



Check Mate (체크메이트)

사용 설명서

제작자

Leemited (이은서, 이중권, 정현경, 한혜윤)

작성일자

2024.06.01

배포일자

2024.06.03

목차

1. [프로그램 설명](#)
2. [프로그램 사용 방법](#)
 - 2.1. [메인 화면](#)
 - 2.2. [데이터 메뉴](#)
 - 2.2.1. [데이터 추가](#)
 - 2.2.2. [데이터 라벨링](#)
 - 2.2.3. [데이터셋 생성](#)
 - 2.3. [학습 메뉴](#)
 - 2.3.1. [YOLO 학습](#)
 - 2.3.2. [EfficientAD 학습](#)
 - 2.4. [불량 검출 메뉴](#)
 - 2.4.1. [YOLO 이미지 검출](#)
 - 2.4.2. [YOLO 실시간 영상 검출](#)
 - 2.4.3. [EfficientAD 검출](#)
3. [참고 사항](#)

1. 프로그램 설명

"체크메이트"는 사용자가 직접 제품 이미지를 추가하고 라벨링 하여 모델을 학습시킬 수 있는 **UI**를 제공합니다. 이를 통해 사용자는 모델 학습 전 과정을 관리할 수 있습니다. 또한 훈련된 모델을 통해 이미지와 실시간 영상을 분석하여 불량 여부를 화면에 표시합니다. 특히 이 프로그램은 데스크톱 앱으로, 사용자는 개인용 **PC**를 통해 프로그램을 실행할 수 있습니다.

해당 프로그램은 체크메이트로 2024학년도 1학기 명지대학교 컴퓨터공학과 캡스톤디자인(0791) 수업을 통해 제작하게 되었습니다. 제작자는 명지대학교 컴퓨터공학과에 재학 중인 4인(이은서, 이종권, 정현경, 한혜윤)으로, **Leemited**라는 팀명으로 약 4개월간 활동하였습니다.

딥러닝을 활용한 불량품 검출 프로그램을 제작하는 본 프로젝트는 딥러닝 모델을 활용해 사용자가 인식할 제품을 직접 학습시킨 후 실시간 이미지 인식을 통해 정상 제품과 불량 제품을 구별하고 제품 누락 등을 표시하는 프로그램을 만드는 것을 목표로 두었습니다. 설문조사를 통해 불량 상품을 발견하는 데 어려움을 겪은 판매자와 이에 따라 피해를 본 소비자의 사례를 수집하였습니다. 이를 토대로 사업 규모가 크고 신뢰성 있는 결함 확인 시스템을 갖춘 판매자보다는 제품을 직접 생산하고 판매하는 과정에 참여해 결함 확인이 미흡한 소규모 사업자를 대상으로 선정하였습니다. 특히, 결함 확인 과정에서는 별도의 고가 기계를 사용하지 않고도 **PC**, 노트북 등 사용자가 일반적으로 휴대할 수 있는 기계에서 프로그램을 실행할 수 있게 함으로써 높은 접근성으로 상용화를 기대할 수 있습니다.

이번 프로젝트는 팀원들의 열정과 헌신, 그리고 지도 교수님의 귀중한 조언으로 진행되었습니다. 조장으로서 일정 조율, 팀 간 커뮤니케이션을 담당하고 이상 탐지 부분 담당 이은서 님, **yolo** 객체 탐지와 데이터 셋을 만드는 부분 담당 이종권 님, 사용자 인터페이스 디자인과 사용자 경험 개선 담당 정현경 님, 데이터 라벨링 부분과 전체적인 프로젝트 완성 담당 한혜윤 님, 각 팀원의 헌신적인 노력과 협업 덕분에 프로젝트가 잘 마무리되었습니다. 특히 프로젝트 진행 중 여러 문제를 함께 해결하며 끊임없이 노력해 준 팀원들에게 감사의 마음을 전합니다. 또한, 프로젝트를 총괄 지도해주신 김상귀 교수님께 깊은 감사를 드립니다. 교수님의 귀중한 조언과 지도 덕분에 프로젝트를 마무리할 수 있었습니다. 모든 분께 진심으로 감사드리며, 앞으로도 함께 더 큰 도전을 이뤄나가길 기대합니다.

감사합니다.

※프로그램 이용을 시작하기에 앞서※

제공된 폰트는 프로그램에 사용된 폰트입니다. 설치하지 않을 경우 기본 폰트를 제공합니다.

사용된 폰트

- [Noto Sans Korean - Google Fonts](#)

NVIDIA GPU(그래픽 카드)가 장착되어 있는 기기를 사용하여 본 프로그램을 이용하실 경우, GPU를 사용한 빠른 학습과 검출을 진행할 수 있도록 **CUDA**를 설치하는 것을 권장합니다. 또한 사용하는 GPU에 따라 호환되는 버전이 다르므로 더 정확한 설치 방법 및 내용은 공식 문서에서 확인하시길 바랍니다.

