

# 2023년 한이음 ICT멘토링 프로젝트 결과보고서

프로젝트명	반려견 목줄 탐지 서비스
-------	---------------

## 요 약 본

프로젝트 정보	
프로젝트명	반려견 목줄 탐지 서비스
주제 영역	<input checked="" type="checkbox"/> 생활 <input type="checkbox"/> 업무 <input checked="" type="checkbox"/> 공공/교통 <input type="checkbox"/> 금융/핀테크 <input type="checkbox"/> 의료 <input type="checkbox"/> 교육 <input type="checkbox"/> 유통/쇼핑 <input type="checkbox"/> 엔터테인먼트
기술 분야	<input checked="" type="checkbox"/> SW-AI <input type="checkbox"/> 방송·콘텐츠 <input type="checkbox"/> 블록체인·융합 <input checked="" type="checkbox"/> 디바이스 <input type="checkbox"/> 차세대보안 <input type="checkbox"/> 미래통신·전파
달성 성과	<input type="checkbox"/> 논문게재 및 포스터 발표 <input type="checkbox"/> 앱등록 <input type="checkbox"/> 프로그램등록 <input checked="" type="checkbox"/> 특허 <input type="checkbox"/> 기술이전 <input checked="" type="checkbox"/> 실용화 <input checked="" type="checkbox"/> 공모전( 2023년 한이음 ICT멘토링 공모전 ) <input type="checkbox"/> 기타(   )
프로젝트 소개	AI 기반 IoT 장비를 통해 반려견 목줄 미착용 상황을 탐지하고, 적절한 조치를 취해 목줄 미착용으로 인한 개 물림 사고를 예방한다. 기존의 복잡했던 목줄 미착용 반려견 신고 절차를 간소화하여 효율적인 단속 시스템을 공공기관에 제공한다.
개발 배경 및 필요성	목줄 착용 의무 관련 동물보호법 시행 규칙이 개정되어도 개 물림 사고가 지속적으로 발생하고 있다. 또한, 단속 인력 부족과 단속 시스템의 비효율성으로 인해 반려견 목줄 착용 의무가 지켜지고 있지 않다.
프로젝트 특징	현재까지 목줄 미착용 반려견 단속을 위한 AI 시스템은 국내외로 관련 사례가 전무하다. 본 프로젝트는 목줄 미착용 반려견에 대한 정부의 통합적 관리와 대응뿐만 아니라 정리된 통계 자료 구축을 가능하게 한다.
주요 기능	IoT 장비로 입력되는 영상을 AI 모델을 통해 반려견을 탐지한 후, 탐지된 반려견의 목줄 착용 여부를 파악한다. 목줄 미착용 판단 시 적절한 조치를 취하고, 착용 시는 일반적인 CCTV 기능을 수행한다. 목줄 미착용 반려견 정보를 공공기관 담당 부처에게 웹사이트를 통해 제공한다.
기대효과 및 활용 분야	공공장소 등에서 활용하여 목줄 미착용 반려견 탐색 시 즉각적인 목줄 착용 요청 및 자동 신고를 통해 반려인들의 목줄 착용의 인식을 향상시키고, 이를 통해 목줄 미착용으로 인한 사고를 방지할 수 있다.

# (본문) 프로젝트 결과보고서

## I. 프로젝트 개요

### 1. 프로젝트 소개

#### 1) 기획 의도

본 프로젝트는 반려견 목줄 미착용 상황을 탐지함으로써 견주에게 경각심을 고취 및 목줄 착용 문화 형성

- 미리 적절한 조치를 통한 사고 예방
- 목줄 미착용 반려견의 견주 단속

#### 2) 프로젝트 내용

- ① AI 모델을 통한 반려견 객체 탐지 및 목줄 착용 여부 분류
  - 객체 탐지 AI 모델을 통해 카메라로 입력되는 영상에서 반려견 탐지
  - CNN 기반 이미지 분류 AI 모델을 통해 반려견의 목줄 착용 여부 판단
- ② IoT 장비를 통해 반려견 목줄 착용 권장 안내 방송 송출
  - AI 모델 추론 후 목줄 미착용 반려견 탐지 시 IoT 장비를 통해 즉각적으로 목줄 착용 요청 방송 송출
- ③ 반려견 출입 시설 관리자용 웹페이지 제공
  - 목줄 미착용으로 판단된 객체의 정보를 담당 부서에게 실시간으로 전송
  - 목적에 따라 안내 방송 멘트 설정 가능
  - 목줄 미착용 반려견이 적발된 시간대와 위치 정보 통계 제공

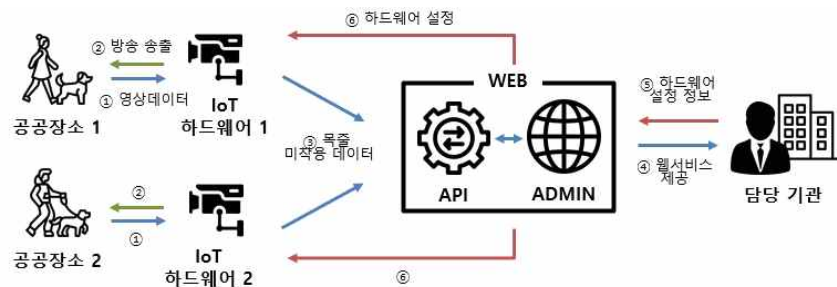


그림 1. 프로젝트 내용

## 2. 개발 배경 및 필요성

### 1) 프로젝트 제작 동기

- 국내 반려동물 양육 비율이 증가하는 가운데 반려견 목줄 미착용으로 인한 개 물림 사고가 지속적으로 발생하고 있다.
- 개 물림 사고는 견주의 반려견 목줄 착용 중요성 인식 부족에서 비롯된다.

진돗개와 산책하던 견주가 목줄이 풀린 개에게 공격을 받은 사건을 전했다. 지난해 동물보호법 시행규칙이 시행되며 반려견 보호자는 외출 시 반드시 **목줄을 2m 이내로** 채워야 하지만 현장에서는 잘 지켜지지 않아 종종 개 물림 사고가 발생하고 있다.

(출처 : [이데일리] 진돗개 견주, 목줄 풀린 핏볼과 헐투)

- 정부에서는 반려견 목줄 착용 의무를 법(동물 보호법 시행규칙 제 12조 1항)으로 제정하여 목줄 미착용 반려견을 신고 대상으로 취급하여 견주를 처벌하고 있다. 하지만 단속 시스템이 비효율적이라는 문제가 있어 실질적으로 처벌 효과가 미비하다.

처벌을 받게 하려면 신고자가 증거까지 확보하면서 처리 상황을 확인해야 한다. 신고자가 직접 근처 CCTV 영상을 확보하거나 사진 또는 동영상을 촬영하는 등 증거를 수집해 담당 부서에 처벌을 촉구하는 식이다. 1

(출처 : [단비뉴스] 목줄 안 한 개, 신고하려면 주소까지 알아야)

### 2) 프로젝트 목적

- 본 프로젝트는 AI 기술을 통해 단속 시스템의 한계를 극복하는 것을 목적으로 한다.
- 기존의 시민 신고에 의존적이고 복잡한 목줄 미착용 신고 시스템을 자동화 및 단순화함으로써 효율적인 단속 시스템을 구축한다.

