

# Capstone Design Project Report

Project Title : Object detection automation solution with  
FFmpeg and YOLO  
(FFmpeg과 YOLO를 이용한 객체 탐지 자동화 솔루션)

Lecture Class	Capstone Design 1-03 (2021 Spring)		
Term	2021.03.02. ~ 2021.06.14		
Supervisor	Class Advisor		Son YoonSik
	Industry Mentor	Industry Mentor	TECHNONIA
	Sung WonYong		
Team Member Students	Student ID No.	Name	E-mail
	2017112106	Shin SoHui	skysohe@naver.com
	2017112094	Lee EunYoung	eyey0811@naver.com
	2018113338	Jo SeongWoon	jsw166098@gmail.com
	2017112100	Han EuGene	hhdbwls97@naver.com

## <Summary in English>

Project Goal and Objectives	<p>In machine learning, we want to automate the process, considering that the process of collecting and labeling data for learning takes a huge amount of human capital and time. In other words, it aims to automate the process of collecting the objects desired by the user and also providing the position where they exist within the image as bounding box coordinates.</p> <p>This task aims to establish an interactive website that allows users connected to the Internet to use the service without any space-time constraints, and outputs results by inputting the images or images they want. Moreover, in order to provide higher satisfaction in exploiting the service, we aim to provide only the dataset results of detected objects with high accuracy, and to build a large amount of image datasets by users.</p>		
Project Contents	<p>This project is a web service that provides the coordinate values of the extracted image and bounding box by detecting the object that the user wants to find as a check box. In the input process, users can optionally attach their own videos or images.</p> <p>Except for the user's input portion, the task is accomplished through the following five processes : Image gathering, Image labeling, Build model, Object detection, Image augmentation. In order to learn, we need to collect images to learn, such as other artificial intelligence projects. In the stage of building learning images, images are collected using public images, OIIV4 toolkit, etc., provided by various search engines.</p> <p>Once the collection of image data is complete, it is necessary to label where the object exists in the image. During labeling, the bounding box is displayed for images collected using Ybat, which is a tool that supports data labeling.</p> <p>Once image collection and labeling are completed, a learning model is built. Using Google Colab's development environment, it uses the Darknet-based YOLOv4. Modify and learn variables in the cfg file, such as class count and filter count, according to the project's content. When the learning is completed, object detection is performed using the weight file.</p> <p>In order to provide a high-accuracy object detection service to the user, the AP, which is a performance evaluation index, is measured and presented for the learning model. IoU (Intersection over Union) is changed to three threshold values of 0.5, 0.7, and 0.9, and the mAP for 32 classes provided as a result is measured. It guarantees the accuracy of 85% or more of mAP, and extracts the coordinates of the object in the image and provides it to the user. If the user wants more images, the image augmentation function can be additionally selected.</p> <p>In conclusion, when a user enters the object he or she wants to find into a checkbox, that input is passed to the website. As a result, users can receive a large number of image datasets and labelling results required for learning without labor.</p>		
Anticipated Results and Impacts on Industry	<p>Collect large amounts of image data required for learning and labeling can be to automate the process. Therefore, learning by reducing time to build three image data, learning and accuracy can use the time to build a model, technological advances will happen faster.</p> <p>Through the project and the results are from artificial intelligence developers to ordinary people rose up in a chain. Learning in the process of preparing for the costs will diminish the same time, human resources and learning models and algorithms, which is quicker means can be completed within a period of time. As a result, the development of technology also people's lives due to the development of faster and will be comfortable. So, social to economic and social influence through can.</p>		
Keywords	Data gathering Automization Yolo v4	Object Detection Bounding Box	Data labeling Automization

과제의 목적과 목표	<p>기계학습에 있어서, 학습을 위한 데이터를 수집하고 라벨링하는 과정은 막대한 인적 자본과 시간이 소요된다. 점을 착안하여 해당 과정을 자동화하고자 한다. 즉, 사용자가 원하는 객체를 수집하고, 해당 객체가 이미지 내에 존재하는 위치 또한 바운딩 박스 좌표로 제공하는 과정의 자동화를 목적으로 한다.</p> <p>본 과제는 인터넷에 연결된 사용자가 시공간의 제약 없이 서비스를 이용할 수 있도록 웹 호스팅을 하고, 사용자가 원하는 이미지나 영상을 입력으로 해서 결과를 출력하는 상호작용이 가능한 웹 사이트를 구축하고자 한다. 또한, 서비스를 이용함에 있어 보다 높은 만족도를 제공하기 위하여 높은 정확도로 탐지된 객체의 데이터 셋 결과물만을 제공하고, 사용자가 대량의 이미지 데이터 셋을 구축하는 것을 목표로 한다.</p>		
과제의 내용	<p>본 과제는 사용자가 찾고자 하는 객체를 체크박스로 입력하면, 해당 객체를 영상 및 이미지에서 탐지하여 추출한 이미지와 바운딩 박스의 좌표값을 제공하는 웹 서비스이다. 입력 과정에서 사용자는 선택적으로 자신이 소유하고 있는 영상이나 이미지를 입력으로 첨부할 수 있다.</p> <p>사용자의 입력 부분을 제외하면, 본 과제는 다음의 5개의 과정을 통해서 수행된다.</p> <ol style="list-style-type: none"><li>1. 학습 이미지 수집</li><li>2. 이미지 라벨링</li><li>3. 학습 모델 구축</li><li>4. 객체 탐지를 통한 이미지 획득</li><li>5. 이미지 어그멘테이션</li></ol> <p>학습을 시키기 위하여 다른 인공지능 프로젝트와 같이 학습할 이미지를 수집해야 한다. 학습 이미지를 구축하는 단계에서, 다양한 검색엔진에서 제공하는 공개된 이미지, OIDV4 toolkit 등을 이용하여 이미지를 수집한다.</p> <p>이미지 데이터의 수집이 완료되면, 객체가 이미지의 어느 위치에 존재하는지 라벨링을 해야 한다. 라벨링을 하는 과정에서, 별도의 설치 과정 없이 HTML 파일을 다운로드하여 이용할 수 있는 Ybat을 이용하여 수집한 이미지에 대해서 바운딩 박스를 표시한다.</p> <p>이미지 수집과 라벨링이 완료되면, 학습 모델을 구축한다. Google Colab의 개발 환경을 이용하여 Darknet 기반의 Yolo v4를 이용한다. 프로젝트의 내용에 맞추어 class 개수, filter 수 등의 cfg 파일의 변수들을 수정하고 학습을 한다. 학습이 완료되면, weight 파일을 이용하여 객체 탐지를 한다. 사용자에게 정확도 높은 객체 탐지 서비스를 제공하기 위하여, 학습 모델에 대해 성능 평가 지표인 AP를 측정하여 제시한다. IoU(Intersection over Union)를 0.5, 0.7, 0.9의 세 가지 threshold 값으로 변경하며 결과물로 제공하는 32개 클래스에 대한 mAP를 측정한다. mAP 85% 이상의 정확도를 보장하며, 이미지 내의 객체의 좌표값을 추출하여 사용자에게 제공한다. 사용자가 더 많은 이미지를 원한다면, 이미지 어그멘테이션 기능을 추가 선택할 수 있다.</p> <p>결론적으로, 사용자가 찾고자 하는 객체를 체크박스로 입력하면, 해당 입력이 웹 사이트로 전달된다. 그 결과, 사용자는 학습에 필요한 대량의 이미지 데이터 셋과 라벨링 결과를 노동 없이 제공받을 수 있다.</p>		
예상되는 결과와 파급 효과	<p>학습에 필요한 대량의 이미지 데이터를 수집하고 라벨링 하는 과정을 자동화할 수 있다. 따라서, 학습 이미지 데이터 셋을 구축하는 데에 들이는 시간을 줄여 학습 모델을 구축하고 정확도를 높이는 데에 시간을 사용할 수 있으며, 보다 빠르게 기술 발전이 일어날 것이다.</p> <p>해당 프로젝트를 통해서, 효과는 인공지능 개발자부터 일반 사람들에게까지 연쇄적으로 일어날 것이다. 학습을 준비하는 과정에 소요되는 시간, 인적 자원과 같은 비용이 줄어들 것이며, 이는 학습 모델 구축 및 알고리즘을 보다 더 빠른 시간 내에 완성할 수 있음을 의미한다. 그 결과, 기술의 발전 또한 빨라지며 기술의 발전으로 인하여 사람들의 삶도 편안해질 것이다. 이렇듯, 본 과제를 통해서 경제적, 사회적으로 영향을 끼칠 수 있을 것이다.</p>		
중요단어	데이터 셋 구축 자동화	객체 탐지	데이터 라벨링 자동화
	Yolo v4	바운딩 박스	

