2020캡스톤디자인

팀 명 : 모자이크해조

발표자 : 송재우

순 서

1. 서비스 소개

YoMo ?

주요 기능

필 요 성

기대 효과

YoMo ?

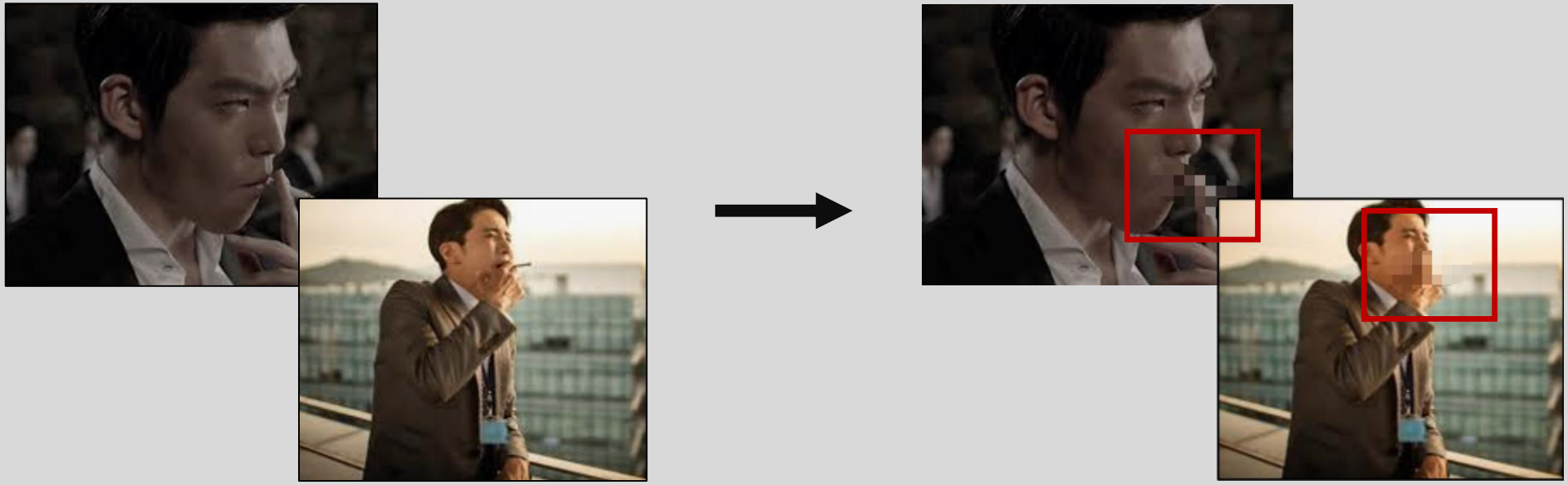
YoMo = You only Mosaic once

YOLO* 알고리즘을 활용한
유해물 및 개인정보 탐지 및 처리 서비스

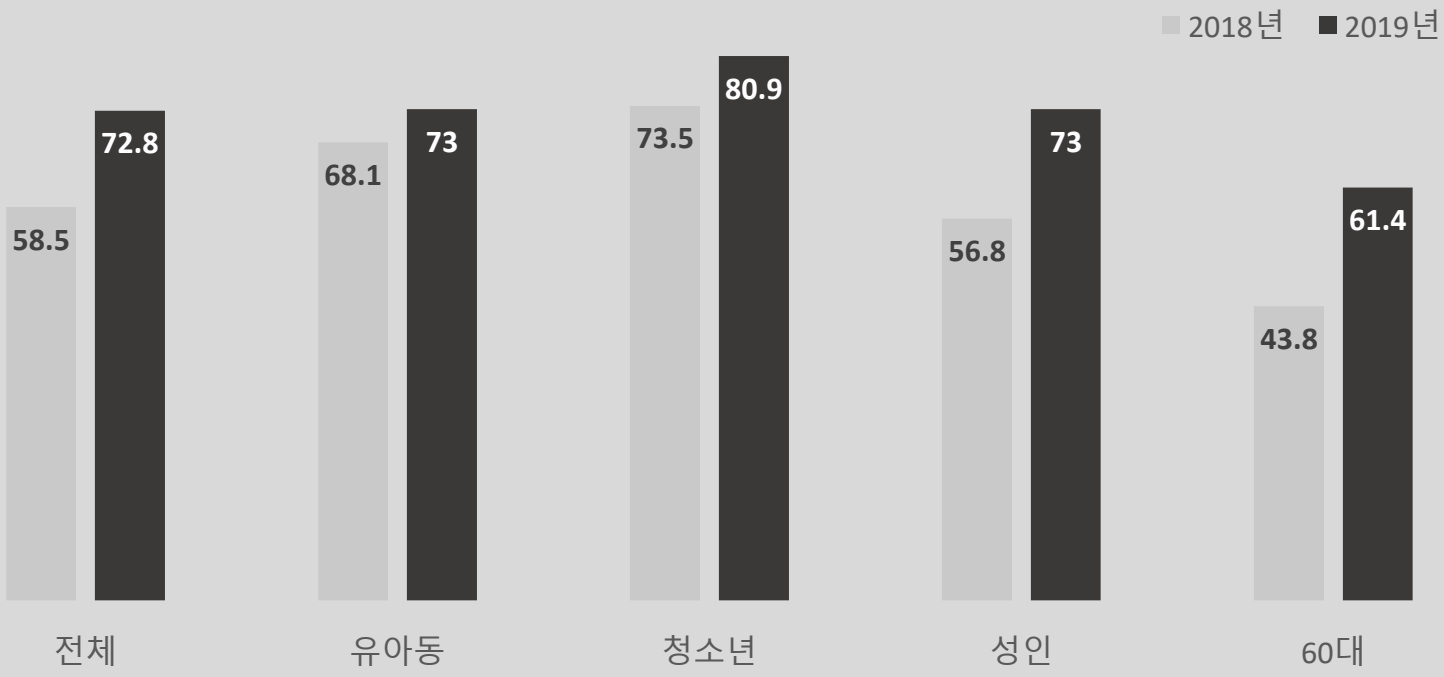
YOLO (You Only Look Once) : 이미지를 여러 그리드로 나누어 Boundary box 생성, 특징 추출,
클래스 분류를 동시에 진행하는 객체탐지 알고리즘으로, 구조가 비교적 간단하고 처리 속도가 매우 빠르다는 장점이 있다.

