

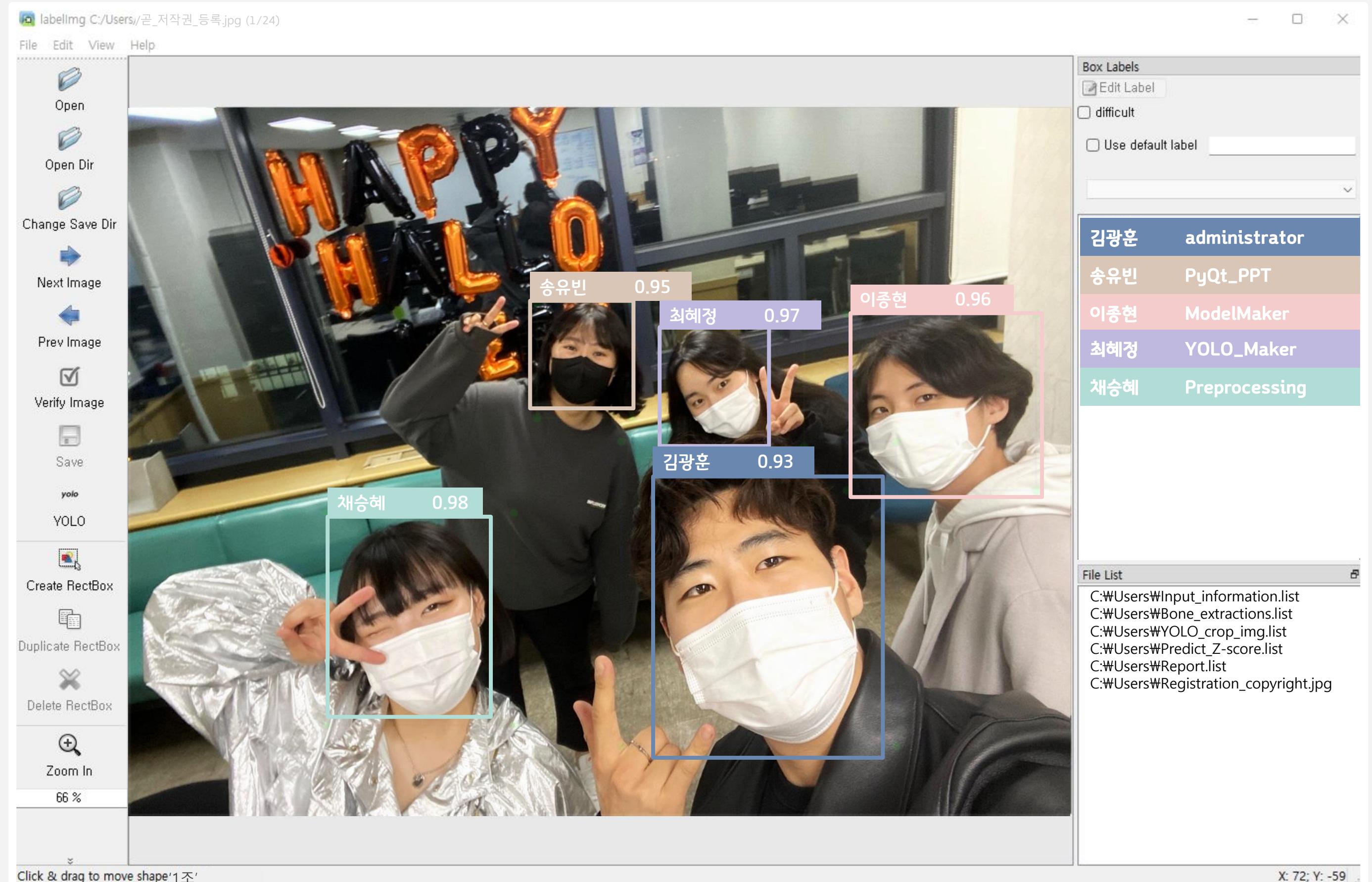
2021.11.12 | cakd3 2차프로젝트

# 골연령 예측

손목, 손가락 관절 탐색 및 인공신경망 모형 제작

1조 | 김광훈, 송유빈, 이종현, 채승혜, 최혜정





**김광훈**  
팀장 / 발표  
<https://github.com/kikiru328>

**송유빈**  
총괄 / PyQt5  
<https://github.com/yibnn>

**이종현**  
모델제작 / PPT  
<https://github.com/Jjongu>

**최혜정**  
YOLOv5 / PyQt5  
<https://github.com/601chl>

**채승혜**  
전처리 / PyQt5  
<https://github.com/SeunghyeChae>

## 제작 환경

Windows 10

OS

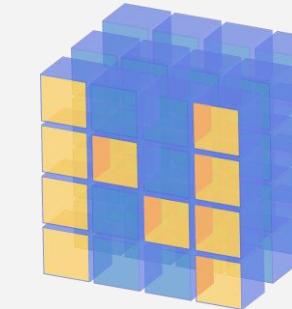


사용언어

ANACONDA®



IDE



OpenCV



PyTorch

분석  
