

올바른 손 씻기 교육을 위한 손 씻기 단계 인식 모델

#개인위생 #올바른손씻기 #6단계 #내몸은 #내가지켜

CV-04

무럭무럭 감자밭

팀 소개



김세영



신승혁



이윤영



조성욱



박성진



이상원



이채윤

목차

01/ Intro

프로젝트 개요

팀 소개

개발 목표

02/ Model Research

데이터셋 준비

모델 연구

최종 모델 선정

03/ Product Serving

시스템 아키텍처

구현

데모

04/ Conclusion

R & R

Future Works

Q & A

1. Intro

1.1 프로젝트 개요



위생 수칙의 기본인 **손 씻기**, 잘 지키고 계신가요?

1.1 프로젝트 개요

올바른 손 씻기를 안했을 때 손에 남는 병원성 미생물의 양

- 대부분 씻기지 않음
- 종종 씻기지 않음
- 전혀 씻기지 않음
- 세균 ZERO

출처
보건복지부·질병관리본부
올바른 손 씻기 홍보리플렛(2016)



손바닥



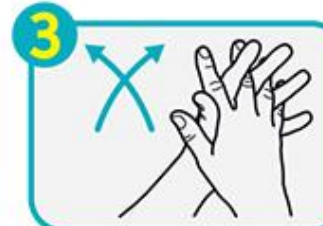
손바닥과 손바닥을
마주대고 문질러 주세요

손등



손등과 손바닥을
마주대고 문질러 주세요

손가락 사이



손바닥을 마주대고
손가락을 끼고 문질러 주세요

두 손 모아



손가락을 마주잡고
문질러 주세요

엄지 손가락



엄지손가락을 다른 편
손바닥으로 돌려주면서
문질러 주세요

손톱 밑



손가락을 반대편 손바닥에
놓고 문지르며 손톱 밑을
깨끗하게 하세요

1.3 개발 목표

- 실시간 영상에서 손 씻기 단계를 판별하고 정부 권장 손 씻기 6단계 지침을 수행할 수 있게 도와주는 서비스
- 다양한 분야에서 사용될 수 있는 서비스
 - 아동 손 씻기 교육을 위한 스마트폰 애플리케이션
 - 음식점, 병원, 공공장소 등에서 사용될 수 있는 손 씻기 검수 애플리케이션
- 기존의 rule-based 방식으로는 다양한 변수를 고려하기 어려운 문제
 - 피부색, 촬영 환경, 개인마다 다른 손 모양 등

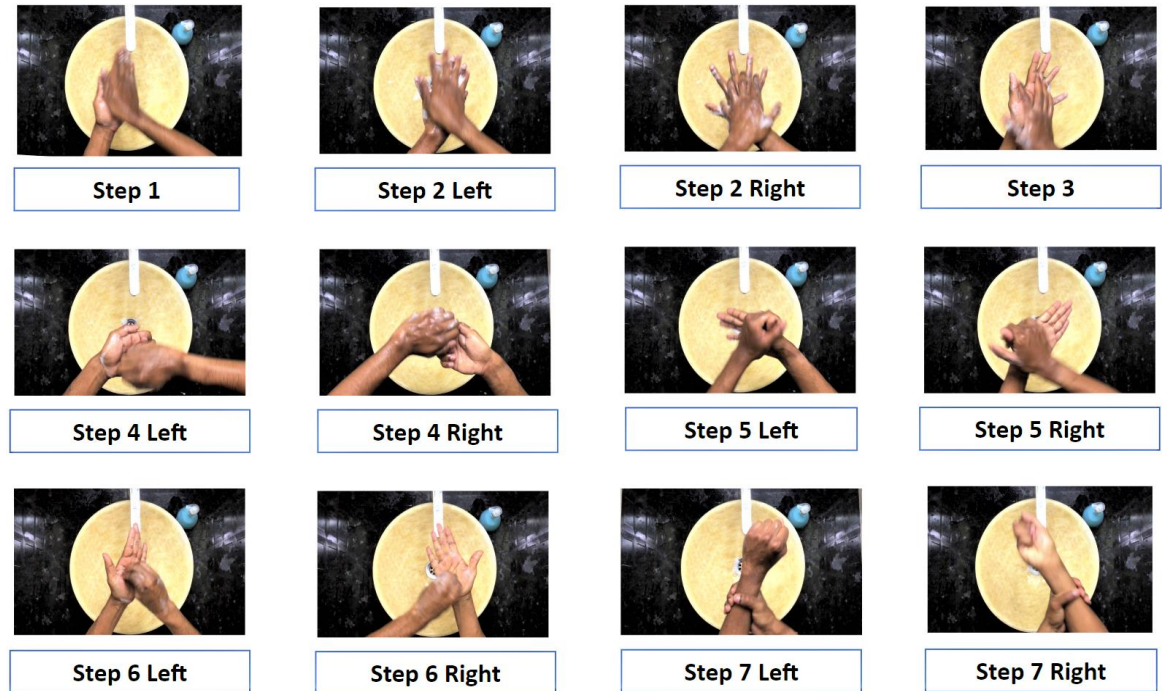
∴ 다양한 데이터를 통해 학습된 딥러닝 모델로 문제 해결

2. Model Research

2.1 Dataset

Kaggle 손 씻기 데이터셋 (링크)