## 3. 프로젝트 특·장점

- 현재까지 목줄 미착용 반려견 단속을 위한 AI 시스템은 국내외의 관련 사례가 전무하다.
- 관련 부처별 정리된 통계 자료가 없어 정부의 통합적 관리와 대응이 불가능했지만, 본 프로젝트를 통해 목줄 미착용과 관련한 가공된 통계 자료를 구축할 수 있다.

## II. 프로젝트 내용

### 1. 프로젝트 구성도

#### 1) 기능 흐름도

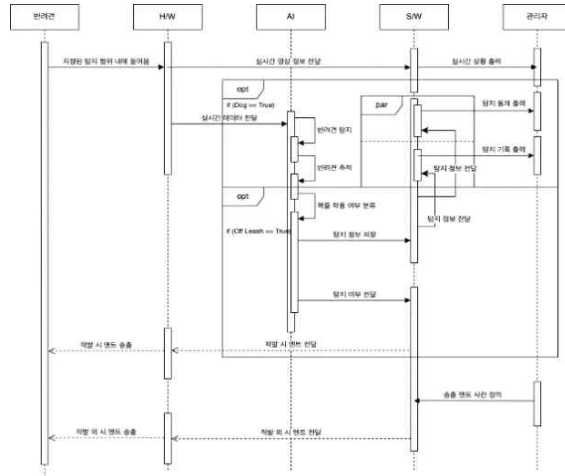


그림 2. 기능 흐름도.

설명.

- 미리 설정된 영역 내에 반려견이 들어오면 감지를 시작한다.
- 감지된 반려견이 지정된 범위 내에 들어오면 시스템은 해당 영역에서 실시간으로 상황을 촬영한다.
- 촬영된 실시간 상황은 사용자가 접근 가능한 인터페이스나 플랫폼을 통해 실시간으로 송출된다.
- 촬영되는 상황에서 시스템은 촬영 객체가 반려견인지 탐지한다.
- 탐지된 반려견 객체를 추적하여 목줄 착용 여부를 판단한다.
- 감지된 정보와 목줄 착용 여부 등의 데이터를 저장한다(차후 탐지 기록 및 통계 분석을 위해 사용).
- 관리자는 시스템을 통해 실시간으로 목줄 미착용 반려견이 탐지됨을 알 수 있다.
- 관리자는 목줄 미착용 반려견 탐지 정보를 기록 및 통계를 통해 확인할 수 있다.
- 시스템은 상황에 따라 사전에 정의된 멘트를 송출할 수 있다.

### 2) AI 흐름도

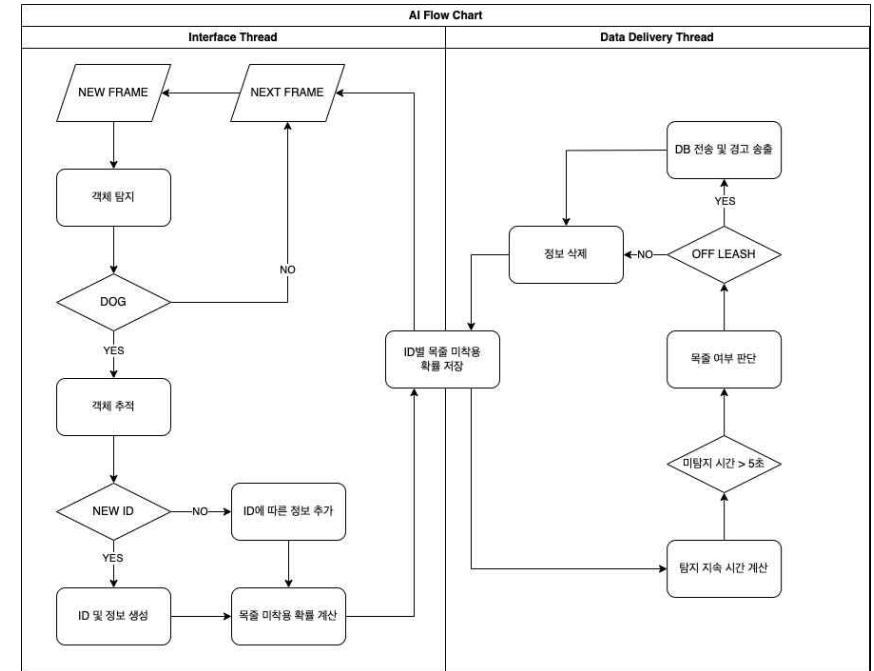


그림 3. AI 흐름도

설명.

- 반려견 정보 저장 공간을 공유하는 2개의 멀티 스레드를 활용하여 기능한다. AI 추론 스레드는 캠 으로부터 입력되는 프레임에 학습된 AI 모델로 분석하여 정보 저장 공간으로 관련 정보를 저장한다. 데이터 전송 스레드는 실시간으로 업데이트되는 저장 공간의 반려견 정보를 ID별로 분석하여 서버로 데이터를 전송하고 임시 정보 저장 공간에서 관련 정보를 삭제한다.

#### AI 추론 스레드 (Inference Thread)

- 영상의 프레임이 입력되면 반려견 객체 탐지 모델의 추론 결과 입력된 프레임 안에 반려견이 존재하면 존재 영역의 경계 상자 좌표를 반환한다.
- 객체 추적 알고리즘을 통해 탐지된 반려견 ID를 부여한다. 이전에 탐지된 반려견과 동일한 반려견일 경우 같은 ID를 부여한다.
- 새로운 ID를 부여받은 반려견의 count 속성을 1로 초기화한다. count는 해당 반려견이 탐지된 횟수를 뜻한다. 기존 ID의 반려견은 count를 1회 증가시킨다.
- 이미지 분류 모델을 통해 반려견 목줄 미착용 확률을 추론한다.
- ID별 반려견의 목줄 미착용 확률을 임시 정보 저장 공간에 저장한다.

### 데이터 전송 스레드 (Data Delivery Thread)

- ID별 반려건의 탐지 지속 시간을 계산하여 5초 이상 새로운 확률 및 좌표 데이터가 업데이트되지 않으면 반려건이 프레임에서 벗어났다고 판단한다.
- ID별 반려건의 저장된 모든 목줄 미착용 확률을 고려하여 최종적으로 반려건의 목줄 착용 여부를 판단한다.
- 목줄 미착용으로 판단 시, DB에 관련 정보를 전송하고 경고 방송을 송출한다.
- ID별 반려건 저장 공간에서 해당 반려건 정보를 삭제한다.

### 3) 하드웨어 흐름도

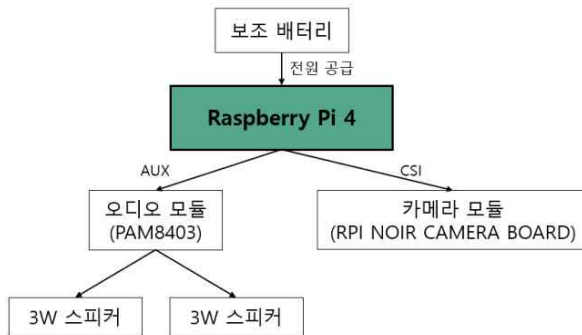


그림 4. 하드웨어 흐름도.

