공학석사 학위논문

드론영상을 위한
앵커기반 물체검출 방법 연구

Anchor based Object Detection for Drone Vision

울 산 대 학 교 대 학 원

전기전자컴퓨터공학과

안 진 수

드론영상을 위한 앵커기반 물체검출방법 연구

지 도 교 수 조 강 현

이 논문을 공학석사학위 논문으로 제출함

2021 년 1 월

울 산 대 학 교 대 학 원
전기전자컴퓨터공학과
안 진 수

안진수의 공학석사 학위 논문을 인준함

심사위원장	강 희 준	(인)
심사위원	조 강 현	(인)
심사위원	서 영 수	(인)

울 산 대 학 교 대 학 원

2021 년 1 월

감사의 글

대학원 석사 과정을 하며 저에게 많은 도움을 주신 모든 분들께 감사드립니다.

먼저, 지능시스템 연구실에서 인공지능의 세계로 이끌어 주시고 늘 마음을 다잡을 수 있도록 이끌어 주신 지도교수 조강현 교수님께 감사의 말씀을 전합니다.

연구실 생활을 잘 할 수 있게 늘 곁에서 응원해주고 격려해준 율경이형 덕분에 이 연구를 잘 마무리할 수 있었습니다. 어떤 일이 있어도 포기하지 않고, 언어장벽도 넘을 수 있게 도와준 율경이형 정말 감사합니다.

연구실 생활에 잘 적응할 수 있게 도와준 연구실 친구들 락소노, 탕칭, 아즈말, 탄, 푸트로에게 깊은 감사를 드립니다.

학부생 때부터 늘 마음의 편안을 유지할 수 있게 좋은 말씀해주신 손보영 선생님 감사합니다.

다른 연구실이었지만 늘 함께 서로를 도와주었던 수현이, 관후, 민혁아 너무 고맙다!!

그리고 이제 석사과정을 시작하는 제환아, 꾸준하고 열심히 한번 달려보자!

마지막으로 항상 저를 믿고 지켜봐준 가족들에게 이 영광을 돌립니다.

감사합니다.

[국문 요약]

드론영상을 위한 앵커기반 물체검출방법 연구

울산대학교 대학원
전기전자컴퓨터공학과
안 진 수

최근 스마트 드론이 물품수송, 재해감시, 시설물 안전진단, 순찰, 레저 스포츠 등 굉장히 다양한 분야에서 활용되고 있다. 그리고 드론이 촬영하는 영상의 시점은 기존 지상의 영상 촬영보다 훨씬 넓은 장소를 한번에 볼 수 있는 장점이 있다. 본 연구에서는 드론 영상으로 컨볼루션 신경망을 학습시켜 물체 검출하는 방법을 제안하였다.

본 연구는 산림지, 관광지, 도심지를 대상으로 드론을 통해 취득한 4k 고화질 영상 데이터를 기반으로 물체검출 알고리즘(SSD, Faster R-CNN, RetinaNet, Bottleneck을 적용한 SSD)을 활용하여 각 모델별 실험 결과를 비교 및 분석하였다. 산림지, 관광지, 도심지는 각각 6, 17, 17개의 물제 종류를 가지고 있으며, 30m, 60m, 100m 이하의 고도별 데이터도 포함하고 있다. 도심지 기준 각 모델별 결과값은 0.801, 0.964, 0.696, 0.918 mAP 순으로 Faster R-CNN이 가장 높았으며, one-stage 검출 알고리즘인 SSD, RetinaNet보다 two-stage 검출 알고리즘인 Faster R-CNN이 드론 영상에서 물체를 검출하는데 더 좋은 성능을 보인다는 것을 확인하였다. 하지만 Faster R-CNN의 처리속도는 실시간에 적용될 수 없을 만큼 느렸다. SSD에 bottleneck을 적용하여 연상량을 절감시킴으로써, 실시간 처리속도를 확보하며 학습모델의 성능을 0.918mAP까지 향상시켰다.

[영문 요약]

Anchor based Object Detection for Drone Vision

Jinsu An
School of Electrical Engineering,
The Graduate School,
University of Ulsan
Supervised by Prof. Kang-Hyun Jo

ABSTRACT

Recently, smart drones have been used in a wide variety of fields such as goods transportation, disaster monitoring, facility safety diagnosis, patrols, and leisure sports. And the viewpoint of the video taken by the drone has the advantage that it can see a much wider place at once than the existing ground video. In this research, we proposed a method for detecting an object by learning a convolution neural network with the drone dataset.

In this research, we utilize object detection algorithms (SSD, Faster R-CNN, RetinaNet, SSD with Bottleneck) based on 4k high-quality video data acquired via drones for forest, tourist destination, city. The experimental results for each model were compared and analyzed. Forest, tourist destinations, and city areas have 6, 17, and 17 classes, respectively, and data for altitudes of 30m, 60m, and 100m or less are also included. The values of the results for each model in the city center are 0.801, 0.964, 0.696, 0.918 mAP in order, For each model based on city areas, Faster R-CNN is the highest in the order of 0.801, 0.964, 0.696, and 0.918 mAP. It was confirmed that it shows better performance in detecting the object in the drone dataset. However, the processing speed of Faster R-CNN was too slow to be applied in real time. By applying bottleneck to SSD to reduce the amount of computation, real-time processing speed was secured and the performance of the learning model was improved to 0.918mAP.

목 차

표 목 차

그 림 목 차

Ⅰ. 서 론

1.1. 연구배경

최근 들어 많은 기업과 기관에서 영상에 의한 주변 환경 감시를 위해 CCTV 카메라를 활용해왔다. 2016년 기준 전세계적으로 약 3억 5천만대의 카메라가 설치된 것으로 추정된다. 방대한 양의 데이터가 실시간으로 생성되고, 일반적으로 보안담당자가 영상을 보고 범죄자, 수상한 행동, 또는 안전 행동을 감시한다. 수작업으로 이루어지는 모니터링 시스템은 비용이 많이 들며 수많은 인적 오류를 유발할 수 있다. 2008년 영국 노팅엄 대학의 “Automatic Surveillance and CCTV Operator Workload” 연구에 따르면, 영상 감시 집중력은 12분에 45%나 감소하고 22분 이후로는 95%나 감소된다고 보고되고 있다[1]. 이 때문에 수작업 모니터링 시스템에는 많은 휴식이 필요하다. 4차 산업혁명시대에 접어들면서, 기존 IT 시스템에 다양한 변화가 생기고, 새로운 형태의 시스템이 증가하고 있다. 그 중 컴퓨터 비전(Computer Vision)을 활용한 영상인식 및 분류기법을 사용하여 수작업 모니터링 시스템의 비용 및 인적 오류를 해결하고자 한다. 본 연구를 통해 영상에서 곧바로 물체 검출(Object Detection)을 할 수 있게 되면, 수작업 모니터링 시스템에서 발생하는 비용을 줄일 수 있고, 인적 오류 발생 가능성도 제거된다. 딥러닝(Deep Learning) 적용을 통한 영상에서의 물체 검출은 인간의 뇌가 동작하는 방식을 모방한 머신러닝(Machine Learning)의 한 종류이다. 딥러닝은 많은 사례에서 인간보다 뛰어난 정확도를 보여주며, 기존의 컴퓨터 비전과 자연어처리 기술을 빠른 속도로 대체하고 있다. 그 예로 캘리포니아에 위치한 AIRLITIX는 농작물을 재배하고 있는 온실 속을 비행하는 드론을 개발해 딥러닝 기반의 인공지능을 적용하였다. 장애물을 감지하고 회피하는 기능과 정밀한 호버링이 가능하며, 건강 상태를 실시간으로 모니터링해서 조기에 질병을 감지하여 농작물의 피해를 최소화하고 있다. 또한 Advanced Computer Vision을 통해 해충을 정확하게 감지하고 박멸함으로써 병충해를 막아주는 기능도 갖고 있다. Smart Base Station에 드론이 착륙하면 사람이 수동으로 배터리를 교체할 필요 없이 자동으로 배터리까지 교체해주는 기술도 제공한다. 이와 같이 인공 지능의 실시간 머신 러닝 기술과 드론의 뛰어난 탐색 능력이 결합된다면 강력한 시너지 효과를 보여줄 것이다. 인공 지능과 결합된 드론은 주변 환경을 스스로 인식하고 환경을 매핑하고, 물체를 검출, 탐지 및 추적하며 실시간으로 피드백 데이터를 제공할 수 있을 것이다.

본 연구에서는 드론영상을 활용하여 인체 및 물체를 포함하는 객체 검출을 위한 연구를 진행한다. 물체 검출 연구는 주어진 영상 혹은 이미지에서 물체가 위치해 있는 영역을 예측하고, 예측한 영역에 대해 어떤 물체가 존재하는지 분류하는 컴퓨터 비전 기술이다. 본 연구에서는 앵커 박스(Anchor Box)를 기반으로 하는 물체 검출 방법을 사용했다. 앵커 박스는 물체 검출에서 손실 함수 수렴과 정확성을 높이는데 큰 도움을 준다. 방법으로는 앵커 박스를 일정 비율로 만드는 방법과 유클리디안 거리 및 IoU(Intersection over Union)를 통해 클러스터링 기법을 사용하여 생성하는 방법이 있다. 그리고 머신러닝의 한 분야인 인공신경망 컨볼루션 신경망(Convolutional Neural Network)이 물체 검출 및 분류 연구에서 부각을 보이기 시작하면서 기존의 컴퓨터 비전과 자연어 처리 기술을 대체하고 있다.

그림 1.1 AIRLITIX사의 농작물을 재배하고 있는 온실 속을 비행하는 드론

1.2. 연구 목표 및 내용

물체 검출은 영상에서 고정된 혹은 움직이는 관심 물체를 배경과 구분해 검출하고 식별하는 기법이다. 컴퓨터 비전에서 올바른 물체 검출을 위해서는 사각 테두리를 활용해 물체를 나타내는 사물의 클래스와 연관시킨다. 아래의 그림1.2와 같은 컨볼루션 신경망은 컴퓨터 비전에서 주목받는 핵심 기술이다. 물체 검출에서 가장 많이 겪는 문제는 바로 프레임별로 달라지는 물체의 개수이다. 영상에 하나의 물체만 존재하는 경우는 쉽게 클래스를 분류할 수 있지만, 다양한 분야에 활용하기 위해서는 영상 내에 존재하는 2개 이상의 물체를 검출하고 분류할 수 있어야 한다. 그림1.2의 컨볼루션 신경망을 이용하여 많은 물체를 검출하기에는 어려움이 있다.