- 총 25개의 손 씻는 영상을 각 영상마다 12개의 손 씻기 단계별로 분할 저장된 데이터셋
- [손씻기 6단계 + 손목 씻기]를 오른손, 왼손에 따라 12단계로 세분화
- 정부 권장 6단계에 포함되지 않는 step 7을 제외
- 데이터의 통일성을 위해 좌, 우 라벨 통합
- 데이터의 품질은 매우 좋지만, **한 사람의 손만 존재**



2.1 Dataset

병원 손 씻기 데이터셋 ([링크](#))

- 총 3185개의 영상으로 kaggle 데이터셋보다 10배 이상 더 많고 다양한 인종, 구도를 가진 데이터셋
- 평소 의료진이 일할 때 손 씻는 모습
- [손씻기 6단계 + 타올로 수도꼭지 잠그기]에 따라 프레임 별로 라벨링
- 여러 명의 annotator 존재
- 영상이 매우 많지만 부정확한 라벨링과 좋지 않은 품질의 데이터가 다수 존재

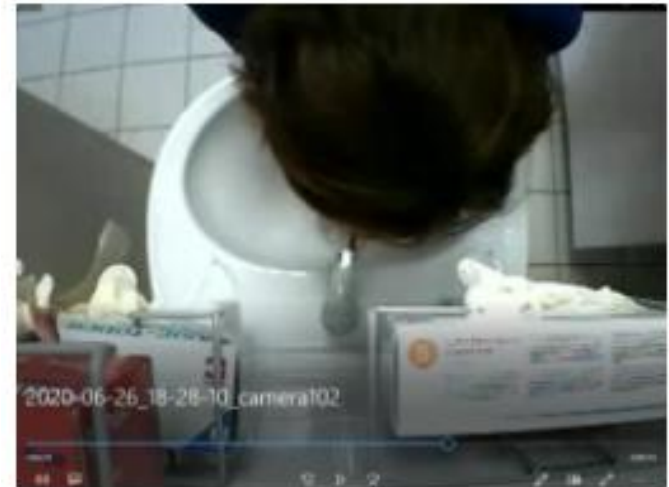


2.1 Dataset

이상치 삭제

- 병원 데이터셋에 라벨링 잘못 된 경우
 1. 동작 라벨이 일치하지 않음
 2. 수도꼭지 또는 머리에 가려서 손이 보이지 않음
- Annotator들끼리 라벨이 일치하지 않은 경우
- Annotator가 한명인 경우
- 프레임-라벨 조금씩 밀려 있는 오류 → 뒤 10프레임 씩 제거
- 손씻기 6단계에 해당하지 않는 데이터 제거

∴ 총 121,053 프레임 추출



2.1 Dataset

Validation용 golden dataset

- 팀원들&가족들이 직접 촬영 및 라벨링
- 다양한 환경에서 촬영
- 교육용을 고려하여 아동 손 씻기 영상 촬영
- 총 984 프레임 수집



2.2 Model Research

문제 정의

- 손을 씻는 실시간 영상에서 현재 진행중인 손 씻기 단계 구분하기
- Online serving

문제 해결 접근법

- 영상 단위 입력
 - Video action classification
- 프레임(사진) 단위 입력
 - Image classification
 - Object detection

2.2.1 Video Action Classification

모델 선정 배경

- 여러 프레임을 묶어 동작을 구분하는 video action classification 모델을 사용하여 문제를 해결할 수 있다고 추측

실험내용

- 대표적인 video task 데이터셋인 kinetics 400 데이터셋 형식을 따라 기존 데이터셋 정제
- Accuracy ↔ Complexity의 trade - off 고려한 X3D 모델을 사용하여 classification 실험

결과 & 한계점

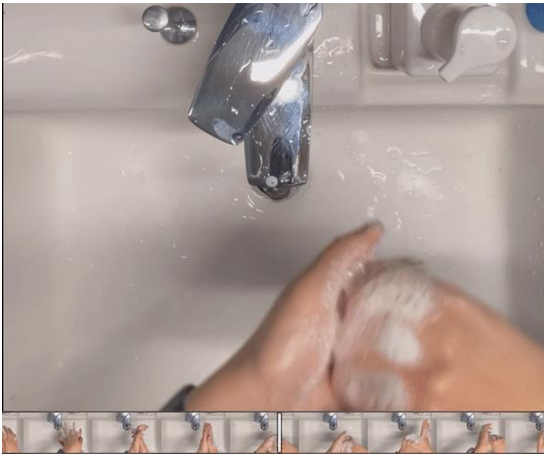
- Validation을 확인했을 때 영상마다 성능의 편차가 큼, 훈련 데이터셋에 과적합이 빠르게 됨
- 기간 내에 목표 성능을 달성하기 어려울 것이라 판단

2.2.2 Image Classification

모델 선정 배경

- 가지고 있는 데이터셋은 영상의 프레임마다 동작이 라벨링된 **classification용 데이터셋**
- 각 **동작의 차이가 크다**고 판단되어 단순한 Image classification으로도 적당한 성능이 나올 것이라 추측

데이터셋 제작



Video

Frame #	Step
...	...
112	4
113	4
114	5
115	5
...	...