#### 설명.




- 카메라 모듈을 CSI의 핀을 이용해서 라즈베리파이와 연결한다.
- 라즈베리파이 4의 전원을 C-type핀을 이용해 보조배터리 혹은 어댑터를 이용해서 전원 공급한다.
- 오디오 모듈에 왼쪽과 오른쪽을 각각 담당하는 스피커를 연결하고 이를 라즈베리파이4로부터 전원을 공급받아서 구동한다.


## 2. 프로젝트 기능

### 1) 전체 기능 목록

구분	기능	설명	현재진척도(%)
AI	반려견 객체 탐지	입력되는 영상에서 반려견 경계 상자 좌표 추론	100%
	목줄 착용 여부 분류	인식된 반려견의 목줄 착용 여부를 판단	100%
	반려견 객체 추적	입력되는 영상에서 프레임별로 동일 객체(반려견)를 추적	100%
	경량화	AI 모델 양자화를 통해 메모리 사용량과 연산 속도 개선	100%
S/W	데이터 관리	목줄 미착용 반려견 탐지 시 들어오는 정보를 MySQL을 이용하여 DB에 저장. DB의 정보를 가공하여 API로 전달.	100%
	통계	관리자의 가독성을 높이기 위해 그래프와 목록으로 정보 전달.	100%
	실시간 관리 감독	목줄 미착용 반려견 탐지 시 실시간 상황 송출.	90%
	하드웨어 설정	각각의 하드웨어에 따라 음향, 무음 시간 설정, 안내멘트 설정.	90%
H/W	목줄 착용 권장 멘트 송출 (H/W 구매 완료)	목줄 미착용 반려견 탐지 시 목줄 착용을 요청하는 안내멘트를 송출	100%
	카메라 구동 (H/W 구매 완료)	반려견 탐지 및 목줄 미착용 여부를 파악하기 위한 영상 입력 장치를 구동	100%
	3D 모델링	3D 모델링을 통해 적당한 크기로 제작.	100%

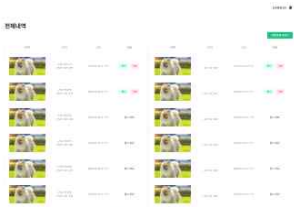
## 2) AI 주요 기능

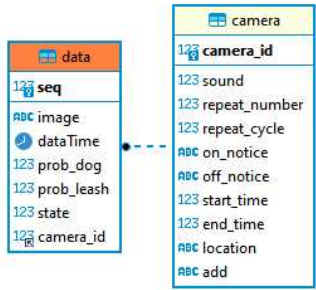

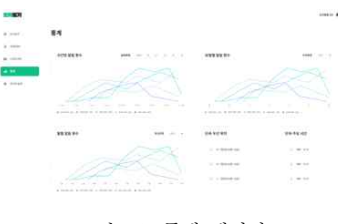
기능	설명	프로젝트 실물 사진
반려견 객체 탐지	<ul style="list-style-type: none"> <li>- 입력되는 영상으로부터 반려견 경계 상자 좌표를 추출하기 위하여 객체 탐지 모델을 학습시킨다*.</li> <li>- 정확도와 FPS(초당프레임수)를 고려하여 최적의 모델을 선정한다.</li> <li>- 파라미터수가 작은 사전 학습된 모델을 반려견 이미지 데이터셋으로 전이학습시켜 객체 탐지 정확도를 향상시킨다.</li> </ul>	 <p>그림 5. 객체 탐지 결과</p>
목줄 착용 여부 분류	<ul style="list-style-type: none"> <li>- 객체 탐지 모델로부터 추론된 반려견 경계 상자를 crop한 이미지로 반려견의 목줄 착용 여부를 판단하는 CNN 모델을 학습시킨다*.</li> <li>- Grad-CAM 모델 시각화 기술을 통해 AI 모델이 반려견의 목줄을 집중적으로 탐색하는지 확인한다.</li> <li>- 모델의 정확도와 Grad-CAM 시각화 정보를 통해 최적의 CLS 모델을 선정한다.</li> </ul>	 <p>그림 6. 목줄 착용 여부 파악</p>
반려견 객체 추적	<ul style="list-style-type: none"> <li>- 객체 추적 기술로 프레임 별로 탐지된 반려견의 특징적인 정보의 유사도를 이용하여 동일 객체를 추적한다.</li> <li>- 동일한 강아지 정보가 DB에 반복적으로 전송되는 것을 방지하는 기능을 한다.</li> <li>- 사람이나 사물에 의해 목줄이 가려져 목줄 미착용으로 판단하는 것을 방지한다. 한 프레임이 아니라 동일 객체의 여러 프레임을 분석하여 목줄 미착용 여부 판단 정확성을 높여주는 기능을 한다.</li> </ul>	 <p>그림 7. 객체 추적 결과</p>

경량화	<ul style="list-style-type: none"> <li>- 모델 경량화를 위한 방법 중 하나인 양자화(Quantization)를 통해 이미지 분류 모델의 가중치 값을 더 가벼운 정밀도로 표현한다. 이 과정을 통해 모델의 메모리 사용량과 연산 속도를 줄인다.</li> <li>- 양자화된 모델을 전이학습시키는 방법, 학습된 모델을 양자화하는 방법, 양자화를 고려한 학습법을 시도하여 AI 모델의 정확하고 빠른 추론을 가능하게 하는 기능이다.</li> </ul>	 <p>그림 8. 모델 경량화 코드</p>
-----	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------

- 객체 탐지 모델 훈련 시 사용되는 커스텀 데이터셋은 AI-hub에서 제공하는 <공원 주요시설 및 불법행위 감시 CCTV 영상 데이터>, <Small Object Detection을 위한 이미지 데이터>로, 10,524장의 반려견 이미지로 구성되어 있다. 이미지 증강 기술을 활용해 모델 훈련에 28,603장의 데이터가 사용되었다.
- 분류 모델 훈련 시 사용되는 데이터셋은 <공원 주요시설 및 불법행위 감시 CCTV 영상 데이터>, <Small Object Detection을 위한 이미지 데이터> 중 반려견 이미지 경계 상자의 상하좌우로 경계 상자보다 30% 넓은 영역을 포함하도록 crop하여 구성하였다. 이는 배경의 영향을 최소화하여 데이터의 편향성을 극복하기 위함이다.