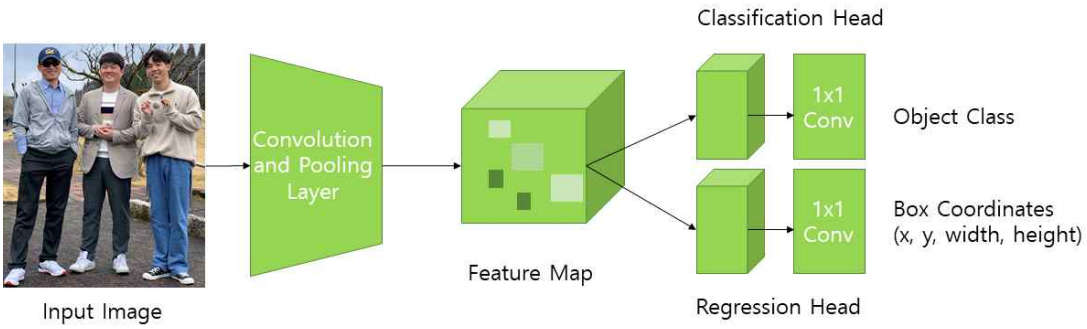
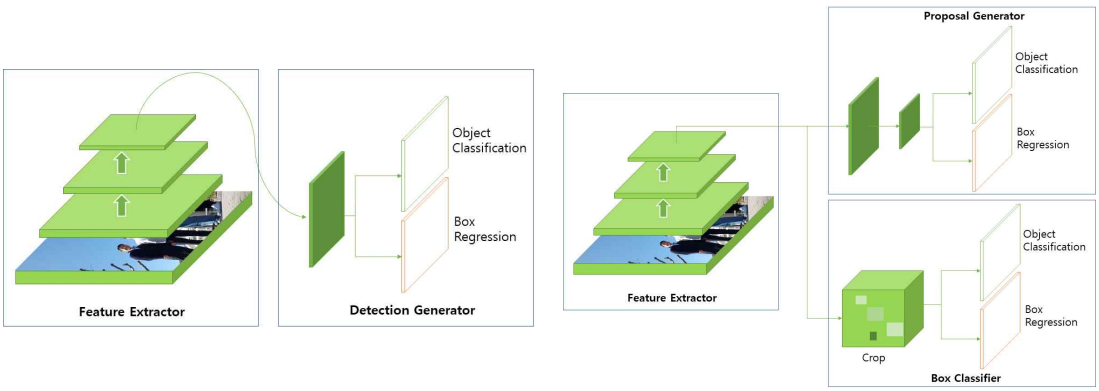


그림 1.2 영상 내 하나의 물체를 검출하는 CNN

다양한 윈도우(Window) 위치와 크기 및 비율을 고려하여, 이 정보를 물체의 종류(Class)를 분류하는데 활용할 수 있다. 이 윈도우를 모두 활용하게 되면 계산량이 너무 많아지기 때문에, 윈도우의 일부만을 활용하여 물체 검출을 효과적으로 할 수 있는 방법으로 Single-Stage Method와 영역제안(Region Proposal)을 활용한 Two-Stage Method가 있다.



(a) One-Stage Detector (b) Two-Stage Detector

그림 1.3 Single-Stage & Two-Stage Method

이후 물체 검출을 활용하여 인간 행동 분석 분야에도 활용할 수 있다. 인간 행동 분석

(Human Behavior Analysis)이란 영상에서 관찰할 수 있는 인간의 행동을 자동적으로 검출하는 기술이다. 인간 행동 분석은 드론을 통한 감시/보안/정찰을 위한 감시 시스템(Surveillance System) 구축, 위험상황 감지, 로봇과 사람의 상호작용, 자동차, IoT장치 등에 응용할 수 있다. 전통적인 인간 행동 분석 방법들은 분석 대상의 행동이나 사건이 끝난 시점에서 행동의 검출을 목표로 한다. 행동의 정보를 모두 담고 있는 영상이 주어진 후에 인식을 시도한다. 하지만 위험상황 감지, 감시 시스템에서는 행동을 미리 예측하여 미연에 방지하는 것이 중요하다.

본 논문에서는 드론 영상을 활용하여 물체 검출 알고리즘을 분석하여 컨볼루션 신경망을 학습한다. 데이터셋(Dataset)은 드론을 통해 촬영한 4k 고화질 영상을 활용한다. 데이터셋은 산림지, 관광지, 도심지로 구성되어 있으며 각 지역별로 6, 17, 17개의 종류로 구성되어 있다. 산림지의 클래스는 나무, 사람, 동물, 정상표식, 자전거, 관리소로 구성되어 있다. 관광지, 도심지의 클래스는 나무, 사람, 동물, 주택, 아파트, 학교, 관리소, 교통표지판, 신호등, 가로등, 현수막, 다리, 승용차, 버스, 트럭, 오토바이로 구성되어 있다.

본 연구는 드론 영상을 활용하여 물체 검출 알고리즘을 분석하고 컨볼루션 신경망을 사용하여 모델을 학습하는 연구를 제안한다. 여러 학습 모델의 결과를 비교하고 각 모델의 성능을 비교하여 분석한다.

1.3. 논문 구성

본 논문은 5장으로 구성되어 있으며, 다음과 같은 순서로 구성되어 있다.

1장에서는 연구배경과 목표 및 내용을 포함한 연구 논문 전체의 배경과 구성방식을 설명하고,

2장은 영상의 이해를 위한 영상의 구성, 종류, 물체검출을 위한 방법을 설명하고 있다.

3장은 드론영상을 활용한 물체 검출 방법에 대해 상술하여 이를 위해 최근 연구되고 있는 다양한 심층학습법의 개론과 활용방식을 설명하였다.

4장에서는 구축한 데이터셋을 간략하게 소개하고 실험을 진행한 하드웨어 구성에 대해 설명한다. 실험 방법 및 실험을 통해 얻은 결과를 분석한다.

5장에서는 고도 촬영에 따른 드론영상이 가지는 성질을 활용한 객체검출 실험결과를 바탕으로 정리하고 본 논문의 효용성을 설명하고 정리한다.

Ⅱ. 영상의 이해

2.1. 영상처리를 통한 물체검출

먼저 영상처리를 통해 물체를 검출하기 위해서는 이미지를 여러 방법으로 가공해서 특징을 뽑아내야 한다. 미분연산을 수행하여 이미지에 발생하는 변화를 알아내거나, 이진화를 통해 이미지를 극적으로 만들어 특징을 두드러지게 만드는 방법이 있다. 예를 들어 숫자를 인식하기 위해서는, 숫자와 배경을 구분하는 엣지를 추출하고 엣지의 모습과 일치율이 높은 숫자로 인식하는 방법이다. 숫자와 비교할 수 있는 대표적인 엣지를 만들기 위한 알고리즘을 개발자가 만들어줘야 한다. 이렇게 추출된 특징들이 인식률에 상당한 영향을 미쳤다.

전통적인 특징 추출 방법으로는 유사Haar특징(Haar-like feature)[2], 경사방향성분포법(HOG:Histograms of Oriented Gradients)[3], 크기불변성특징변환(SIFT:Scale Invariant Feature Transform)[4], 국소이진화패턴법(LBP:Local Binary Pattern)[5], MCT(Modified Census Transform)[6] 등이 있다. 특징을 추출한 이후, 특징들의 분포에서 경계 설정을 하는 알고리즘은 지지기반 벡터기구(SVM:Support Vector Machine)[7], Adaboost[9] 등과 같은 알고리즘을 사용하였다. 특징의 분포가 어떤 물체를 표현하는지 알아내는 것으로 물체를 검출했다.

2.1.1. Haar-like feature

사람의 얼굴에는 공통적으로 가지는 패턴이 있는데, 대부분의 모든 사람들은 두 개의 눈, 눈썹, 하나의 코, 입을 가지고 있다. 두 눈은 명암이 어둡고 코는 상대적으로 밝다. 이와 같이 명암을 이용해 패턴을 구하는 방식이며 이것을 유사Haar특징이라고 한다.

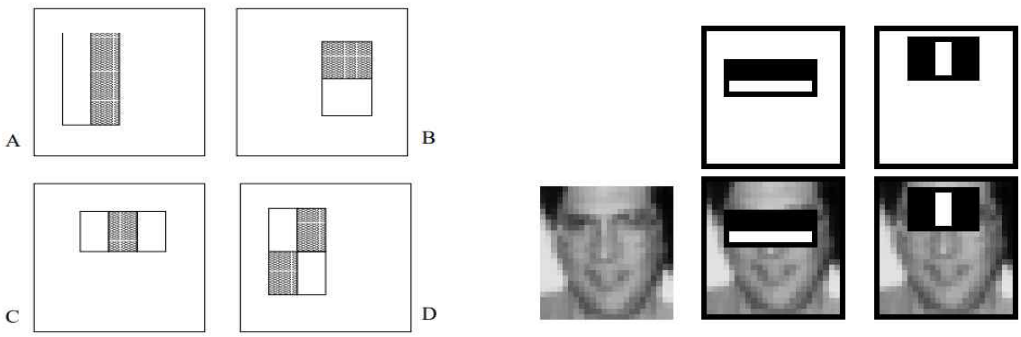


그림 2.1 Haar-like feature

2.1.2. HOG(Histograms of Oriented Gradients)

형상 기술자(Feature Descriptor)는 유용한 정보를 추출하고 필요 없는 정보를 버림으로써

이미지를 단순화하는 이미지 혹은 이미지 패치를 표현한 것이다. HOG 형상 기술자에서는, Gradients의 방향의 분포(Histograms)가 특징으로 사용된다. 모서리와, 모서리 주위에서 Gradients의 값이 커지므로, 평면 영역보다 물체 모양에 대한 정보를 더 많이 표현한다.

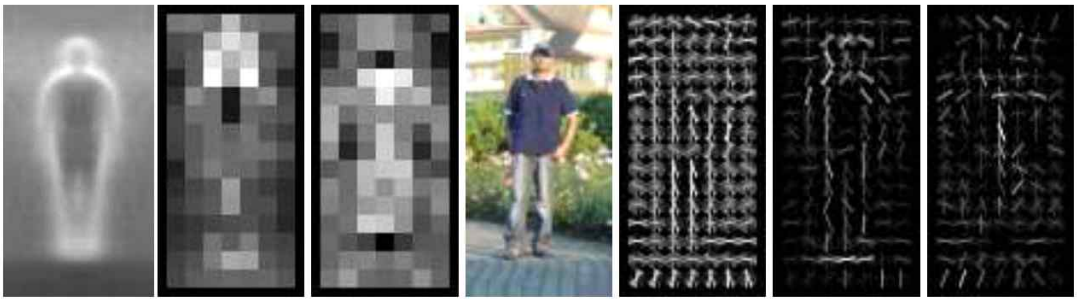


그림 2.2 Histograms of Oriented Gradients

2.1.3. SIFT(Scale Invariant Feature Transform)

SIFT는 특징을 추출하는 대표적인 알고리즘 중의 하나이다. 이미지의 크기(scale), 회전(rotation)에 대해 불변하는 특징을 추출하기 때문에, 이미지가 회전해도 크기가 변해도 항상 그 위치에서 같은 특징을 찾아내는 알고리즘이다.