라이브러리

# CONTENT

01.

## 주제선정 이유

- # Background
- # Pain points

02.

## 프로젝트 진행

- # Project Progress
- # Image Preprocessing
- # Yolov5
- # TJ-Net

03.

## 함수 자동화 및 GUI

- # 프로그램 자동화
- # PyQt를 이용한 GUI 제작

01.

---

## 주제선정 이유

# 성장평가를 실시하는 이유



1

아동에게서 성장이란 구조적, 기능적, 행동적 변화를 의미

2

신체 구조의 변화 뿐 아니라 심리적, 행동, 적응과정에 이르기까지 총체적인 변화를 의미

3

아동기에는 성장발달의 범위가 넓고 빠른 속도로 진행되어 인간 발달학에서 아동기를 중요한 시기로 다루고 중요한 기초로 봄

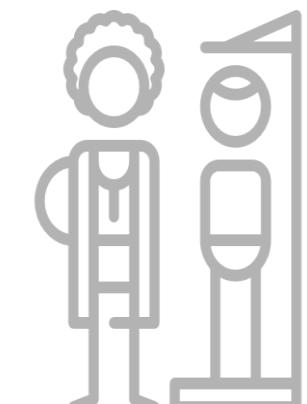
**\*불가역성의 원리** : 특정시기의 발달에 결함이 생기면 후에 아무리 많은 경험을 제공하고 보상해도 원래의 시기로 되돌려 만회할 수 없음