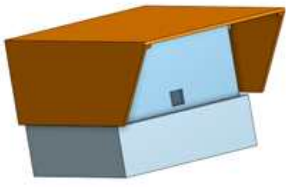
## 3) S/W 주요 기능

기능	설명	프로젝트 실물 사진
실시간 관리 감독	<ul style="list-style-type: none"> <li>- Django 프레임워크를 이용하여 공공장소에서 탐지된 목줄 미착용 반려견 데이터를 전달하는 웹페이지를 제작한다.</li> <li>- AI로부터 전달받은 데이터는 MySQL을 이용하여 DataBase를 설계하여 관리한다.</li> <li>- 목줄 미착용 반려견은 실시간으로 웹페이지로 전달한다.</li> <li>- 전달된 데이터는 이미지, 시간, 날짜 순으로 확인되며 대응 부분에서 단속 처리 더블체크가 가능하다.</li> </ul>	 <p>그림 9. 상세보기 페이지</p>

데이터 관리	<ul style="list-style-type: none"> <li>- MySQL에 들어오는 데이터를 가공하여 각 기능에 맞추어 설계를 진행한다.</li> <li>- 데이터를 기능에 맞게 SQL을 가공하여 API로 전달한다.</li> <li>- API에서 전달된 정보를 웹페이지 내에 자바스크립트로 전달받아 사용한다.</li> <li>- API 사용으로 기능을 모듈화하여 제공하므로 다른 어플리케이션에도 해당 기능을 활용할 수 있고, 코드의 중복이 적어지며, 재사용성을 높인다.</li> </ul>	 <p>그림 10. DATABASE TABLE</p>
통계	<ul style="list-style-type: none"> <li>- API에서 가공된 정보를 받아 각 위치에 맞게 적용한다.</li> <li>- 대시보드 페이지는 실시간으로 전달된 오늘의 정보를 그래프와 목록을 제공한다.</li> <li>- 목록에서 접수 또는 삭제 버튼을 통해 관리자는 실시간으로 단속이 가능하다.</li> <li>- 통계 페이지는 실시간으로 특정 시점의 정보를 그래프로 제공한다.</li> <li>- 통계 페이지에서 특정 날짜/주차/년도를 선택하여 원하는 데이터를 확인할 수 있다.</li> </ul>	 <p>그림 11. 대시보드 페이지</p>  <p>그림 12. 통계 페이지</p>

#### 4) H/W 주요 기능

기능/부품	설명	프로젝트실물사진
목줄 착용 권장 멘트 송출	<ul style="list-style-type: none"> <li>- 스피커 모듈(DC 5V 전원)과 3W의 앰프를 이용해서 스피커를 제작하고 이를 라즈베리파이와 연동하여 사용자 정의 안내멘트를 송출한다.</li> </ul>	 <p>그림 13. 구매한 하드웨어.</p>  <p>그림 14. 하드웨어와 결합한 사진.</p>
카메라 구동	<ul style="list-style-type: none"> <li>- 라즈베리파이 카메라 모듈 V2를 이용해서 라즈베리파이와 연결하여 AI 모델이 반려견과 목줄 착용 여부를 파악할 수 있도록 영상 데이터를 제공한다.</li> </ul>	<p>Video Streaming Demonstration</p>  <p>그림 15. 카메라 구동 확인.</p>  <p>그림 16. 라즈베리파이와 카메라 연결.</p>

3D 모델링	<ul style="list-style-type: none"> <li>- 3D 모델링을 통해 설계도를 만들고, 3D 프린터를 이용해서 출력한다.</li> <li>- Onshape의 플랫폼을 이용해 제작하였다.</li> </ul>	 <p>그림 17. 3D 설계도</p>
--------	---------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------

### 3. 주요 적용 기술

#### 1) 객체 탐지 (Object Detection) - YOLO(You Only Look Once)

- YOLO는 객체 탐지를 위한 신경망 알고리즘이다. 이미지를 한 번만 보고 객체의 위치와 클래스를 동시에 판별한다. 이 점은 다른 객체 탐지 방법들이 두 단계로 나뉘서 처리하는 것과 대비된다. YOLO 모델은 전체 아키텍처가 end-to-end 학습되므로 속도가 빠르고 사용자 맞춤 전이 학습이 편리하다. 이러한 특징 때문에 실시간 시스템에 많이 사용되는 AI 모델이다. YOLO의 주요 구조와 작동 원리는 아래와 같다.

- (1) 그리드 분할(Grid Division): YOLO는 입력 이미지를  $S \times S$  크기의 그리드로 분할한다. 각 그리드 셀은 B 개의 경계 상자와 해당 경계 상자들이 특정 클래스에 속할 확률을 예측한다.
- (2) 경계 상자 예측: 각 그리드 셀은 B개의 경계 상자를 예측하며, 각각에 대해 5개의 예측값(x, y, w, h, 신뢰도 점수)을 생성한다.
  - (x, y): 경계 상자 중심의 위치를 나타내며 해당 그리드 셀 내에서의 상대적인 위치
  - (w, h): 전체 이미지에 대한 경계 상자 너비 및 높이 비율
  - 신뢰도 점수: 이 특정 경계 상자가 객체를 포함하고 있을 확률과 해당 경계 상자가 얼마나 잘 맞추어져 있는지(Intersection over Union-IoU 메트릭) 두 가지 요소로 구성
- (3) 클래스 확률(Class Probability): 그리드 셀 당 C 개 클래스에 대한 조건부 확률을 계산한다.
- (4) 결합된 예측값: 각 그리드셀에서 B개 경계 상자에 대해  $P(\text{Class}|\text{Object}) * \text{Confidence}(\text{Object})$ 의 형태로 최종 결과값을 계산하여 반환한다.

다음은 YOLOv8\_n 모델을 수집한 데이터로 전이 학습시킨 결과이다.

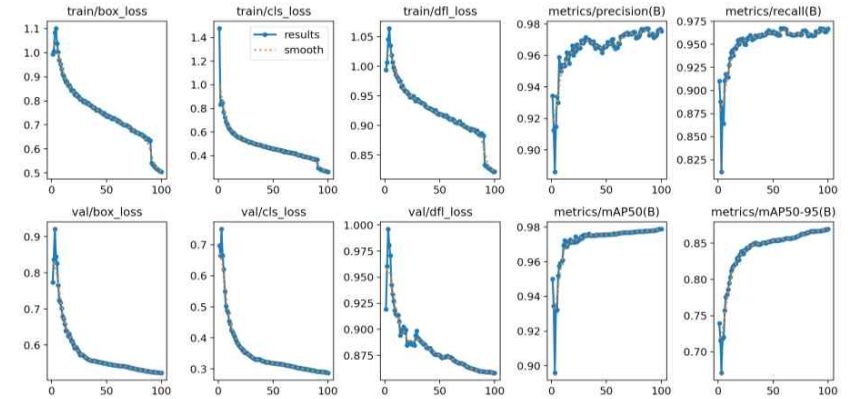


그림 18. 객체 탐지 모델 학습 결과

metrics	Precision	Recall	mAP50	mAP50-95
결과	0.966	0.966	0.98	0.87

#### 2) 객체 추적(Object Tracking) - ByteTracker

- 다중 객체 추적(Multi Object Tracking; 이하 MOT)은 입력 영상에서 여러 객체를 동시에 추적하는 기술이다. 비디오나 실시간 영상에서 객체를 식별하고 움직임을 추적하여 객체의 위치와 경로를 파악한다. MOT 분야에서는 객체 탐지기로부터 전달받은 과거 프레임의 객체와 현재 프레임의 객체를 유사도 기반으로 연결하는 정보 연결(Data Association) 알고리즘을 사용하는 것이 일반적이다. 프레임 내의 유사한 객체들을 연결할 때 사용되는 데이터는 경로 정보(tracklet)와 경계 상자이다.