Annotation.csv



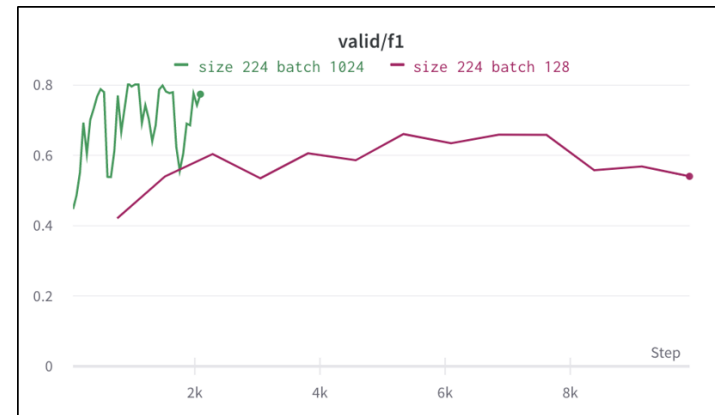
2.2.2 Image Classification

실험내용

- 영상을 잘라 만든 데이터셋의 특성 상, batch size가 작을수록 영상의 앞, 뒤 부분을 미리 학습해 local minima에 빠질 것이라 추측
 - Batch size 증가시켰을 때 가장 높은 성능 향상 (f1 score 0.65 -> 0.80)

결과 & 한계점

- 모델 성능 자체는 일반적인 상황에서 충분히 실생활에서 사용 가능
 - 단 테스트를 진행할수록 배경이 특이하면 예측 성능을 보장해주지 못하는 문제 발견
- "시작" 버튼을 눌러야만 판별할 수 있는, 즉 시작 타이밍을 선택해주지 못하는 한계
 - 손을 탐지할 수 있는 object detection 모델의 필요성



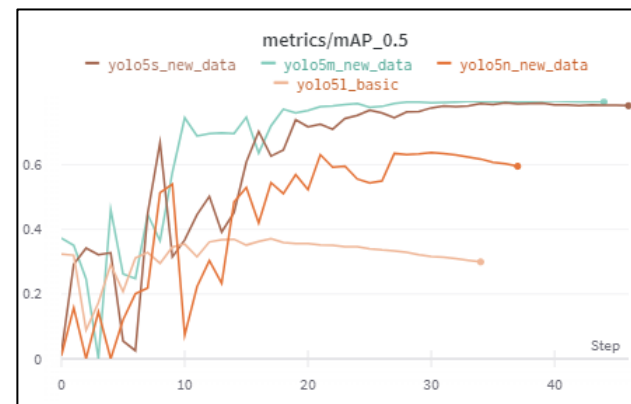
2.2.3 Object Detection

모델 선정 배경

- 손 객체를 탐지함과 동시에 동작 분류 가능
- 처리 속도, 실시간성, background error를 고려해서 1-stage 모델로 접근

YOLOv5

- Real-time object detection task 대표 모델
- 활용도가 높고 접근성이 좋아 customized dataset을 실험하기에 편리한 프레임워크
- mAP@50을 모델 성능 평가 metric으로 설정
- 다양한 모델 중 크기, 성능 고려하여 YOLOv5s와 YOLOv5m으로 실험



2.2.3 Object Detection

Train 데이터 셋 pseudo labeling & 검수

- Hand-pretrained model을 이용하여 동영상 프레임별 **bbox pseudo labeling** 진행
- 데이터 검수 가이드라인
 1. 두 손만 타이트하게 잡은 경우, 손동작이 식별 가능한 경우만 보존
 2. 위에서 아래로 찍은 구도가 아닌 경우 삭제

∴ 121,053장 → 54,410장 확보



2.2.3 Object Detection

Object detection용 valid 데이터셋 제작

- Golden dataset에 대해 **annotation tool**을 이용하여 **bbox 라벨링**
- 라벨링 가이드라인
 1. 손목은 포함하지 않는다
 2. 두 손을 최대한 타이트하게 잡는다



2.2 최종 모델 선정

Model		mAP@50
<i>YOLOv5s</i>	Batch size 144	0.715
	+ Brightness Aug	0.7457
	+ Mosaic ↑ , Mixup ↑	0.7906
<i>YOLOv5s Multiscale</i>	Batch size 64	N/A
	+ Brightness Aug	0.8643
	+ Mosaic ↑ , Mixup ↑	0.8753
<i>YOLOv5m</i>	Batch size 100	N/A
	+ Brightness Aug	0.7966
	+ Mosaic ↑ , Mixup ↑	0.8375