호환되는 **CUDA** 버전 체크

- cmd 창에 `nvidia-smi` 입력

CUDA 버전에 맞는 **Compute capability** 확인

- [CUDA - Wikipedia](#)

CUDA Toolkit 설치

- [CUDA Toolkit Archive | NVIDIA Developer](#)

cuDNN 설치 및 해당 폴더 **CUDA Toolkit** 폴더에 덮어쓰기 (**bin, include, lib**)

- [cuDNN Archive | NVIDIA Developer](#)

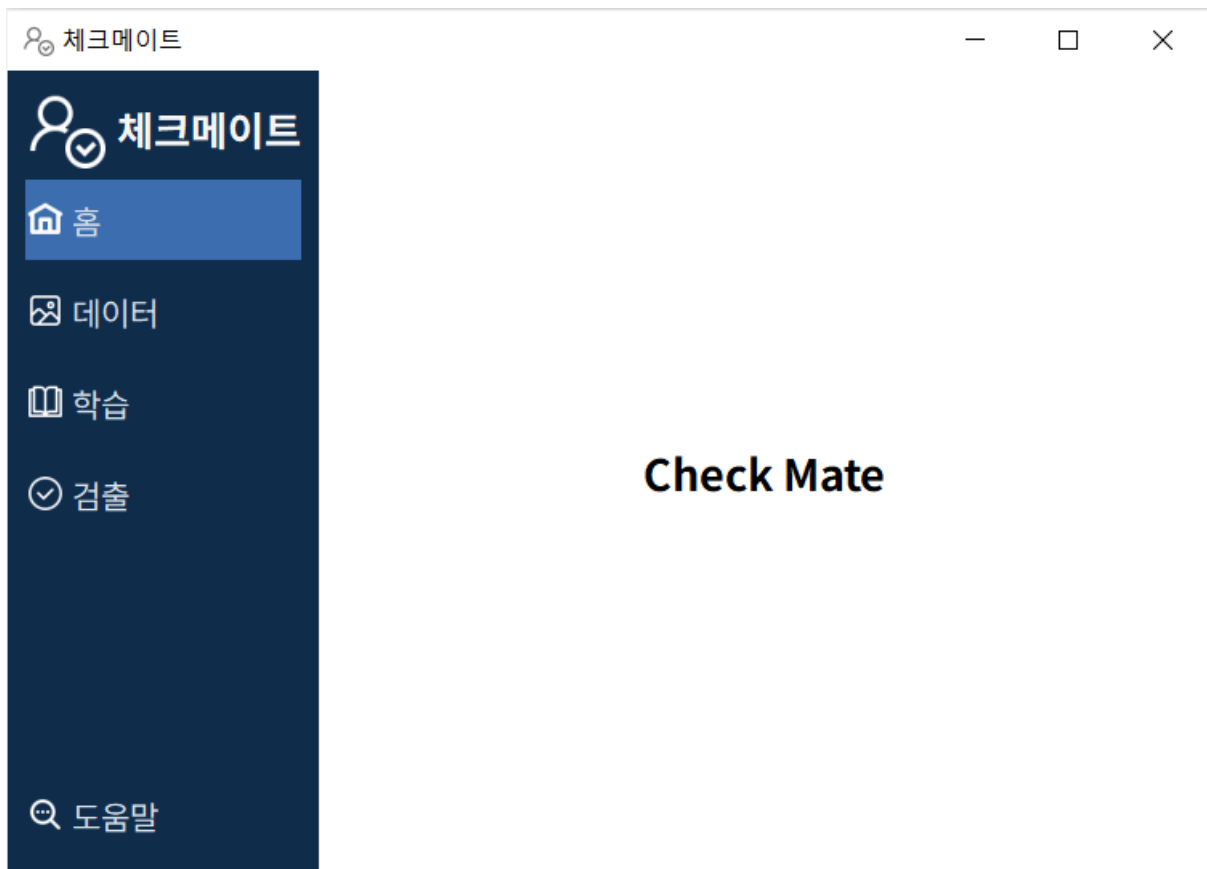
PATH 변수에 **CUDA**의 폴더가 추가되었는지 확인

설치한 **CUDA**와 호환되는 **Pytorch install**

- [Previous PyTorch Versions | PyTorch](#)