2.1.4. HOG-SVM

HOG-SVM 알고리즘은 HOG와 SVM(Support Vector Machine)을 결합한 물체 검출 알고리즘이다. HOG를 이용해서 이미지의 지역적인 기술기를 해당 이미지의 특징으로 추출하여 학습 기반 분류기인 SVM에 입력한다. HOG는 물체 영역을 일정 크기의 셀로 분할하고 셀 내의 윤곽선의 양과 방향을 계산하여 히스토그램으로 생성한다. SVM은 패턴인식을 위한 지도학습 알고리즘이다. 주어진 데이터셋을 기반으로 새로운 입력 데이터가 어느 종류(Class)인지 판단하는 선형 분류 학습 모델이다. 그림 2.3과 같이 마진이 최대인 선형 초평면(Linear Hyperplane)을 결정하여 주어진 데이터들을 분류한다. HOG가 이미지 내 물체에 대한 HOG 특징과 비객체에 대한 HOG을 추출하여 SVM에 입력 데이터로 넘겨주면, SVM에서 두 특징을 기반으로 이미지 내 객체의 유무를 판단하도록 학습한다.

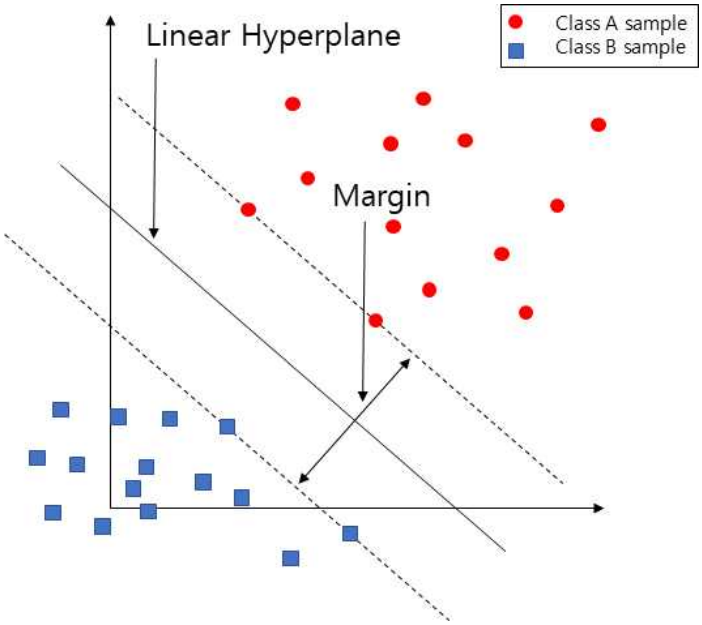


그림 2.3 지지기반 벡터기구법(Support Vector Machine) 설명

2.2. CNN 기반 물체 검출 연구 현황

컴퓨터 비전 분야에서는 오랫동안 얼굴, 사람/보행자, 차량 등을 포함한 다양한 객체를 검출 및 인식하는 연구가 진행되어왔다. 전통적으로 SIFT, HOG, DPM(Deformable Part Model)[9]과 같은 객체가 가지는 특징을 추출해 객체를 검출하기 위해 이용하였다. 데이터보다는 도메인의 지식에 많이 의존하는 방법이었으며, 추출한 특징이 여러 가지의 파라미터를 가지면 수동적으로 개발자가 파라미터를 수정해주어야 한다는 단점이 있다. 일반적으로 객체를 검출하기 위해서는 영상에서 (1) 객체 후보 영역을 선정, (2) 객체 후보 영역의 특징을 추출, (3) 추출한 특징에 분류기를 적용하여 객체 후보 영역의 클래스를 분류한다. (4) 이후, Bounding Box Regression과 같은 후처리를 통해 Localization 성능을 높인다.

최근 CPU 및 GPU 등의 하드웨어의 발전으로 인해 대용량의 데이터(Big Data)를 처리할 수 있게 되었고, 신경망(Neural Network)[11]분야가 재조명 받으면서 컨볼루션 신경망 기반의 물체 검출 연구가 활발히 진행된다. 특히 2012년 Krizhevsky et al,의 연구에서는 대용량 데이터셋(ImageNet)[10]에 대해 컨볼루션 신경망을 활용하여 분류(Classification) 문제를 다루었는데, 전통적인 영상처리와 비교해 상당한 성능 향상을 이루었다. 컨볼루션 신경망은 시각피질의 뉴런이 어떻게 반응하는지 알아보기 위해 개발되었다. 고양이에게 단순한 모양의 여러 가지 패턴을 보여주고 시각피질의 반응을 살펴보며, 패턴에 따른 뉴런의 반응을 관찰하는 실험이었다.

시각 피질 안의 많은 뉴런이 작은 local receptive field를 가진다는 것을 증명하였고, 뉴런들이 시야의 일부 범위 안에 있는 시각 자극에만 반응하는 것을 알아냈다. 뉴런의 receptive

field는 서로 겹칠 수 있으며, 겹쳐진 receptive field들이 전체 시야를 이루고 있다. 특정 뉴런이 넓은 receptive field를 가져 저수준의 패턴(edge, blob 등)이 조합되어 복잡한 패턴(texture, object)에 반응한다. 이런 아이디어를 기반으로 개발된 것이 컨볼루션 신경망이다.

컨볼루션 신경망은 합성곱층(Convolutional Layer), 풀링층(Pooling Layer)이 전체 신경망의 앞쪽에 위치하며 층 사이의 노드를 모두 연결하는 완전연결층(Fully Connected Layer)이 신경망의 뒤쪽에 이루어져 있으며, 심층신경망(Deep Neural Network)보다 특징을 잘 보존하며 학습하고, 필터를 공유하여 특징을 추출하기 때문에 학습에 요구되는 파라미터 수 또한 적다. 합성곱의 연산은 다음 그림 2.4처럼 나타낼 수 있다.

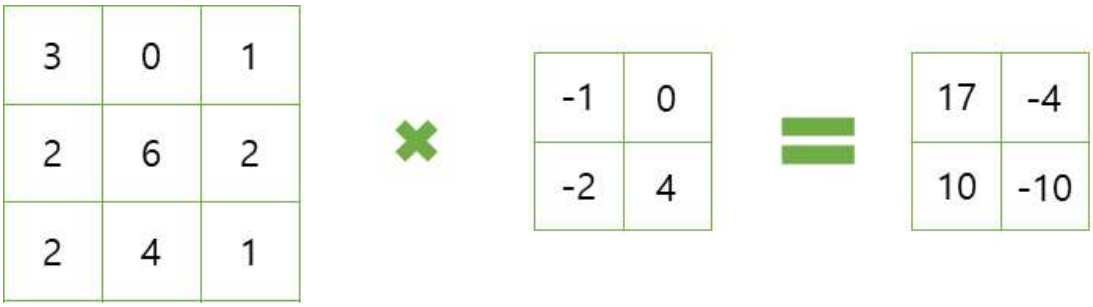


그림 2.4 합성곱 연산 예시

이 합성곱 연산은 다음 수식 (2.1)과 같이 나타낼 수 있다.

$$y[i,j] = (x*k)[i,j] = \sum_{n=0}^{w}\sum_{m=0}^{h} x[i-m, j-n]k[m,n] \tag{2.1}$$

수식 (2.1)에서 w는 합성곱 필터의 너비, h는 합성곱 필터의 높이를 의미하며 i와 j는 각각 가로, 세로 방향 요소를 의미한다.

풀링 연산은 주로 합성곱 연산 바로 다음에 진행되며, 여러 픽셀 값 중 한 개 픽셀 값만을 추출하는 연산을 말한다. 풀링의 종류는 최대 풀링(Max pooling), 평균 풀링(Average pooling), 최소 풀링(Min pooling)이 있다. 이름에서 알 수 있듯이 최대 풀링은 배치의 최대 픽셀 값만 남기고 모든 값을 지워버리는 풀링입니다. 평균 풀링은 배치의 모든 픽셀의 평균 값을 출력하며, 최소 풀링은 배치의 최소 픽셀 값을 출력하는 풀링이다. 여기서 배치란, 이미지 크기에 따라 결정되는 필터의 크기와 동일한 크기의 픽셀 그룹이다. 풀링 연산의 장점은 이미지로부터 가장 특징이 될 만한 픽셀 값을 추출하여 불필요한 정보를 생략하는 것이다. 풀링 연산은 $W\times H\times C$ 인 입력 영상의 픽셀(i,j)를 중심으로 하는 $N\times N$ 크기의 정사각형 배치 영역에 포함되는 픽셀을 P_{ij}로 표현한다. 이 P_{ij}내의 픽셀에 대해 채널 C마다 N^2개의 픽셀값으로부터 하나의 픽셀값 $y_{p(avg)}$ 을 구한다. 그림 2.5은 최대풀링의 예시이다.

$$y_{p(\max)} = \max P_{ij} \tag{2.2}$$

$$y_{p(avg)} = \frac{1}{MN}\sum_{M}\sum_{N} P_{ij} \tag{2.3}$$

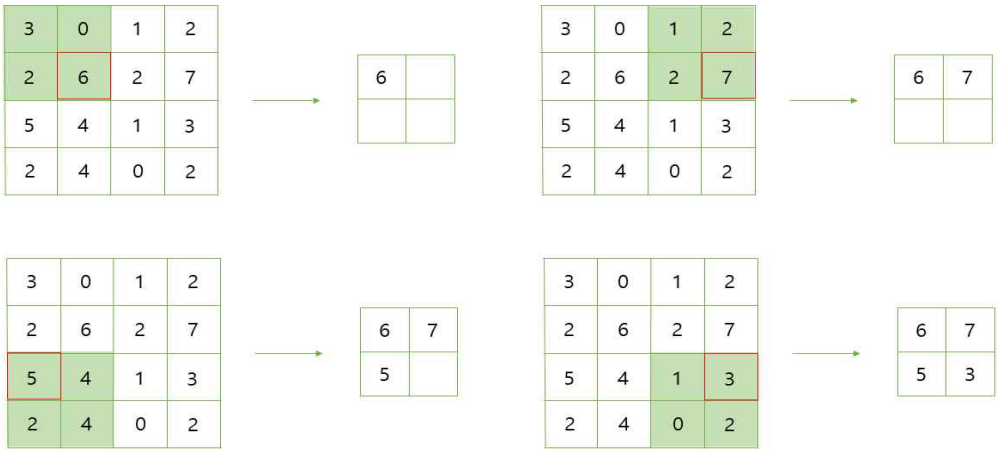


그림 2.5 최대 풀링(Max Pooling) 예시

컨볼루션 신경망의 합성곱 연산과 풀링 연산에는 스트라이드(Stride)와 패딩(Padding) 파라미터가 필요하다. 스트라이드는 이미지 내 연산을 수행하면서 픽셀의 이동 범위를 말한다. 그림 2.5에서 스트라이드는 2이며, 패딩은 0, 풀링커널 사이즈는 2×2이다. 스트라이드를 크게 설정하면 추출할 특징을 생략할 수 있고, 적게 설정하면 출력 특징맵들의 픽셀값 분포가 유사하게 추출될 수 있다. 패딩은 이미지의 가장자리의 정보를 유지하며, 패딩값을 줌으로써 출력 이미지의 크기를 조절할 수 있다.