# 골연령 검사

아동의 키는 실제 연령보다 골연령 및 골성숙과 깊은 연관이 있음

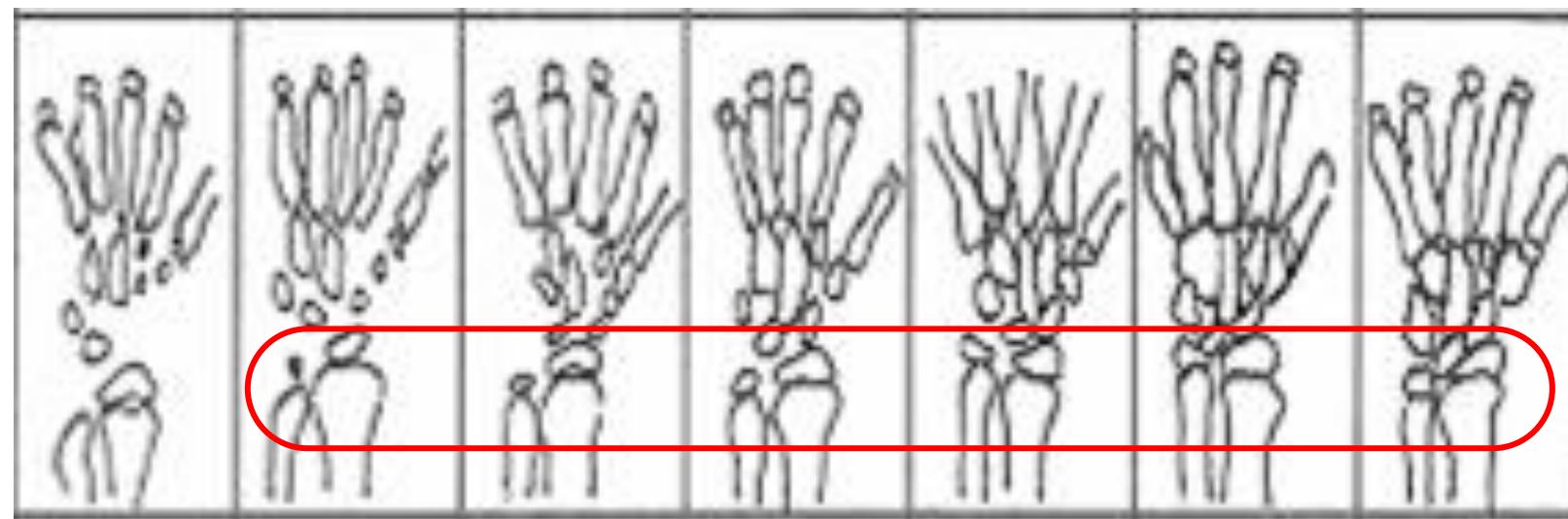
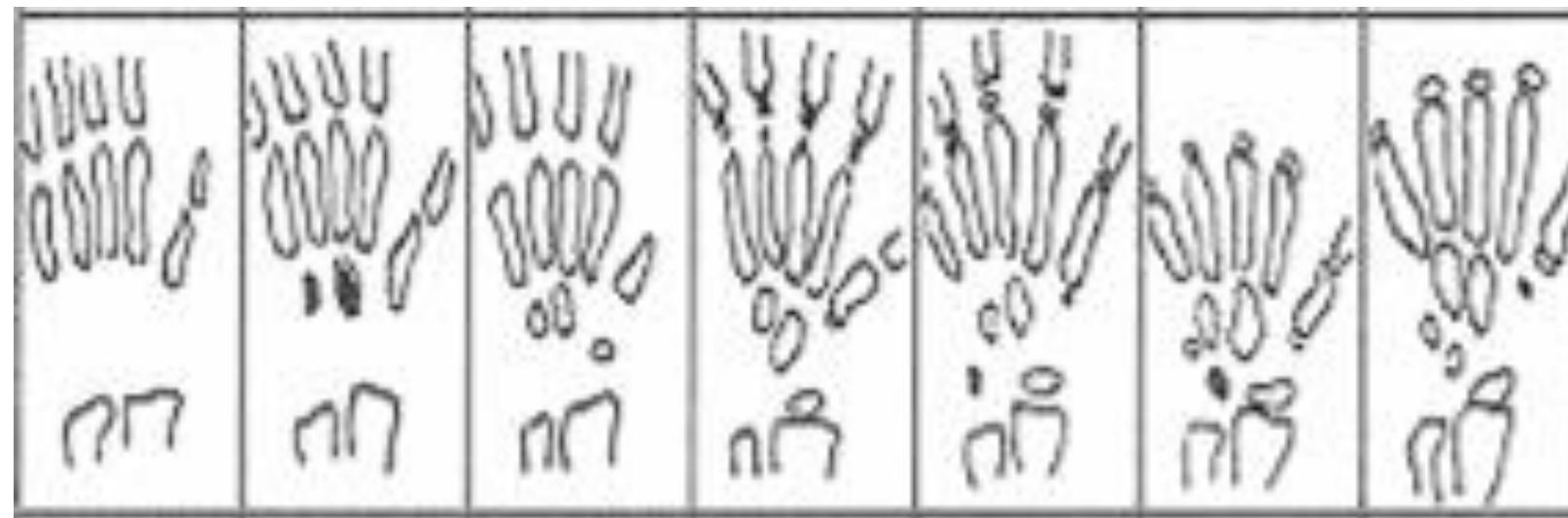
일반적으로 왼쪽 손등과 손가락을 촬영하여 골연령을 예측하여 개인의 성장 잠재력과 성인기를 예측할 수 있음

