

# 산학연계 캡스톤 프로젝트 최종보고서

**팀별 작성, 반드시 한글파일로 제출**

과제 수행원 현황						
수행 학기	<input type="checkbox"/> 2022년 3월 ~ 2022년 6월 (2022학년도 2학기)					
프로젝트명	Object Detection Model 비교					
팀명	모로					
	학과	학번	성명	학년	연락처	E-mail
팀장	컴퓨터전자시스템공학과	201903868	황가은	4	010-2559-5347	idhge12@gmail.com
팀원	컴퓨터전자시스템공학과	201900919	김영서	4	010-3930-0586	serakim002@hufs.ac.kr
	컴퓨터전자시스템공학과	201901489	박서린	4	010-6671-1566	seorin0313@naver.com
	컴퓨터전자시스템공학과	201903566	최서영	4	010-8762-9339	226tjdud@naver.com
학부생멘토						
지도교수	교과목명	캡스톤설계및실습				
	소속	<input checked="" type="checkbox"/> 컴퓨터전자시스템공학부 <input type="checkbox"/> 정보통신공학과 <input type="checkbox"/> AI교육원 <input type="checkbox"/> 기타( )				
	성명	두일철				
산업체	기업명	(주)디엠항공교역				
	멘토 성명	김태진	멘토 직위		대표	

과제 일반 현황					
작품(과제)명	Object Detection Model 비교				
특허·실용신안					
포상여부	상격	기관	행사명	수상일시	부상내역
	※ 포상실적은 해당사항이 있을시 필히 기재 요망. ※ 포상실적을 허위로 기재시 신청인은 포상대상에서 제외됨 ※ 타기관에서 이미 수혜받은 정부포상 과제는 포상대상에서 제외됨				

## 보고서

작품명  
(프로젝트명)

Object Detection Model 비교

# Key Words

RetinaNet

YOLO v3

DETR

Faster R-CNN

Cascade R-CNN

1. 개발동기/  
목적/필요성및  
개발 목표

\* 개발동기 및 목적:

항공기 활주로, 계류장, 공항 주변에는 지형 지물인 건물과 산, 지상 이동체인 항공기와 차량 등이 혼재한다. 이처럼 시각화된 3차원 공간을 정량적으로 평가하기 위해서는 시각 영상의 데이터화가 필요하다.

\* 개발목표:

Object Detection Model 간의 성능 평가를 통해 계류장과 공항이라는 배경에서 목적에 맞는 최적의 모델 선정

2. 최종  
결과물 소개

- API 성능 비교 사진

5. 통합

	카카오 API	ETRI API	Google API	Microsoft API
활용 용도	상품, 얼굴, 텍스트 검출 등 다양한 기능이 있지만 필요한 기능이 아님.	비행기 검출 가능, 사람, 자동차 등 다양한 종류의 객체 인식.	비행기, 버스, 트럭 등 인식. 로고나 문구도 인식 가능.	객체 감지, 이미지 분류 및 설명, 텍스트 추출 등 여러가지 기능을 제공한다.
활용 가능성	비행기라는 객체를 인식을 해야 하는데 객체 인식을 못함. 그러므로 활용 가능성이 없음.	유용하다고 판단함. - 비행기 외에도 총 80개의 카테고리 객체 인식 가능. - 그러나 비행장 관련 활주로, 건물 등은 포함되지 않는다는 점이 아쉬움. - 위치 정보(x, y 좌표)나 신뢰도 정보 등 부가 정보를 제공하므로 유용할 것으로 보임.	다양한 객체를 인식 가능함. 그러나 활주로나 건물 등은 인식 안 됨. 겹쳐있는 것도 인식 잘 안 됨.	비행기, 하늘 인식, 하늘을 날고 있는 것까지 인식이 가능하다. 따라서 신뢰도가 있고 활용 가능성도 있어 보임.
주제와 취지의 부합성	주제와 취지에 부합하지 않은 API임.	“비행장, 공항 이미지를 활용한 데이터화”라는 주제를 바탕으로, 주제에 특화된 API는 아니지만 효과적으로 활용될 가능성이 있다고 생각함.	비행기 검출 목적으로는 취지와 부합하다고 할 수 있으나 활주로 등은 인식하지 못함.	비행기 검출이 주 목적이므로 주제와 취지에 부합한다고 볼 수 있음.
가격	확인 안됨.	1일 250번 제한	단위 1,000개 미만 무료 (월)	월별 트랜잭션 5000개 무료

결론

동일한 사진이어도 객체 검출되는 API가 있고, 되지 않는 API가 존재함

어떤 것이 항공 객체 검출에 효율적인지 확실하게 판단 불가.