- (1) Motion model 활용(경로 정보 기준): MOT는 객체의 경로를 예측하여 추적 대상 객체가 다음 프레임에서 어디에 있을지 예측하는 모델인 Motion model을 활용한다. 이때, 주로 칼만 필터(Kalman Filter)가 사용된다. 예를 들어, 직진 중인 차량이 다음 프레임에서 어느 영역에 존재할지 예측하는 기능을 한다.
- (2) Appearance 활용(경계 상자 정보 기준): Appearance 기반 MOT는 객체 탐지기의 결과로 전달받은 경계 상자 내부의 개체 외형을 기준으로 여러 프레임 간의 객체 매칭을 수행한다. 이 방법은 장기 폐색(long-term occlusion: 특정 개체가 한 프레임에서 탐지된 후 다시 탐지되는 프레임 사이에 시간적 공백이 긴 경우)과 같은 상황에서도 동일한 객체로 판단할



수 있도록 한다.

- ByteTracker는 다른 추적 알고리즘과 구별되는 특징을 가지고 있다. ByteTracker는 신뢰도 점수가 낮아서 제외되었던 경계 상자들도 정보 연결 알고리즘에 포함시킨다. 일반적인 MOT 알고리즘이 낮은 신뢰도 점수를 가진 경계 상자 정보(a, Frame t2 참조)를 제외하고 데이터 연결 알고리즘을 수행한다면(b), ByteTracker는 사물에 가려져서 신뢰도 점수가 낮더라도 해당 경계 상자들까지 데이터 연결 시도한다(c). 이러한 접근 방식으로 ByteTracker는 모든 경계 상자들의 데이터 연결을 수행하며 보다 정확한 객체 추적을 가능하게 한다. 따라서 다른 객체 추적 알고리즘에 비해 정확도가 높고 객체 간 ID가 교체되는 ID Switching 문제 해결 기능이 우수하다.

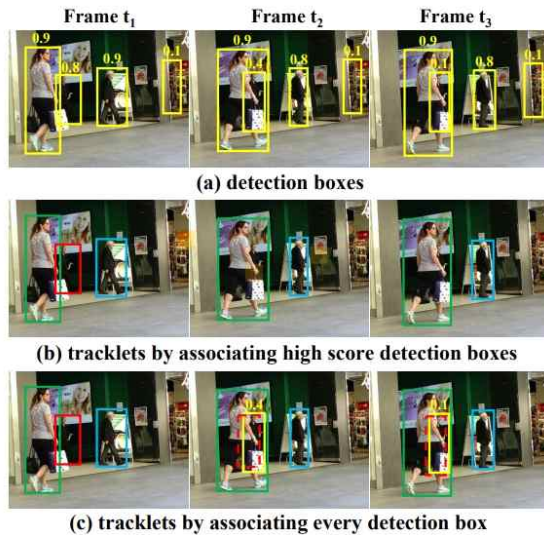


그림 19. ByteTracker의 특징(출처: ByteTrack: Multi-Object Tracking by Associating Every Detection Box)

- 반려견이 목줄을 착용했음에도 불구하고, 사람이나 주변 사물에 의해 목줄이 가려져 일시적으로 목줄 미착용으로 잘못 판단되는 문제가 있었다. 이로 인해 잘못된 정보가 서버로 전송되는 상황을 해결하기 위해 ByteTracker를 객체 추적 알고리즘으로 활용하였다. ByteTracker의 도입은 단일 프레임 기반의 목줄 착용 여부 판단에서 벗어나, 동일한 반려견을 연속된 여러 프레임에서 추적하여 목줄 착용 여부를 보다 정확하게 판단할 수 있게 해준다.

다음은 커스텀 데이터셋으로 학습시킨 YOLO 모델을 객체 탐지기로 하여 ByteTrack 알고리즘을 수행한 결과이다.



그림 20. 서로 다른 프레임에서의 반려견 추적 결과

### 3) CNN(Convolution Neural Network)기반 이미지 분류

- CNN은 Deep Neural Network에서 큰 계산량을 해결하기 위해 등장한 Neural Network로 Weight Shared(가중치 공유)와 같은 특징을 갖는다. 영상 및 사진에서 객체, 클래스, 범주 인식을 위한 패턴을 찾는 데 매우 유용하다. 본 프로젝트에서는 탐지한 반려견의 목줄 착용 여부를 분류하기 위해서 CNN이 사용되었으며, 특히 프로젝트를 위해 학습시켰던 모델들 중 (참고, 기타사항 AI관련)에서 좋은 성능을 보인 SuffleNet\_v2를 이용하여 목줄 착용 여부를 훈련시켰다.
- SuffleNet은 주로 모바일과 같은 자원이 제한된 장치에서 효율적으로 작동하도록 설계된 신경망이다.

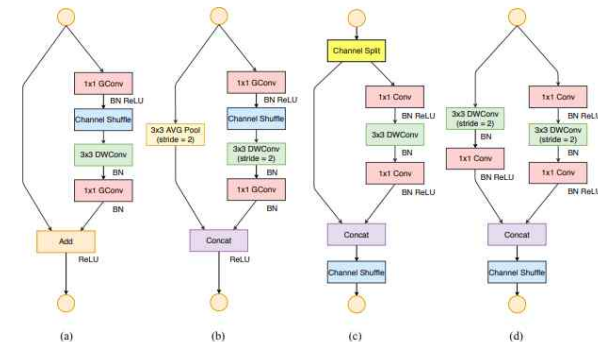


그림 21. ShuffleNet V1, ShuffleNet V2

(출처. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design)



그림 21에서 (a), (b)가 ShuffleNet의 기본 구조를 보여준다. MoblieNet에 ResNet의 잔차 연결(residual connection)을 추가하고, 일반적인 1x1 합성곱 대신 그룹 합성곱(grouped convolution)을 사용해 계산량을 줄였다. 또한, 입력 데이터를 섞어주는 채널 셔플링(channel shuffling)도 도입했다.

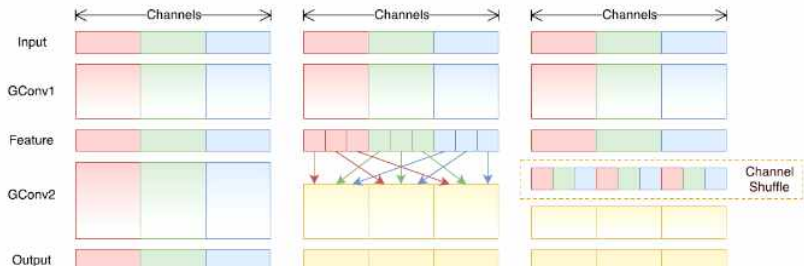


그림 22. 채널 셔플링 과정

(출처. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile)

그림 22은 채널 셔플링 과정을 보여준다. 입력 데이터가 들어오면 각각의 채널(빨강, 초록, 파랑 등)이 그룹으로 나뉘지고, 이후 이 그룹들 내부의 데이터가 섞인다. 이때, 단순히 무작위로 섞는 것이 아니라 각 그룹에 모든 종류의 데이터가 고르게 분포하도록 섞는다.