2. 프로그램 사용 방법

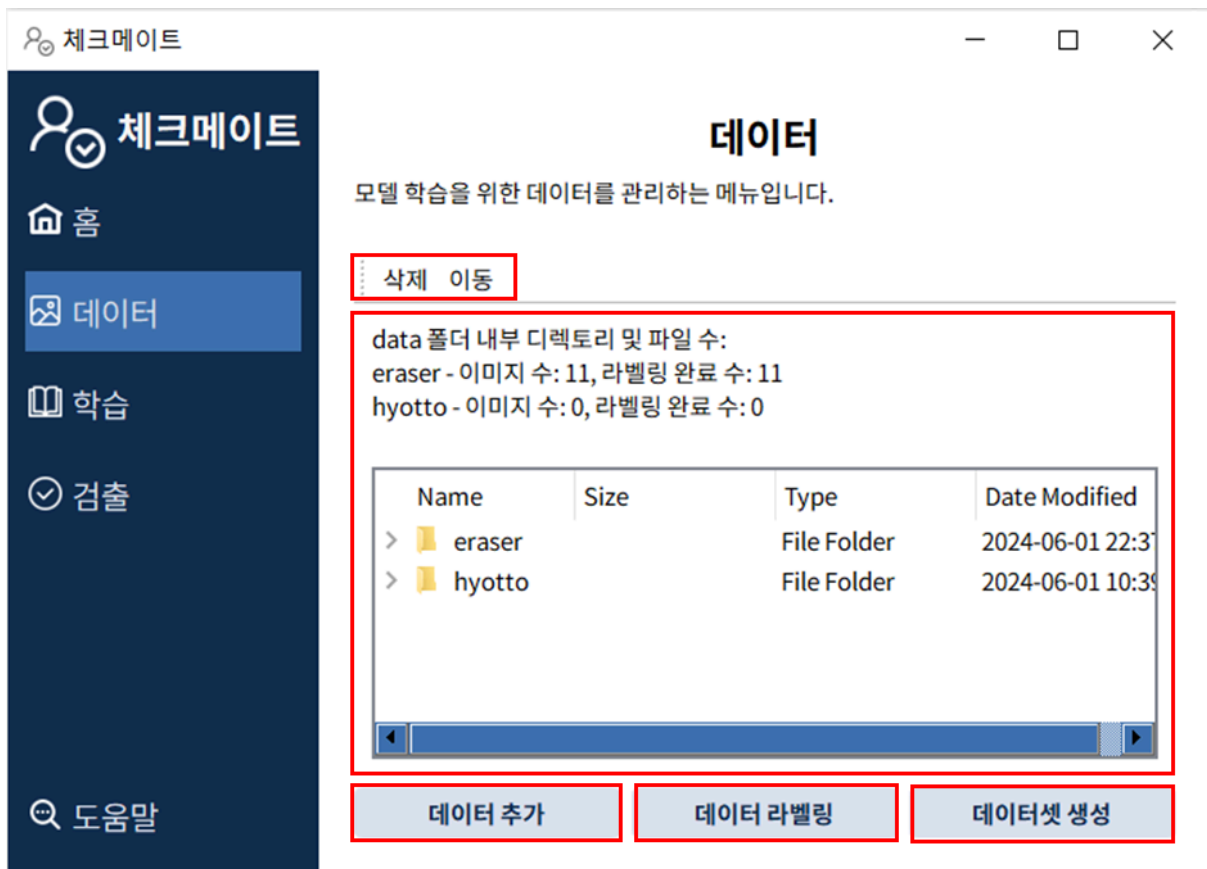
2.1. 메인 화면



전체적인 프로그램 이용 순서는 다음과 같습니다.

1. 데이터 페이지에서 물품 이미지 데이터를 추가합니다.
2. 데이터 페이지에서 추가된 이미지 라벨링 및 모델 학습 데이터셋을 생성합니다.
3. 학습 페이지에서 YOLO, EfficientAD 모델 학습을 진행합니다.
4. 검출 페이지에서 YOLO, EfficientAD를 활용해 불량을 탐지합니다.

2.2. 데이터 메뉴



체크메이트

데이터

모델 학습을 위한 데이터를 관리하는 메뉴입니다.

삭제 이동

data 폴더 내부 디렉토리 및 파일 수:
eraser - 이미지 수: 11, 라벨링 완료 수: 11
hyotto - 이미지 수: 0, 라벨링 완료 수: 0

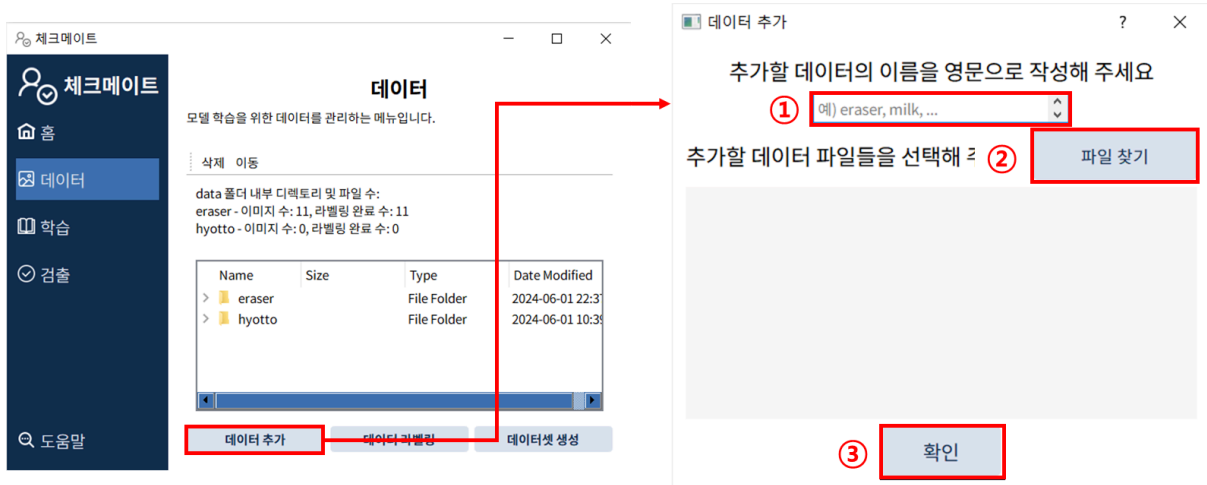
Name	Size	Type	Date Modified
> eraser		File Folder	2024-06-01 22:31
> hyotto		File Folder	2024-06-01 10:39

데이터 추가 데이터 라벨링 데이터셋 생성

데이터 페이지에서는 다음과 같은 작업을 수행할 수 있습니다.