1. Project Goals and Objectives (과제의 목표와 목적)

1.1 Project Goal (과제의 목적)

머신러닝에 있어서, 학습 이미지의 획득이 머신러닝 프로젝트의 성패를 좌우한다. 학습 데이터를 획득하는 과정은 막대한 시간과 노력이 소요되는데, 이 과정을 자동화하여 쉽게 얻을 수 있는 방법이 필요하다고 생각하였다. 본 프로젝트의 목적은 기계학습에 필요한 대량의 학습 이미지 데이터 셋을 자동으로 구축해 주는 서비스를 구현하는 것이다.

전 세계적으로, 인공지능 로봇과 자율주행 자동차 등 다양한 분야의 기업들이 머신러닝 서비스를 연구하고 있다. 현재 많은 자동차 기업들이 자율주행 서비스를 시도하고 있으며 서비스가 실제로 상용화되고 있지만, 자율주행 자동차의 사고로 이슈가 끊이지 않는다<sup>1)</sup>. 아직은 학습이 완전하지 않은 상태임이 분명하다. 이 점에 착안하여 자율주행 개발자들이 자율주행 자동차의 학습에 필요한 이미지 데이터를 보다 편리하고 빠르게 수집할 수 있도록 도와주고자 한다. 따라서, 이미지 데이터 셋을 구축하는 도메인을 공통으로 정하여 자율주행 학습에 필요한 객체들을 자동으로 구축하고자 하는 것이 본 프로젝트의 최종 목적이다.

1.2 Project Objectives (과제의 목표)

본 프로젝트를 통해서, 다음과 같은 4개의 세부 목표들을 달성할 것이다.

1. 웹 호스팅을 통한 결과물 제공
  - 클라우드 기반의 node.js를 사용하여 인터넷에 연결된 사용자가 시공간의 제약 없이 서비스를 이용할 수 있으며 최종적으로 구축된 이미지 데이터 셋과 라벨링 파일을 다운로드할 수 있도록 한다.
2. 사용자의 영상 및 이미지를 입력으로 한 결과물 산출
  - 사용자가 직접 소유하고 있는 동영상과 원하는 객체를 선택하여 입력하면, 객체를 탐지 결과의 이미지와 bounding box의 좌표 값 또한 산출할 수 있도록 한다.
  - 사용자가 직접 소유하고 있는 이미지와 원하는 객체를 선택하여 입력하면, 객체를 탐지한 결과 이미지의 bounding box 좌표 값을 산출할 수 있도록 한다.

3. 데이터 셋 결과물의 정확도

- 객체 탐지 모델의 성능을 precision-recall 곡선과 average precision(AP)로 평가하여 각 클래스의 평균을 계산한 mean average precision(mAP)로 나타낼 것이다. mAP 평가 방식으로는 대표적으로 Pascal VOC, MS COCO 방식이 있다. Pascal VOC metric은 1개의 IoU 값(0.5) 즉, single threshold 값으로만 측정하므로 정확도 측정에 있어 느슨하다는 단점이 있다. MS COCO

1) “테슬라 반자율주행 ‘불안불안’... 미국서 또 사고”, <조선비즈>, 2021.03.18., <[https://biz.chosun.com/site/data/html\\_dir/2021/03/18/2021031801749.html](https://biz.chosun.com/site/data/html_dir/2021/03/18/2021031801749.html)>, <2021.06.03.>

metric은 IoU를 0.5부터 0.05 간격으로 0.95까지 변경하며 측정한 값의 평균을 내는 방식인데, IoU 기준을 높이다 보니 상대적으로 classification 정확도는 무시되며 IoU가 높아짐에 따라 동일한 가중치를 적용한다는 단점이 있다. 본 프로젝트에서는 0.5, 0.7, 0.9의 세 가지 threshold 값으로 변경하며 mAP를 측정한다. 각각의 threshold 값은 ‘다중 센서를 사용한 주행 환경에서의 객체 검출 및 분류 방법’의 논문들<sup>2)</sup> 참고하여 정하였다. 참고 자료를 기반으로, 제공하는 전체 클래스의 mAP를 85%로 정하였으며, 대개 80% 이상이면 높은 정확도라고 한다.<sup>3)</sup> 따라서, mAP를 85%로 하여 정확도 높은 데이터 셋을 제공할 수 있도록 한다.

- 사용자에게 제공하는 최종 이미지 데이터 셋은 객체 탐지 시 confidence score가 0.8 이상의 확률로 일치하는 객체만을 탐지하여 결과로 반환하도록 한다.

#### 4. 대량의 이미지 데이터 셋 구축

- 사용자가 최종 결과물 데이터 셋의 수량이 부족하다고 생각하면, augmentation 기능을 체크박스로 입력하여 데이터를 대량으로 증폭시켜 제공받을 수 있도록 한다.
- augmentation 기능을 선택한 경우, 본 프로젝트 내에서 bounding box의 좌표 값 변환도 자동으로 제공하여 사용자가 별도의 라벨링 과정 없이 곧바로 학습에 사용할 수 있도록 한다.