다음으로 ShuffleNet V2를 살펴보면 (c)에서 볼 수 있듯이, 입력 데이터가 들어오면 '채널 분할(channel split)' 연산을 통해 특징 맵이 절반으로 나뉜다. 이렇게 나뉜 특징 맵은 좌우로 전달되며 각각 1x1 합성곱 연산을 수행한다.

각 부분에서 독립적으로 합성곱 연산이 수행되었으므로 전체 MAC(Memory Access Cost)를 줄일 수 있다.

나뉜 특징 맵들은 Concat연산을 통해 병합해 입력과 출력 채널을 같아지도록 하였다. 그리고 (d)는 stride를 2로 설정하여 특징 맵의 크기를 절반으로 줄인다. 이후 Concat연산을 수행하여 크기가 줄어든 특징 맵들을 병합한다. 이때, (c)와 다르게 채널 분할은 수행되지 않으므로 출력층의 채널 수는 입력 때보다 두 배가 된다. 이런 방법을 통해 기존 버전보다 향상된 성능과 효율성을 가진다.

#### 4) Grad-CAM을 활용한 AI 모델 해석

- Grad-CAM (Gradient-weighted Class Activation Mapping) 기술은 딥 러닝 모델의 해석 가능성을 향상시키는 방법이다. 인공지능망(ANN, Artificial Neural Networks)은 딥 러닝의 성장과 함께 광범위한 분야에서 다양한 문제 해결에 활용되고 있다. 그러나 복잡한 모델들로 인해 이들 네트워크의 내부 동작 메커니즘은 블랙박스과 같이 불투명한 상태로 남아 있어, 예측 결과에 대한 해석이 어려운 상황이다. 이에 대한 해답으로, Grad-CAM 기술은 딥 러닝

모델의 해석을 도와주며 모델이 어떤 특징에 주목하여 결정을 내린 것인지를 시각화할 수 있다. Grad-CAM 알고리즘의 세부 동작 과정은 아래와 같다.

- (1) 모델의 마지막 합성곱 층에서의 특성 맵을 얻는다.
- (2) 이 특성 맵에 대하여, 예측한 클래스에 대한 각 특성 맵의 기울기(gradient)를 구한다.
- (3) 기울기 값을 각 특성 맵에 가중치로 적용하여 가중 특성 맵(weighted feature map)을 생성한다.
- (4) 만들어진 가중 특성 맵을 더하여 최종 Grad-CAM 히트맵을 얻는다.

- Grad-CAM 시각화 결과를 통해 CNN 기반 이미지 분류 모델의 적절성을 판단하였다. 다음은 분류 모델을 Grad-CAM 시각화 기술을 통해 AI 모델이 집중하고 있는 이미지의 영역을 히트맵으로 작성한 결과이다. AI 모델이 반려견의 목줄에 집중하고 있으며, 목줄 미착용 반려견의 경우 반려견 주변 영역의 히트맵이 활성화되었음을 확인할 수 있다.



그림 23. 목줄 미착용



그림 24. 목줄 착용

Ground-Truth	추론 결과	목줄 착용 확률	목줄 미착용 확률
목줄 미착용(상)	목줄 미착용	0.0117	0.9883
목줄 착용(하)	목줄 착용	0.9966	0.0035

## 5) 모델 경량화 - Quantization(양자화)

- 양자화는 모델의 가중치(weight)를 더 작은 비트로 표현해, 모델의 크기를 줄이고, 계산 효율성을 높이는 기법이다. 양자화에는 가중치 양자화(Weight Quantization)와, 활성화 양자화(Activation Quantization) 두 가지 유형이 있는데, 가중치 양자화는 32bit 소수형(float) 자료형을 8bit의 정수(int)형으로 전환하는 방법이고, 활성화 양자화는 계산된 활성화(activation) 값을 양자화하는 방법이다. 특히 본 프로젝트에서 라즈베리파이를 사용하기 때문에 작동 성능이 제한되어 있어 양자화 처리는 매우 중요하다. 특히, 본 프로젝트에서는 가중치 양자화를 통해 정확도는 거의 같게 유지하면서, 모델의 크기와 메모리 사용량을 감소시켰다. Pytorch에서 제공되는 양자화된 모델을 불러와서 학습을 진행하고, 다시 양자화하는 과정을 거쳐서 진행하였다.

### - 결과분석(양자화 전, 후 비교)

양자화를 사용하면 사용하기 이전보다 FPS가 상승하는 것을 확인할 수 있다.

모델명	양자화 전 FPS	양자화 후 FPS
shufflenet_v2_x0.5	82.94	85.48
shufflenet_v2_x2_0	43.34	49.61
shufflenet_v2_x1_5	54.9	59.96
shufflenet_v2_x1_0	55.5	62.89
mobilenet_v2	39.43	44.21
mobilenet_v3	45.34	46.27

(분류 모델만 비교했으며, 환경은 AMD 3700x CPU를 사용하였다.)

## 6) OpenCV(Open Computer Vision)

- 기존의 OpenCV는 오픈소스화된 컴퓨터 비전처리 라이브러리로, 컴퓨터로 영상처리를 하게 된다면, 가장 많이 사용하는 라이브러리로 실시간 처리에 중점을 두고 설계되어 빠른 속도와 효율성을 자랑하여 이를 사용하게 되었다. 이진화, 노이즈 제거, 외곽선 검출, 패턴인식, 기계학습 등 다양한 기능 있지만, 본 프로젝트에서는 라즈베리파이의 카메라를 학습된 AI 모델에 영상 데이터를 입력하기 위해 사용된다.

## 7) SSH(Secure Shell)통신

- 원격 호스트 컴퓨터에 접속하기 위해 사용되는 인터넷 프로토콜로, 기존의 텔넷이 암호화가 이루어지지 않아 암호화 기능이 추가된 프로토콜이다. 본 프로젝트에서는 노트북에서 라즈베리파이와의 연결과, 노트북과 AWS 서버

컴퓨터에 접속하기 위해 사용하였다. 특히, AWS 원격 서버와 연결은 Private key를 이용해서 접속하였다.

## 8) Flask (Micro Web Frameworks)

- Flask는 Python으로 웹 애플리케이션과 API를 구축하기 위한 웹 프레임워크이다.
- API Server를 가볍게 구현할 수 있으며, 코드의 길이가 짧아 가독성이 높다.
- 간결함, 유연성 및 최소주의적인 디자인이며, 웹 애플리케이션을 빠르고 효율적으로 만들 수 있다.

## 9) MySQL

- 오픈 소스 관계형 데이터베이스 관리 시스템(RDBMS)으로, 데이터 저장, 검색 및 관리를 위한 웹 애플리케이션 및 소프트웨어 시스템이다.
- 확장성, 안전성 및 높은 성능을 제공하는 데이터베이스 시스템이다.
- 본 프로젝트에서는 MySQL을 Dbeaver tool을 이용하여 DataBase를 구축하였고, 쿼리문을 작성하여 DB에 있는 정보를 가공하여 API로 전달한다.