완전연결층의 뜻은 한 층(Layer)의 모든 뉴런이 다음 층의 모든 뉴런과 연결된 상태를 말한다. 1차원 배열의 형태로 이미지를 분류하는데 사용된다. 그림 2.6은 완전연결층의 구조를 나타낸다.

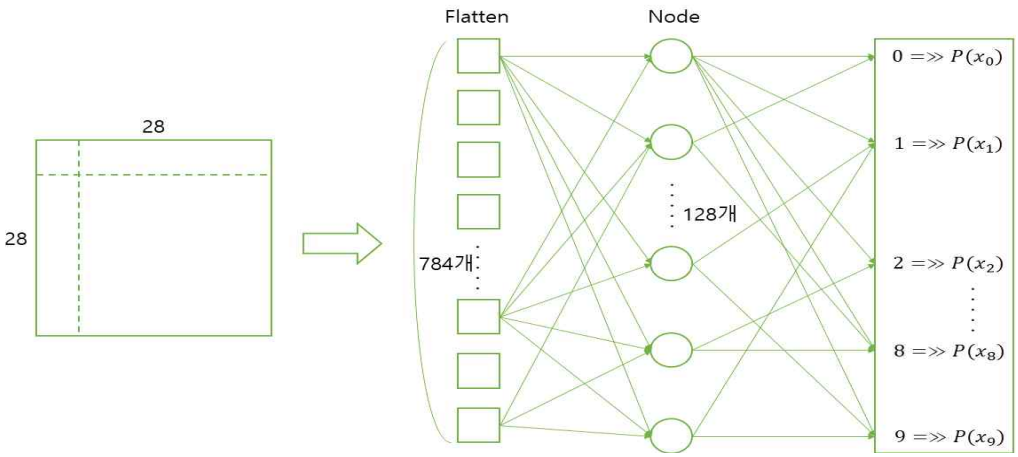


그림 2.6 완전연결층(Fully Connected Layer) 구조

2.2.1. 딥러닝 네트워크 구조

딥러닝은 깊은 신경망(DNN: Deep Neural Network) 알고리즘과 이를 학습하는 방법을 의미하며 인공신경망에서 출발했다. 입력층(Input Layer), 물체 종류 분류를 하는 출력층(Output Layer)과 특징을 추출하는 은닉층(Hidden Layer)이 2개 이상인 신경망으로 1980년대에 처음 제안되었으나 학습에 오랜 시간이 걸리고 학습데이터에 과적합(Overfitting)되는 단점 때문에 사용하지 않았다. 하지만 2000년대 이후 병렬 연산이 가능한 GPU의 발전으로 과적합을 방지할 수 있는 방법들이 제안되며 해결되었다. 인공신경망의 특징은 입력값에 일정한 가중치(Weight)를 곱해 목표값과 비교하여, 출력값을 목표값과 근접하도록 가중치를 조절하는 것이다. 대표적인 방법으로는 출력층에서 하나씩 입력층으로 되돌아가며 각 층의 가중치를 수정하는 역전파(Backpropagation)이다. 그림 2.7은 딥러닝의 한 종류로 영상인식에 주로 사용되는 컨볼루션 신경망 구조를 나타낸다.

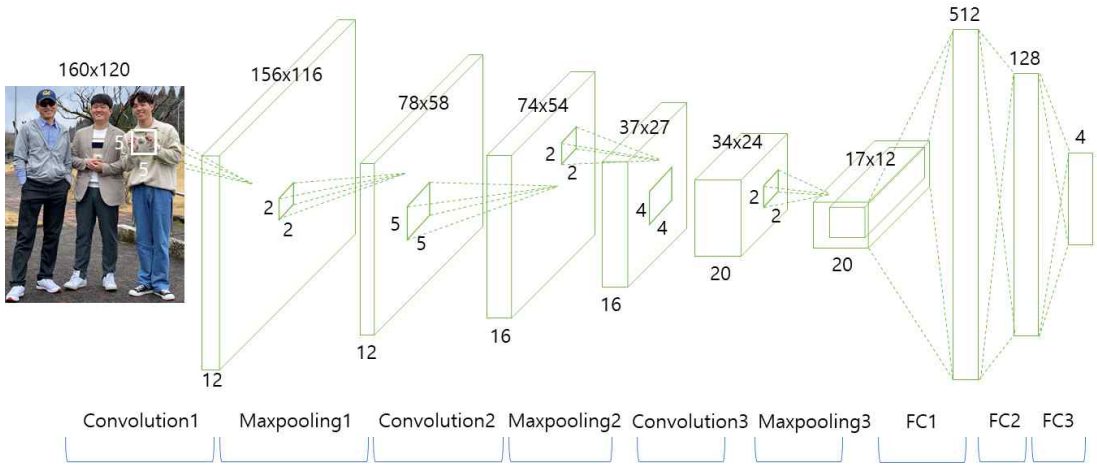


그림 2.7 컨볼루션 신경망(Convolution Neural Network)[12] 구조

2.2.2. 물체 검출 네트워크 연구 현황 분석

컨볼루션 신경망이 ImageNet 2012 대회에서 기존 방식의 성능을 뛰어넘는 결과를 보여주면서, 컨볼루션 신경망을 이용한 물체 검출 방법이 주목을 받기 시작한다. 물체의 위치를 검출하는 방법에 대한 연구로 R-CNN은 딥러닝 회귀(Regression) 방법을 사용했다. R-CNN의 단점이었던 느린 검출 속도를 보완하기 위해 Fast R-CNN과 Faster R-CNN이 개발되었지만, 자율주행과 같은 실시간으로 데이터를 처리해야하는 응용분야에 적용하기에는 충분하지 못했다. 이러한 처리속도 문제를 해결하기 위해 제안된 방법이 one-stage 방법이다. 대표적인 신경망으로는 SSD, YOLO(You Only Look Once) 등이 있다. 본 연구에서는 SSD과 Faster R-CNN 그리고 RetinaNet을 활용하여 물체 검출을 수행하였다.

2.2.2.1. R-CNN

R-CNN[13]은 크게 세 가지 모듈로 구성되어 있다. 첫 번째는 카테고리 독립적인 RP를 만드는 모듈이다. Selective search라는 알고리즘을 이용해 이미지에서 물체가 있을 것으로 추정되는 지역(Region)을 도출한다. 이 지역을 region proposals(RP)이라고 부른다. 두 번째 모듈은 각 RP에서 고정된 크기의 특징벡터를 추출하는 CNN이다. 각 지역에 대해 이미지 분류용 CNN을 이용해서 특징벡터를 추출한다. 세 번째는 클래스별 선형 서포트벡터머신(Support Vector Machine, SVM)이다. 서포트벡터머신을 활용하여 특징벡터와 관련된 클래스를 예측한다. 예측된 물체의 위치를 정확하기 파악하기 위해 Bounding Box regression을 통해 물체가 Bounding Box의 중앙에 위치하도록 조정한다.

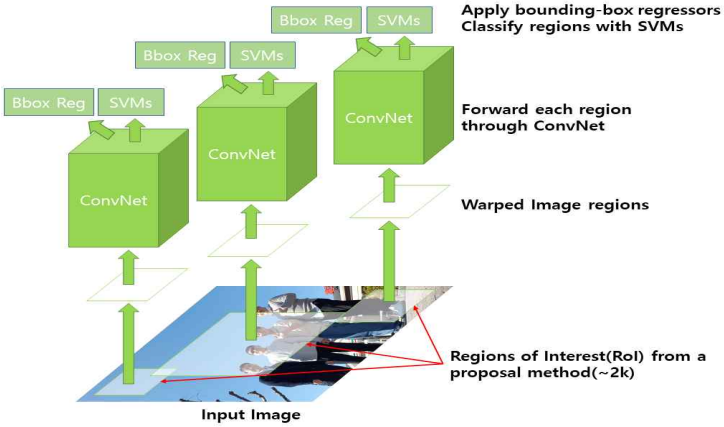


그림 2.8. R-CNN 구조

2.2.2.2. Fast R-CNN

SPPNet은 기존 R-CNN이 seletive search로 찾아낸 모든 RoI에 대해 CNN inference하는 문제를 전체 이미지에 1회만 수행하고, 이 피쳐맵을 공유함으로써 해결했다. 하지만 fully connected layer만 학습시킬 수 있는 한계점이 있었고, Fast R-CNN[14]에서는 하나의 모델에서 feature extraction, classification, bounding box regression을 학습시켜 문제를 해결하려 시도했다. Fast R-CNN의 가장 큰 특징은 end-to-end로 엮어서 모델을 학습시키며 학습 속도, 인퍼런스 속도, 정확도 모두를 향상시켰다는 것이다.

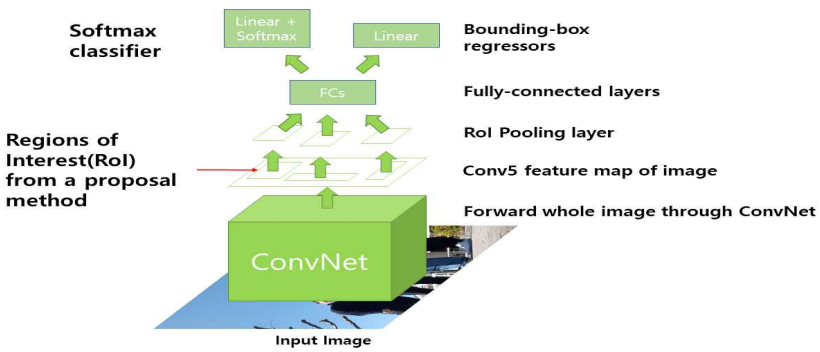


그림 2.9. Fast R-CNN 구조

2.2.2.3. Faster R-CNN

영역제안에 대해 CNN inference를 전체 이미지에 1회만 수행하고, 특징맵을 공유하면서 많은 시간을 절약했지만, 여전히 영역제안을 생성하는데 많은 시간이 소요된다. Faster R-CNN[15]에서는 영역제안네트워크(RPN: Region Proposal Network)를 학습시킴으로써, selective search를 수행하는 영역제안이 네트워크 외부에 존재하기 때문에 발생하는 bottleneck문제를 해결하였다. Faster R-CNN의 전체 구조는 다음과 같다.