## 2. Background of the Project (과제의 필요성)

### 2.1 Motivations (필요성)

조사에 따르면, 인공지능 프로젝트에 소요되는 시간 중 약 35%의 비율이 데이터 수집과 데이터 라벨링에 소요된다고 한다. 반면에, 알고리즘 개발과 모델 학습에 소요되는 시간은 약 13%의 비율만을 차지한다고 한다<sup>4)</sup>. 즉, 학습 알고리즘과 모델을 개발하는 것보다 데이터를 수집하고 라벨링을 하는 데에 약 2.7배의 시간이 더 소요된다는 점을 알 수 있었다. 일각에서는 데이터 수집과 라벨링은 ‘인공지능(AI) 학습을 위한 인간의 노동’이라고 칭한다. 따라서, 단순하고 반복적인 업무에 사용되는 시간, 인적 자원과 같은 비용을 줄이고자 본 프로젝트를 구상하였다.

인공지능 기술이 발달하면서 사람이 하던 주문, 청소 등의 반복적이고 단순한 업무는 기계가 대체하고 있다. 기계학습에 있어서, 데이터 수집과 라벨링은 고도의 기술이 필요 없는 단순 반복적인 작업<sup>5)</sup>이므로 해당 작업 또한 자동화가 필요한 부분이다.

본 프로젝트를 통해서 데이터 수집과 라벨링이 자동화가 된다면, 데이터 수집과 라벨링에 소요되는 약 35%의 시간 비율은 그 미만으로 줄어들 것이다. 이는, 해당 기술의 구현이 더 빠르게 완성된다

2) 김정언, 강행봉, 「다중 센서를 사용한 주행 환경에서의 객체 검출 및 분류 방법」, Journal of Korea Multimedia Society Vol. 20, No.8, 2017, 9쪽.

3) 김영연, 최인훈, 김명현, 김승직, 허의남, 「딥러닝 기반 객체 인식을 통한 실내 위치 정보 탐색 자율 주행 로봇 개발」, 경희대학교, 2020, 2쪽.

4) “AI 학습용 데이터 구축, AI 강국의 기반 다진다”, <컴퓨터 월드>, 2021.01.31., <<https://www.comworld.co.kr/news/articleView.html?idxno=50123>>, <2021.06.03.>

5) “AI 시대 숨은 주역, ‘데이터 노동자’를 아시나요”, <정보통신 신문>, 2020.10.20., <<https://www.koit.co.kr/news/articleView.html?idxno=79944>>, <2021.06.03.>

는 점을 시사한다. 따라서, 인공지능 학습을 통한 여러 기술의 출현 속도가 빨라질 것이며 이는 사회적으로 사람들의 삶을 더 편하게 만들어줄 것이다.

결론적으로, 인력이 기계로 대체되어 인건비가 줄어들고 시간이 절약되는 경제적인 장점이 있고, 데이터 수집과 라벨링에 투자하는 시간이 적어져서 기술이 그만큼 더 빨리 발전할 수 있어 사회적으로 장점이 있다. 이러한 사회적 및 경제적 이점을 바탕으로, 대량의 이미지를 구축하고 이미지에 존재하는 지정 객체들을 라벨링하는 과정을 자동화하는 서비스를 구축하는 동기가 되었다.

### 2.2 Significances (중요성)

인공지능 개발 과정에서 데이터 셋의 양과 품질은 인공지능경망 훈련 결과에 영향을 미치므로, 중요한 위치를 차지한다<sup>6)</sup>. 대부분의 데이터 셋은 인적 자원을 활용하여 구축되고 있으며, 많은 인적, 시간적 측면의 자원을 소비한다. 학습을 하기 위한 충분한 양의 데이터가 수집이 되어 있지 않다면, 오류가 커지므로<sup>7)</sup> 학습이 미숙하게 될 것이다. 결론적으로, 학습을 위해서는 충분한 양의 데이터 셋 구축이 중요하고, 본 프로젝트를 통해 해당 과정을 자동화하여 대량의 데이터 셋을 구축함으로써 문제를 해결하고자 한다.

## 3. Related Works (관련연구)

### 3.1 Related Techniques

본 과제인 “FFmpeg과 YOLO를 이용한 동영상 내 객체 탐지 자동화 솔루션”과 관련된 기술들로는 Google과 NAVER가 제공하는 이미지 검색 기능, MS COCO data를 제공하는, COCO dataset 웹사이트, OIDv4 toolkit 등 다양한 서비스와 툴이 존재한다. 그러나 기존의 서비스들은 단순히 이미지를 입력으로 넣었을 때 추가적으로 이미지를 출력해주거나, 인적 자원에 기반해 데이터 셋을 구축하고 정적으로 결과물을 제공해준다. 그러나 대부분의 오픈 데이터 셋이 제공하는 클래스와 annotation 자동화 툴이 인식할 수 있는 클래스가 한국이 아닌 외국의 객체에 초점이 맞춰져 있다. 국가를 “한국”으로 한정했을 때는 도로 상황이 다르므로 자율주행 자동차 학습 데이터 셋이 부족한 실정이다.

위의 관련 연구와 본 과제는 다음과 같은 차이점을 가지고 있으며, 본 과제의 신규성을 보장한다.

#### 1) 경제성

- 현재 한국의 데이터 비즈니스 플랫폼의 경우, 학습 데이터 셋을 얻기 위해서는 클라이언트 입장이 되어 플랫폼과 이해관계를 맺고 유료 비용을 지급해야 한다. 이런 데이터 비즈니스 플랫폼과 대조되게 본 과제는 도로 주행 객체를 무료로 얻을 수 있다. 또 사용자가 영상

6) 송순용, 「이미지 데이터셋 제작을 위한 Annotation 자동화 기술 분석」, ITFIND 중간기술동향 1987호, 한국전자통신연구원, 2021, 2쪽.

7) “딥러닝(Deep Learning), 인공지능의 핵심 기술 알아보기”, LG디스플레이기업블로그, 2016.06.16. <<https://blog.lgdisplay.com/2016/06/deep-learning/>>, 2021.06.03

에서 객체를 탐지하고 싶다면 웹 사이트상에서 실시간으로 획득할 수 있다는 점에서 경제적이다.

### 2) 효율성

- 학습 이미지 데이터 셋을 구축하는데에 가장 인적 자원, 시간 비용이 많이 필요한 과정이 이미지 annotation 단계이다. 본 과제는 이 단계를 자동화할 수 있으므로 전체 비용이 감소한다.

### 3) 편리성

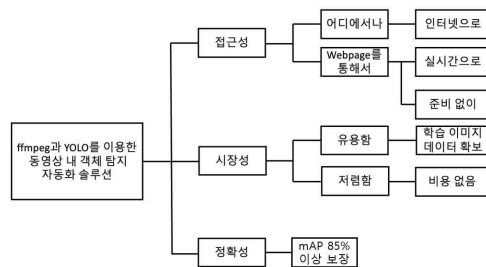
- 별도의 환경설정이나 작동 방식을 이해하는 시간 비용을 들이지 않고 웹 사이트상에서 클릭을 통해 데이터 셋을 확보할 수 있다. 또한 augmentation 옵션을 넣어 데이터 셋을 추가 확보할 수 있어 사용자가 코드를 이용해 추가 작업을 해야 하는 수고로움을 덜어준다.