#### 4. 프로젝트 개발 환경

구분		상세내용
S/W 개발환경	OS	Windows, Mac
	개발환경(IDE)	Google Colab : GPU 기반 AI 모델 학습 RoboFlow : AI 이미지 데이터 관리 툴 VScode : 웹 개발 DBeaver : DataBase 관리
	개발도구	Pytorch : 객체(반려견) 탐지 및 추적 모델 훈련 Django : 웹페이지 제작 Flask: API 프레임워크.
	개발언어	Python : AI 모델 학습 및 백엔드 구현 JavaScript : 반응형 웹 구현 Html/CSS : 웹 페이지 디자인 구성. MySQL : DataBase 구축.
	운영환경 배포	AWS : 가상 머신 서버 이용. DNS : 도메인 주소를 IP주소로 변환.
	기타사항	Figma : 웹페이지 디자인 협업 툴
H/W 구성장비	디바이스	Raspberry-pi : 하드웨어 장치(카메라&스피커) 구동 및 반려견 목줄 탐지 서비스 구현을 위한 디바이스 스피커 모듈 : 반려견 목줄 미착용 시 안내멘트 송출
	센서	카메라 센서 : 반려견 객체 탐지를 위한 영상 입력 장치
	통신	SSH 통신 : 라즈베리파이와 직접 연결할 모니터의 부재로 인해 개인 노트북을 이용한 구동을 위해 사용
	언어	Python : SW 파이프라인 구축 및 작동
	기타사항	Linux: 라즈베리파이 개발환경
프로젝트 관리환경	형상관리	Notion : 회의록 작성 및 전체 프로젝트 관리 GitLab : 프로젝트 저장소 관리
	의사소통관리	Slack : 멘티-멘토 소통 및 공지사항 전달 카카오톡 : 멘티들간 소통 및 프로젝트 진행도 파악 Zoom : 온라인 미팅 및 공동 작업
	기타사항	해당사항 없음

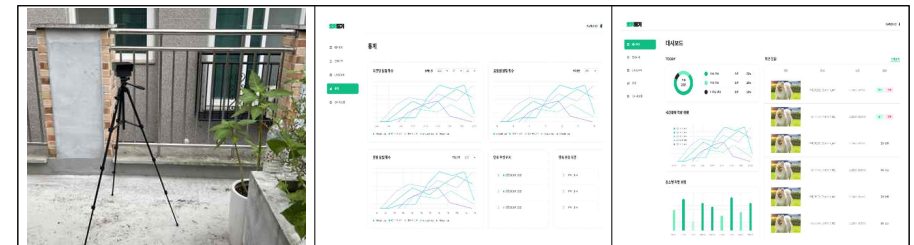
#### 5. 장비(기자재/재료) 활용

번호	품명	작품에서의 주요기능
1	라즈베리파이	- 카메라 및 스피커 구동을 위한 디바이스, AI 모델 추론을 위해 사용.
2	PAM8403 (스피커모듈)	- 목줄 미착용 반려견 탐지 시 경고 멘트 출력.
3	RPI NOIR CAMERA BOARD (카메라 모듈)	- 목줄 미착용 반려견 탐지를 위한 영상 데이터 입력.

#### 6. 프로젝트 작동 동영상

- <https://www.youtube.com/watch?v=aDSMdX6uOjQ>

#### 7. 결과물 상세 이미지



## 8. 달성 성과

<input type="checkbox"/> 논문게재 및 포스터발표	게재(발표)자명	논문(포스터)명	게재(발표)처	게재(발표)일자
				2023. 00. 00.
<input type="checkbox"/> 앱(APP) 등록	등록자명	앱(APP)명	등록처	등록일자
				2023. 00. 00.
<input type="checkbox"/> 프로그램 등록	등록자명	프로그램명	등록처	등록일자
				2023. 00. 00.
<input type="checkbox"/> 특허/실용신안 출원	출원자명	특허/실용신안명	출원번호	출원일자
				2023. 00. 00.
<input type="checkbox"/> 기술이전	기술이전기업명	기술명	금액	이전일자
				2023. 00. 00.
<input checked="" type="checkbox"/> 공모전	구분(교내/대외)	공모전명	수상여부(출품/수상)	상격
	대외	2023 한이음 ICT멘토링 공모전	수상	장려상
<input type="checkbox"/> 실용화				
<input type="checkbox"/> 기타				

## III. 프로젝트 수행 내용

### 1. 업무분장

번호	성명	역할	담당업무
1	이호진	멘 토	멘토링
2		지도교수	
3	조세은	팀 장	- 주제선정, 팀 통솔, MySQL 이용 django와 API로 연결(백엔드)
4	강지원	팀 원2	- 백엔드/웹 설계/서버구축/파이프라인 설계 및 구축
5	노지예	팀 원3	- 데이터 수집 및 전처리/AI 모델 학습 및 결과 분석/프론트엔드/서비스 알고리즘 설계 및 구현
6	오승범	팀 원4	- 데이터 수집 및 전처리/프론트엔드/AI 모델 학습 및 결과 분석/백엔드 구현/ 하드웨어 3D 도면 설계 및 출력

## 2. 프로젝트 수행일정

구분	추진내용	수행일정									
		3월	4월	5월	6월	7월	8월	9월	10월	11월	
계획	프로젝트 일정 수립										
분석	시스템 기술 분석										
설계	소프트웨어 개발 설계										
	하드웨어 개발 설계										
개발	AI 모델링 개발										
	웹 페이지 개발										
	하드웨어 개발										
테스트	인공지능 웹 하드웨어 테스트										
종료	서비스 런칭										
온오프라인 미팅	온오프라인 미팅										

## 3. 프로젝트 추진 과정에서의 문제점 및 해결방안

### 1) 프로젝트 관리 측면

- 프로젝트의 시작은 4월부터였지만, 중간 모두 개개인의 대학교 시험으로 미팅날짜와 시간을 조정하여 회의 및 프로젝트를 유연하게 진행했다. 시험 기간에는 시험에 집중하고 시험이 끝나는 순간부터 모두 집중적으로 프로젝트에 집중하여 기존에 세웠던 일정을 수정하지 않고 원하는 목표대로 진행할 수 있었다.

### 2) 프로젝트 개발 측면

- 학습 데이터셋 구축  
목줄 탐지 여부를 학습하기 위해 AI hub의 데이터를 이용하였으나, 목줄 착용 여부에 대해 분류가 되어 있지 않아, 간단한 GUI를 통해서 빠르게 분류하여 학습용 데이터셋을 구축할 수 있었다.
- 이미지 데이터 전처리  
목줄 미착용 반려견 이미지 데이터의 상당수가 실내에서 촬영된 이미지로 구성됨. 초기 목줄 착용 여부 분류 모델이 반려견의 목줄이 아닌 배경에 따라 목줄 착용 여부를 추론했음. 배경의 영향을 최소화하기 위해 반려견 경계 상자

주위를 crop하여 분류 모델 전용 데이터셋을 새로 구축하였음. 전처리된 데이터로 분류 모델을 학습시킨 결과 배경과 무관하게 목줄 미착용 반려견을 분류할 수 있었다.