** Default Augmentation*

Flip, Rotate, HueSaturationValue, Clahe, Blur, Mosaic, Mixup

2.2 최종 모델 비교



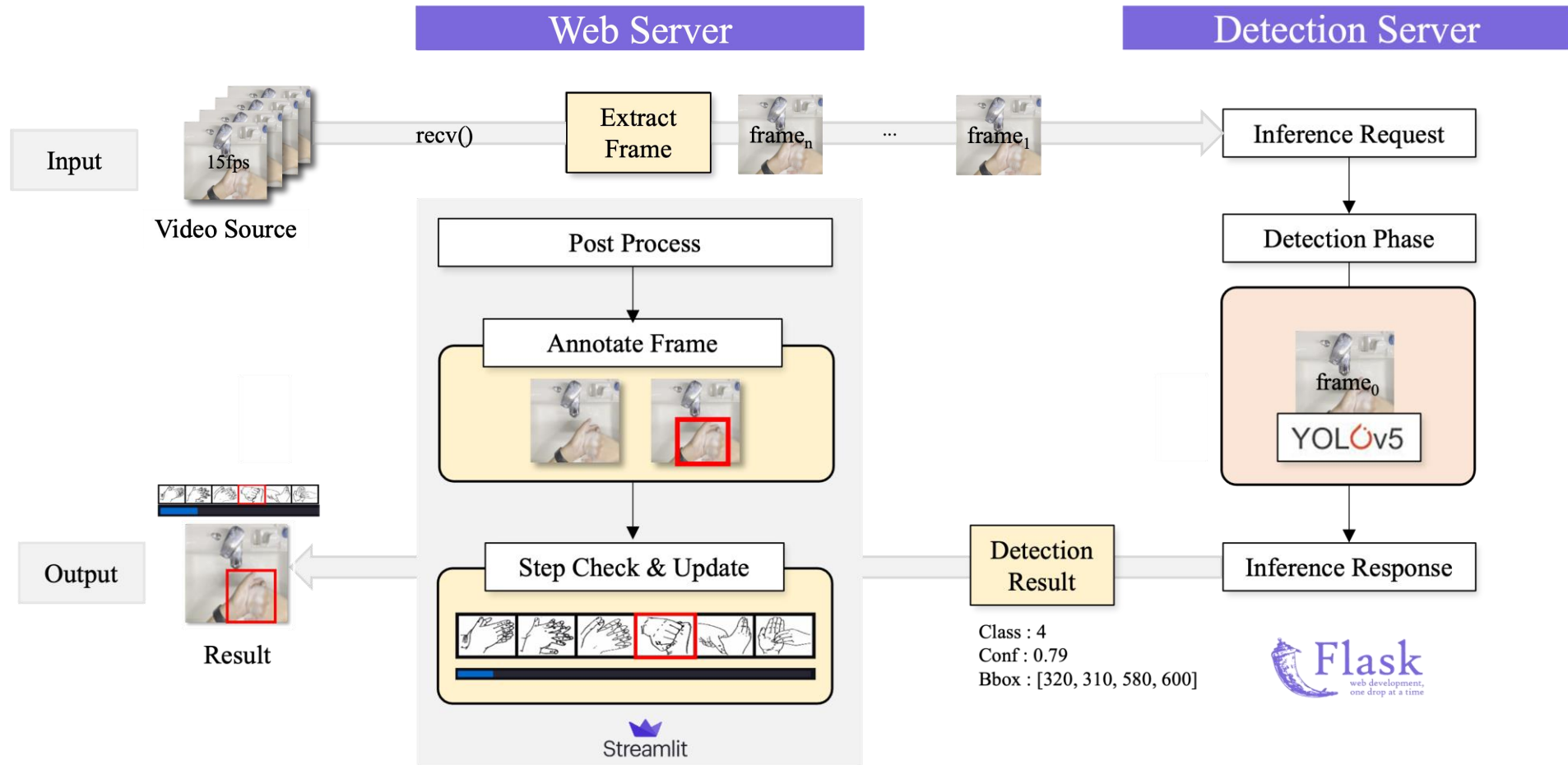
Base model



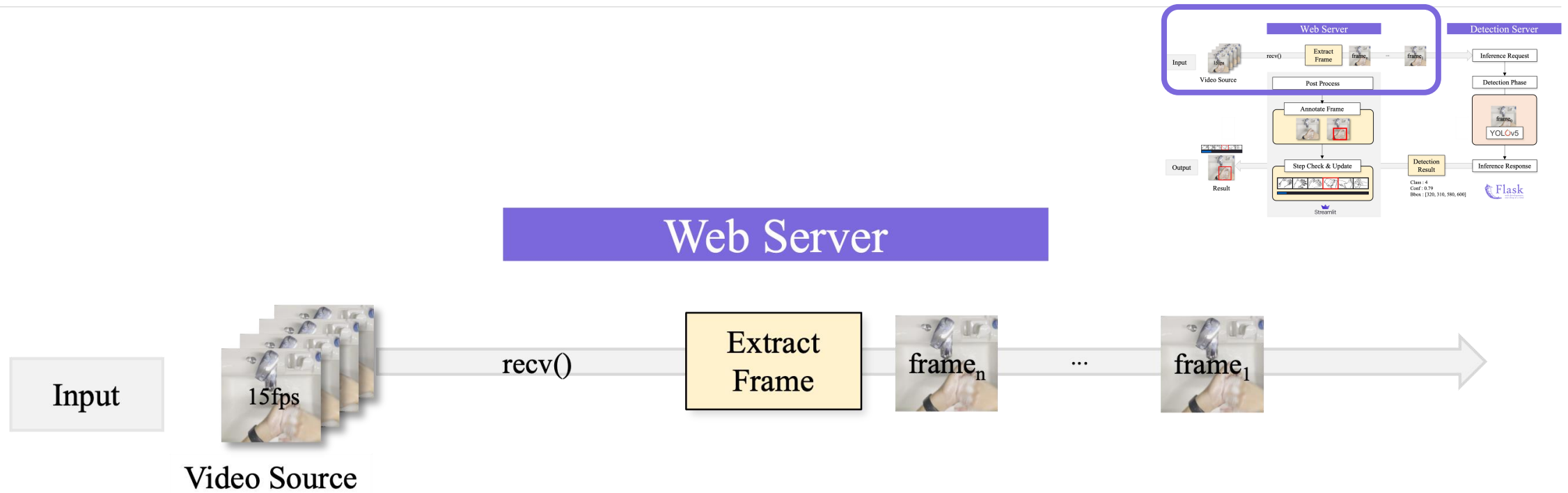
Multiscale model w/ augment

3. Product Serving

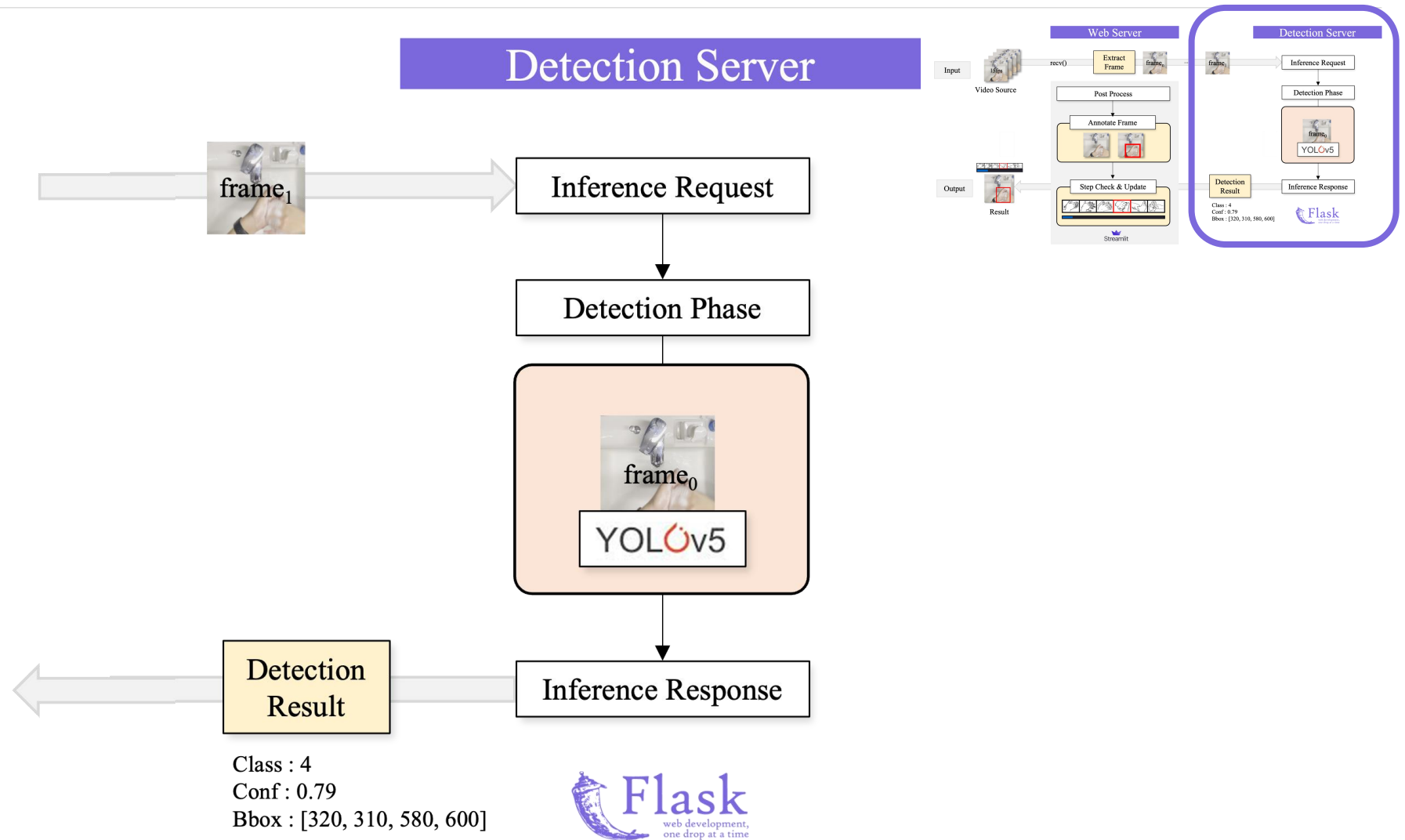
Overall Architecture



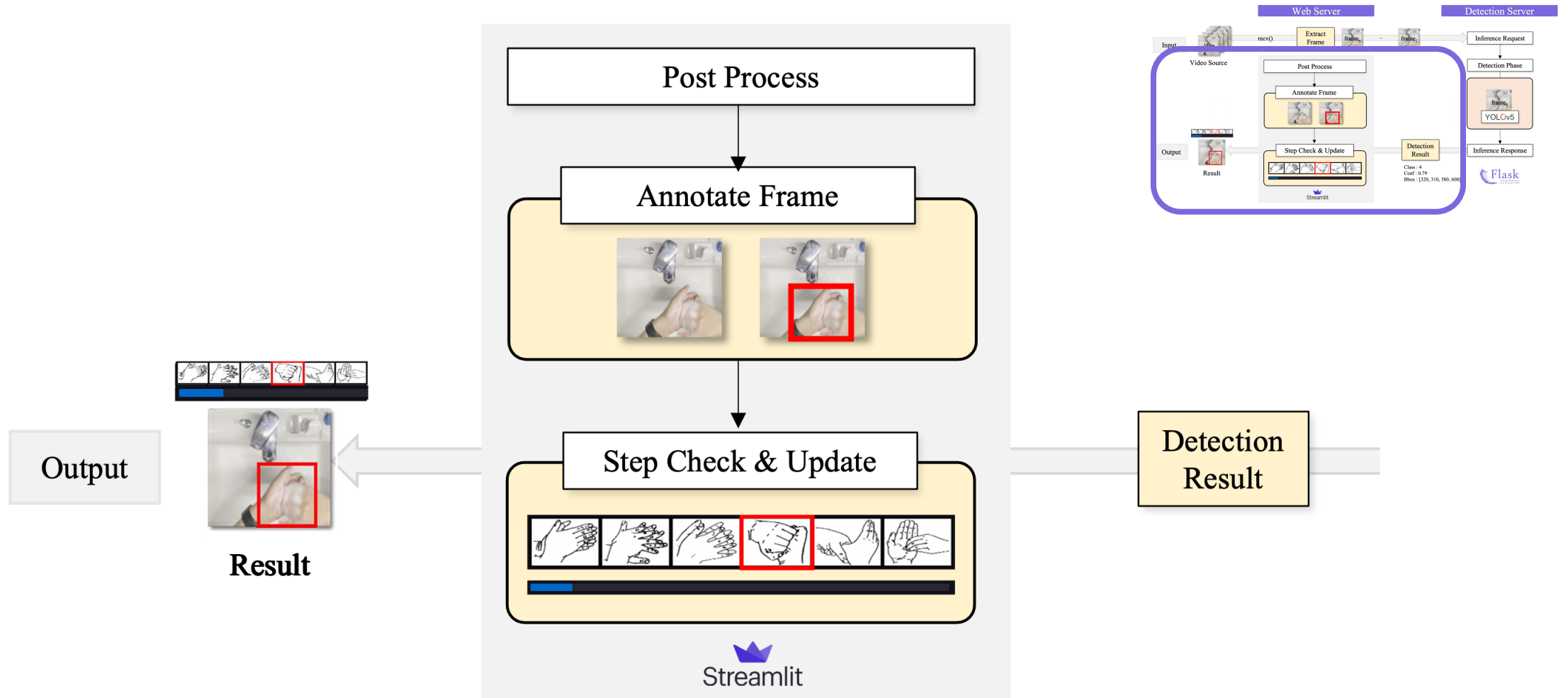
