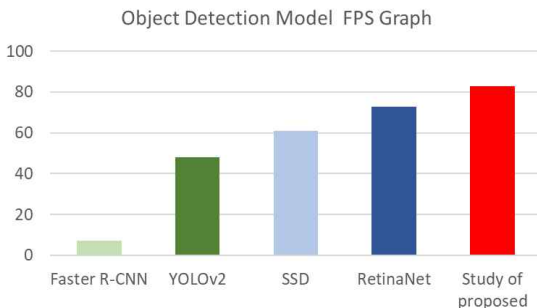


Fig. 5. Object Detection Model FPS Graph.

5. 결 론

객체탐지와 관련된 연구는 작물의 병해충 탐지, 작물 과실 탐지, 드론을 활용한 객체탐지, 스마트폰, 자율주행 자동차를 위한 객체탐지, 보안 등 다양한 분야에 적용하기 위한 연구가 진행되고 있다. 이러한 객체탐지와 관련된 연구는 실시간으로 정확하게 객체를 탐지하기 위한 연구가 진행되고 있으며 빠른 속도를 가진 1단계 기반의 객체탐지 알고리즘을 개선하여 높은 정확도를 제시하는 연구가 꾸준히 진행되고 있다. 본 논문에서는 기존 1단계에 적용되는 RetinaNet 알고리즘을 재구조화하고 객체 인식의 성능을 개선하기 위한 재구조화된 객체탐지 알고리즘을 제시하였다. 제시한 알고리즘을 평가하기 위하여 1단계(One-Stage)가 아닌 2단계(Two-Stage)형태로 저장된 X-Ray 이미지 데이터를 활용하였다. 기존 객체탐지 모델의 구조를 변경하여 여러 객체탐지 모델들과 성능평가를 진행하였고, 성능평가 결과 제안하는 객체탐지 모델의 성능이 향상된 것을 확인하였다.

향후 연구에서는 데이터 저장장치뿐만 아니라 항공기 반입금지 품목들을 추가하여 객체를 탐지하는 연구가 진행되어야 한다. 또한, 제안하는 객체탐지 모델을 개선하여 공항 검색대뿐만 아니라 스마트팜의 병해충 및 과실 객체탐지, 드론을 활용한 객체탐지, 대규모 공장에서의 불량품 검출, 자율주행 자동차의 사고 예방 등에 활용할 수 있는 연구가 필요하다.

REFERENCE

- [1] F. Shkurti, W. Chang, P. Henderson, M. Islam, J. Higuera, J. Li, et al., "Underwater Multi-Robot Convoying Using Visual Tracking by Detection," *IEEE International Workshop on Intelligent Robots and System*, pp. 4189-4196, 2017.
- [2] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Communications of the ACM*, Vol. 6, No. 6, pp. 1097-1105, 2012.
- [3] D. Berckmans, "Automatic On-line Monitoring of Animals by Precision Livestock Farming," *Livestock Production and Society*, pp. 27-30, 2004.
- [4] J.Y. Jeon, D.Y. Park, S.Y. Lim, Y.S. Ji, and S.B. Lim, "Automatic Extraction of Hangul Stroke Element Using Faster R-CNN for Font Similarity," *Journal of Korea Multimedia Society*, Vol 23, No. 8, pp. 953-964, 2020.
- [5] K.S. Yoon and J.Y. Choi, "Compressed Ensemble of Deep Convolutional Neural Networks with Global and Local Facial Features for Improved FaceRecognition," *Journal of Korea Multimedia Society*, Vol. 23, No. 8, pp. 1019-1029, 2020.
- [6] J.Y. Kim, S.A. Hong, and H.M. Kim, "A Style GAN Image Detection Model Based on Convolutional Neural Network," *Journal of Korea Multimedia Society*, Vol. 22, No. 12, pp. 1147-1456, 2019.
- [7] H. Im, S. Cho, J. Quan, J. Kim, C. Lee, and G. Lee, "Prediction of Aviation Security Technologies through Cluster Analysis of Patent Big Data," *Journal of the Korean Air Management Association*, Vol. 15, No. 5, pp. 43-63, 2017.
- [8] G. Nowacki and B. Paszukow, "Security Requirements for New Threats at International Airports," *TransNav the International Journal on Marine Navigation and Safety of Sea Transportation*, Vol. 12, No. 1, pp. 187-192, 2018.
- [9] Homeland Security Research Corporation, *X-Ray Baggage, Cargo, People, Container & Vehicle Screening Market - 2017-2021*, 2017.
- [10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779-788, 2016.
- [11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, et al., "SSD:Single Shot Multi Box Detector," *Proceeding of European Conference on Computer Vision*, pp. 21-37, 2016.
- [12] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 42, Issue 2, pp. 318-327, 2018.
- [13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *Proceeding of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587, 2014.
- [14] R. Girshick, "Fast R-CNN," *Proceeding of IEEE International Conference on Computer Vision*, pp.1440-1448, 2015.
- [15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, Issue 6, pp. 1137-1149, 2017.
- [16] J.W. Hong, J.H. OH, and H.K. Lee, "Smart

Airport and Next Generation Security Screening Technology,” *Electronics and Telecommunications Trends*, Vol. 34, Issue 2, pp. 73-82, 2019.



김 준 영

2019년 2월 순천대학교 멀티미디어공학과 졸업(공학사)
2019년 3월~현재 순천대학교 대학원 멀티미디어공학과 석사과정

관심분야 : IoT 상황인식, 빅데이터 처리 및 분석, 딥러닝



정 세 훈

2010년 2월 순천대학교 멀티미디어공학과 공학사
2012년 2월 순천대학교 멀티미디어공학과 공학석사
2017년 2월 순천대학교 멀티미디어공학과 공학박사

2015년 1월~2018년 8월 광양만권SW융합연구소 팀장
2018년 9월~2020년 2월 영산대학교 빅데이터융합전공 조교수
2020년 3월~현재 안동대학교 창의융합학부 조교수
관심분야 : 최적화 알고리즘, 강화학습, 블록체인



심 춘 보

1996년 2월 전북대학교 컴퓨터공학과 공학사
1998년 2월 전북대학교 컴퓨터공학과 공학석사
2003년 2월 전북대학교 컴퓨터공학과 공학박사

2005년 3월~현재 순천대학교 정보통신·멀티미디어공학부 교수
관심분야 : 빅데이터 시스템, 머신러닝, IoT/IoE 플랫폼, 멀티미디어