카카오 API는 연구 주제의 취지와 부합하지 않아 제외, 나머지 3개의 API는 무료로 사용 가능

상단에 우리가 설정한 기준 표를 기반으로 3가지 중 Microsoft API가 가장 적합하다고 판단함. 하지만 실제적으로 구성을 이해하고 조정하기에 Open API를 다루는 것이 올바르지 않다고 생각  
→ 2차적으로 Open Source를 조사하기로 결정

- 최종 결과물은 레포트 형식으로 레포트의 일부분 사진으로 첨부하며, 해당 결과물은 개별적으로 제출하였습니다.

## Object Detection

- 캡스톤 설계 및 실습 -



14조 모로  
201900919 김영서  
201901489 박서린  
201903566 최서영  
201903868 황가은

### III. One-stage Detection

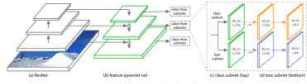
#### 1. RetinaNet

a) 등장 배경

One-stage detector는 Two-stage detector보다 속도는 빠르지만 성능은 떨어진다는 단점이 존재한다. RetinaNet 자체는 one-stage detector의 낮은 정확도의 원인은 객체와 배경 클래스 불균형이 원인이라는 것을 발견하였다. Focal Loss는 one-stage detector에서도 클래스 불균형 문제를 해결할 수 있도록 한다. Loss function을 수정하여 예측하기 쉬운 example에는 0에 가까운 loss를 부여하고 예측하기 어려운 negative example에는 기존보다 높은 loss를 부여한다. Focal Loss의 효과를 입증하기 위해 one-stage detector인 RetinaNet을 설계한다.

b) 구조 및 진행 과정

하나의 backbone network와 classification 및 bounding box regression을 수행하는 2개의 subnet으로 구성한다. Backbone network는 convolutional feature map을 추출한다. 정해진 subnet은 object classification을 수행한다. 두 번째 subnet은 bounding box regression을 수행한다. RetinaNet은 일정한 예측하기 어려운 hard example에 집중하도록 하는 Focal Loss를 제안한다. RetNet과 FPN을 활용하여 구축한 one-stage 모델인 RetinaNet은 focal loss를 사용하여 two-stage 모델 Faster R-CNN의 정확도를 높인다. 이를 통해 One-stage detector의 빠른 detection 시간의 장점을 가지면서 One-stage detector의 detection 성능 저하 문제 개선하였다.



(그림 1) RetinaNet Architecture

1. Li, L., P. Qian, R. Girshick, A. R. He and P. Dollar "Focal Loss for Dense Object Detection"

---

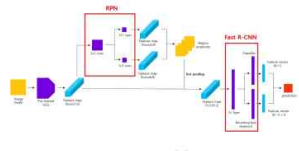
### IV. Two-stage Detection

#### 1. Faster R-CNN

a) 등장 배경

기존 Fast R-CNN 모델은 selective search 알고리즘을 통해 region proposals을 추출하기 때문에 학습 및 detection 속도를 향상시키는데 한계가 있다. 또한 detection 과정을 end-to-end 방법으로 수행하지 못한다는 문제가 있다. 이러한 문제를 해결하기 위해 여러 속도와 모델의 한성도를 개선한 Faster R-CNN 모델을 등장하였다.

b) 구조



(그림 1) Faster R-CNN 구조

Faster R-CNN 모델을 간략하게 보면 RPN+Fast R-CNN이라고 볼 수 있다. 그동안 Selective Search를 사용하여 계산량 많은 Region Proposal 단계 Neural Network 인으로 공여해서 긴장한 것의 end-to-end Object Detection 모델을 제시하였으며, 기존 Fast R-CNN 구조를 개선하면서 selective search를 대신 RPN을 통해서 RoI를 계산한다. 이로 인해 GPU를 통한 RoI 계산이 가능해졌으며, RoI 계산

1. Ren, S., K. He, R. Girshick, J. Sun "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks"


### V. 성능 비교

Model	Backbone	Memory(GB)	Box AP
RetinaNet	R-50-FPN	3.8	36.5
YOLO v3	Darknet-53	3.8	30.9
DETR	R-50	7.9	40.1
Faster R-CNN	R-50-FPN	4.0	37.4
Cascade R-CNN	R-50-FPN	4.4	40.3