- 삭제/이동 버튼: 폴더 삭제 및 이동
- 파일 관리 박스: 물품의 이미지가 저장된 폴더와 폴더 내의 이미지, 라벨링 된 이미지 확인
- 데이터 추가 버튼: 물품의 사진이 저장된 폴더를 열어 이미지 등록
- 데이터 라벨링 버튼: 데이터셋 생성을 위한 데이터 라벨링
- 데이터셋 생성 버튼: 라벨링이 완료된 폴더를 지정하여 학습을 위한 데이터셋 생성

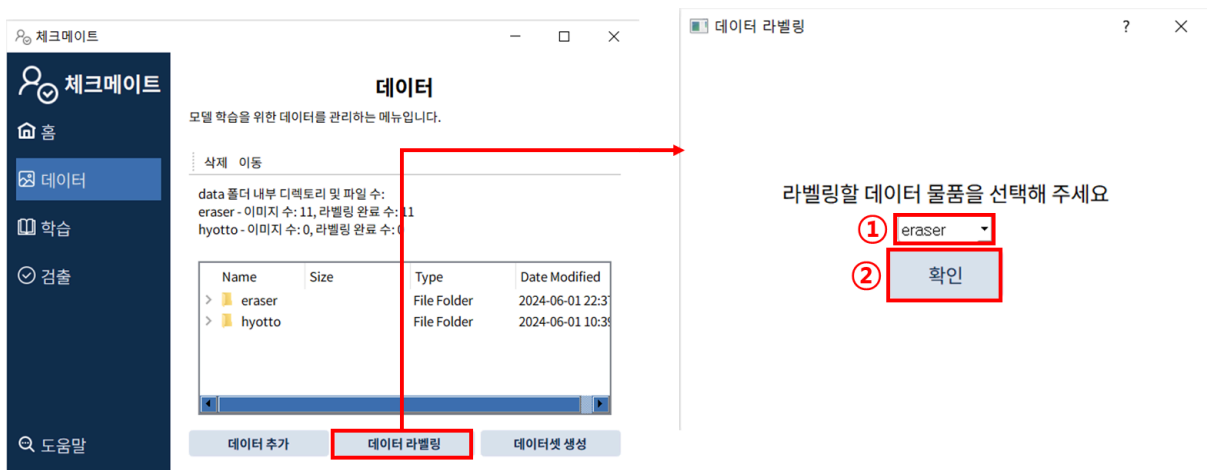
2.2.1. 데이터 추가



데이터 추가는 다음과 같은 순서로 진행합니다.

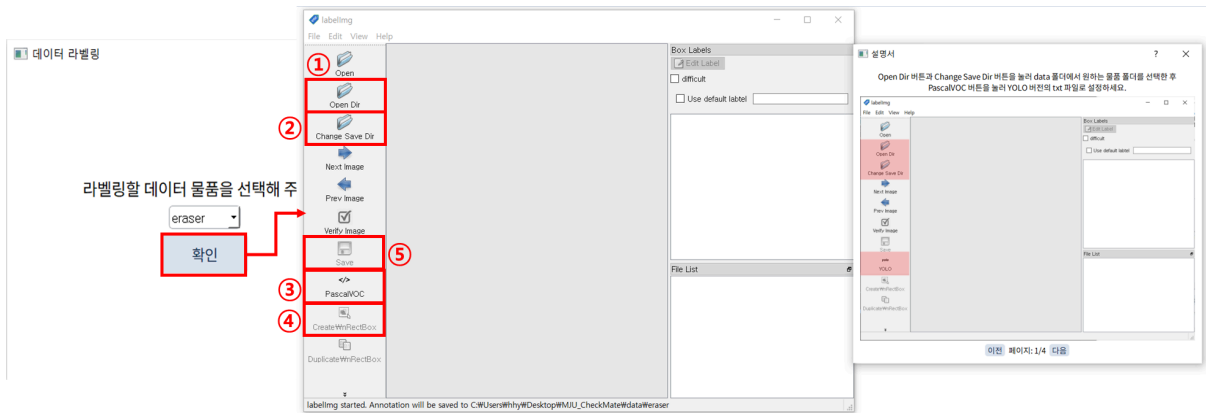
1. 추가할 데이터의 이름을 영문으로 작성합니다.
2. 파일 찾기 버튼을 통해 물품의 이미지를 불러옵니다.
3. 확인 버튼을 눌러 프로그램 내의 **data** 폴더에 추가합니다.

2.2.2. 데이터 라벨링



데이터 라벨링 창에서는 다음과 같은 작업을 수행할 수 있습니다.

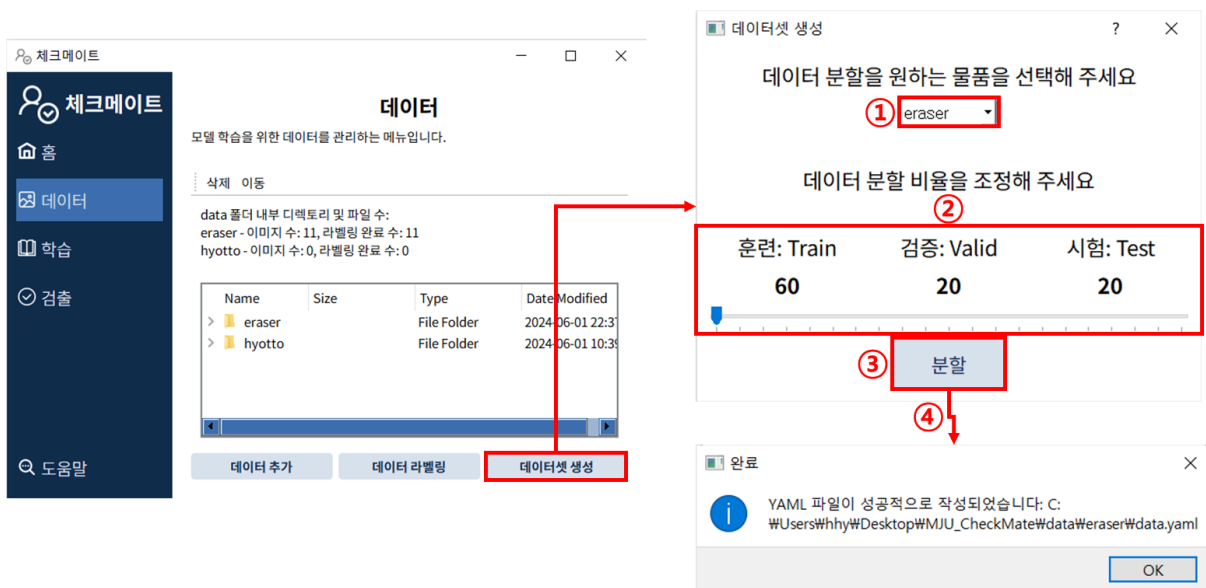
1. 라벨링 할 물품을 선택합니다.
2. 확인 버튼을 누르면 라벨링 툴 창이 생성됩니다.