따라서 골연령을 측정함으로써 성장기 아동과 청소년의 성장을 판단하는데 유용한 검사임



# 성장기 범주별 특징

생후 5주

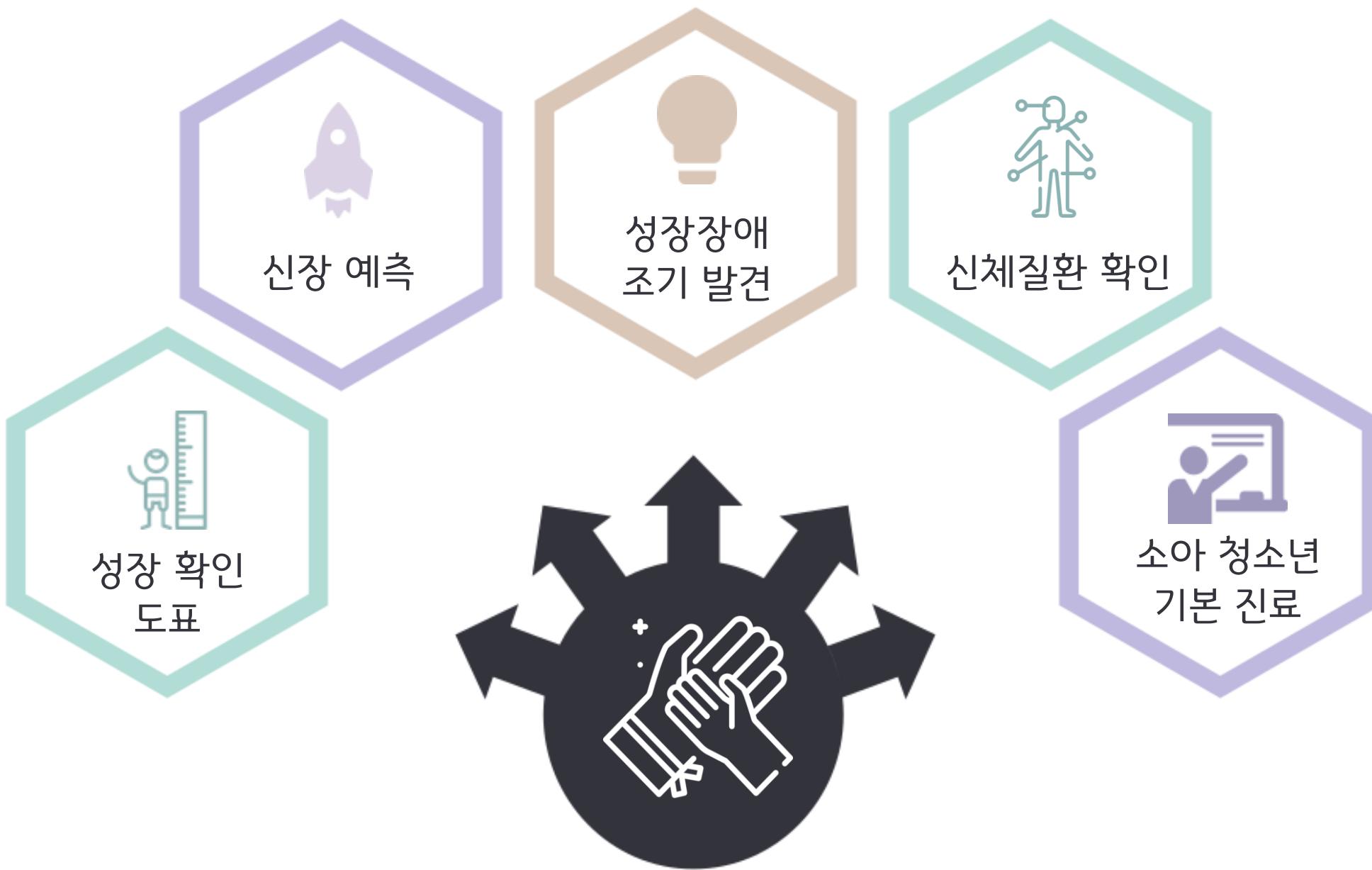


13세

18세

## 손목과 손가락의 성장기 범주별 특징

- 생후5주 후부터 손목과 손가락의 석화(뼈의 생성)가 시작됨
- 태어나서 13세까지 손목뼈의 성장으로 골연령을 판독
- 13세부터 18세까지 손가락 관절의 성장으로 골연령을 판독



## 골연령 검사의 활용

빠른 검사를 통한 조기 진단은 빠른 회복을 가져올 수 있음

02.

---

## 프로젝트 진행

## 프로그램 제작 이유



측정하는 시간이 오래 걸려 외래에서 제한된 시간에 측정이 불가

경험이 많고 숙련된 전문의가 시행해야 정확하게 가능

숙련되는데 상당한 경험을 필요로 함

전문의라도 같은 사진에 대한 성숙에 대한 해석이 판독때마다 달라질 수 있음

# 목표

1

- 1) 골연령 예측 정확도를 높이기 위해 보다 정말하게 전처리.
- 2) 손목 및 손가락 관절 탐색을 위한 인공신경망 제작.