출처: COCO Object Detection model zoo

a) 예시 ①

- 원본 이미지



### 3. 프로젝트 추진 내용

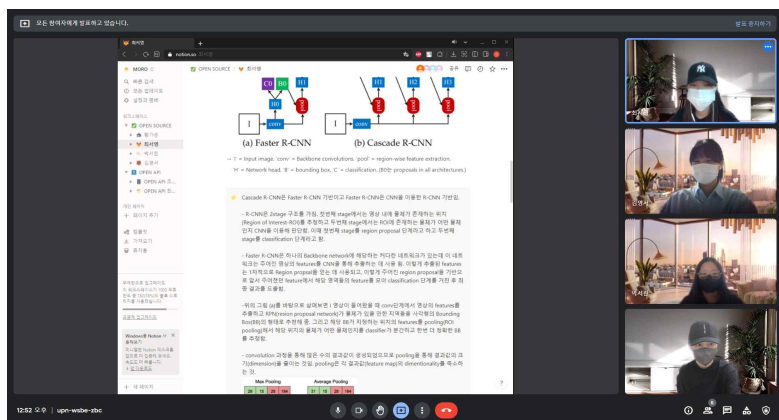
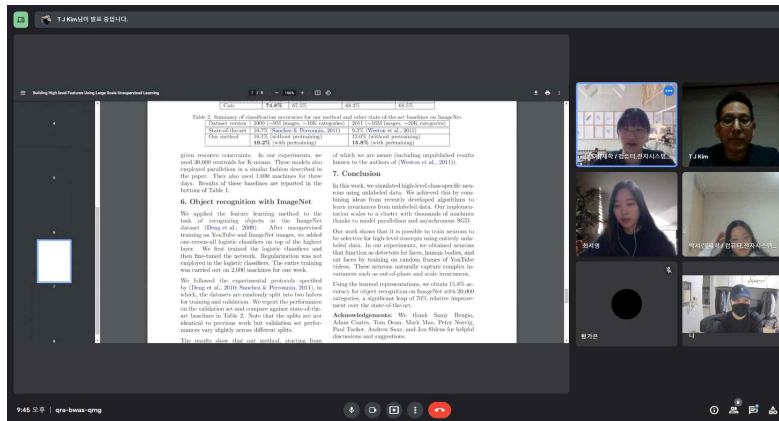
#### 1. 프로젝트 진행과정

- 항공 산업에 사용될 만한 Open Object Detection API를 조사 및 성능 비교
  - 카카오프API, ETRI API, Google API, Microsoft Azure API
- Open API를 조사하였으나 전문성이 부족하다고 생각하여 Open Source로 되어있는 Object Detection 모델 조사 및 비교
  - RetinaNet, Yolo v3, DETR, Faster R-CNN, Cascade R-CNN 조사 및 정리
  - 각 모델 간의 성능을 비교하여 항공 산업 배경에서의 적절한 모델을 최종 선정

	<p><b>2. 프로젝트 구현과정</b></p> <p>1. '시각 영상의 데이터화'를 주제로 시장 및 문헌 자료 분석</p> <ul style="list-style-type: none"> <li>- 항공 산업에 실질적으로 사용되는 시각 영상 분석 프로그램이 보편적으로 갖춰져 있지 않다는 사실을 알게 되었다. 프로그램을 비교, 분석할 대상이 마땅히 존재하지 않아 프로그램 소스에 해당하는 결과물을 생산하는 것보다 앞으로 어떤 모델이 항공 산업에 활용되면 좋을지에 대한 유의미한 결과 자료를 공유하기로 결정하였다.</li> </ul> <p>2. 객체 검출에 이용되는 open API 분석</p> <ul style="list-style-type: none"> <li>- 이미지 분석을 수행하는 오픈 API를 조사하고 현실적으로 응용이 가능한지 활용 용도, 활용 가능성, 가격 등 새롭게 지정한 기준에 맞추어 비교 분석을 진행하였다. 그러나 오픈 API는 어떻게 동작하는지 내부 구성을 알 수 없으며 이를 다루는 것만으로는 항공 객체 검출에 기능을 적용하기에 정보가 부족하다고 판단하였다. 따라서 모두가 쉽게 접근할 수 있는 open Source를 조사하는 것으로 진행 방향을 수정하였다.</li> </ul> <p>3. open Source 조사 및 여러 객체 검출 모델의 배경 및 성능 분석</p> <ul style="list-style-type: none"> <li>- object detector 기능을 제공하는 오픈 소스를 조사하였고 이는 객체 검출 모델을 어떻게 사용하는지에 대한 내용임을 인지하였다. 이후 오픈 소스에 등장하는 여러 객체 검출 모델 중 일부를 선정하여 배경 지식을 조사하였다. 모델에 사용되는 이론 개념을 체계적으로 나열함으로써 각 모델 사이의 공통점, 차이점을 도출할 수 있었다.</li> <li>- 모델의 기본 성능과 더불어 실제 항공 데이터셋을 적용하여 나오는 결과를 바탕으로 새로운 기준에 맞춰 성능을 비교, 분석한 후 항공 산업에 어떤 모델을 적용하는 것이 적합한지 결론을 도출하는 것까지 달성하였다.</li> </ul>
<p><b>4.기대효과</b></p>	<p><b>* 경제적/ 사회적인 측면</b></p> <p>시중에는 항공 산업 관련 객체 검출 서비스에 대한 내용을 찾아보기 어렵다. 그만큼 현재까지도 해당 기술은 항공 산업에서 보편화되어 있지 않은 상태이다. 항공 산업에서 활용될 수 있는 시각 영상 분석, 객체 검출 모델의 배경 및 성능을 분석하고 비교하였다. 기존에 존재하는 다양한 모델을 분석한 결과를 공유하고 최종적으로 항공기 운영 환경에 응용되는데 적합한 모델을 결론 내려 방향성을 제시함으로써 항공 산업 분야의 사회, 상업적인 측면에 큰 도움을 제공한다.</p>
<p><b>5.산학협력</b></p>	<p>김영서 : ETRI API, openCV를 이용한 Object Detection과 YOLO v3 조사 및 분석          박서린 : Microsoft Azura API, RetinaNet, YOLO v5 조사 및 분석          최서영 : Google API, Cascade R-CNN, YOLO v3, Faster R-CNN 조사 및 분석          황가은(팀장) : Kakao API, DETR 조사 및 분석, 산업체 멘토님에게 공유할 자료 통합 및 연락</p>