라벨링 툴 창에서는 다음과 같은 순서로 진행합니다.

1. Open Dir 버튼을 눌러 **data** 폴더에서 원하는 물품 이미지 폴더를 선택합니다.
2. Change Save Dir 버튼을 눌러 **data** 폴더에서 원하는 물품 이미지 폴더를 선택합니다.
3. PascalVOC 버튼을 눌러 YOLO 버전의 **txt** 파일로 설정합니다.
4. Create Rect Box 혹은 w 버튼으로 객체의 영역을 지정합니다.
이때 최대한 물품만 크롭합니다.
5. bad/good를 선택한 후 Save 버튼이나 **Ctrl+s**로 저장합니다.
6. Next Image 버튼이나 Prev Image 버튼을 이용하여 모든 이미지에 4번과 5번 과정을 반복합니다.
7. 모든 이미지를 라벨링 한 후 프로그램을 닫고 라벨링이 모두 완료되었는지 파일 관리 박스의 폴더를 눌러 확인합니다.

2.2.3. 데이터셋 생성

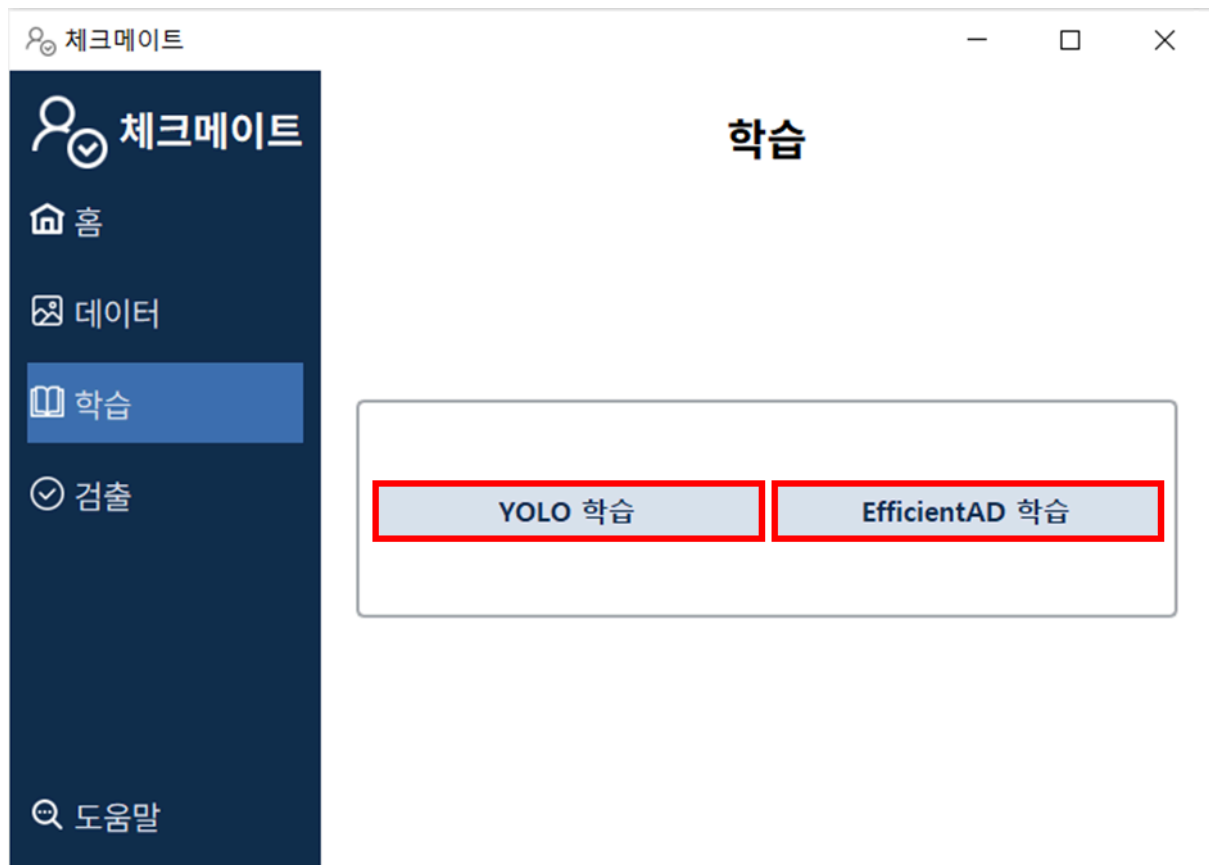


데이터셋 생성 창에서는 다음과 같은 순서로 진행합니다.