Overall Architecture



Overall Architecture



Overall Architecture

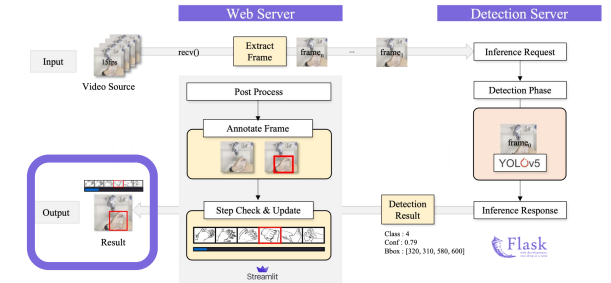


Overall Architecture

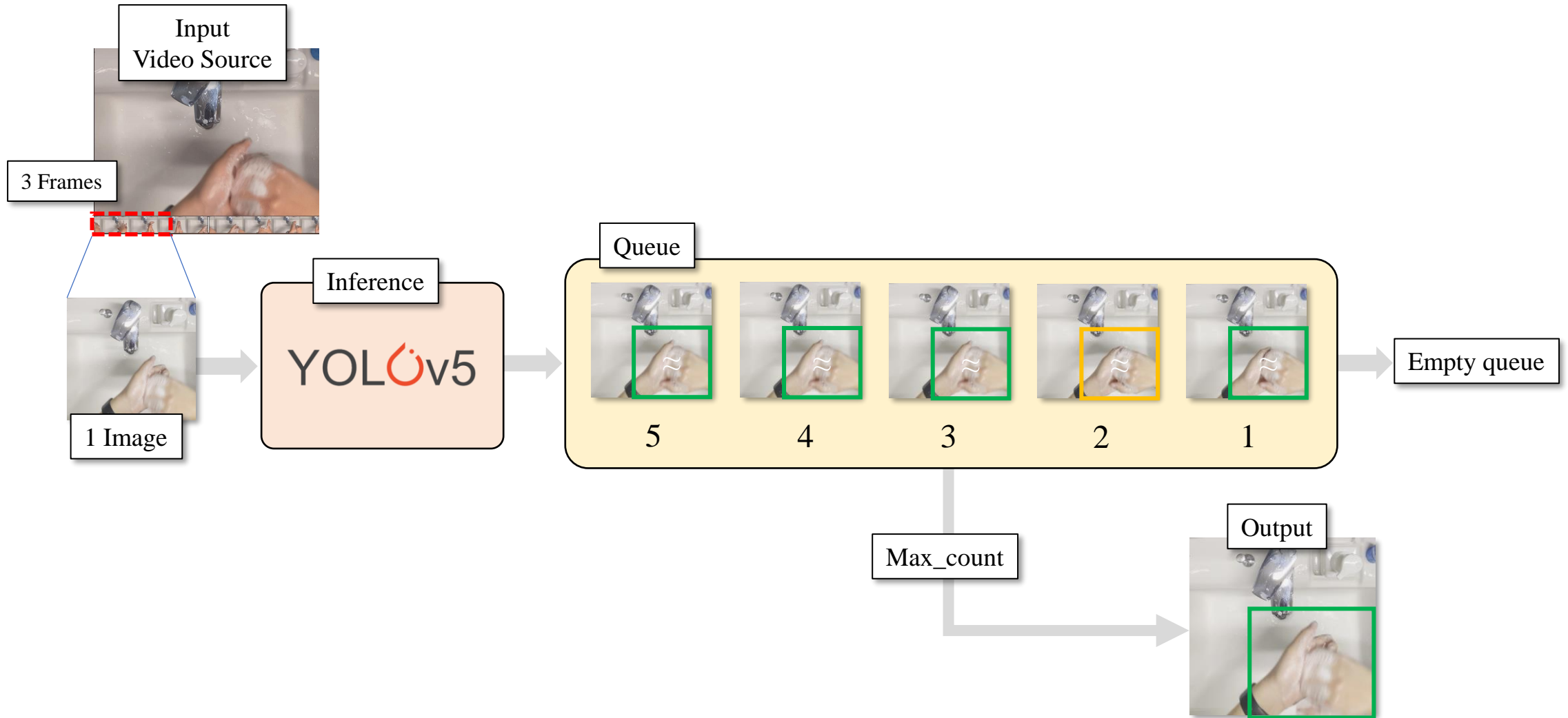
Output



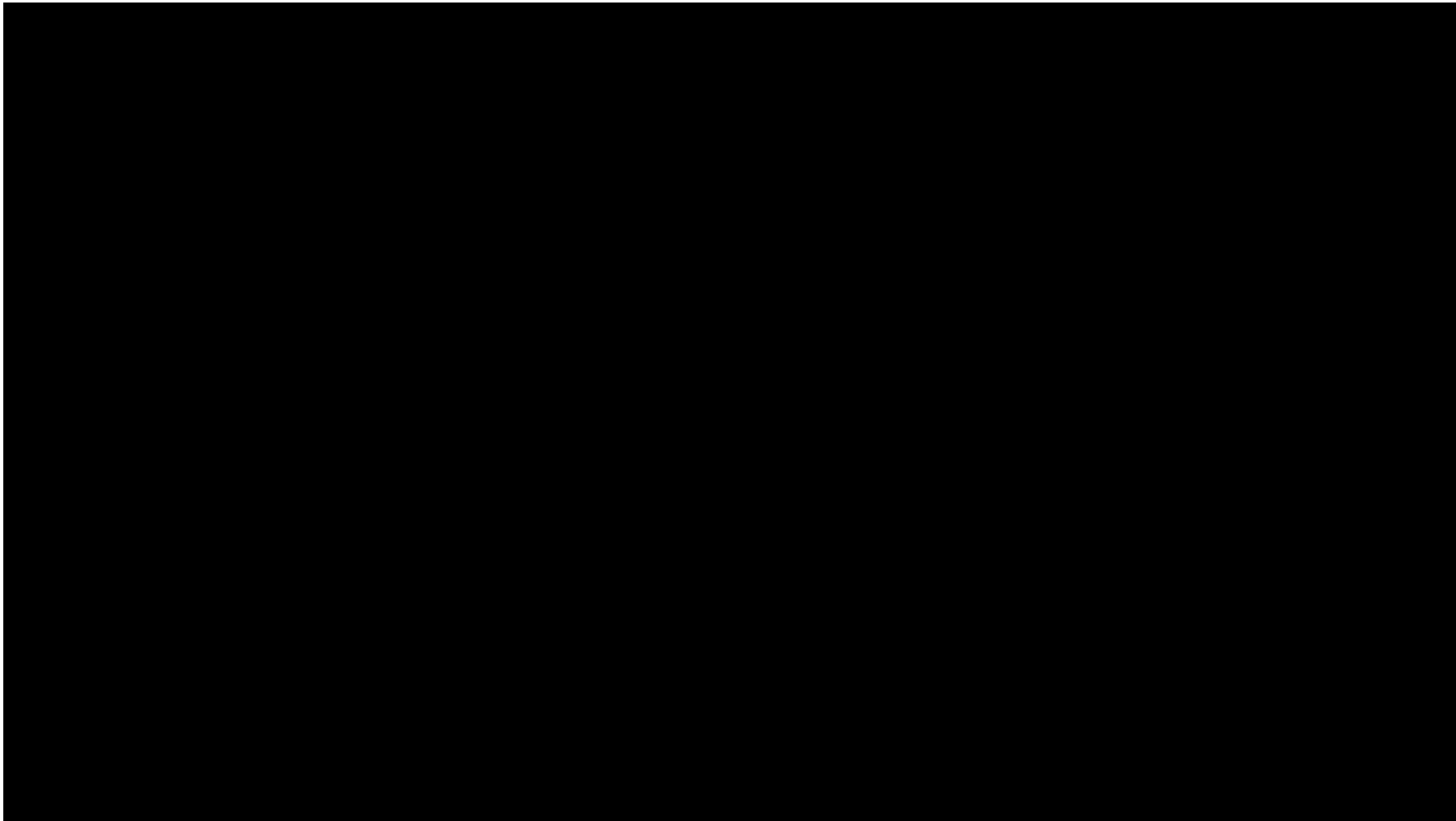
Result



2.4 Input/Output



2.4 Demo



4. Conclusion

4. Future Works

Server-side에서 client-side로 전환

- 현재 server-side로 구현 되어있음 → client-side로 전환