주요 기능

동영상 속 유해 이미지(칼, 담배 등) 및
개인정보 이미지(차 번호판, 도로명 주소판 등)를 탐지하여 모자이크 처리



필요성



< 1인 미디어 콘텐츠 이용률(%) >

(과학기술정보통신부, 한국정보화진흥원, 2020)

필 요 성

개인정보 보호 및 안전 센터 > 개인정보 보호 자료 > **신원 보호**

YouTube는 사용자의 안전을 중요하게 생각합니다. 그러므로 사이트의 동영상이나 댓글이 개인정보를 침해하거나 안전을 위협하는 경우가 있으면 알려주시기 바랍니다.

누군가가 허락 없이 내 개인정보를 게시했거나 내가 등장하는 동영상을 업로드한 경우 먼저 업로더에게 직접 연락하여 콘텐츠를 삭제해 달라고 요청하세요. 업로더와 합의점을 찾을 수 없거나 업로더에게 문의하기 불편한 상황이라면 YouTube 개인정보 보호 가이드라인에 따라 콘텐츠 삭제를 요청할 수 있습니다.

콘텐츠 삭제 기준

YouTube 개인정보 보호 가이드라인에서는 개인정보 침해 신고 절차에 대한 자세한 설명과 개인정보 침해 신고 평가 시 고려 요인에 대해 다룹니다.

개인을 고유하게 식별할 수 있어야 콘텐츠 삭제를 고려할 수 있습니다. 개인정보 침해 신고 절차를 사용하고 싶다면 진행에 앞서 신고할 콘텐츠에서 나를 고유하게 식별할 수 있는지 확인하세요. 주체의 고유 식별 여부를 평가할 때 고려되는 요인은 다음과 같습니다.

- 이미지 또는 음성
- 이름
- 금융 정보
- 연락처 정보
- 기타 개인 식별 정보

YouTube 정책 > **유해하거나 위험한 콘텐츠 관련 정책**

아래에 설명된 내용 중 어느 하나라도 해당한다면 콘텐츠를 YouTube에 게시하지 마세요.

- **매우 위험한 도전**: 위급한 신체적 상해의 위험이 있는 도전
- **위험하거나 위협적인 장난**: 피해자가 위급하고 심각한 신체적 위험에 처했다고 믿게 만들거나 미성년자에게 심각한 정신적 고통을 초래하는 장난
- **사람을 죽이거나 해치는 방법**: 다른 사람을 죽이거나 불구로 만들기 위한 행위를 하는 방법을 시청자에게 알려주는 내용. 다른 사람을 해치거나 죽이기 위한 폭발물을 만드는 방법을 알려주는 행위를 예로 들 수 있습니다.
- **중독성 마약 흡입 또는 제조**: 코카인 또는 오피오이드와 같은 중독성 마약을 남용하는 모습을 묘사하거나 제조하는 방법을 알려주는 콘텐츠. 대개 신체적 중독에 이를 수 있는 마약을 중독성 마약으로 정의합니다.
- **섭식 장애**: 거식증이나 기타 섭식 장애를 칭송하거나, 미화하거나, 따라하도록 시청자를 독려하는 콘텐츠. 섭식 장애는 음식이 아닌 물질을 먹는 것을 포함하여 건강에 부정적 영향을 주는 비정상적이거나 잘못된 식습관을 의미합니다.
- **폭력 사건**: 학교 총격 사건과 같은 폭력적인 참사를 조장하거나 미화하는 내용
- **절도 또는 속임수 방법 안내**: 시청자에게 실물 상품을 훔치는 방법을 알려주거나 부정 행위를 조장하는 콘텐츠
- **해킹**: 사용자 인증 정보를 훔치거나, 개인 정보를 손상시키거나, 소셜 미디어 계정을 해킹하는 등 다른 사람에게 심각한 피해를 발생시킬 의도로 컴퓨터나 정보 기술을 사용하는 방법을 알려주는 내용
- **디지털 콘텐츠 또는 서비스 결제 우회:** 일반적으로 요금 결제가 필요한 오디오 콘텐츠, 시청각 콘텐츠, 정식 버전의 비디오 게임, 소프트웨어 또는 스트리밍 서비스에 무료로 무단 액세스할 수 있는 웹사이트, 앱 또는 기타 정보 기술의 사용 방법을 시청자에게 알려주는 내용
- **유해한 약물 또는 치료법 홍보**: 유해한 약물 또는 치료법이 건강에 도움이 될 수 있다고 주장하는 콘텐츠

심각한 신체적 손상 또는 사망으로 이어질 수 있는 위험하거나 불법적인 행동을 독려하는 콘텐츠는 YouTube에서 허용되지 않습니다.