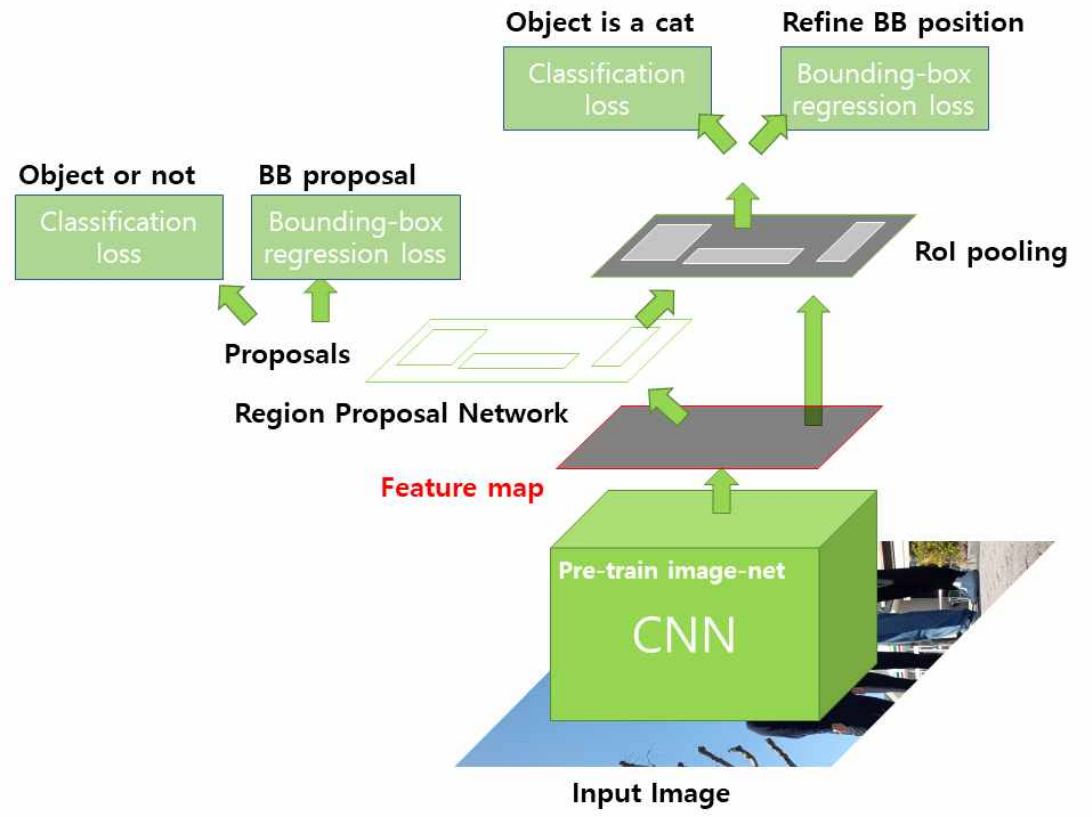


그림 2.10. Faster R-CNN 구조

2.2.2.4. Single Shot Detector(SSD)

SSD[16]는 VGG16 network의 conv5_3까지를 기본구조로 사용하며, 이후 single-shot detector와 Multi-scale feature maps을 이용하여 성능을 향상시켰다. Single-shot detector는 말 그대로 사진의 변형 없이 한 장의 사진으로 훈련 및 검출하는 detector이다. SSD에서는 다양한 크기의 feature map을 만들어서 큰 map에서는 작은 물체를 검출하고, 작은 map을 통해서는 큰 물체를 검출하는 Multi-scale feature maps 방법을 활용했다. Single Shot Detector(SSD)는 one-stage 물체 검출 네트워크로 VGGNet 등 특징 추출 레이어(Feature Extraction Layer)의 뒷부분에 연결되어 물체의 위치추정을 위한 별도의 네트워크를 사용하지 않고 Convolutional Layer에서 물체의 위치 영역 정보를 포함하는 특징맵(Feature Map)을 추출하기 때문에 빠르게 물체의 위치와 종류를 판별한다. SSD에서는 많이 발견되는 물체의 모양과 유사한 Default Box를 정의하고, Ground Truth Box와 Default Box의 차이를 학습하며, 다양한 해상도를 갖는 여러 단계의 특징맵에 연결되어 다양한 크기의 물체를 검출할 수 있다.

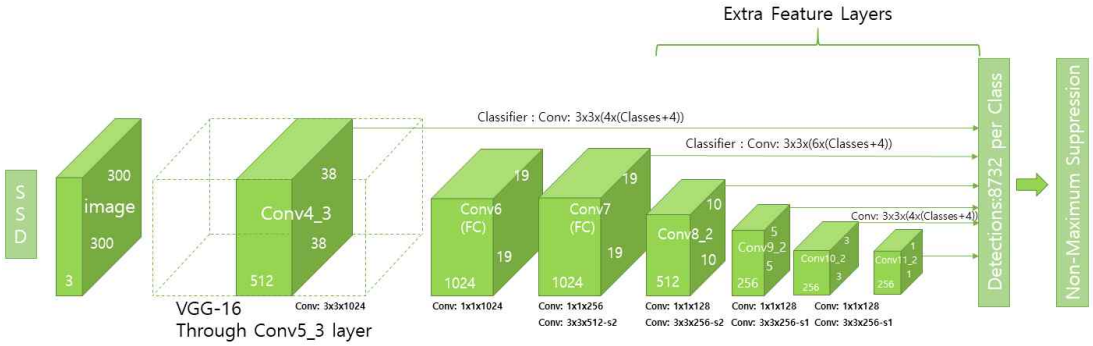


그림 2.11 Single Shot Detector(SSD) 구조

2.2.2.5. RetinaNet

RetinaNet[17]은 컨볼루션 신경망을 이용한 물체 검출 알고리즘이 가졌던 'Easy Negative Sample'이 모델을 압도해버리는 문제를 손실함수를 대체함으로써 해결했다. 분류기의 손실함수로 cross entropy 함수를 focal 손실 함수로 제안해, 종류 불균형(Class imbalance) 문제를 해결했다. Easy Negative Sample은 실제값도 negative이고 예측도 negative인 예측이 쉬운 데이터다. 컨볼루션 신경망이 RPN에서 2000개에 달하는 관심영역(RoI:Region of Interest)을 추출하는 과정에서 생산된다. 물체 후보(Object Proposal)이 생성될 때, 물체가 너무 크거나 작을 경우, 그리고 잘려 있어 일부만 보이는 경우에는 물체의 사각 테두리(Bounding Box)와 종류를 구별하기 어렵다. 반면에 적당한 크기의 물체들은 대부분 사각 테두리 안에 들어있습니다. 구별하기 쉬운 샘플(Easy Sample)과 어려운 샘플(Hard Sample) 갯수 비율의 차이 때문에, 손실 함수는 구별하기 쉬운 샘플에 쉽게 압도당해 모델의 성능이 더 이상 향상되지 않게 된다. 검출기의 성능은 어려운 샘플도 잘 검출하고 분류하는 것이기 때문에, 손실 함수에 끼치는 쉬운 샘플의 영향을 약화하고 어려운 샘플의 영향을 크게 하는 것이 RetinaNet에서 제안한 focal 손실함수이다.

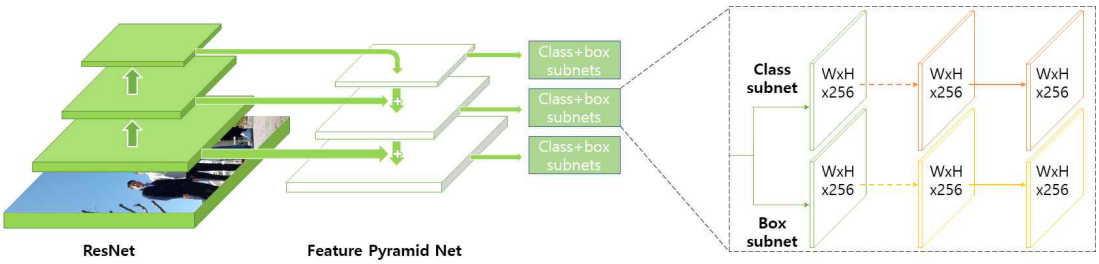


그림 2.12 RetinaNet 구조

Ⅲ. 드론영상을 활용한 물체 검출 연구

본 연구에서는 드론을 이용하여 데이터를 수집하고, 드론 영상과 컨볼루션 신경망을 이용하여 물체 검출을 하고자 한다. 사용한 컨볼루션 신경망은 SSD, Faster R-CNN, RetinaNet 등이 있으며, 각 컨볼루션 신경망을 이용하여 물체 검출에 적합한 CNN을 비교분석한다. 영상 내 물체 검출 문제는 존재 물체의 분류(Classification)와 위치(Position)의 두 문제를 해결하는 것을 의미하며, 결과로 물체의 위치정보를 사각 테두리로 나타내는 것이다. 물체의 분류는 영상에 있는 물체의 종류가 무엇인지 알아내는 문제이고, 위치(Position)추정은 영상에 있는 물체의 위치를 찾는 문제이다. 물체마다 모양과 크기, 내부 텍스처가 다르고 카메라와 물체사이의 거리, 각도에 따라 같은 물체도 다르게 보일 수 있기 때문에 물체 검출을 위해서 물체의 종류를 제한하여 구분지어 물체 검출을 진행한다. 연구 상세내용은 다음과 같다.

첫째, 드론을 이용하여 산림지, 관광지, 도심지 등 다양한 지역의 4k 고화질 영상을 취득하여, 18가지의 종류로 레이블링 작업하였다. 둘째, 각 지역별로 3가지 모델을 사용하여 학습을 진행한다. 셋째, 학습된 모델의 성능을 향상시키기 위해 병목(Bottleneck)기법을 사용한다.

구축된 전체 데이터셋은 학습(Train), 검증(Validation), 실험(Test) 데이터셋으로 나누게 된다. 모델학습을 위해 각 모델별로 영상의 크기가 다르게 전처리되어 사용되었다. 물체 검출을 위해 사용되는 영상의 종류는 카메라영상, CCTV영상, 드론영상 등이 있다.

3.1. 영상의 종류

영상의 종류는 일반 카메라(스마트폰, DSLR 등)영상, CCTV영상, 드론 촬영 영상으로 구성되어 있으며, 차이점은 촬영 각도 및 고도가 있다.

3.1.1. Camera 영상

일반적인 카메라 영상은 사람의 눈높이를 크게 벗어나지 않는 범위의 영상을 말하며, 사람이 바라보는 방향에 있는 물체의 정면, 측면, 후면 정보를 다루는 영상이다.

그림 3.1 사람 눈높이에 있는 영상

3.1.2. CCTV 영상

CCTV의 설치 높이는 대략 3~4m 사이이며, 대부분의 CCTV의 영상은 사람의 눈높이보다 높은 위치에서 물체를 아래로 내려 보기 때문에 영상은 조감이 된다. 이러한 조감영상으로부터 사물을 탐색추출, 인식한 방법은 정면 영상의 다른 촬영 각도가 적용된다.