- 경량화 기법 시도

동작 세분화

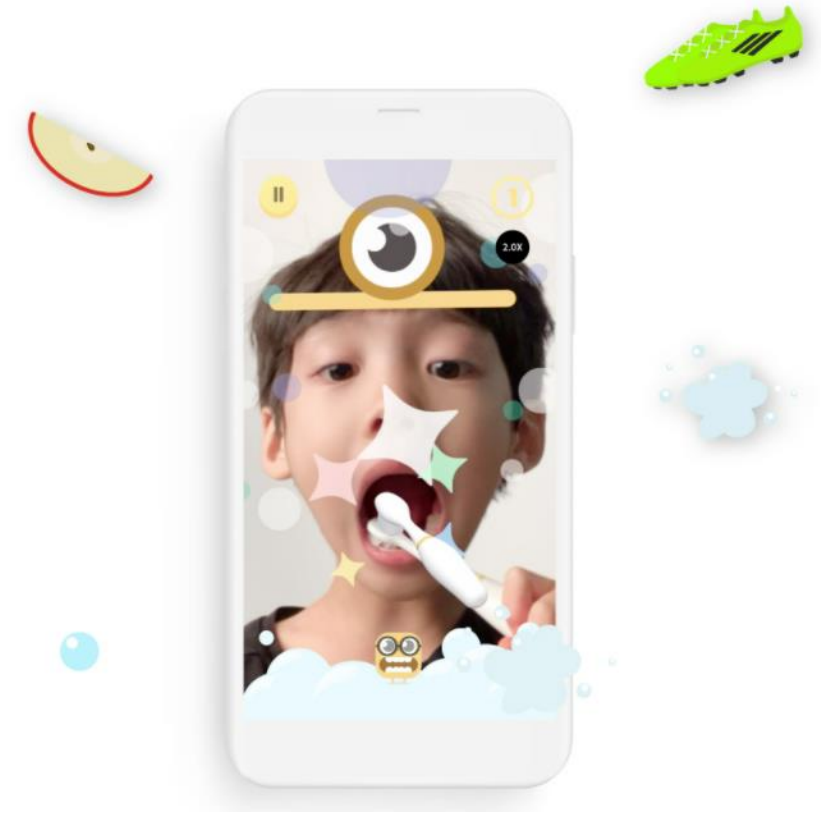
- 현재는 동작이 6단계 중 어떤 단계에 포함되는지 분류
- 왼손, 오른손이 구분되는 동작의 경우 단계를 세분화해서 정확도를 높이는 방법 고려

Video task로 해결

- 현재는 프레임 단위 object detection → 앞뒤 맥락을 고려하지 못함

4. Future Works

- 교육용 콘텐츠로서 재미요소 가미
 - 실시간 촬영 영상 이외에 UI에서 세균 애니메이션 추가
 - 손 씻기가 잘 진행된다면 세균이 사라지도록 구현
 - 2D 이미지 인식에서 3D 인식으로 발전해 AR 기술로도 첨가 가능



출처: 브러쉬 몬스터
<https://brushmon.kr/>

Q&A

Appendix I. 팀 내 역할



김세영

- Object Detection 실험 리딩
- Pseudo Labeling
- Object Detection용 데이터셋 정제·배포
- Multiscale Model 실험
- Project Manager



신승혁

- Video Classification 실험
- Pretrained Hand Detection 모델 탐색 및 실험
- YOLOv5 실험



이윤영

- Video Classification 베이스라인 코드 작성 및 실험
- Object Detection용 데이터셋 정제·배포
- Background Image 추가
- YOLOv5 실험



조성욱

- 데이터 분석
- 데이터셋 정제 모듈 구축
- Image Classification 연구 베이스라인 구축 및 실험
- 프로젝트 구조 설정
- AWS 환경 구축

Appendix I. 팀 내 역할



박성진

- Post Processing 구현
- 시작 Trigger 구현
- UI 개선
- Demo 영상 제작
- Project Manager



이상원

- 자료조사 (데이터, 모델)
- Test UI 구현(PyQt5)
- Post Processing 구현
- Web 기반 UI 구현



이채윤

- 자료조사 (웹 서버)
- 스트림릿 프로토타입 개발
- Web 기반 UI 구현 및 개선
- 통합테스트 실험