〈 유튜브의 개인정보 보호 자료 중 신원 보호 및 유해물 관련 규제 〉

(유튜브 고객센터 홈페이지, 2020)

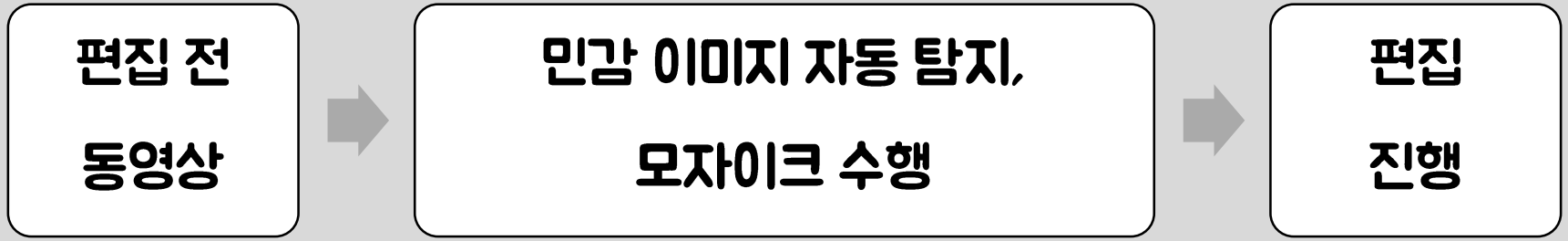
필요성

〈 동영상 속 유해 이미지 및 개인정보 이미지의 모자이크 처리 현황 예 〉

기대 효과

1. 불특정 다수의 미디어 이용자에게 유해 이미지 및 개인정보 이미지 유출을 사전에 차단할 수 있다.

2. 동영상 편집자에게 편집 전 하나의 전처리 기능으로써 반복 작업을 없애, 피로감을 낮춰줄 수 있다.