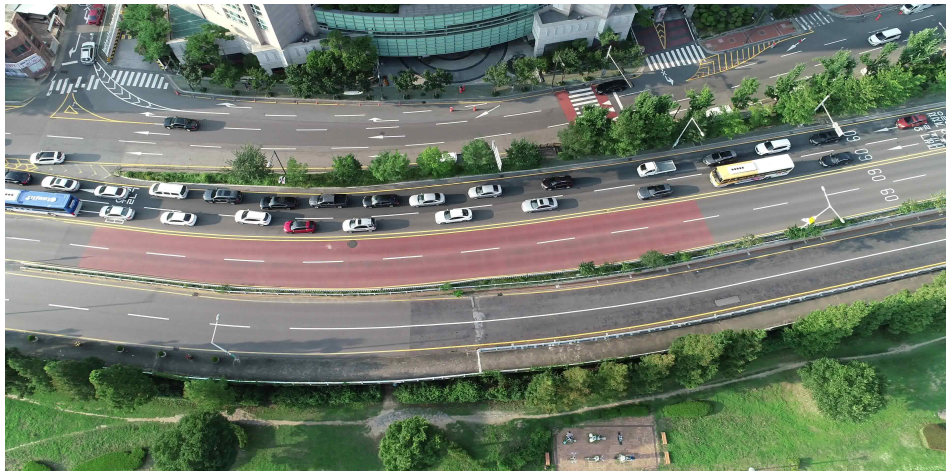
그림 3.2 CCTV 영상[18]

출처: https://aihub.or.kr/aidata/139

CCTV 영상에서 움직이는 물체를 검출하기 위해서는 크게 3단계를 거친다. 각 단계는 영상 영역 분할(Segmentation), 배경 추출(Extraction), 물체 영역 다듬기(Refinement)이며, 같은 물체를 나타내는 픽셀들은 대체적으로 유사한 색상값을 가지고 비슷한 영역에 위치한다. 따라서 비슷한 영역의 유사한 색상값을 가지는 픽셀들을 그룹화하여 물체의 후보들을 만드는 영역 분할을 우선적으로 실시한다.

3.1.3. 드론 영상

본 연구에서 사용할 드론 영상은 30m, 60m, 100m 등 다양한 고도에서 촬영하였으며, 물체의 측면과, 상평면의 정보를 담고 있다. 같은 물체라도 드론의 고도, 영상촬영 각도에 따라 물체 이미지의 해상도와 비율이 다양하게 나타날 수 있으며, 물체 검출에 많은 영향을 미친다. 다음 그림은 고도와 촬영 각도에 따른 촬영 이미지다.



a) 고도: 80m, 각도: 60도

b) 고도: 30m, 각도: 15도

c) 고도: 100m, 각도: 90도
그림3.3 고도와 각도에 따른 드론 영상

3.2. 컨볼루션 신경망을 이용한 물체 검출

본 연구에서는 드론을 이용하여 데이터를 수집하고, 드론 영상과 컨볼루션 신경망을 이용하여 물체 검출을 하고자 한다. 사용한 컨볼루션 신경망은 SSD, Faster R-CNN, RetinaNet 등이 있으며, 각 CNN을 이용하여 물체 검출에 적합한 CNN을 비교분석한다.

3.2.1. SSD

구축한 데이터 셋의 객체 이미지의 해상도와 비율이 다양하고 드론 영상 데이터의 해상도가 너무 높기 때문에, 컨볼루션 신경망 입력을 위해 전체 크기의 이미지를 일정한 크기로 변환한다. 다음 그림 3.4는 본 논문에서 사용한 CNN 구조이다. MMdetection 툴을 활용하여 SSD를 간략화한 구조이다. 입력 이미지는 2048 x 2048, 1024 x 1024 픽셀 크기의 정방형 컬러 이미지이며, VGG16의 Conv4_3 layer를 base network로 두고 처리하면 2048 x 2048 x 3, 1024 x 1024 x 3의 이미지가 256 x 256 x 512, 128 x 128 x 512의 특징맵을 뽑아낸다. SSD에서는 multi feature maps에 해당하는 부분으로 256 x 256, 128 x 128, 64 x 64, 32 x 32, 16 x 16, 8 x 8의 특징맵을 활용하여 결과값을 도출한다. 각 특징맵에서 conv 연산을 통해 예측하고자하는 bounding box의 class 점수와 offset을 계산한다. Convolution filter size는 3 x 3 x (# of BBox x (class score + offset)) 이다. 계산된 bounding box의 결과값을 다 활용하면 계산량이 너무 많아지기 때문에, 각 특징맵에 다른 스케일을 적용해 default box 사이의 IOU를 계산하고, 0.5 이상 되는 box들만 대상에 포함시키고 나머지의 박스는 모두 고려하지 않는다. 이후 Non-maximum suppression을 통해 최종 detect된 결과값을 얻을 수 있다.

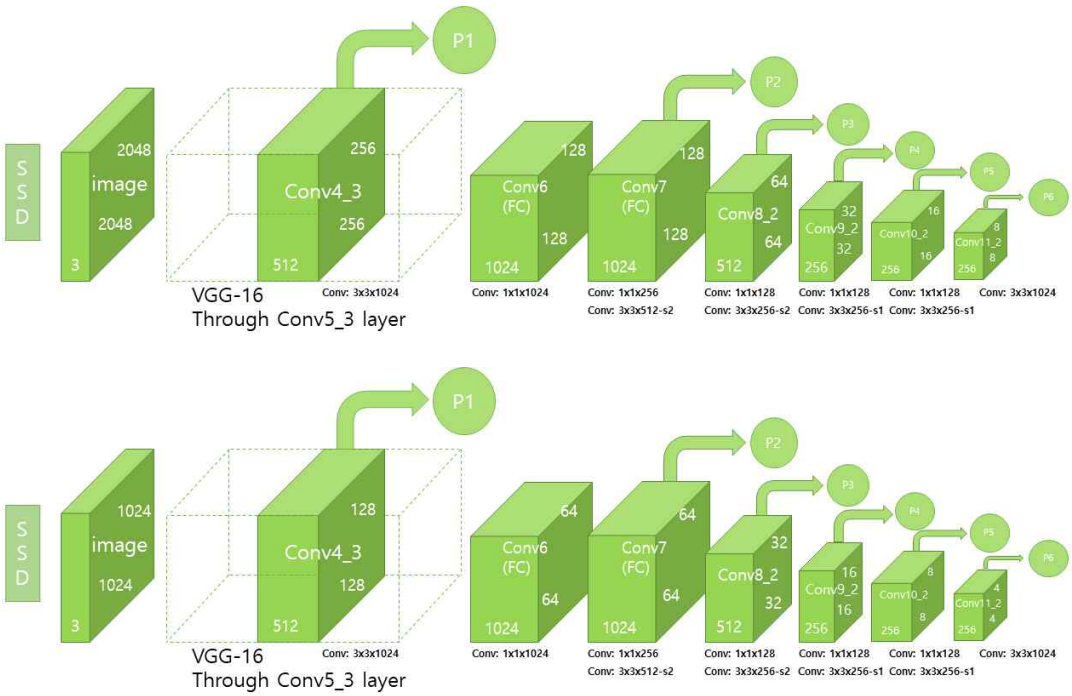


그림3.4 2048, 1024 입력이미지를 활용한 Single Shot Detector Architecture

3.2.2. Faster R-CNN

그림 3.5은 Fast R-CNN을 더욱 속도를 실험에 사용된 Faster R-CNN의 구조이다. 입력 이미지는 2880 x 1620 픽셀 크기의 컬러 이미지이며, ResNeXt의 Conv5를 backbone network로 두었다. 다음 그림의 backbone 부분에서 확인할 수 있듯, 2880 x 1620 이미지가 720 x 405, 360 x 202, 180 x 101, 90 x 50의 크기로 축소되며 각각의 채널은 256, 512, 1024, 2048개의 채널수를 가진다. Neck부분에서는 Feature Pyramid Network(FPN)를 활용했으며, Backbone에서 추출된 특징맵의 채널수를 256으로 통일시킨다. 이후 Region Proposal Network의 Head 부분에서 채널수를 256으로 통일시키며, proposal 할 영역을 학습하여 anchor ratio의 x_min, y_min, x_max, y_max와 같은 위치 정보를 추출한다. 추출된 위치 정보를 기반으로 특징맵을 cropped시킨 후 7 x 7크기로 제한된 크기의 특징맵으로 구성한다. 그리고 변환된 특징맵을 통해 물체의 class를 분류한다. feature map size는 5가지, anchor ratio는 3가지, class는 1가지, 그리고 x, y좌표의 값을 총 4가지 가지게 된다.