1. 분할하기를 원하는 물품을 선택합니다.
2. 데이터 분할 비율을 조정합니다.
3. 분할 버튼을 눌러 데이터셋 생성을 완료합니다.
4. 성공적으로 데이터셋이 생성되었다면 **YAML** 파일이 생성됩니다.
5. **EfficientAD**를 위한 데이터셋도 생성됩니다.

주의※ 선택한 이미지를 모두 라벨링 하지 않거나 **bad**와 **good**에 이미지가 존재하지 않으면 분할이 진행되지 않습니다.

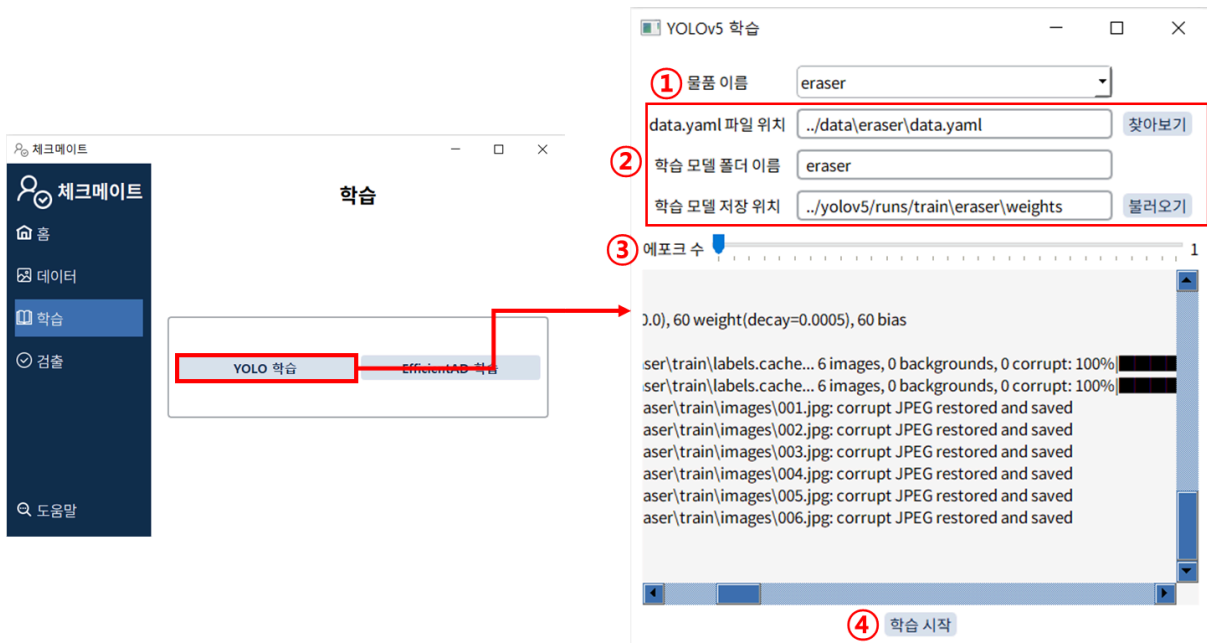
2.3. 학습 메뉴



학습 페이지에서는 다음과 같은 작업을 수행할 수 있습니다.

- YOLO 학습 버튼: YOLO 학습을 진행
- EfficientAD 학습 버튼: EfficientAD 모델 학습을 진행

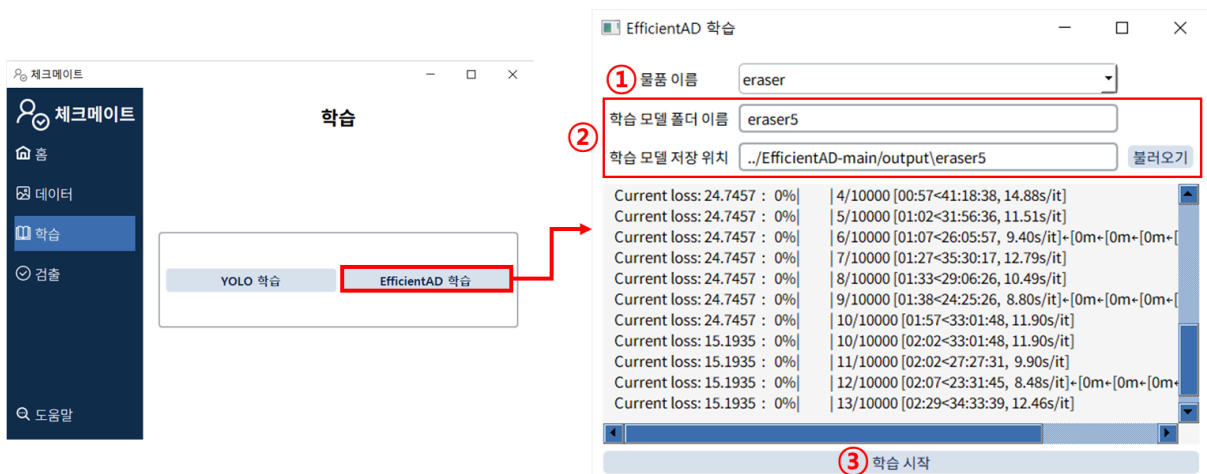
2.3.1. YOLO 학습



YOLO 학습 창에서는 다음과 같은 순서로 진행합니다.