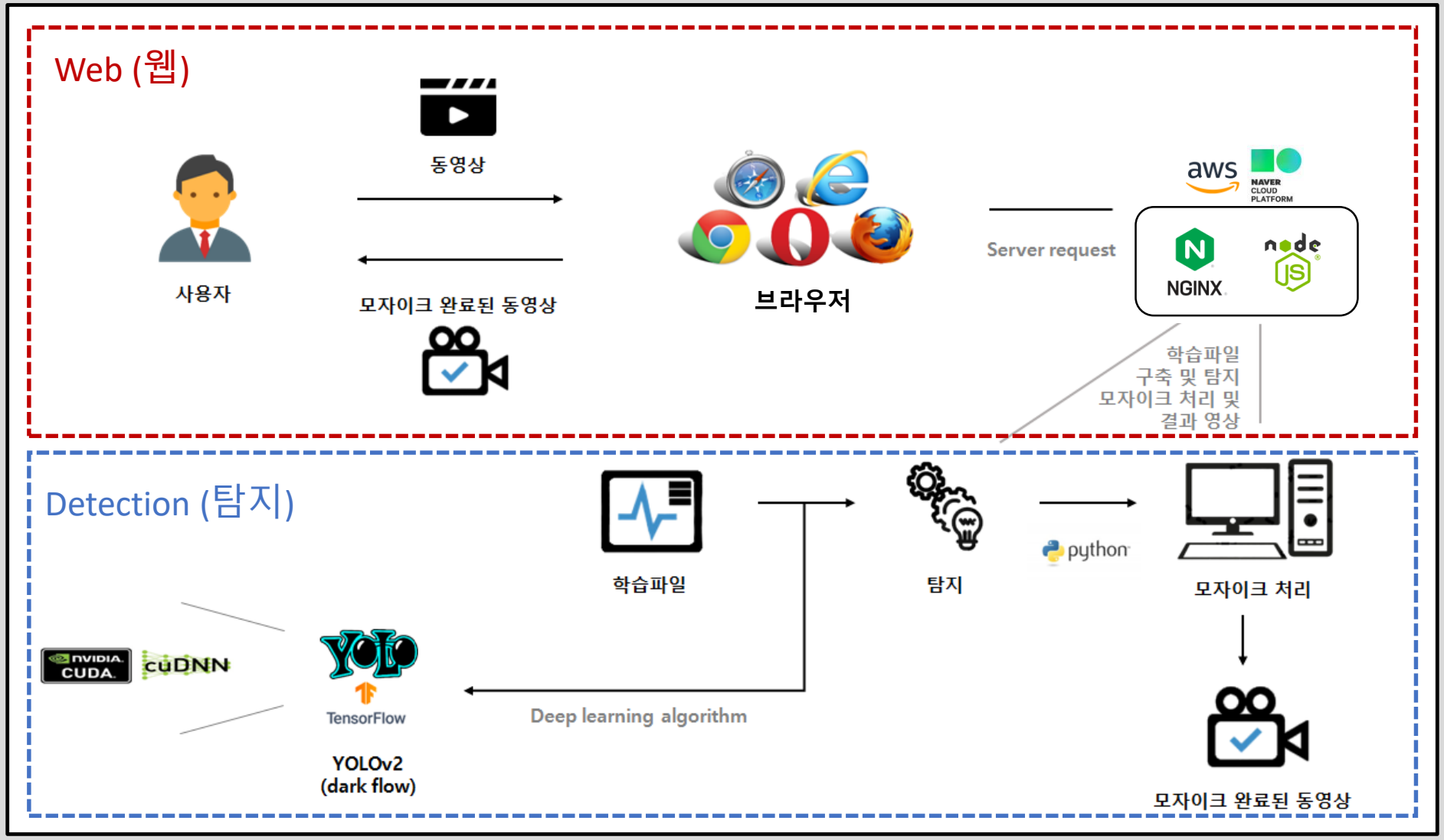
2. 시스템 구성 및 구현

시스템 구성도

시스템 구현

테스트 영상

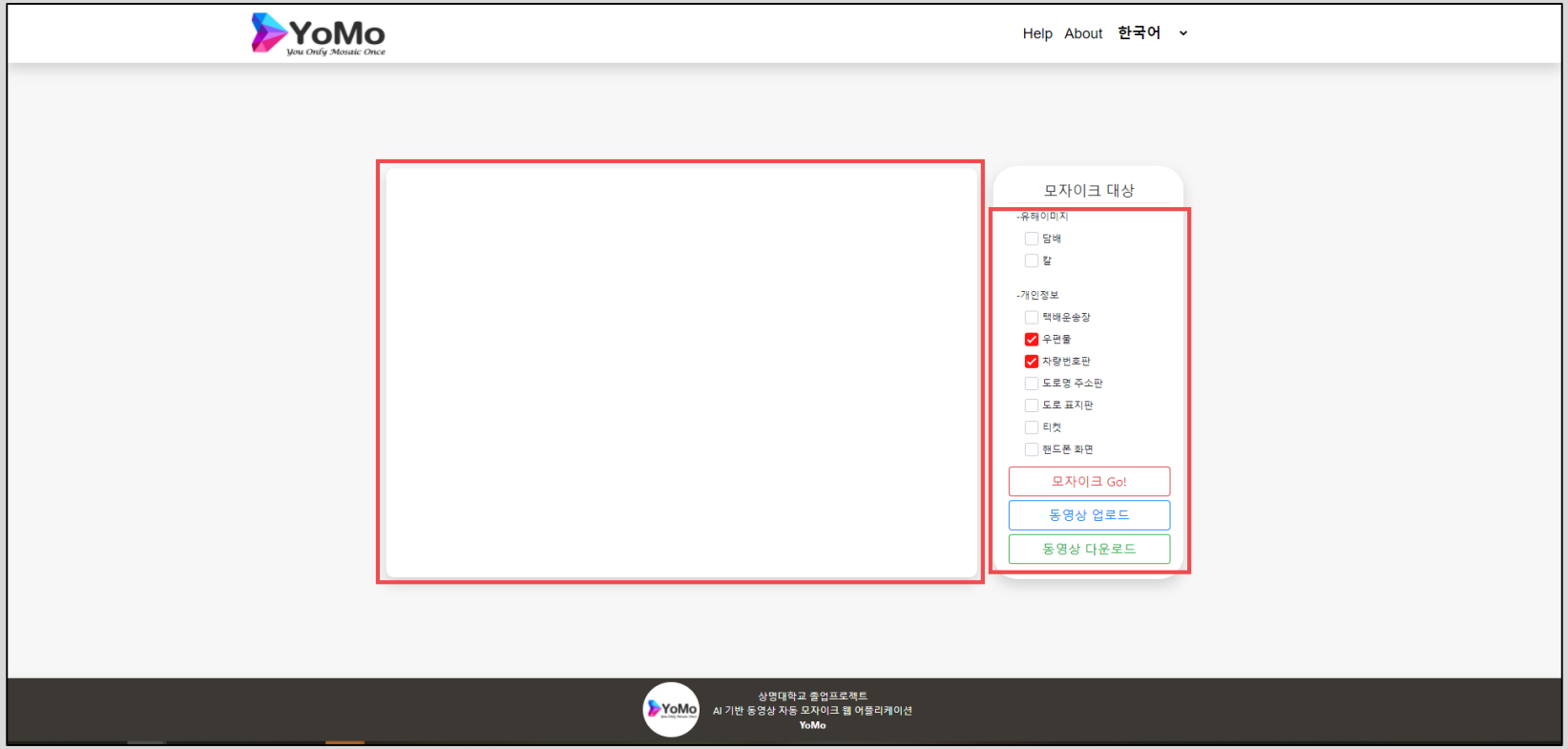
시스템 구성도


Web (웹)
동영상
사용자
모자이크 완료된 동영상
브라우저
Server request
aws
NAVER CLOUD PLATFORM
NGINX
node
학습파일
구축 및 탐지
모자이크 처리 및
결과 영상
Detection (탐지)
학습파일
탐지
python
모자이크 처리
NVIDIA CUDA
cuDNN
YOLO
TensorFlow
YOLOv2
(dark flow)
Deep learning algorithm
모자이크 완료된 동영상

시스템 구현 :: Web (웹)