### 4) 기능

- 실시간으로 동영상에서 객체 탐지를 할 수 있고, augmentation을 해주기 때문에 사용하기에 편리하다.

## 4. Requirement Analysis (요구 분석)

### 4.1 Objective Tree (목적나무)



### 4.2 Functional Requirement Analysis (요구 분석)

#### 4.2.1 Function-Related Requirement (기능 관련 요구사항)

요구기능 중 주요 기능은 총 5개이다.

##### ● 주요 상세 기능 #1) 웹 서버 호스팅 수행 기능

정의: 주요 상세기능 #1은 사용자들이 브라우저를 통해 본 프로그램 웹 사이트를 접속한다. (프로그램이 동작할 서버를 구축하는 작업이다.)

상세 설명: 사용자 개인 PC의 브라우저에서 본 프로그램의 웹 사이트로 접속 가능하게 한다. 본 프로그램의 서비스를 이용할 수 있도록 서버를 구축하고 프로그램을 서버에 올린다.

##### ● 주요 상세 기능 #2) 사용자 항목 선택 기능

정의: 주요 상세기능 #2는 본 프로그램의 YOLO 모델에서 이미 학습된 클래스 중 사용자가 원하는 클래스 종류 정보를 입력받는 기능이다.

상세 설명: 기존에 이미 학습이 진행된 클래스 명 또는 객체 이름이 체크박스와 함께 명시되어 있다. 사용자는 원하는 항목을 체크박스를 통해 복수의 클래스를 선택할 수 있다.

##### ● 주요 상세 기능 #3) 기존에 탐지하기로 지정한 객체에 대한 Yolo 학습 기능

정의: 프로그램에서 사용할 Yolo 모델을 학습하여 학습 모델을 생성하는 기능이다.

상세 설명: Yolo 학습은 이미지를 수집하고, 이미지에 객체에 대한 라벨링을 진행하고, 얻은 좌표값의 파일과 설정 파일로 학습을 진행하는 순서로 진행된다. 이미지 수집 단계를 거치고, 이미지 라벨링 단계에서는 지정된 객체를 bounding box로 구분하여 학습에 필요한 파일을 생성하는 과정을 거친다. 이 때 학습 데이터 셋이 적으면 학습 데이터 셋의 수를 늘리기 위해 수직 뒤집기, 수평 뒤집기 등의 image augmentation을 통해 이미지를 늘린다. 그리고 Yolo 학습을 진행하게 된다. 학습 모델을 생성하는 Yolo Training에서는 darknet을 이용하여 Yolo 학습 모델을 생성하고 최종 학습 모델인 가중치 파일과 설정 파일을 얻는다.

##### ● 주요 상세 기능 #4) 영상 데이터에서 Yolo Detection 수행 기능

정의: DB에 저장된 자율주행 영상 데이터 셋에 대해 Yolo Detection을 진행하며 객체가 탐지된 프레임 정보를 얻어낸다.

상세 설명: YOLO 모델을 학습시킨 이후 DB에 저장된 UC Berkeley BDD100K, AIhub 데이터 셋에 대해 객체를 탐지한다. UC Berkeley BDD100K, AIhub 데이터 셋에서 사용자가 원하는 객체가 담긴 프레임 정보를 얻는다. 이때, UC Berkeley BDD100K, AIhub 데이터 셋에 있는 주행 영상이 YOLO 학습 모델에 적용할 수 없는 형식일 때, FFmpeg을 이용해 YOLO 학습 모델에 적용할 수 있는 형식으로 변환한다. 또한 사용자가 동영상 및 이미지를 입력하면 입력한 영상 내에서 객체를 탐지하고 결과 이미지를 획득할 수 있다.

##### ● 주요 상세 기능 #5) 학습용 이미지 데이터 셋 제공 기능

정의: 사용자가 원하는 객체가 Detection된 이미지를 사용자에게 제공하는 기능이다.

상세 설명: 도로상 데이터의 객체 분류표에 기반해 사용자가 원하는 객체가 Detection된 이미지를 다운받을 수 있도록 제공하는 단계이다. 이 때, 사용자는 augmentation 체크 박스의 여러 옵션을 체크하여 추가적인 학습 이미지 데이터셋을 얻을 수 있다.

위 기능은 다음과 같은 고려사항이 필요하다.

1) 주요 상세 기능 #1에서 사용자들이 프로그램을 이용할 수 있도록 프로그램을 올릴 서버를 구축한다. 다른 요구 기능들과는 독립적으로 프로그램 완성 전까지 #1이 구축되어야 하며 서버는 클라우드 상의 node.js 기반으로 구축되어야 한다. 또 사용자들이 사용하기에 안전하고 안정적이어야 한다. 사용자들은 서버 주소를 통해 사이트에 접속해서 프로그램을 이용할 수 있게 된다.

2) 주요 상세 기능 #3은 객체를 검출하기 위한 Yolo 학습 모델을 생성하는 기능으로 프로그램의 가장 핵심 부분이며 #1 기능이 완성된 후 서버 위에 올라가야 한다. Yolo 학습 모델의 클래스를 미리 지정하고 학습을 한다. 미리 지정된 객체의 종류로는 정보통신단체 표준으로 자율주행 자동차의 객체 인식 기술에 필요한 도로상 데이터의 객체 분류 체계 중 선택된 것으로 보행자, 차, 표지판, 신호기, 차선 등과 관련된 세부 클래스들이 있고 후에 프로그램 실행 시 기본적으로 선택할 수 있게 표시된다. 크롤링과 공개 데이터셋을 활용하여 이미지를 확보하고, 객체를 라벨링하는 과정을 거친 다음, 학습 이미지의 수를 늘리는 과정을 거치고 난 뒤, Google Colab에서 Darknet 프레임워크의 YOLOv4 모델로 학습이 수행된다.

3) 주요 상세 기능 #1과 #3이 정상적으로 수행된 이후에 주요 상세 기능 #2 기능이 수행된다. 사용자가 정상적으로 페이지에 접속하였다면 프로그램에서 탐지할 수 있는 클래스의 리스트가 화면에 뜨게 되고 그중 원하는 클래스를 선택하여 체크한다.

4) 주요 상세 기능 #1, #2, #3이 모두 수행된 이후 주요 상세 기능 #4가 수행된다. 사용자 입력이 서버로 넘어가면 서버에서는 저장된 학습 이미지 데이터 셋을 웹 페이지에서 다운로드 할 수 있도록 응답한다. 만약 사용자의 동영상 입력이 있다면 FFmpeg을 이용하여 파일 형식을 변환하고 실시간으로 동영상 내에서 탐지한 뒤 결과 데이터셋을 출력할 수 있도록 한다.

5) 주요 상세 기능 #5는 앞선 모든 기능이 수행된 후에 수행된다. 사용자는 결과물을 웹 페이지에서 다운로드 할 수 있다. 만약 얻어진 결과물에 대해 추가적인 augmentataion을 수행한 결과물을 원한다면 augmentation 체크 박스에 원하는 변환 방법을 체크하고 해당 변환 방법이 적용된 결과물을 얻을 수 있다.