1. 학습을 진행할 물품을 선택합니다.
2. data.yaml 파일 위치와 학습 모델 저장 위치는 자동으로 지정되지만, 불러오기 버튼을 통해 저장 위치를 변경할 수 있으며, 학습 모델 폴더 이름을 직접 변경하면 자동으로 학습 모델 저장 위치도 변경됩니다.
3. 에포크 수를 조절합니다. 에포크 수를 늘릴수록 학습에 많은 시간이 소요됩니다.
4. 학습 시작 버튼을 눌러 학습을 진행합니다.
5. “학습이 완료되었습니다.” 창이 뜨면 학습을 종료합니다.

2.3.2. EfficientAD 학습



EfficientAD 학습 창에서는 다음과 같은 순서로 진행합니다.

1. 학습을 진행할 물품을 선택합니다.
2. 학습 모델 저장 위치는 자동으로 지정되지만, 불러오기 버튼을 통해 저장 위치를 변경할 수 있으며, 학습 모델 폴더 이름을 직접 변경하면 자동으로 학습 모델 저장 위치도 변경됩니다.
3. 학습 시작 버튼을 눌러 학습을 진행합니다.
4. “학습이 완료되었습니다.” 창이 뜨면 학습을 종료합니다.

주의※ 이미 학습이 완료된 물품을 다시 학습하면 파일이 이미 존재한다는 문구와 함께 덮어쓸 것인지 묻는 경고 문구가 나옵니다. 덮어쓰는다면 다시 학습을 진행할 수 있습니다.

주의※ 학습 진행 중 닫기 버튼을 누르면 학습이 진행 중인데 종료할 것인지 묻는 경고 문구가 나옵니다. **No**를 선택할 경우 학습은 계속 진행되며, **Yes**를 선택하면 학습이 종료됩니다. 이때 지금까지 진행된 학습은 정상적으로 저장됩니다.

2.4. 불량 검출 메뉴

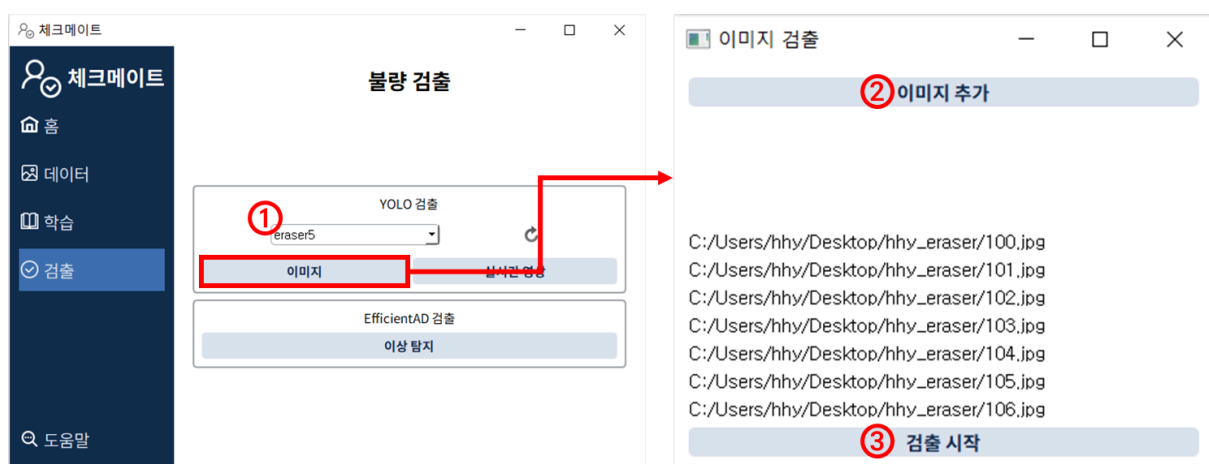


참고※ YOLO는 *EfficientAD*보다 탐지 결과 신뢰도가 낮기에 YOLO 분석 후 *EfficientAD* 이상 탐지를 진행하는 것을 권장합니다.

검출 페이지에서는 다음과 같은 작업을 수행할 수 있습니다.

- YOLO 검출 박스: 검출 진행 물품 선택, 이미지 또는 실시간 영상 중 선택
- EfficientAD 검출 박스: 이상 탐지 진행

2.4.1. YOLO 이미지 검출



주의※ 라벨링 된 불량 종류만 인식 가능합니다.

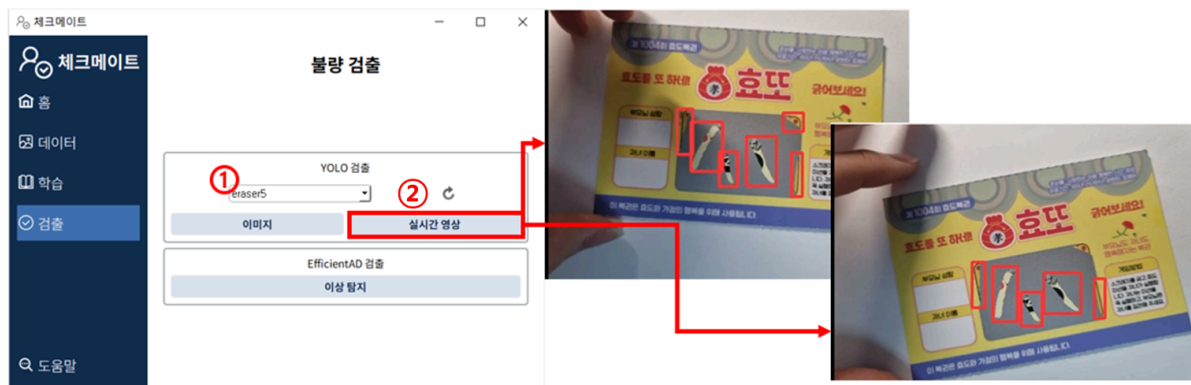
이미지로 YOLO 검출을 진행하는 창에서는 다음과 같은 순서로 진행합니다.