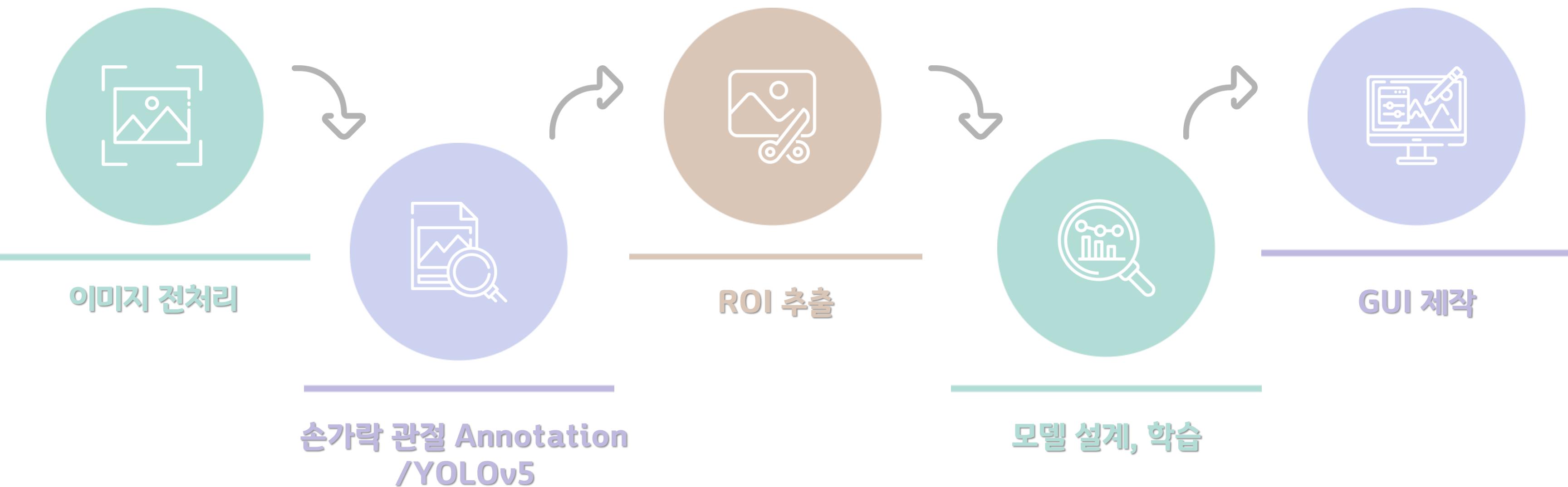
2

TJ-Net 인공신경망을 이용한 예측 모델 제작.