#### 4.2.2 Data-Related Requirement (자료 관련 요구사항)

##### ● 데이터 종류

: 데이터는 크게 5가지 종류가 있다. 1) 객체가 포함된 이미지 데이터, 2) 라벨링 작업 거친 YOLO 학습 데이터, 3) UC Berkeley BDD 100K 데이터, 4) 사용자 입력 동영상 데이터, 5) 인식된 객체가 담긴 이미지, 좌표 데이터 6) 증폭 작업을 거친 데이터

##### ● 데이터 수집

1)은 객체가 담긴 이미지 데이터로 COCO Dataset, OVIDv4, AIhub 등 기업들이 공개한 데이터를 수

집한다. 2)는 1)의 데이터에서 annotation 작업을 통해 확보한다. 1)의 이미지 데이터와 bounding box 좌표가 담긴 txt 파일이 이에 해당한다. 3)은 UC Berkeley BDD100K의 오픈 소스로부터 획득하며, 4)의 데이터는 사용자로부터 입력받는다. 5)는 YOLO v4 모델을 통해 3), 4) 데이터를 이미지로 분할 후 객체를 인식해서 수집한다. 6)의 데이터는 5)의 데이터를 image augmentation 기법을 통해 이미지 데이터를 증폭시켜 확보한다.

##### ● 데이터 입출력 및 저장

사용자로부터 입력받는 데이터는 4)이다. node.js 기반 웹 브라우저로 업로드된 후 서버로 전송된다. 출력 데이터는 5), 6)으로 이미지 데이터와 객체의 좌표를 제공한다.

##### ● 데이터 사용

1)은 학습 데이터를 구축하기 위해 사용된다. annotation 작업을 거친 후 좌표 값을 포함해 2) 데이터를 생성한다. 2)는 YOLO v4 모델 학습을 위해 사용되며 학습된 모델은 3), 4)를 이용해 객체 인식 작업을 진행한다. 5)는 객체 인식 결과 데이터로 image augmentaion 작업 진행 시 사용된 후 6)을 생성한다.

#### 4.2.3 Interface-Related Requirement (인터페이스 관련 요구사항)

##### ● 사용자의 편의성을 고려

: 사용자는 웹 브라우저의 직관적인 ui를 통해 이미지를 다운로드할 수 있어야 한다.

##### ● 사용자 인터페이스 학습성 만족

: 해당 서비스가 제공하는 'How to do?'의 웹 사용 설명서를 통해 처음 접속하는 사용자도 쉽게 사용할 수 있어야 한다.

#### 4.2.4 User-Related Requirement (사용자 관련 요구사항)

##### ● 네트워크 연결을 통해 웹 브라우저로 접속이 가능한 사용자

: 사용자는 네트워크 연결이 가능한 PC를 통해 웹 브라우저로 접속할 수 있어야 한다.

##### ● 출력값을 저장하기 위해 충분한 컴퓨터 메모리를 보유한 사용자

: 출력값을 저장할 수 있는 충분한 메모리가 탑재된 컴퓨터를 보유해야 한다.

##### ● 개인 동영상을 입력할 사용자

: 개인 동영상에서 객체를 인식하려는 경우 사용자는 입력할 동영상을 확보해야 한다.

### 4.3 Realistic Constrains (현실적 제한조건)

#### 4.3.1 Constraints for Available Resources (가용 자원에 대한 제한조건)

- 다수의 사용자를 수용할 수 있는 서버

: 다수의 클라이언트가 서버로 진입할 때 모든 서비스 요청을 처리할 수 있어야 한다.

- 외부에서 입력되는 데이터와 내부에서 발생하는 데이터를 수용할 수 있는 서버

: 사용되는 서버는 사용자가 입력한 동영상 데이터를 저장할 수 있어야 한다. 또한 UC Berkeley BDD100K 데이터 셋, AIhub 제공 데이터와 Yolo Detection 출력 결과물 그리고 augmentation 결과 데이터를 저장할 수 있어야 한다.

#### 4.3.2 Constraints for Minium Performance (최소 성능에 대한 제한조건)

- 높은 정확도로 인식된 객체 이미지 제공

: 본 프로젝트에서는 AP 측정 방식을 사용하여 IoU(Intersection over Union)를 0.5, 0.7, 0.9의 세 가지 threshold 값으로 변경하며 결과물로 제공하는 32개 클래스에 대한 mAP를 측정한다. 각각의 threshold 값은 '다중 센서를 사용한 주행 환경에서의 객체 검출 및 분류 방법'의 논문을<sup>8)</sup> 참고하여 정하였다. 참고 자료를 기반으로, 제공하는 전체 클래스의 mAP를 85%로 정하였으며, 대개 80% 이상이면 높은 정확도라고 한다.<sup>9)</sup> 따라서, mAP를 85%로 하여 정확도 높은 데이터 셋을 제공할 수 있도록 한다.

- 빠른 시간 내에 객체 이미지 제공

: 사용자 요청에 따른 이미지 제공 과정이 빠른 시간 내에 이뤄져야 한다. YOLO 모델 학습과 객체 인식 작업을 사전에 진행하며 사용자 요청이 발생하면 저장된 객체 이미지를 곧바로 전송해야 한다.

- 데이터 손실 방지

: 사용자에게 제공되는 최종 결과물 데이터는 무결해야 한다. 내부 시스템에서 작업이 진행되거나 사용자에게 전송될 때 데이터 손실이 발생하면 안 된다.

#### 4.3.3 Constraints for Security (보안에 대한 제한조건)

- 사용자 입력 동영상 유출 방지

: 사용자 입력 동영상이 외부로 유출되는 경우를 방지해야 한다. 웹 브라우저를 통해 서버로 전송되

8) 김정언, 강행봉, 「다중 센서를 사용한 주행 환경에서의 객체 검출 및 분류 방법」, Journal of Korea Multimedia Society Vol. 20, No.8, 2017, 9쪽.

9) 김영인, 최인훈, 김명현, 김승직, 허의남, 「딥러닝 기반 객체 인식을 통한 실내 위치 정보 탐색 자율 주행 로봇 개발」, 경희대학교, 2020, 2쪽.

는 과정에서 외부에서의 접근을 차단해야 한다.

- 네트워크 공격에 대한 방어

: 사용자는 외부 네트워크를 통한 공격으로부터 보호받아야 한다. 웹 브라우저로 접속하는 과정에서 사용자 개인 정보가 유실되는 사태를 방지한다.

- 시스템 내부 데이터 보호

: 객체 검출 이미지가 포함된 시스템 내부 데이터는 외부의 공격으로부터 보호받아야 한다.

#### 4.3.4 Constraints for Compatibility (호환성에 대한 제한조건)

- 내부 시스템 개발 환경에 대한 호환성

: YOLO 학습 모델을 구축하기 위해 호환 가능한 프레임워크를 사용해야 한다. 그리고 정해진 프레임워크에서 사용 가능한 tool들을 선택해야 한다.