	<p>멘토링은 일정이 가능한대로 최소 주 1회 진행하였다. 멘토링에서는 주로 주간 수행했던 과제에 대해 발표를 하고 그에 대해 멘티마다 개인 피드백을 세세하게 전달하였다. 또한, 향후 과제의 방향성에 있어서 멘토와 멘티의 의견을 수렴하였다.</p> <p>제출용 과제(보고서)에 대해서는 팀원이 모여 작성한 과제물을 멘토님께 전달한 후 검토해주셨다.</p>
6.참고문헌	<p>* 저자, 제목, 출판사, 페이지, 년도 순으로 작성</p> <p>자료 조사를 기반으로 하여 참고문헌이 매우 많아 자주 참고한, 그리고 주요하게 생각되는 문헌에 대해서만 일부 작성하였습니다.</p> <p>[Retinanet]</p> <ul style="list-style-type: none"> <li>- Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar "Focal Loss for Dense Object Detection, arXiv.org, 2017</li> <li>- 김상현, "RetinaNet 논문 리뷰", <a href="https://velog.io/@skhim520/RetinaNet-%EB%85%BC%EB%AC%B8-%EB%A6%AC%EB%B7%B0">https://velog.io/@skhim520/RetinaNet-%EB%85%BC%EB%AC%B8-%EB%A6%AC%EB%B7%B0</a>, 2021</li> </ul> <p>[Yolov3]</p> <ul style="list-style-type: none"> <li>- Joseph Redmon and Ali Farhadi, "YOLOv3: An Incremental Improvement", arXiv:1804.02767, 2018.</li> <li>- Ayoosh Kathuria, "Series: YOLO object detector in PyTorch", <a href="https://blog.paperspace.com/tag/series-yolo/">https://blog.paperspace.com/tag/series-yolo/</a>, 2018</li> </ul> <p>[DETR]</p> <ul style="list-style-type: none"> <li>- Xizhou Zhu, Weiye Su, Lewei Lu, Xiaogang Wang, and Jifeng Dai, "End-to-End Object Detection with Transformers", arXiv:2005.12872, 2020</li> <li>- 한땀컴비 외 6명, "한땀한땀 딥러닝 컴퓨터 비전 백과사전 – DETR", <a href="https://wikidocs.net/145910">https://wikidocs.net/145910</a>, 2021</li> </ul> <p>[Faster R-CNN]</p> <ul style="list-style-type: none"> <li>- Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", arXiv:1506.01497, 2016</li> <li>- herbwood, "Faster R-CNN 논문(Faster R-CNN: Towards Real-Time ObjectDetection with Region Proposal Networks) 리뷰", <a href="https://herbwood.tistory.com/10">https://herbwood.tistory.com/10</a>, 2020</li> </ul> <p>[Cascade R-CNN]</p> <ul style="list-style-type: none"> <li>- Zhaowei Cai and Nuno Vasconcelos, "Cascade R-CNN: Delving into High Quality Object Detection", arXiv:1712.00726, 2017</li> <li>- brianjaum, "Cascade R-CNN: Delving into High Quality Object Detection", <a href="https://blog.lunit.io/2018/08/13/cascade-r-cnn-delving-into-high-quality-object-detection/">https://blog.lunit.io/2018/08/13/cascade-r-cnn-delving-into-high-quality-object-detection/</a>, 2018</li> </ul>
7.R&D성과	<p>산업체 측에서 제안한 프로젝트의 방향성은 Development보다 Research 중심으로, 주어진 과제를 진행하였고 제출한 최종 결과물이 이에 대한 보고서에 해당합니다.</p>