1. YoMo 웹 어플리케이션

- 사용자가 별도의 프로그램을 설치하지 않아도 서비스를 이용할 수 있도록 웹 사이트를 제작



〈 YoMo 웹 페이지 전체 화면 〉

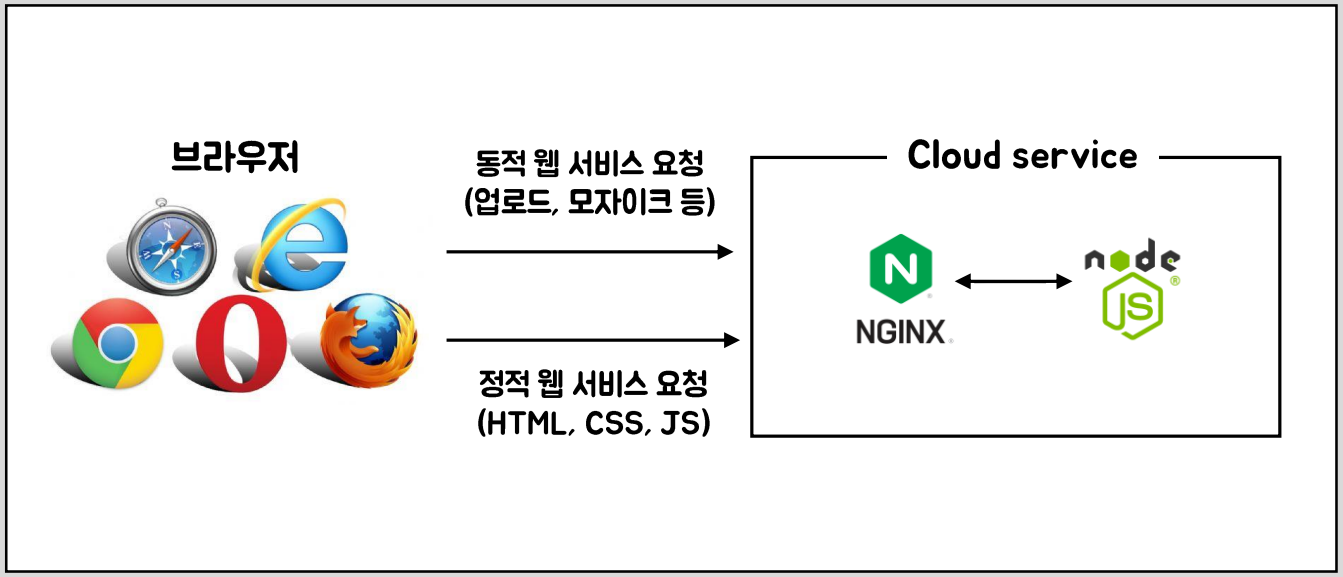
시스템 구현 :: Web (웹)

2. 구성 요소

- Cloud service
- Nginx (웹 서버)
- Node.js (자바스크립트 런타임)
- 브라우저

3. 사용 언어

- JavaScript
- CSS
- HTML



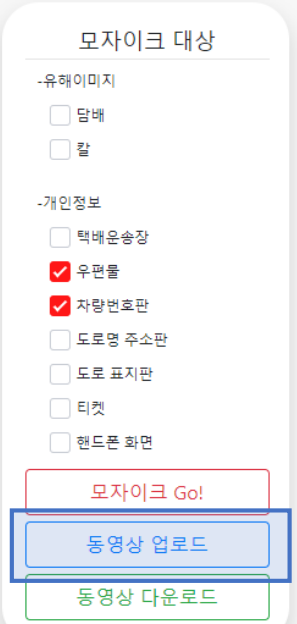
시스템 구현 :: Web (웹)

4. 동작

- 동영상 업로드

Front

3 응답으로 온 파일의 Path를 이용하여 업로드한 영상을 화면에 출력



1 요청
POST/upload
body
파일

2 응답
body
파일 Path

Server

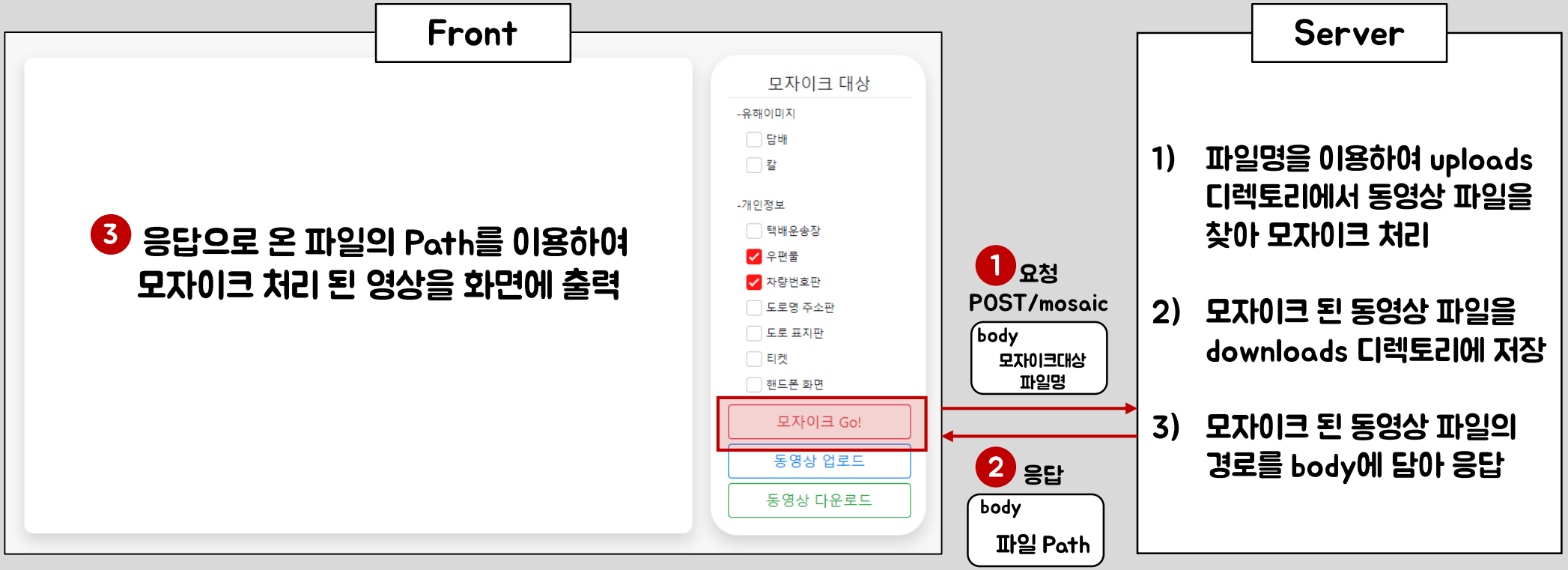
1) 파일명을 다음과 같이하여 Uploads 디렉토리에 파일을 저장
'filename YYYY.MM.dd HH:mm:ss'