- 이식 가능한 학습 모델의 호환성

: 학습된 YOLO v4 모델은 웹 사이트가 아닌 또 다른 프로그램 형식으로 이식할 수 있어야 한다. 앱 형식과 같은 다른 프로그램 형식에서도 호환성을 보장해야 한다.

#### 4.3.5 Constraints for Ethics (윤리성에 대한 제한조건)

- 사용자 입력 데이터 유출 방지

: 사용자가 입력한 영상 데이터는 본 프로그램의 작업 과정에서만 사용되어야 한다. 외부로 유출되는 경우를 방지해야 한다.

- 사용자의 개인 정보를 받아들이지 않는 서비스 제공

: 회원 가입 절차를 거치지 않고 본 프로그램의 서비스를 이용할 수 있어야 한다. 민감한 개인 정보는 받아들이지 않으며 입력 데이터는 사용자의 동영상 데이터로 한정 지어야 한다.

## 5. Project Contents (과제 내용)

### 5.1 Design in Detail (상세 설계)

#### 5.1.1 System Structure (시스템 구조)

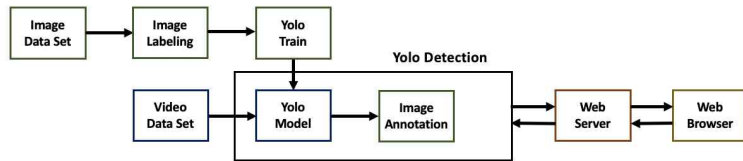


그림 2 Overall System Structure

**Image Data Set** : Yolo 학습 모델을 생성할 때 사용되는 학습용 이미지 데이터 셋이다. Yolo 학습 모델의 성능을 위해 MS COCO dataset과 유사한 형태의 학습용 이미지 데이터 셋을 구축한다.

**Image Labeling** : 앞서 구축해놓은 학습용 이미지 데이터 셋을 Yolo 학습에 사용될 데이터 형식에 맞춰 Image Labeling을 진행한다. 클래스는 크게 자동차, 보행자, 차선, 신호등, 표지판, 노면 표시, 노면 화살표로 분류되며 각 클래스는 다시 세부적으로 분류된다.

**Yolo Train** : Labeling된 이미지 데이터로 Yolo 학습을 진행한다. 수행 후 생성된 Yolo 학습 모델은 추후 영상 데이터 셋에서 Yolo Detection을 수행하는 데에 사용된다.

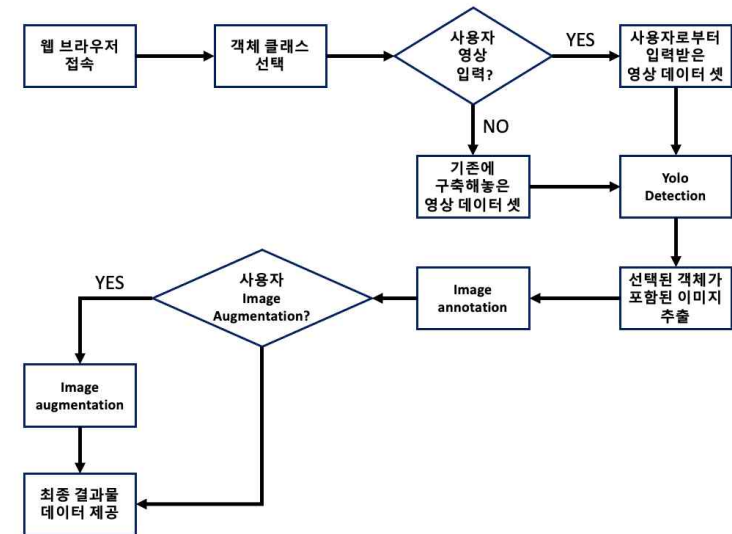
**Yolo Detection** : 미리 생성해놓은 Yolo 학습 모델을 이용하여 이미지에서 객체를 탐지하고, 동시에 탐지된 객체의 bounding box 좌표 값도 제공하는 annotation을 진행한다. Web Server에게 수행 결과물인 이미지 데이터와 annotation 정보를 전달한다.

**Video Data Set** : Yolo Detection이 수행될 대상 영상 데이터 셋이다. 미리 수집해놓은 영상 데이터가 존재하고, 사용자가 자신이 원하는 영상을 직접 입력할 때는, 사용자가 입력한 영상에서 Yolo Detection이 수행된다.

**Web Server** : 프로그램은 Web Server에 배포되어 있고 사용자는 해당 서버 주소를 통해 웹 사이트에 접속할 수 있다.

**Web Browser** : Web Browser를 통해 사용자에게 프로그램을 보여주게 된다. Web Server에게 사용자로부터 받은 HTTP 요청을 하게 되며 Web Server로부터 전달받은 내용을 화면에 렌더링하여 사용자에게 보여준다.

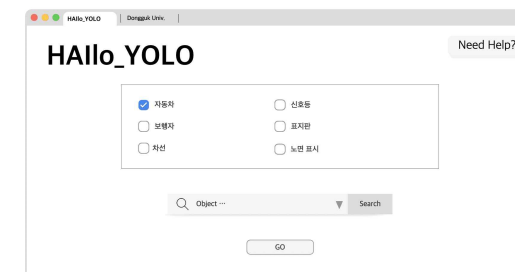
#### 5.1.2 Module Specification (모듈 설계)



### 5.2 Implementation (구현)

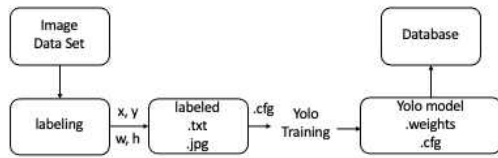
#### 5.2.1 Implementation Result (구현 결과)

##### 5.2.1.1 웹 서버 호스팅



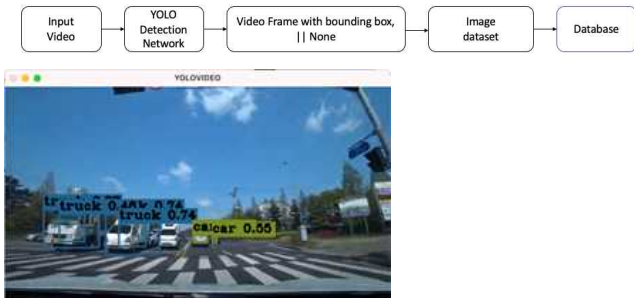
사용자는 웹 서버 호스팅 후 웹 브라우저를 통해 사이트에 접속할 수 있다. 웹 브라우저는 웹 서버로부터 전달받은 내용을 화면에 렌더링한다.