3

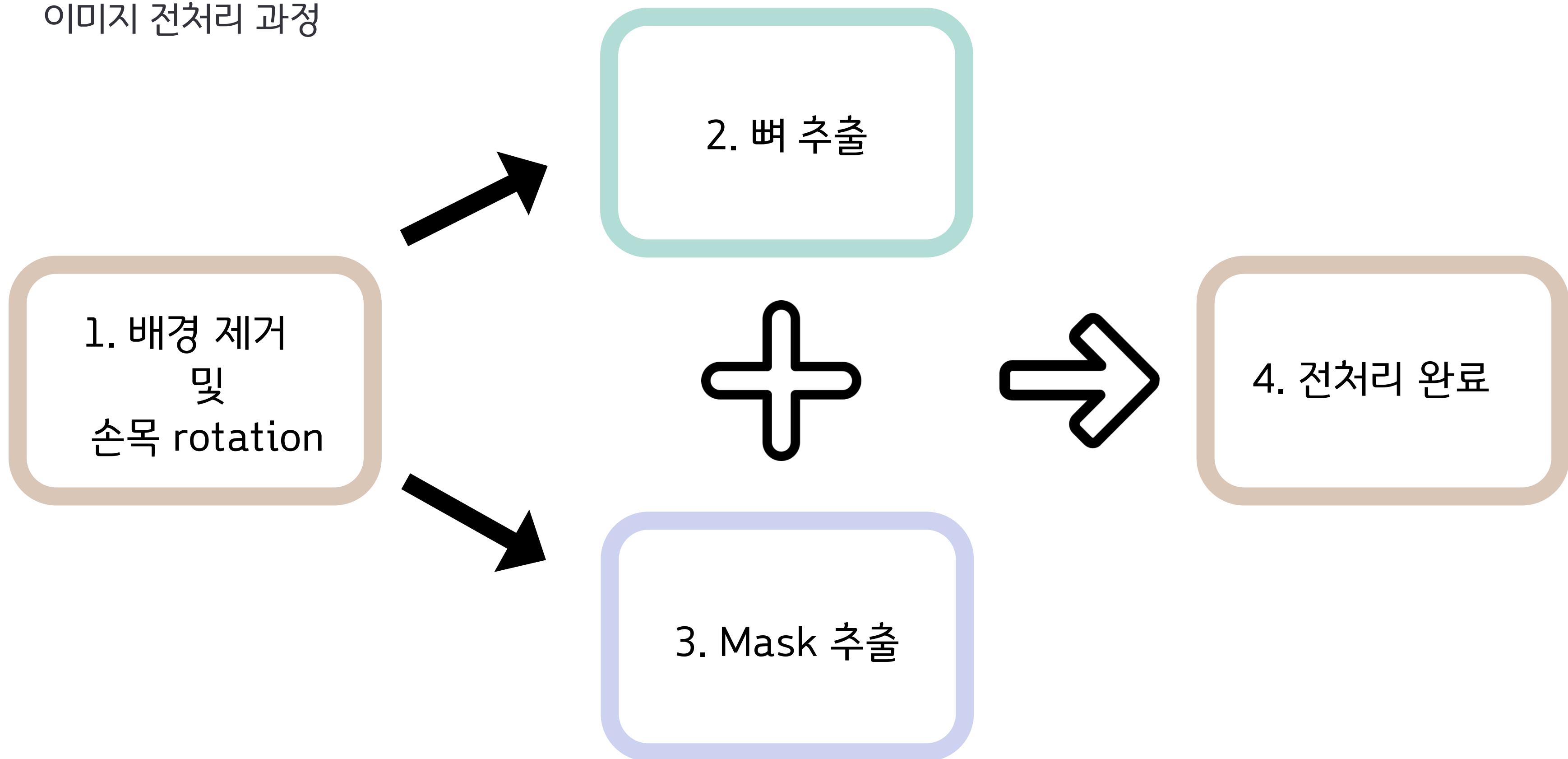
사용자 편의를 위한 PyQt5를 이용해 GUI 제작.

