

샌드위치 AI 유스케이스

AI를 활용한 자동차 외관 부품 품질 검사 플랫폼

8조 샌드위치

컴퓨터공학과 18011881 김정민

컴퓨터공학과 19011579 서해영

컴퓨터공학과 19011554 하소영

경제통상학과 18012248 신민기

1. 시스템 개요

샌드위치 AI는 이전에 사람을 통해 행해지던 자동차 외관 검수 과정을 AI로 대체하기 위해 개발되었다. 이를 통해 외관 검수의 효율성을 증진시키고, 비용을 절감할 수 있을 것이다.

본 시스템은 플랫폼 기반 S/W이며 데이터는 사용자에게 의해 촬영된 자동차 외관 영상을 기반으로 한다. 주요 사용자는 자동차 검수 단계 임무를 수행하는 관리자이며, 본 시스템을 통해 외관 부품 불량 검출에 대한 업무 부담을 줄일 수 있다.

2. Use Case

가) Actor 목록

Actor	구분	설명
생산직 종사자	시스템사용자	영상 업로드, 결과값 관찰
애플리케이션 서버	시스템SW	영상 관리, 애플리케이션 시스템 서버
AI 서버	시스템 AI 서버	영상 처리, 차량 불량 판단 주체

그림1. Actor 목록

나) 차량 불량 검사 시스템

1) 개요

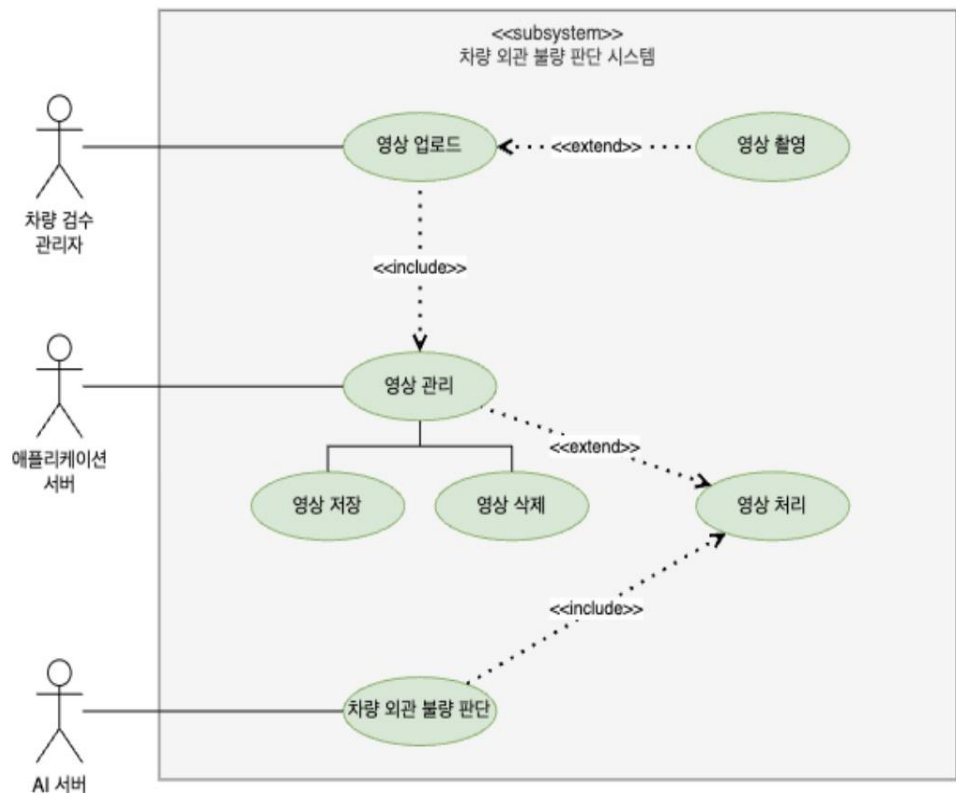


그림2. 유스케이스 다이어그램

사용자는 자신이 찍은 자동차 외관 영상을 업로드한다. 업로드를 하면 로딩이 되는 동안은 로딩창이 뜬다. 그 동안, 애플리케이션 서버로 영상이 보내지고, 이를 AI서버로 보내 외관 불량을 탐지한다. 이후 AI 서버는 자동차에 외관 불량을 시각적으로 표시한다. 표시된 영상을 애플리케이션 서버로 다시 보낸다. 애플리케이션 서버에서는 영상을 저장 및 관리한다. 저장된 영상은 프론트 엔드에 보내준다. 이를 바탕으로 유저는 결과값을 관찰한다.