## 5.2.1.2 Yolo Train



학습을 수행할 때 사용될 이미지 데이터 셋에서 지정된 객체에 대한 라벨링을 진행한다. 이때 데이터를 높은 품질로 생성하기 위해, 인공지능 데이터 구축 가이드라인<sup>10)</sup>을 참고하여 객체 영역의 외곽에 맞춰 최대한 타이트하게 라벨링을 진행한다. 라벨링 Tool은 Yolo 학습에 사용되는 데이터 형식과 같은 결과물을 제공할 수 있는 Ybat Tool을 사용한다. 라벨링의 결과로 생성된 이미지의 annotation 정보와 이미지 데이터 셋을 이용하여 Colab 환경에서 Darknet 프레임워크를 사용하여 Yolo 학습을 수행한다. 이때 설정 파일인 .cfg 파일을 학습을 위한 값을 조절한다. Yolo 학습이 끝난 뒤에는 학습의 결과물인 .weights 파일을 얻게 된다. 최종 Yolo 학습 모델인 .weights 파일과 .cfg 파일은 프로그램의 Yolo Detection 기능에 사용될 수 있도록 서버상의 데이터베이스에 저장된다.

## 5.2.1.3 Yolo Detection



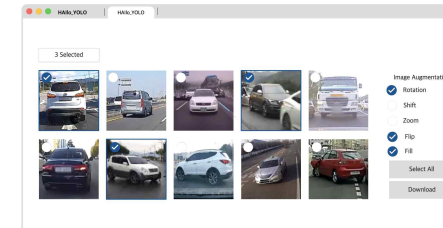
5.2.1.2의 결과물인 Yolo 학습 모델을 이용하여 Yolo Detection이 수행된다. 영상 데이터 셋에서 Yolo Detection을 수행하며 객체가 포함된 순간의 프레임 정보(이미지 데이터)와 객체의 bounding box 정보를 얻을 수 있다. Yolo Detection 수행 중 얻은 이미지와 bounding box 정보는 Database에 저장되며 수행이 완료되면 해당 Database에 저장된 이미지 데이터 셋과 annotation 정보는 사용자에게 제공된다.

10) 정하옥 외, 인공지능 데이터 구축, 활용 가이드라인 19. 주행환경 정적객체 영상, PDF file, <[https://aihub.or.kr/sites/default/files/Sample\\_data/구축활용가이드북\\_2020-02/054.차선\\_횡단보도\\_인지\\_영상%28수도권%20외%29\\_데이터\\_구축\\_가이드라인.pdf](https://aihub.or.kr/sites/default/files/Sample_data/구축활용가이드북_2020-02/054.차선_횡단보도_인지_영상%28수도권%20외%29_데이터_구축_가이드라인.pdf)>, 2021.03.12.

## 5.2.1.4 Image augmentation

5.2.1.3이 완료된 후 사용자가 image augmentation 옵션을 선택한다면 5.2.1.3의 결과물인 이미지 데이터 셋과 annotation 정보는 albumentations tool을 사용하여 augmentation이 진행된다. 이때 이미지에 대한 augmentation뿐만 아니라 해당 이미지에 대한 annotation 정보(bounding box 정보)도 함께 augmentations을 수행해주어야 한다.

## 5.2.1.5 이미지 데이터 셋 제공



사용자는 웹 브라우저를 통해 최종 결과물인 이미지 데이터 셋과 이미지 annotation 정보를 다운 받을 수 있다.

## 5.2.2 Implementation Tools (구현 도구)

## Framework

- Darknet : Yolo 학습에 사용된 프레임워크. Yolo가 리눅스에서 Darknet 기반이며 비교적 빠른 속도를 낼 수 있다.
- NodeJS : 웹 사이트는 NodeJS 기반의 서버를 구축하여 진행한다.

## Labeling Tool

- Ybat : 별도의 설치 없이 html 파일에서 실행할 수 있으며 Yolo 학습에 사용될 데이터 형식과 같은 결과물을 제공한다.

## Library

- OpenCV : 컴퓨터 비전 오픈 소스 라이브러리로 Yolo Detection 수행에 사용된다.
- Albumentations : image augmentation에 사용되는 파이썬 라이브러리로 사용자의 선택에 따라 image augmentation을 진행할 때 사용된다.

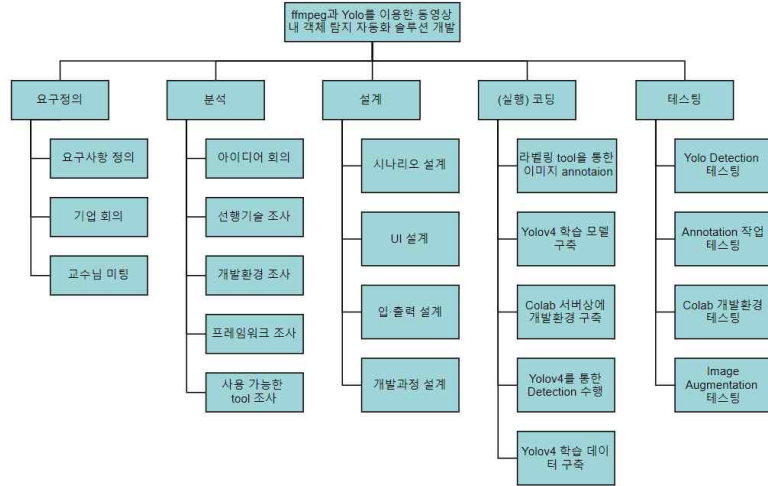
## Infra

- cloud : 클라우드 상에 웹 사이트를 구성한다.



### 5.3 Tasks and Task Assignments (과업 및 과업 분담)

#### 5.3.1 Work Breakdown Structure (과업 분해)



#### 5.3.2 Task Assignment and Project Timeline (과업 분담 및 과제 진행)

Task	작업자	진행 상태	March				April				May				June			
			W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4
요구정의																		
요구사항 정의	All	진행 완료																
기업 회의	All	진행 완료																
교수님 미팅	All	진행 완료																
분석																		
아이디어 회의	All	진행 완료																
선행기술 조사	All	진행 완료																
개발환경 조사	All	진행 완료																
프레임워크 조사	All	진행 완료																
사용 가능한 tool 조사	All	진행 완료																
설계																		
시나리오 설계	All	진행 완료																
UI 설계	이은영, 한유진	진행 완료																
입출력 설계	All	진행 완료																
개발과정 설계	All	진행 완료																
구현																		
라벨링 tool을 통한 이미지 annotation	신소희	진행 중																
YOLOv4 학습 모델 구축	신소희, 조성운	진행 중																
Colab 서버상에 개발환경 구축	한유진	진행 중																
YOLOv4를 통한 Detection 수행	이은영	진행 중																
YOLOv4 학습 데이터 구축	조성운	진행 중																
테스팅																		
YOLO Detection 테스트	이은영	진행 완료																
Annotation 작업 테스트	조성운	진행 중																
Colab 개발환경 테스트	한유진	진행 완료																
Image Augmentation 테스트	신소희	진행 중																