# 프로젝트 진행 순서



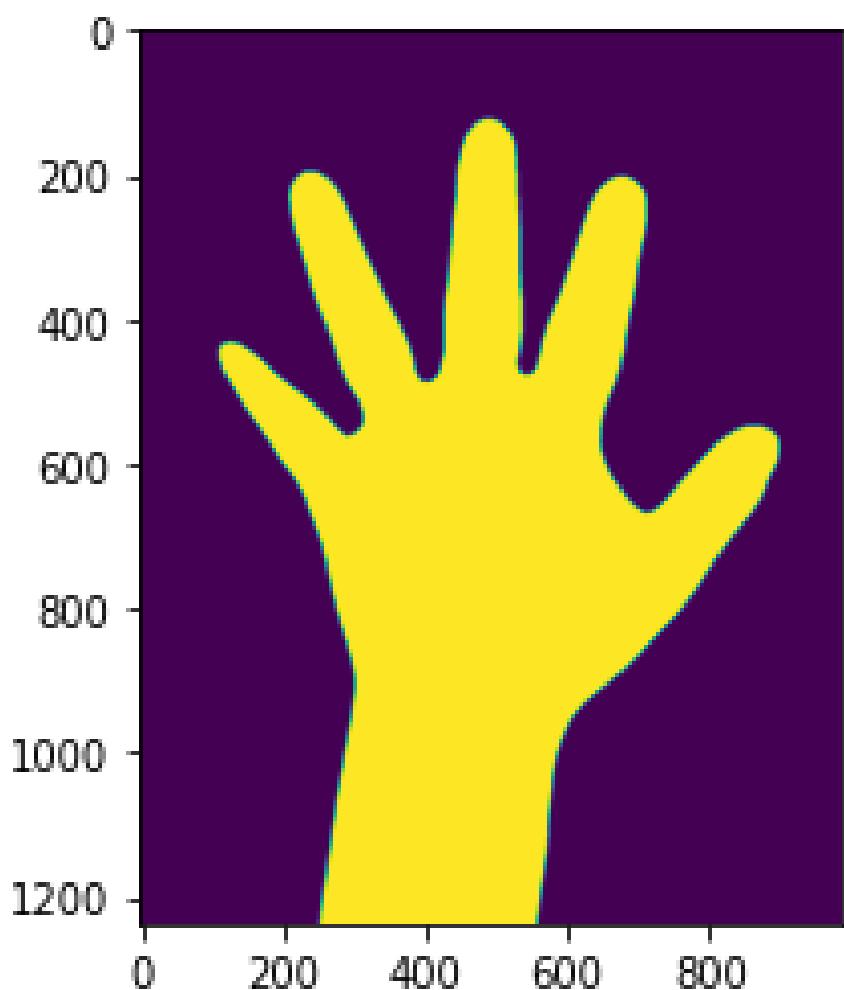
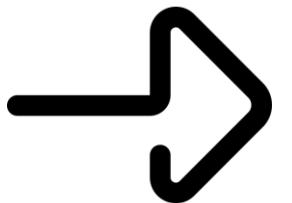
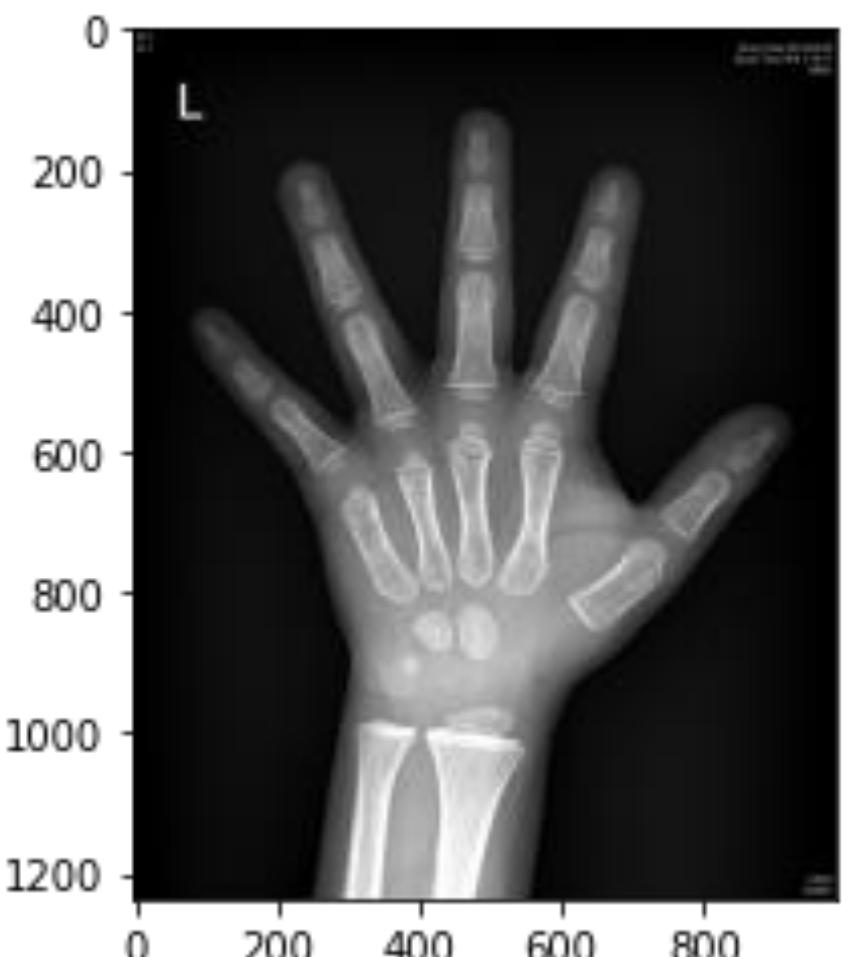
## Bone extraction