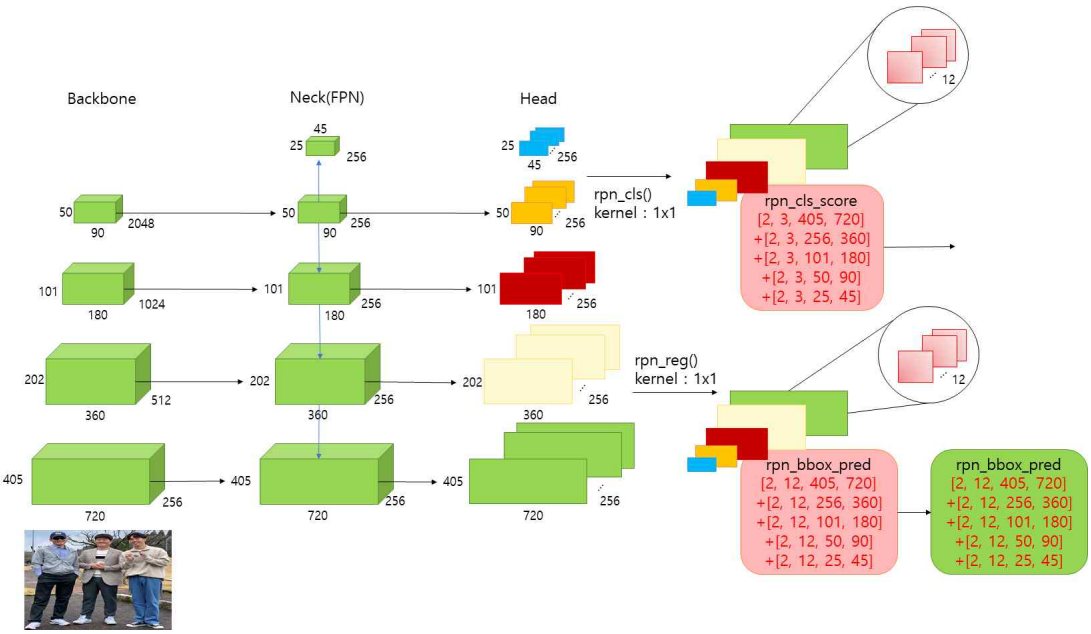


그림 3.5 2880 x 1620 입력이미지를 활용한 Faster R-CNN Architecture

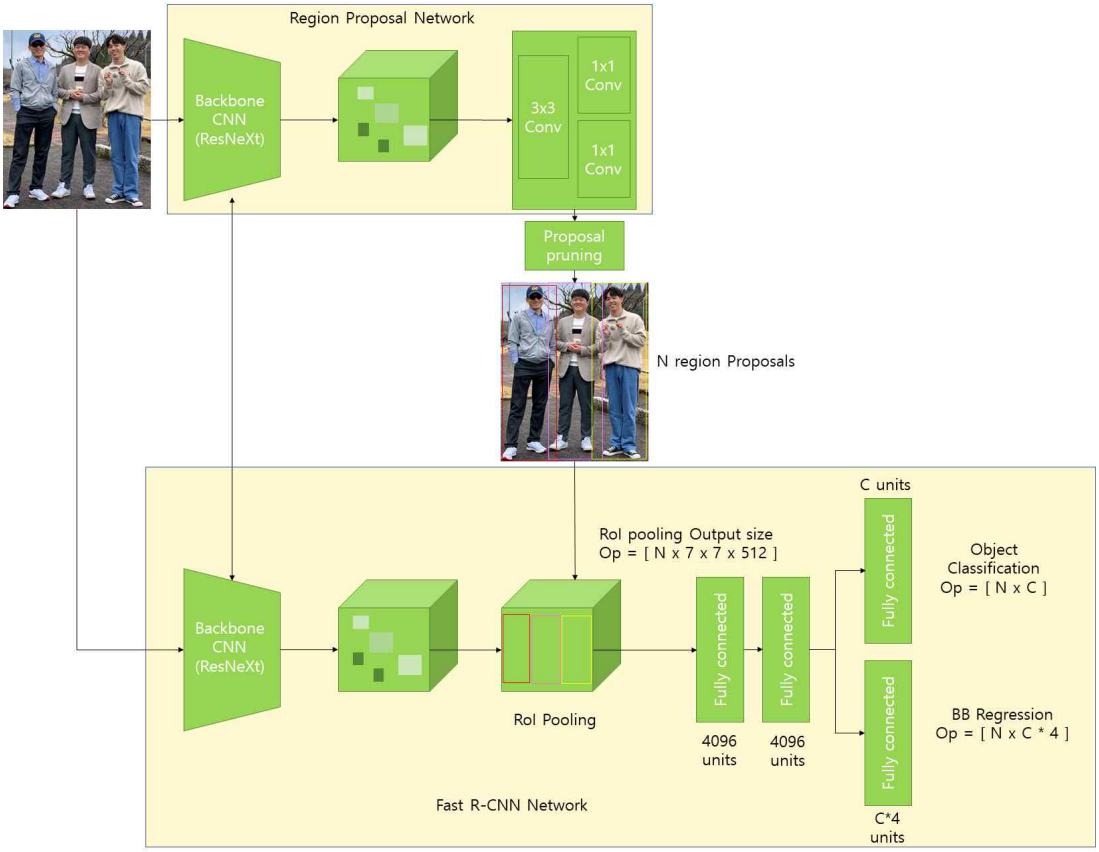


그림 3.6 Faster R-CNN Architecture

3.2.3. RetinaNet

그림 3.7는 RetinaNet의 구조이다. 입력이미지는 1024 x 1024 픽셀 크기의 컬러 이미지를 사용했으며, backbone network로 특징피라미드네트워크(FPN: Featrue Pyramid Network)를 사용했다. 특징피라미드네트워크를 사용하여 풍부한 다중 스케일 컨볼루션 특징 피라미드를 생성합니다. 특징 피라미드에 두 개의 하위 네트워크를 연결하여, 하나는 앵커박스를 분류하기 위한 네트워크이고, 다른 하나는 앵커박스에서 실제 물체 박스로 회귀하기 위한 네트워크이다. 피라미드 레벨은 P3에서 P7이 되도록 사용했으며, 모든 피라미드 레벨의 채널은 256로 동일하다.

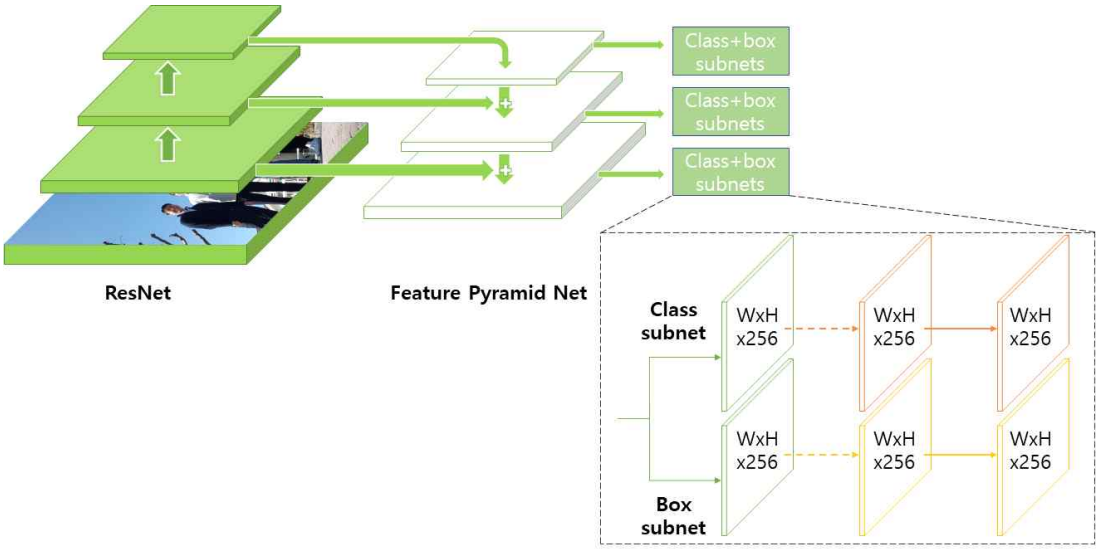


그림 3.7 1024 x 1024 입력 이미지를 활용한 RetinaNet Architecture

Ⅳ. 실험결과

4.1. 데이터셋 구성 및 실험환경

본 연구에서는 드론을 이용하여 산림지, 관광지, 도심지 등 다양한 지역의 비행 도정에서 고화질 영상을 취득하며, 취득 영상을 물체 검출의 학습 및 시험 데이터로 활용한다. 드론의 기본 비행 즉, 이착륙, 고도안전비행 및 임무비행을 위해 다양한 상황 영상을 취득할 필요가 있었고, 상황별 안전한 비행을 위해 물체 검출 및 인식이 가능하도록 인공지능 학습 모델용 영상 데이터를 수집하였다. 드론 1회 비행당 5분 이상, 10분 이상, 20분 이상 등으로 구별된 비행영상으로 구성되어 있으며, 드론의 비행성능과 안전성을 평가할 수 있는 항목, 지표, 시험방법 등 성능평가체계 기준이 되는 데이터, 내풍성, 비행시간, 최대하중, 통신성능, 충동 방지 및 회피의 정보를 동시에 수집하였고, 비행시각, 고도, 위도, 경도 등 드론의 경로 정보(Track record)를 구축하였다. 데이터 포맷은 다양한 시스템에서 널리 사용되고 있는 JSON 포맷을 사용하였으며, 다른 포맷(TFRecord. hdf5 등)의 형태로 변환하여 사용이 가능하다. 산림지, 관광지, 도심지에서 쉽게 발견할 수 있는 물체들을 물체 검출 대상으로 정의했으며, 정의된 물체 정보는 다음과 같다.

표 4.1 지역별 물체 종류

구분 / 번호	관광지	도심지	산림지
1	나무	나무	나무
2	사람	사람	사람
3	동물	동물	동물
4	주택	주택	관리소
5	아파트, 빌딩	아파트, 빌딩	정상표식
6	학교	학교	오토바이, 자전거
7	관리소	관리소	
8	교통표지판	교통표지판	
9	신호등	신호등	
10	가로등, 전신주	가로등, 전신주	
11	현수막	현수막	
12	다리	다리	
13	상징탑	상징탑	
14	승용차	승용차	
15	버스	버스	
16	트럭	트럭	
17	오토바이, 자전거	오토바이, 자전거	

드론 촬영 영상은 3840x2160의 해상도를 가진 4k로 촬영되었으며, 학습 모델을 훈련시키기 위해서 SSD에서는 1024x1024, 2048x2048로 입력 이미지 크기를 수정하여 사용했다. Faster R-CNN은 2880x1620의 입력 이미지를 사용했으며, RetinaNet에서는 1024x1024의 입력 이미지만을 사용했다.

CNN 코드는 Pytorch를 사용하여 작성되었다. OS는 리눅스 우분투 18.04버전을 사용했으며, 실험장비로는 Intel I-9 10세대 10900k, Geforce RTX 2080Ti 11GB 4개, Intel Xeon Gold, NVIDIA Tesla V100 32GB를 사용하여 실험을 진행했다. 학습을 위해 전체 데이터의 60%(16,743장)를 랜덤으로 선택하여 학습데이터로 사용하였고, 나머지 40%(11,160장)의 20%(5,580장)를 각각 검증, 테스트용으로 사용하였다.

표 4.2 물체 데이터 정보

번호	종류	Train	Validation	Test	Total
1	나무	20,064	3,953	5,367	29,384
2	사람	348	17	1,961	2,326
3	동물	-	-	-	-
4	주택	101	53	78	232
5	아파트, 빌딩	759	1,168	1,079	3,006
6	학교	-	-	-	-
7	관리소	-	-	-	-
8	교통표지판	786	488	605	1,879
9	신호등	1,415	5,137	3,423	9,975
10	가로등, 전신주	4,849	4,113	5,761	14,723
11	현수막	1,079	533	172	1,784
12	정상표식	-	-	-	-
13	다리	9	2	3	14
14	상징탑	-	-	-	-
15	승용차	63,388	48,114	42,460	153,962
16	버스	1,943	1,337	1,550	4,830
17	트럭	4,411	6,016	3,757	14,184
18	오토바이, 자전거	533	670	533	1,736

아파트

배너

전봇대

사람

표지판

자동차

오토바이

신호등

버스

트럭

집

다리

그림 4.1 물체 검출을 위해 제안된 드론 영상 데이터셋

4.2. 물체 검출 성능 평가지표

본 실험에서는 물체 검출 알고리즘의 성능을 평가하기 위해 mAP를 사용하였다. mAP값을 구하기 위해서는 정밀도(Precision)과 재현율(Recall) 값이 필요하다.