## 8.첨부



## 산학연계 프로젝트 결과보고서 (요약)

과 제 명	Object Detection Model 비교		
협력기관명	(주)디엠항공교역	수행기간	2022년 3월 ~ 2022년 5월
참여인원	(총 5 명) 멘토 1 명, 참여교수 0 명, 학부 수강생 4 명		
	구분	성명/소속(직급)	
	멘토	김태진/디엠항공교역(대표)	
	교수		
	학생	김영서/컴퓨터전자시스템공학부(4학년)	박서린/컴퓨터전자시스템공학부(4학년)
		최서영/컴퓨터전자시스템공학부(4학년)	황가은/컴퓨터전자시스템공학부(4학년)
추진배경	<p>고정 지형 지물, 지상 이동체 등이 혼재하는 항공기 활주로, 계류장, 공항 주변을 배경으로 3차원 공간의 시각 영상을 데이터화하는 것이 중요하다. 그런데 현재 항공 산업 관련 객체 검출 서비스가 아직 보편화되지 않았다는 시장 배경을 인지하였다. 따라서 우리가 기존의 다양한 객체 검출 모델을 연구하고 분석함으로써 앞으로 항공 산업 객체 검출 서비스에 도움되는 정보를 공유하기로 결정하였다.</p>		
목표 및 내용	<p>Object Detection 모델을 연구하는데 앞서 객체 검출의 데이터 처리 과정을 심층적으로 학습하고, 이론 배경을 객체 검출 모델에 적용함으로써 전문 지식을 함양한다.</p> <p>여러 객체 검출 모델을 비교 분석하고 실제 데이터셋을 적용하여 결과를 도출한다. 항공기 운영 환경에 사용될 수 있는 객체 검출 모델에 대한 유용한 정보를 제공하고 세부적인 방향성을 제시한다.</p>		
수행결과	<p>객체 탐지기의 종류는 크게 One-Stage Detector와 Two-Stage Detector로 나뉜다. 먼저 One-Stage Detector는 Regional proposal과 Classification이 동시에 이루어진다. 이에 속하는 모델은 YOLO계열, SSD계열(RetinaNet, SSD), DETR 등이 있다. Two-Stage Detector는 그와 반대로 순차적으로 이루어지는 구조를 가지고 있으며 이에 속하는 모델은 R-CNN 계열이 있다.</p> <p>이러한 구조의 차이로 One-Stage Detector는 비교적 검출 속도가 빠르지만 정확도가 낮고 그와 반대로 Two-Stage Detector는 검출 속도가 느리지만 정확도가 높다는 특징을 가지고 있다. 이들은 각각 장단점이 있어 하나를 선택하게 된다면 편향적인 정보를 제공하게 된다. 따라서 신뢰도가 높은 정보를 제공하기 위해 두 가지 종류를 모두 다양하게 비교할 수 있도록 각각 3개, 2개의 모델을 조사하였다. One-Stage Detector 모델은 Retinanet, Yolo v3, DETR을, Two-Stage Detector는 Faster R-CNN, Cascade R-CNN가 있다.</p> <p>여러 이미지를 각 모델에 입력하여 실행한 결과 Cascade R-CNN 모델이 기준에 포함된 주요 대상을 가장 잘 검출하는 것으로 파악하였다. 최종적으로 DETR과 Cascade R-CNN의 결과 성능이 유사하나 Cascade R-CNN을 선택한 이유는 동일한 객체에 대해 출력되는 박스의 개수, 즉, 하나의 객체에 대해 중복적으로 검출되는 현상이 DETR보다 현저히 적기 때문이다. 또한, 실행 시간을 기준으로 접근하였을 때 YOLO v3를 제외하고는 실행 시간이 비슷하므로 Cascade R-CNN이 가장 적합하다는 결과를 도출하였다.</p>		