2) 사건흐름

- 기본흐름

(ㄱ) (유저) 영상 업로드

- ① 차량 품번 입력
- ② 업로드 버튼 클릭
- ③ 영상 파일 선택
- ④ 파일 명이 뜨면 업로드 버튼 클릭

(ㄴ) (프론트 엔드) 영상 애플리케이션 서버로 전송

- ① HTTP 헤더 생성
- ② HTTP 바디에 영상 넣기
- ③ Request 전송
- ④ 전송 후 대기 시간 동안 로딩 화면 송출

(ㄷ) (애플리케이션 서버) 영상 형식 확인

- ① 파일 형식 및 영상 크기 확인
 - i. 잘못된 파일 형식인 경우 예외 생성
 - ii. 영상 크기가 너무 큰 경우 예외 생성
 - iii. 프론트 엔드에 예외 반환

(ㄹ) (애플리케이션 서버) AI 서버로 외관 검수 요청

- ① HTTP 헤더 생성
- ② HTTP 바디에 영상 넣기
- ③ Request 전송

(ㄴ) (AI 서버) 애플리케이션 서버로부터 영상 수신

(ㄷ) (AI 서버) 프레임 단위 이미지로 영상 분할

(ㄱ) (AI 서버) 한 이미지 당 불량 외관 검출

① 불량 부품을 영상 내에 시각적으로 표시

(ㅇ) (AI 서버) 불량 판단 영상 애플리케이션 서버에 전송

① HTTP 헤더 생성

② HTTP 바디에 추론 결과 넣기

③ Response 전송

(ㄲ) (애플리케이션 서버) 불량 판단 영상 스토리지에 저장

(ㄴ) (애플리케이션 서버) 저장된 영상 url 추출

(ㄷ) (애플리케이션 서버) url을 프론트 엔드에 전송

(ㄹ) (프론트 엔드) 유저에게 결과값 제공

(ㅍ) (유저) 결과값 관찰

- 대안흐름

(ㄱ) 영상의 파일 형식이 mp4가 아닐 경우

- 애플리케이션 서버에서 MediaTypeException 및 에러 메시지 반환
- HTTP STATUS CODE 415(UNSUPPORTED_MEDIA_TYPE)

(ㄴ) 영상 파일 크기가 50MB를 초과할 경우

- 애플리케이션 서버에서 FileSizeException 및 에러 메시지 반환
- HTTP STATUS CODE 414(REQUEST_ENTITY_TOO_LARGE)

```
java
public class ApiException extends Exception{

    public CollectionException(String message) {
        super(message);
    }

    public static String MediaTypeException(String fileName) {
        return fileName+"의 파일 형식이 맞지 않습니다.";
    }

    public static String FileSizeException(String fileName) {
        return fileName + "의 파일 크기 50MB를 초과합니다.";
    }

}
```

그림3. 에러 메시지 코드

- 사전조건

(ㄱ) 유저는 자동차 외관을 한바퀴 두르는 영상을 찍어 준비한다.

(ㄴ) 업로드 되는 영상은 자동차의 외관을 카메라로 찍은 mp4 형식의 영상이어야 한다.

(ㄷ) 영상의 크기는 50MB이하로 제한한다.

- 사후조건

(ㄱ) 입력된 영상에 결함이 존재한다면 결함과 관련된 정보를 차트와 리포트 형식으로 사용자에게 보여준다.

(ㄴ) 영상에 외관 불량 발견될 경우 차량 검수 관리자는 검수를 abort한다.

다) 관련 데이터

- 데이터 맵

use case 명	AI 서버로 영상 url 전송		
use case 설명	AI 서버로 영상 url을 전송한다.		
Data Name	자료형	크기	설명
carNumber	String	50B	자동차 품번
outputUrl	String	255B	불량 판단한 영상 url
inputVideo	multipart/form-data	50MB	유저가 업로드하는 영상
resultVideo	multipart/form-data	50MB	검증 결과 영상

그림4. 데이터 맵

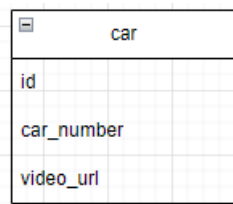


그림 5. ER 다이어그램

- 데이터 통신 구조

(ㄱ) HTTP 통신 구조를 사용

(ㄴ) Request 구성

- Method: POST
- Content-Type: multipart/form-data
- Body: 차량 품번, 업로드 영상 파일

User → Application Server, Application Server → AI Server

동일한 Request 사용

(ㄷ) AI Server → Application Server Response 구성

- Content-Type: multipart/form-data
- Body: 차량 품번, 불량 판단 영상 파일
- HTTP STATUS CODE 200(OK)

(ㄹ) Application Server → User Response 구성

- Content-Type: multipart/form-data
- Body: 차량 품번, 불량 판단 영상 URL
- HTTP STATUS CODE 200(OK)