2) 파일의 경로를 body에 담아 응답

시스템 구현 :: Web (웹)

4. 동작

- 모자이크 처리



시스템 구현 :: Web (웹)

4. 동작

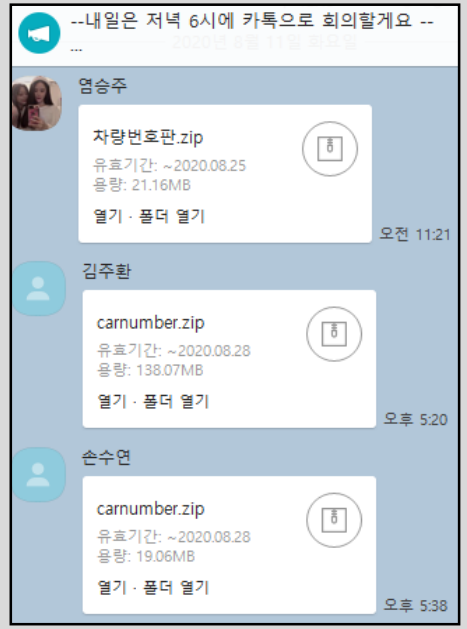
- 동영상 다운로드

시스템 구현 :: Detection (탐지)

1. 이미지 데이터 수집

- 기존 방식 : 웹 크롤링을 통한 '이미지 수집'만을 이용
  - 이미지의 중복, 낮은 품질의 이미지 등으로 인한 문제 발생
  - 때문에, 이미지 개수는 많지만 학습에 유용하지 않음

- 현재 방식 : 크롤링 이미지 수집 + 직접 이미지 촬영
  - 수집 이미지 중 중복, 낮은 품질의 이미지 제거



< 직접 수집한 이미지를 공유하는 채팅 화면 >

2. 이미지 데이터 변환

- 이미지 데이터를 효율적으로 늘리기 위한 과정
  - 대칭 이동
  - 밝기 조절

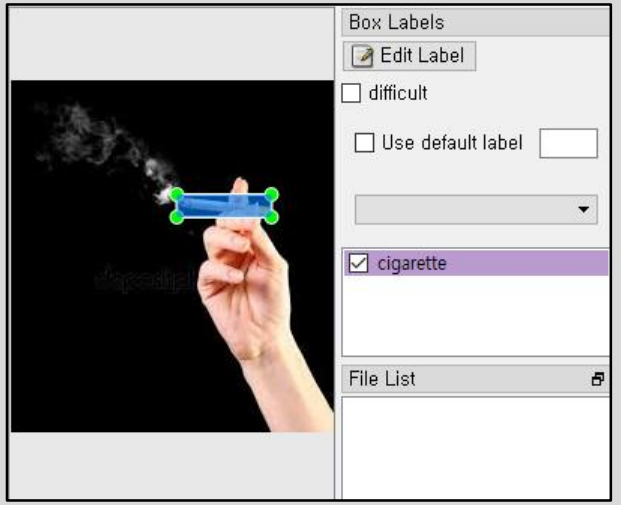
- 총 데이터 개수
  - 담배 : 4129장, 칼 : 2396장
  - 차번호판 : 2252장, 도로명주소판 : 1670장

< 이미지 데이터 변환 예시 >

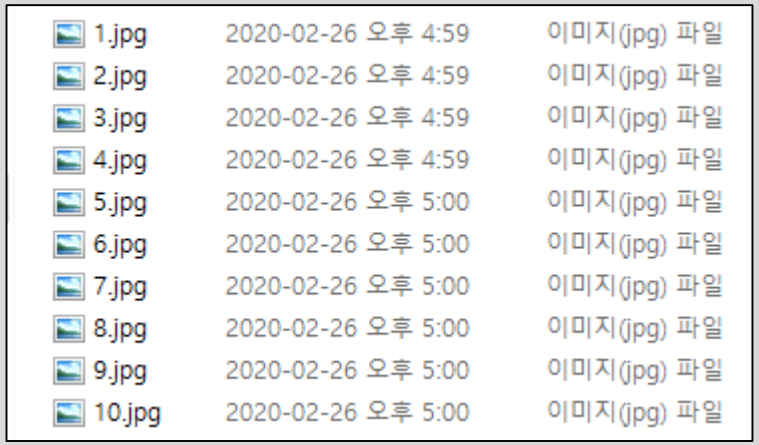
시스템 구현 :: Detection (탐지)