#### - 데이터셋의 편향성 문제

초기 데이터셋을 crop하는 전처리 과정을 추가했음에도 편향성 이슈가 발생했다. 옷을 입은 반려견은 대부분 목줄을 착용했기에, 옷을 입은 목줄 미착용 반려견의 정확한 분류가 불가능했다. Grad-CAM 기술을 통해 CNN 모델이 반려견의 목줄이 아니라 몸통 부분에 집중함을 알게 되어, 옷을 입었으나 목줄을 착용하지 않은 데이터를 추가하여 훈련한 cls 모델은 반려견의 목 주위에 집중하여 추론하는 결과를 확인할 수 있었다.

#### - 객체 추적 과정의 필요성 논의

초기 계획 수립 내용에는 포함되지 않았던 객체 추적 알고리즘의 필요성이 대두되어, 객체 탐지 모델은 입력되는 영상을 프레임 단위로 분할하여 객체 탐지를 수행했다. 같은 반려견임에도 불구하고 객체 탐지 모델은 프레임마다의 객체가 동일 객체라고 판단하지 못해, 객체 추적 알고리즘이 필요하다는 결론을 내리고 이를 AI 모델에 추가하였다.

#### - 라즈베리파이 모니터 연결

라즈베리파이를 직접 연결할 모니터가 없어 mobaxterm을 이용한 SSH 통신을 통해 노트북에서 명령어를 이용해서 라즈베리파이를 사용하는 방법으로 해결할 수 있었다.

## 4. 프로젝트를 통해 배우거나 느낀 점

### 1) 협업 및 커뮤니케이션 능력

프로젝트를 진행하면서 협업 및 커뮤니케이션 능력을 향상시키는 데 큰 도움이 되었다. 팀원들과 원활한 소통을 위해 주기적인 회의를 통해 진행 상황을 공유하고 정보를 공유하였으며, 각 팀원의 의견을 존중하고 잘 들어주며 문제 발생 시 서로 돕고 협력하여 빠르게 해결할 수 있었다. 온라인 환경에서의 협업이 주를 이루었기 때문에, 비대면 상황에서도 효과적으로 팀워크를 발휘하기 위해 다양한 커뮤니케이션 도구를 활용하였다. 예를 들어, Slack과 Zoom을 이용하여 적절한 차원에서 소통하고, Notion과 같은 협업 플랫폼을 활용하여 워크플로우를 개선하였다. 이런 과정을 거치며, 다양한 배경과 경험을 가진 팀원들과 함께 일하는 중요성을 몸소 체험하였고, 서로를 배려하고 존중하는 것이 원활한 업무 진행에 핵심이라는 사실을 깨달았다. 더불어 상대방의 의견을 경청하고 이해할 줄 아는 유연한 사고 방식 수립에 도움이 되었다. 앞으로의 프로젝트에서도 이러한 협업 및 커뮤니케이션 역량을 계속 발전시켜, 팀 내에서 더 나은 성과를 이루도록 힘쓸 것이다.

### 2) 문제해결 능력

프로젝트를 진행하는 과정에서 여러 문제에 직면했으나, 이러한 문제들을 해결하기 위한 전략을 세우고 팀원들과 협력하여 성공적으로 극복할 수 있었다. 이를 통해 문제 해결 능력을 증진시킬 수 있었으며, 다음과 같은 방식으로 발휘하였다.

- 빠른 정보 수집 및 공유: 문제가 발생했을 때, 필요한 정보를 신속하게 수집하고 팀원들과 공유하여 문제에 대한 해결책을 빠르게 찾아내고 이해할 수 있었다.
- 다양한 시각을 고려한 아이디어 제시: 문제 해결에 접근할 때, 팀원들의 다양한 의견과 아이디어를 수렴하여 다각도에서 문제를 바라봐 최적의 해결책을 도출할 수 있었다.
- 문제의 원인 분석: 문제가 발생한 원인을 분석하여 같은 문제가 반복되지 않도록 예방 대책을 세웠다.
- 유연한 대처와 실행: 문제 해결을 위해 빠르게 대처하고 실행하는 능력을 발휘하여 변화하는 환경에 유연하게 적응하여 최선의 결과를 이끌어냈다.

이러한 노력을 통해 프로젝트가 원활하게 진행되도록 도와주었으며, 앞으로도 작업 중 발생할 수 있는 문제들을 끊임없이 나아가는 데 능력을 발전시킬 수 있을 것이다.

## IV. 기대효과 및 활용분야

### 1. 프로젝트의 기대효과

	기존	프로젝트 기대 효과
신고	신고 방법 및 절차 복잡함	자동 신고 가능
단속 시스템	비효율적 단속 시스템	미착용 반려견 빈번 출몰 구역 관리 용이 효율적 단속 인력 배치
인식	목줄 착용 의무에 대한 인식 부족	인식 개선
개물림 사고	목줄 미착용으로 인한 개물림 사고가 줄지 않음	목줄 미착용 반려견에게 즉각적인 착용 권고를 통해 사고 예방

### 2. 프로젝트의 활용분야

#### 1) 공공장소

- 반려견의 목줄 착용 문화 형성: 간단한 자동 신고를 통해 목줄 미착용 반려견을 단속하여 견주들이 자발적으로 반려견의 목줄을 착용시키는 문화 조성을 기대할 수 있다.
- 비반려인의 안정성 확보: 지금까지의 비반려인들의 인식이 목줄 착용이 되고 있지 않아 반려견들이 위험하다고 생각하는 사람들이 많지만, 반려견의 목줄 착용 문화가 잘 형성된다면, 비반려인들도 안정하게 공공장소를 이용할 수 있을 것이다.
- 관리 인원 효율적 배치: 기존에는 공공장소를 이용하는 사람들이 직접 신고 후 출동하는 방법이었으나, 자동 신고 시스템을 통해 인식하고 알람이 송출된다. 관리인이 이를 확인하여 조치를 취하는 방법으로 기존에는 관리 인력 배치에 비효율적이었으나 이를 효율적으로 배치하여 관리가 가능해진다.
- 사고 예방 효과: 반려인이 반려견의 목줄을 착용하지 않을 경우 관리인이 이를 바로 확인하고 조치를 취해 목줄을 착용할 수 있도록 하여, 이를 통해 반려견의 목줄 미착용으로 인한 개 물림 사고가 감소할 것이다.

## V. 참고자료

### 1. 참고 및 인용자료

- 김혜선, “진돗개 견주, 목줄 풀린 핏볼과 혈투 겨우 살려냈다”, 이데일리, 2023.07.04.,  
<https://www.edaily.co.kr/news/read?newsId=01308726635671240&mediaCodeNo=257>
- 지수현, “목줄 안 한 개, 신고하려면 주소까지 알아야”, 단비뉴스, 2023.05.10.,  
<http://www.danbinews.com/news/articleView.html?idxno=23363>
- Yifu Zhang, “ByteTrack: Multi-Object Tracking by Associating Every Detection Box”
- Xiangyu Zhang, “ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices”