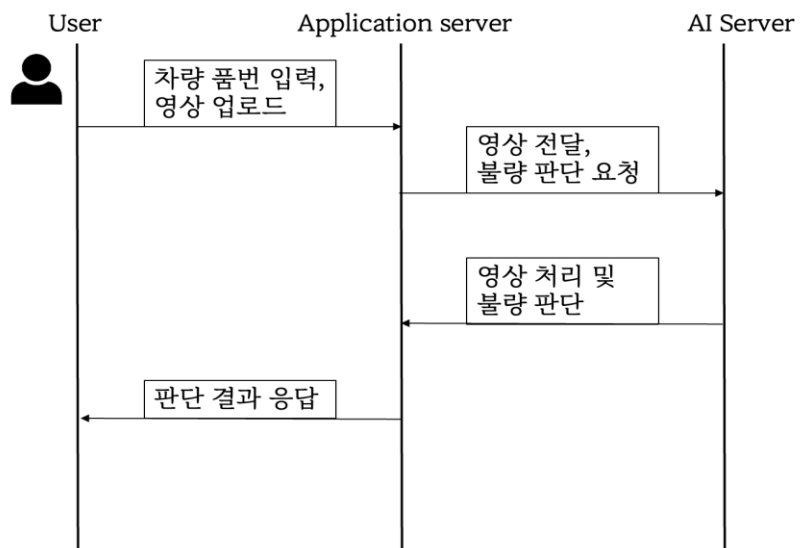


그림6. 데이터 통신 구조도

- 플랫폼 디자인 예시

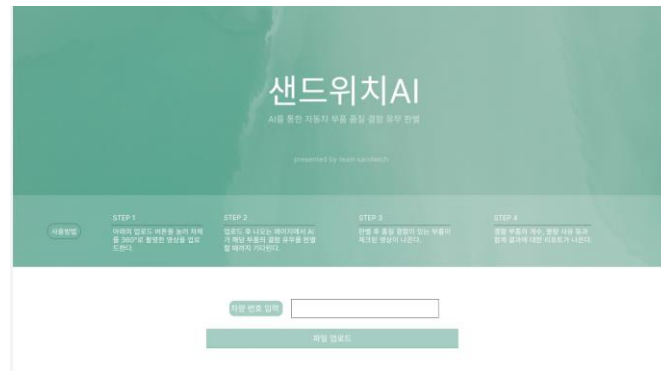


그림7. 부품 품질 검사 플랫폼 첫 화면

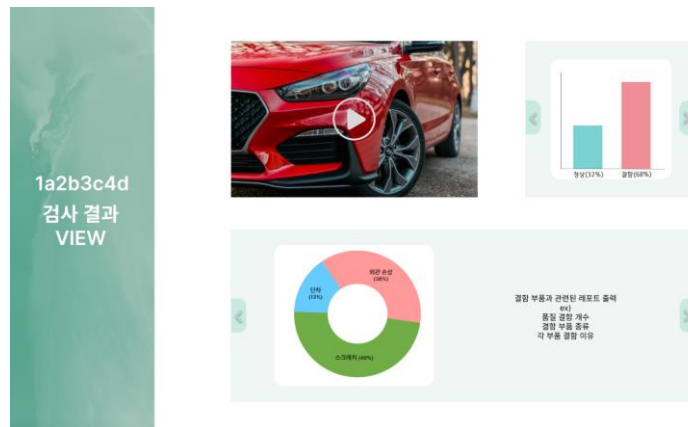


그림8. 부품 품질 검사 플랫폼 결과 화면(결함이 있을 때)

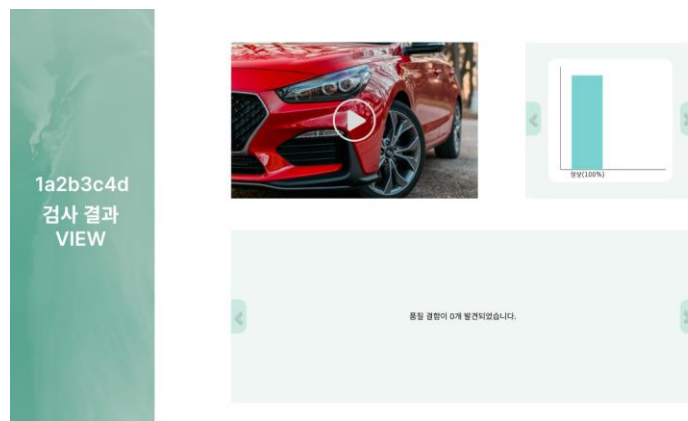


그림9. 부품 품질 검사 플랫폼 결과 화면(결함이 없을 때)