표 4.3

Actual / Predicted	True	False
Positive	True Positive	False Positive
Negative	True Negative	False Negative

정밀도는 물체검출기가 검출한 정보들 중에서 Ground-Truth와 일치하는 비율을 의미한다. 모든 검출 결과 중 제대로 옳게 검출한 비율을 의미하며 다음 수식 (4.1)과 같이 나타낼 수 있다. TP(True Positive)는 “옳은 검출”을 의미하며, FP(False Positive)는 “잘못된 검출”을 의미한다.

$$\mathrm{Pr}ecision = \frac{TP}{TP+FP} = \frac{TP}{All\ Detections} \tag{4.1}$$

재현율은 실제 정답 중 얼마나 제대로 검출에 성공했는지를 나타낸다 다음 수식 (4.2)와 같이 나타낼 수 있다. FN(False Negative)는 “검출되었어야 하는 물체인데 검출되지 않은 것”을 의미한다.

$$Recall = \frac{TP}{TP+FN} = \frac{TP}{All\ Ground\ Truths} \tag{4.2}$$

정밀도 혹은 재현율만으로 물체 검출을 위한 컨볼루션 신경망의 성능을 평가하는 것은 적절하지 않다. 예를 들어, 실제 검출되어야 하는 물체가 10개일 때, 컨볼루션 신경망이 검출해낸 물체가 5개이고 그중 4개가 정답이라고 가정한다. 이때 정밀도 = 4/5 = 0.8이고, 재현율 = 4/10 = 0.4 이다. 정밀도로 보면 성능이 좋아 보이지만, 재현율로 보면 성능이 좋지 않다. 정밀도와 재현율은 항상 0과 1사이의 값을 가지며, 정밀도가 높으면 재현율이 낮은 경향이 있고, 정밀도가 낮으면 재현율이 높은 반비례적 경향이 있다. 따라서 정밀도와 재현율, 두 값을 종합해서 학습 모델을 평가해야한다. 그래서 정밀도-재현율(Precision-recall) 곡선 및 AP가 필요하다. 정밀도-재현율 곡선은 confidence 레벨에 대한 임계값의 변화에 의해 물체 검출기의 성능을 평가하는 방법으로, confidence 레벨이란, 검출한 물체에 대해 검출 알고리즘이 얼마나 확신이 있는지 알려주는 값이다. Confidence 레벨에 대해 임계값을 부여해서 특정값 이상이 되어야 검출된 것으로 인정한다. 이 confidence 레벨에 대한 임계값에 따라 정밀도와 재현율 값들도 달라진다. 이것을 그래프로 표현한 그래프가 정밀도-재현율 곡선이다. 그림 4.2은 정밀도-재현율 곡선 그래프이다. x축은 재현율 값이고, y축은 정밀도 값이다. 정밀도-재현율 곡선에서는 재현율 변화에 따른 정밀도 값 혹은 정밀도 변화에 따른 재현율 값을 확인할 수 있다. AP는 정밀도-재현율 그래프의 그래프 선 아래쪽 면적으로 계산된다.

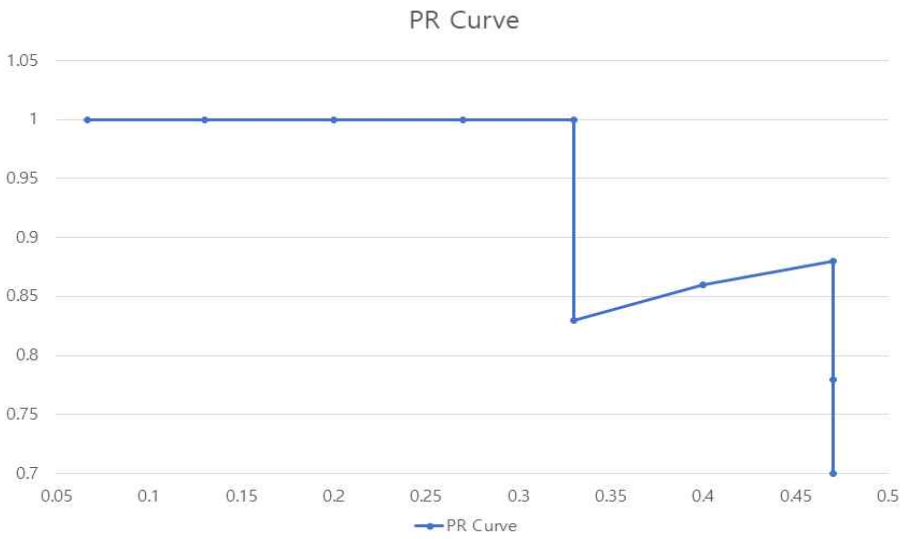


그림 4.2 정밀도-재현율 곡선

물체의 종류가 여러 가지인 경우, 각 종류의 AP를 구하여 평균을 구한 것이 mAP이다.

4.3. 물체 검출 및 분류 모델 결과 및 분석

본 실험에서는 드론영상 데이터셋을 활용하여 물체검출을 위한 컨볼루션 신경망을 학습시켰다. 산림지, 관광지, 도심지 지역별로 모델을 학습시켜 비교하였으며, 학습모델로는 SSD, Faster R-CNN, RetinaNet, Bottleneck을 적용한 SSD 총 4가지를 사용하였다.

표 4.4 지역별 물체 검출 모델 결과

지역	학습모델	BBox mAP	BBox mAP_50	BBox mAP_75	BBox mAP_s	BBox mAP_m	BBox mAP_l
산림지	SSD	0.605	0.935	0.684	0.225	0.555	0.594
	Faster R-CNN	0.714	0.959	0.947	-	0.045	0.717
	RetinaNet	0.621	0.716	0.628.	0.27	0.331	0.744
	SSD Bottleneck	0.695	0.984	0.812	0.479	0.649	0.752
관광지	SSD	0.431	0.717	0.465	0.035	0.266	0.619
	Faster R-CNN	0.711	0.975	0.82	0.433	0.617	0.791
	RetinaNet	0.504	0.657	0.583	0.023	0.264	0.749
	SSD Bottleneck	0.641	0.878	0.742	0.2	0.515	0.769
도심지	SSD	0.513	0.801	0.576	0.075	0.368	0.636
	Faster R-CNN	0.75	0.964	0.876	0.2	0.671	0.82
	RetinaNet	0.55	0.696	0.617	0.143	0.42	0.724
	SSD Bottleneck	0.609	0.918	0.703	0.4	0.473	0.745

V. 결 론

본 연구에서는 드론영상을 활용하여 컨볼루션 신경망을 학습하여 컴퓨터 비전 및 영상처리의 한 분야인 물체 검출 알고리즘을 분석하였다. 사용된 컨볼루션 신경망 모델은 SSD, Faster R-CNN, RetinaNet, Bottleneck을 적용한 SSD 등 one-stage 와 two-stage 네트워크를 모두 사용하여 비교했다. 모델을 학습시키기 위해서 산림지, 관광지, 도심지 각 지역별로 드론을 통해 4k 고화질 영상을 촬영했으며, 지역별로 6, 17, 17개의 종류로 어노테이션 작업을 수행했다. 4k 고화질 영상을 사용하여 모델을 학습시킬 수 없었기 때문에, 각 모델별로 입력 이미지 크기를 수정하여 학습시켰다. SSD, Bottleneck이 적용된 SSD, RetinaNet은 1024 x 1024, Faster R-CNN 2880 x 1620 크기를 입력 이미지로 사용했다. 사용한 드론 데이터셋의 종류별 개수가 균등하지 않은 상태로 SSD 모델을 먼저 학습시켰다. IoU 50에 대한 결과값은 산림지, 도심지, 관광지 순으로 0.935, 0.801, 0.717 mAP를 보여주었으며, 산림지의 mAP가 가장 높았다. Faster R-CNN의 IoU 50에 대한 결과값은 0.959, 0.975, 0.964 mAP로 도심지의 mAP가 가장 높았다. Faster R-CNN은 SSD보다는 결과값이 좋았지만, 실시간에 적용해서 사용할 fps는 나오지 않았다. RetinaNet은 0.716, 0.657, 0.617 mAP로 산림지가 가장 높았다. Bottleneck을 적용한 SSD는 0.984, 0.878, 0.918 mAP로 산림지가 가장 높았다. Faster R-CNN을 제외한 학습모델은 산림지에서 결과값이 가장 좋게나왔다. 산림지의 결과값이 높게 나온 이유는, 산림지 지역 영상에 많은 종류의 물체가 없었기 때문에 상대적으로 관광지 및 도심지보다 모델을 학습하기 쉬웠을 것이다. SSD 모델의 Extra Feature Layers 연산 시간을 줄이기 위해서 1x1 컨볼루션을 활용하여 bottleneck을 적용시켰다. 결과적으로 3x3 컨불루션 2개를 곧바로 연결시킨 구조에 비해 연상량을 절감시킬 수 있었다. 향후에는 데이터확장(Data Augmentation) 알고리즘을 활용하여 지역별 데이터셋의 물체 종류를 골고루 분포되도록 만들어, 모델을 학습시킬 예정이다. 실시간으로 드론영상에서 물체검출을 충분히 수행할 수 있게 되면 향후, 드론영상을 통해 추출한 사람 및 물체의 정보를 이용해 사람의 행동분석에도 활용할 수 있을 것이라 예상된다.

본 연구는 드론의 영상 데이터를 활용하여 컨볼루션 신경망을 학습시켜 물체 검출 연구에 적합한 모델을 제안했다. 제안된 모델을 활용하여 드론에서 실시간 물체검출 및 인간행동 분석이 가능하도록 어플리케이션을 개발할 수 있을 것으로 기대된다.

참 고 문 헌

[1] Nastaran Dadashi, “Automatic Surveillance and CCTV Operator Workload”, University of Nottingham, 2008.

[2] Haar A, “On the Theory of Orthogonal Function Systems”, Mathematische Annalen, 69, pp. 331-371, 1910.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1, pp. 886-893, 2005.

[4] Lowe, David G, "Object recognition from local scale-invariant features", Proceedings of the International Conference on Computer Vision, 2, pp. 1150-1157, 1999.

[5] DC. He and L. Wang, "Texture Unit, Texture Spectrum, And Texture Analysis", Geoscience and Remote Sensing, pp. 509-512, 1990.

[6] B. Froba and A. Ernst, "Face detection with the modified census transform," Sixth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 91-96, 2004.

[7] Cortes, Vapnik, “Support-vector networks”. Machine Learning, 1995.

[8] Freund Yoav, Schapire Robert E, "A decision-theoretic generalization of on-line learning and an application to boosting". Journal of Computer and System Sciences, 55, pp. 119-139, 1997.

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models," IEEE Transactions on Pattern Analysis and Machine Intelligence, 32, pp. 1627-1645, 2010.

[10] A. Krizhevsky, I. Sutskever, G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, Advances in Neural Information Processing Systems, 2012.

[11] Schmidhuber J. “Deep Learning in Neural Networks: An Overview”, 61, pp. 85-117, 2015.

[12] Karen Simonyan, Andrew Zisserman, “ Very Deep Convolutional Networks for Large-Scale Image Recognition”, Computer Vision and Pattern Recognition, 2015.

[13] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation”, Computer Vision and Pattern Recognition, 2013.

[14] Ross Girshick, “Fast R-CNN”, Computer Vision and Pattern Recognition, 2015.

[15] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”, Computer Vision and Pattern Recognition, 2015.

[16] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg, “SSD: Single Shot MultiBox Detector”, Computer Vision and Pattern Recognition, 2016

[17] T. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal Loss for Dense Object Detection," IEEE International Conference on Computer Vision, pp. 2999-3007, 2017.

[18] 이상행동 CCTV 영상 AI데이터, https://www.aihub.or.kr/aidata/139