3. 이미지 라벨링

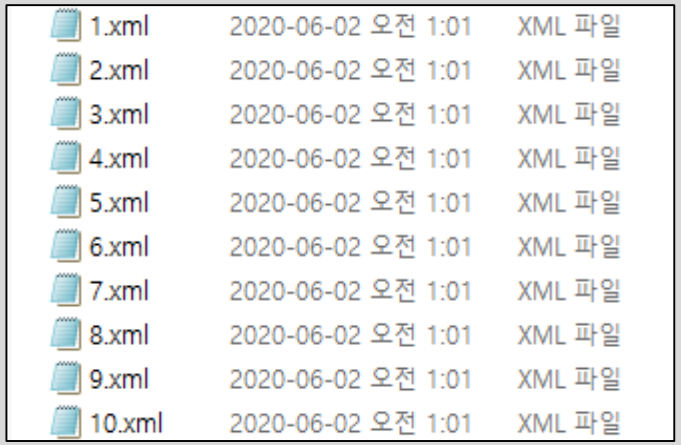
- 수집한 이미지에서 탐지해야 할 영역을 지정
  - 라벨링 도구 [labelimg] 사용
  - 라벨 이름과 설정한 영역의 위치가 저장된 [.xml] 파일이 생성



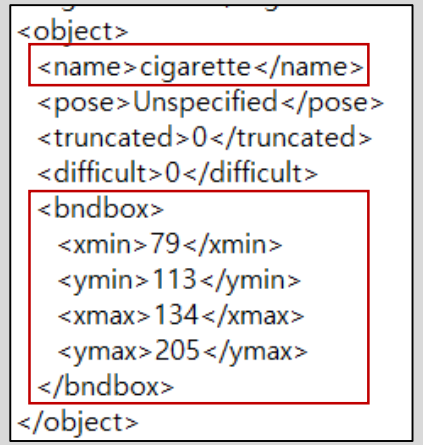
〈labelImg 사용 예시〉



〈 이미지 파일 예시 〉



〈 이미지에 대한 라벨링 완료 후 생성된 xml 파일 예시 〉



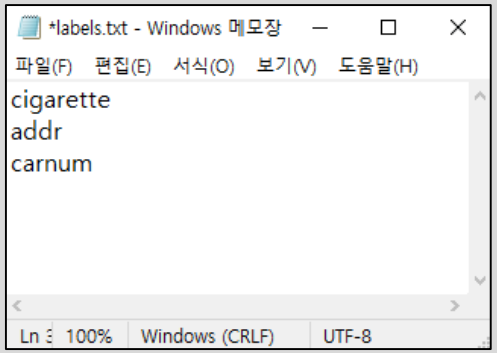
〈 xml 파일 내용 일부 예시 〉

시스템 구현 :: Detection (탐지)

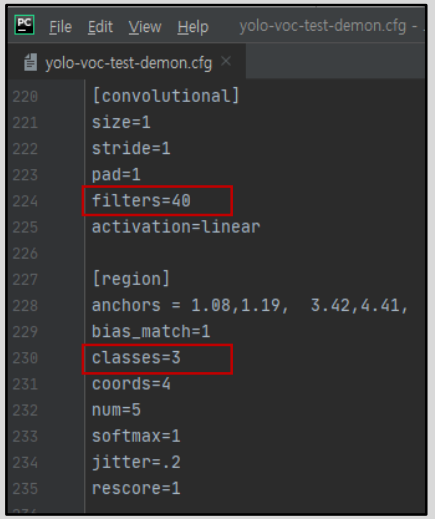
4. 학습 준비

- Darkflow 설치 (Tensorflow로 구현된 YOLO v2)
  - YOLO (You Only Look Once) : 이미지를 여러 그리드로 나누어 연산을 진행하는 객체 탐지 알고리즘
  - URL : https://github.com/thtrieu/darkflow/c

- 필요 SW설치 및 설정
  - GPU 연산을 위한 CUDA, CuDNN 설치
    - CUDA : GPU를 이용한 개발을 돕는 통합 환경
    - CuDNN : GPU 가속화 라이브러리
  - 탐지할 객체 항목의 개수에 맞게 [.cfg] 파일 수정
    - cfg : 모델의 세부 설정을 담은 파일
    - classes = 탐지할 객체의 항목 개수
    - filters = (5+classes) x 5
  - 탐지할 객체들의 이름을 [labels.txt]에 추가
    - 라벨링 시 지정했던 이름으로 설정



〈 labels.txt 파일 내용 예시 〉



〈 .cfg 파일 내용 예시 〉

시스템 구현 :: Detection (탐지)

개발 언어 : Python
IDE : Jupyter Notebook, PyCharm

5. 학습

- 관련 파라미터 설정
  - Model 및 Optimizer
    - Model : 수정했던 cfg 파일
    - Optimizer : Adam
  - Hyper parameters
    - batch size : 4
    - epoch : 총 200 (learning rate마다 다르게 설정)
    - learning rate : 1e-4, 1e-5, 1e-6 으로 점차 줄여가며 설정

```
In [21]: options = {"model": "cfg/yolo-voc-test-addr_adam.cfg",
                    "batch": 4,
                    "epoch": 10,
                    "load" : -1,
                    "gpu": 0.5,
                    "lr" : 1e-5,
                    "train": True,
                    "trainer" : "adam",
                    "annotation": "./data/annotations_addr/",
                    "dataset": "./data/dataset_addr"}

         tfnet = TFNet(options)

         Parsing cfg/yolo-voc-test-addr_adam.cfg
         Loading None ...
```