1. 불량 검출 모델을 선택합니다.
2. 이미지 추가 버튼을 눌러 원하는 이미지를 선택합니다.
3. 검출 시작 버튼을 눌러 검출을 진행합니다.
4. 검출 결과 창이 생성되며 어떤 물품의 이미지가 정상인지 불량인지 확인합니다.
5. 이상 감지 버튼을 누르면 *EfficientAD* 검출을 진행할 수 있는 창이 생성됩니다.

다음은 검출 결과 화면입니다.



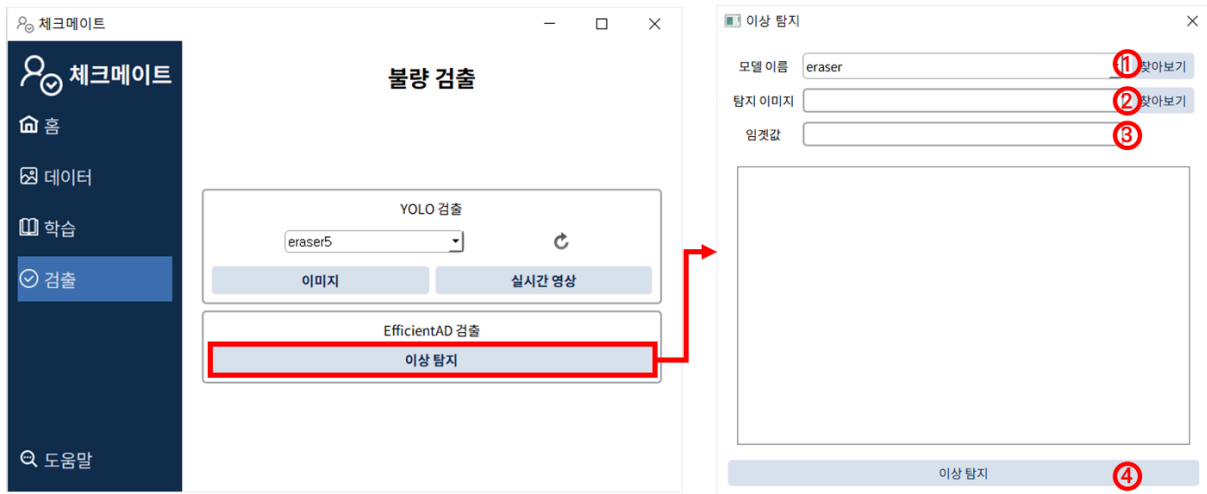
2.4.2. YOLO 실시간 영상 검출



실시간 영상으로 YOLO 검출을 진행하는 창에서는 다음과 같은 순서로 진행합니다.

1. 불량 검출 모델을 선택합니다.
2. 실시간 영상 버튼을 눌러 검출을 진행합니다.

2.4.3. EfficientAD 검출



EfficientAD 검출을 진행하는 창에서는 다음과 같은 순서로 진행합니다.

1. 모델 위치에서 찾아보기 버튼을 누르거나 콤보박스로 모델을 선택하여 불러옵니다.
2. 탐지 이미지에서 찾아보기 버튼을 눌러 하나의 이미지 혹은 한 폴더를 불러올 것인지 선택합니다.
3. 임계값을 설정합니다. (정상의 이미지를 여러 장 골라 임계값을 0으로 설정 후 검출을 진행해 도출된 가장 큰 값 등의 적절한 값을 채택합니다.)
4. 이상 탐지 버튼을 눌러 검출을 진행합니다.
5. 이상 탐지 결과 화면이 표시되고 이때 불량에 대해서는 **Anomaly Map**이 같이 나타나 불량으로 탐지된 부분을 확인할 수 있습니다.

다음은 검출 결과 화면입니다.



참고※ 불량 데이터가 없어도 정상과 다른 부분을 인식하여 **Map**(비정상적인 부분의 위치) 출력을 지원합니다.

3. 참고 사항

- 필요시 특정 하드웨어 사양을 요구할 수 있습니다. (GPU 등)
- 현재 시점에서는 실제 프로그램을 배포하여 소규모 사업자의 판매 전후 운영 실태에 대한 사례를 수집하지 않았기 때문에 안정적인 사업 운영에 대한 평가가 어렵습니다.
- 정상 제품과 불량 제품의 판별 기준이 훈련 데이터나 임곗값에 따라 크게 달라질 수 있어 객관적인 신뢰 판별 기준을 제공하지 않을 가능성이 존재합니다.
- 실시간 영상으로 불량을 탐지하는 기능은 높은 성능의 GPU와 고성능 카메라 사양을 요구하여 사용 환경에 제약이 있습니다.
- 프로그램의 안정성과 환경적 요건이 갖추어진다면 더욱 견고한 모델을 학습시킬 수 있어 지금보다 더 높은 신뢰도를 제공할 것으로 기대합니다.

* 2024.06.02 기준 해당 문서는 1.0.0 입니다.

이외에도 프로그램 관련 오류 제보 및 문의가 필요하신 분은 아래로 연락 부탁드립니다.
감사합니다.

Github 주소

<https://github.com/Meawcat/CheckMate.git>

Leemited

이은서: dldmstj2009@gmail.com

이중권: jk00122681@gmail.com

정현경: jhyunnk30@gmail.com

한혜윤: hhy33719@gmail.com