&lt;Table 1&gt; Peer Evaluation Table

Tasks	Detail Description	Weight (1-10)	신소희 A (%)	이은영 B (%)	조성운 C (%)	한유진 D (%)	Total (%)
요구사항 정의	프로그램의 요구기능 및 동작 절차 결정	7	25	25	25	25	100
기업 회의	기업과 미팅 진행	6	25	25	25	25	100
교수님 미팅	교수님과 미팅 진행	6	25	25	25	25	100
아이디어 회의	아이디어 구체화	8	25	25	25	25	100
선행기술 조사	기존에 이미 존재하는 기술 조사	7	25	25	25	25	100
개발환경 조사	프로그램이 구현될 서버와 개발 언어 조사	8	25	25	25	25	100
프레임워크 조사	서버상에서 YOLOv4 학습 모델을 구현하기 위한 프레임워크 조사	6	25	25	25	25	100
사용 가능한 tool 조사	개발 단계에서 사용 가능한 tool 조사	7	25	25	25	25	100
시나리오 설계	사용자의 구체적인 task 설계	5	25	25	25	25	100
UI 설계	프로그램 실행 시 UI 설계	5	0	50	0	50	100
입출력 설계	프로그램의 입력 및 출력 설계	5	25	25	25	25	100
개발과정 설계	프로그램을 구현하기 위한 개발 단계 구체화	8	25	25	25	25	100
라벨링 tool을 통한 이미지 Annotation	Ybat tool을 이용하여 Annotation 작업 수행	5	20	30	20	30	100
YOLOv4 학습 모델 구축	YOLOv4 모델링 작업 수행	6	25	25	25	25	100
Colab 서버상에 개발환경 구축	Colab 서버에 Darknet 설치	4	30	20	30	20	100
YOLOv4를 통한 Detection 수행	YOLOv4를 통해 객체 인식 작업 진행	5	20	30	20	30	100
YOLOv4 학습 데이터 구축	객체가 담긴 학습 이미지 데이터 수집	4	30	20	30	20	100
YOLO Detection 테스트	객체 인식 작업 실험 진행	4	50	0	50	0	100
Annotation 작업 테스트	Annotation 작업 실험 진행	5	20	30	30	20	100
Colab 개발환경 테스트	Colab 서버에서 프로그램 실행을 통해 실험 진행	4	30	20	20	30	100
Image Augmentation 테스트	이미지 증폭 작업 실험 진행	4	25	25	25	25	100
Individual Sum			525	525	525	525	

## 6. Anticipated Results and Contributions (예상 결과 및 기대 효과)

본 프로젝트는 자율주행 학습에 필요한 대량의 이미지 데이터를 수집하고 라벨링 하는 과정을 자동화하는 웹 사이트를 결과로 도출한다. 프로젝트 구현물의 출력 결과는 사용자가 체크박스를 통해 입력한 객체의 이미지 데이터 셋과 그에 상응하는 바운딩 박스 좌표 값이다. 이미지 데이터 셋과 라벨링이 완료된 데이터만 준비된다면, 바로 학습에 사용할 수 있다. 따라서, 사용자는 데이터 수집과 라벨링을 따로 하지 않고, 해당 결과물을 이용하여 곧바로 학습 모델을 만드는 것이 가능하다.

이 프로젝트를 통해서, 인공지능 개발자들이 학습할 때 걸리는 시간을 최소화시킬 수 있다. 전처리 과정에 소요되는 시간이 줄어들면 학습 모델을 구축하거나 알고리즘을 구현하는 데에 더욱 집중할 수 있고, 그 과정을 빨리 시작할 수 있다. 궁극적으로 해당 데이터를 이용하여 기계학습 기술의 발전이 빨라질 것이다. 기술의 발전이 빨라지면, 그 영향은 인간에게 미친다. 즉, 인간의 삶이 더 편리해질 것이다. 따라서, 본 프로젝트는 인공지능 개발자부터 일반 사람들에게까지 연쇄적으로 영향을 미칠 것이라 생각한다.

향후, 범주를 확장하여 다양한 분야의 학습 이미지 데이터 셋을 구축할 수 있도록 발전시킬 예정이다. 도메인을 확장하여 다양한 범주의 이미지 수집과 라벨링의 자동화가 실현된다면, 머신러닝의 산업은 가속화하여 발전할 것이며, 각종 반복적이고 단순한 업무들은 기계가 사람의 업무를 대신할 수 있을 것이다. 양날의 검과 같이, 기계가 사람을 대체하는 만큼 사람의 일자리가 줄어들다는 부작용이 있지만, 사람의 삶은 조금 더 편해질 것이라는 효과를 기대해볼 수 있다.

## 7. References (참고 서적)

- 1) “테슬라 반자율주행 ‘불안불안’... 미국서 또 사고”, <조선비즈>, 2021.03.18., <[https://biz.chosun.com/site/data/html\\_dir/2021/03/18/2021031801749.html](https://biz.chosun.com/site/data/html_dir/2021/03/18/2021031801749.html)>, <2021.06.03.>
- 2) “AI 학습용 데이터 구축, AI 강국의 기반 다진다”, <컴퓨터 월드>, 2021.01.31., <<https://www.comworld.co.kr/news/articleView.html?idxno=50123>>, <2021.06.03.>
- 3) “AI 시대 숨은 주역, ‘데이터 노동자’를 아시나요”, <정보통신 신문>, 2020.10.20., <<https://www.koit.co.kr/news/articleView.html?idxno=79944>>, <2021.06.03.>
- 4) 송순용, 「이미지 데이터셋 제작을 위한 Annotation 자동화 기술 분석」, ITFIND 주간기술동향 1987호, 한국전자통신연구원, 2021, 2쪽.
- 5) “딥러닝(Deep Learning), 인공지능의 핵심 기술 알아보기”, LG디스플레이 기업 블로그, 2016.06.16. <<https://blog.lgdisplay.com/2016/06/deep-learning/>>, 2021.06.03.
- 6) 서윤경, 조상흠, 황원준, 「최근 정보통신 기술 YOLO 를 통해 자율주행자동차의 object detection 과 classification 에 관한 연구」, 한국통신학회, 2, 3쪽.
- 7) 김영인, 최인홍, 김명현, 김승직, 허의남, 「딥러닝 기반 객체 인식을 통한 실내 위치 정보 탐색 자율 주행 로봇 개발」, 경희대학교, 2020, 2쪽.
- 8) 김정언, 강행봉, 「다중 센서를 사용한 주행 환경에서의 객체 검출 및 분류 방법」, Journal of Korea Multimedia Society Vol. 20, No.8, 2017, 9쪽.

9) 정하옥 외, 인공지능 데이터 구축, 활용 가이드라인 19. 주행환경 정적객체 영상, PDF file, <[https://aihub.or.kr/sites/default/files/Sample\\_data/구축활용가이드북\\_2020-02/054.차선\\_횡단보도\\_인식\\_영상%28수도권%20외%29\\_데이터\\_구축\\_가이드라인.pdf](https://aihub.or.kr/sites/default/files/Sample_data/구축활용가이드북_2020-02/054.차선_횡단보도_인식_영상%28수도권%20외%29_데이터_구축_가이드라인.pdf)>, 2021.03.12.

## 8. Budget (예산)

해당 없음