< 파라미터 설정 코드 예시 >

- 학습 진행
  - 하드웨어 환경
    - CPU : Intel(R) Core(TM) i5-7500T
    - GPU : NVIDIA GeForce GTX 1070
    - Memory : 16.0 GB

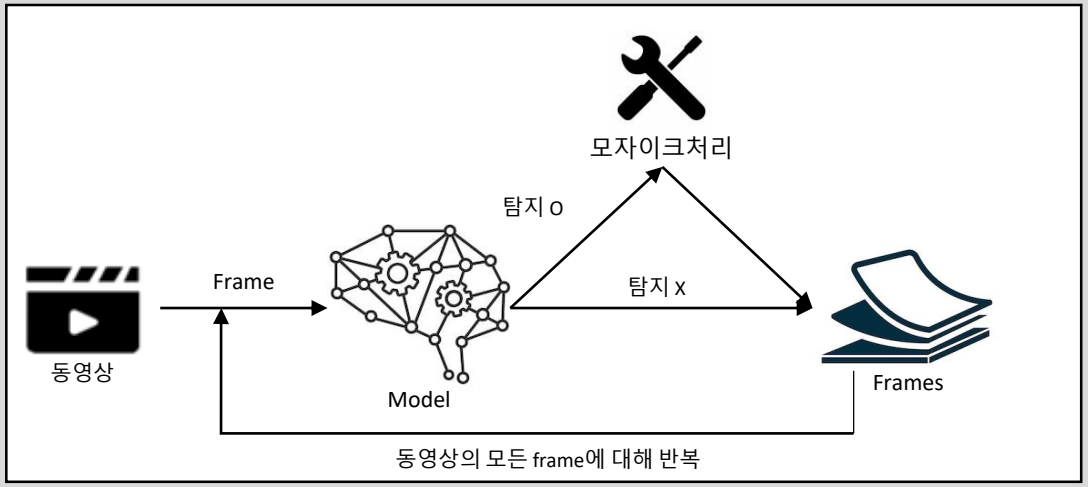
```
[*]: tfnet.train()
     step 54526 - loss 0.8527671098709106 - moving ave loss 0.7037386216748149
     step 54527 - loss 0.7712291479110718 - moving ave loss 0.7104876742984407
     step 54528 - loss 0.4231833815574646 - moving ave loss 0.6817572450243432
     step 54529 - loss 0.2653799057006836 - moving ave loss 0.6401195110919773
     step 54530 - loss 0.618375301361084 - moving ave loss 0.637945090118888
     step 54531 - loss 0.2559196949005127 - moving ave loss 0.5997425505970505
     step 54532 - loss 0.5104020833969116 - moving ave loss 0.5908085038770367
     step 54533 - loss 0.38828206062316895 - moving ave loss 0.5705558595516499
     step 54534 - loss 0.21669812500476837 - moving ave loss 0.5351700860969617
     step 54535 - loss 0.27908676862716675 - moving ave loss 0.5095617543499822
     step 54536 - loss 0.8501402139663696 - moving ave loss 0.5436196003116209
     step 54537 - loss 0.3319041132926941 - moving ave loss 0.5224480516097283
     step 54538 - loss 1.1291472911834717 - moving ave loss 0.5831179755671027
     step 54539 - loss 0.26872003078460693 - moving ave loss 0.5516781810888531
```

< 학습 진행 예시 >

시스템 구현 :: Detection (탐지)

6. 탐지 및 모자이크 처리

- **탐지 및 모자이크 과정**
  1) 영상에서 하나의 frame을 추출
  2) frame을 학습시킨 모델에 통과시킴
  3) 통과시킨 frame에 탐지된 객체가 있다면
  4) 해당 부분을 모자이크 처리
  5) 처리가 완료된 frame을 쌓아 영상으로 만듦



〈 탐지 및 모자이크 과정 그림 예시 〉

```python
# 비디오 코덱은 XVID
fourcc = cv2.VideoWriter_fourcc(*'XVID')
# 30프레임짜리 XVID코덱으로 저장하는 포맷 변수 선언
writer = cv2.VideoWriter('output_addr_0919_2_1_Adam.avi', fourcc, 30.0, (width, height))

while(cap.isOpened()) :
    ret, frame = cap.read()
    if ret == False:
        break

    result = tfnet.return_predict(frame)
    result_img = frame.copy()
    # 모자이크 처리 부분
    if result:
        print(result)
        for res in result:
            x = res['topleft']['x']
            y = res['topleft']['y']
            w = res['bottomright']['x']
            h = res['bottomright']['y']
            target = frame[y:h, x:w]
            flt = cv2.blur(target, (40, 40))  # blur처리 한 후 flt 변수에 저장
            result_img[y:h, x:w] = flt
            tl = (x, y)
            br = (w ,h)
            label = res['label']
            conf = res['confidence']
            result_img= cv2.rectangle(result_img, tl, br, (0, 255, 0), 2)
            result_img= cv2.putText(result_img, label, tl, cv2.FONT_HERSHEY_TRIPLEX, 1,
    else :
        result_img = frame.copy()
```

〈 탐지 및 모자이크 처리 코드 예시 〉

테스트 영상

〈 Before 〉

〈 After 〉

3. 시연 영상


YoMo
You Only Mosaic Once
www.BANDICAM.com
Help
About
한국어
모자이크 대상
-유해이미지
담배
칼
-개인정보
택배운송장
우편물
차량번호판
도로명 주소판
도로 표지판
티켓
핸드폰 화면
모자이크 Go!
동영상 업로드
동영상 다운로드

2020캡스톤디자인

감사합니다.