### 이미지 전처리 과정



### 1-1) 배경 제거

- 이미지 테두리에 x\_ray영상(검정)이 아닌 흰 테두리가 있는 경우를 위해 이미지 테두리 제거
- 손만 추출할 수 있도록 mask 생성

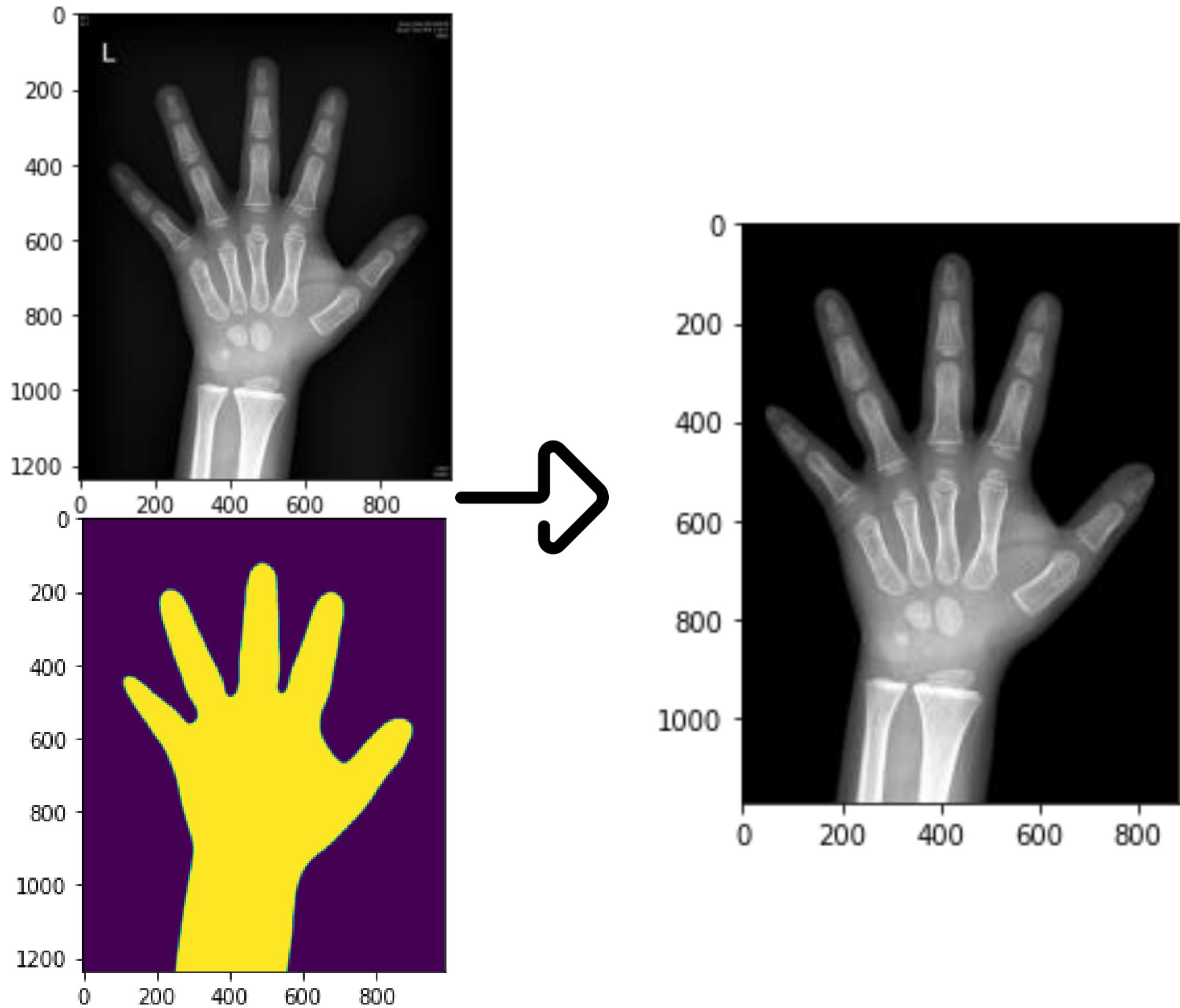


### 1-2) 배경 제거

- 이미지 원본과 이미지 마스크를 활용하여 비트연산을 통해 손을 제외한 배경 이미지

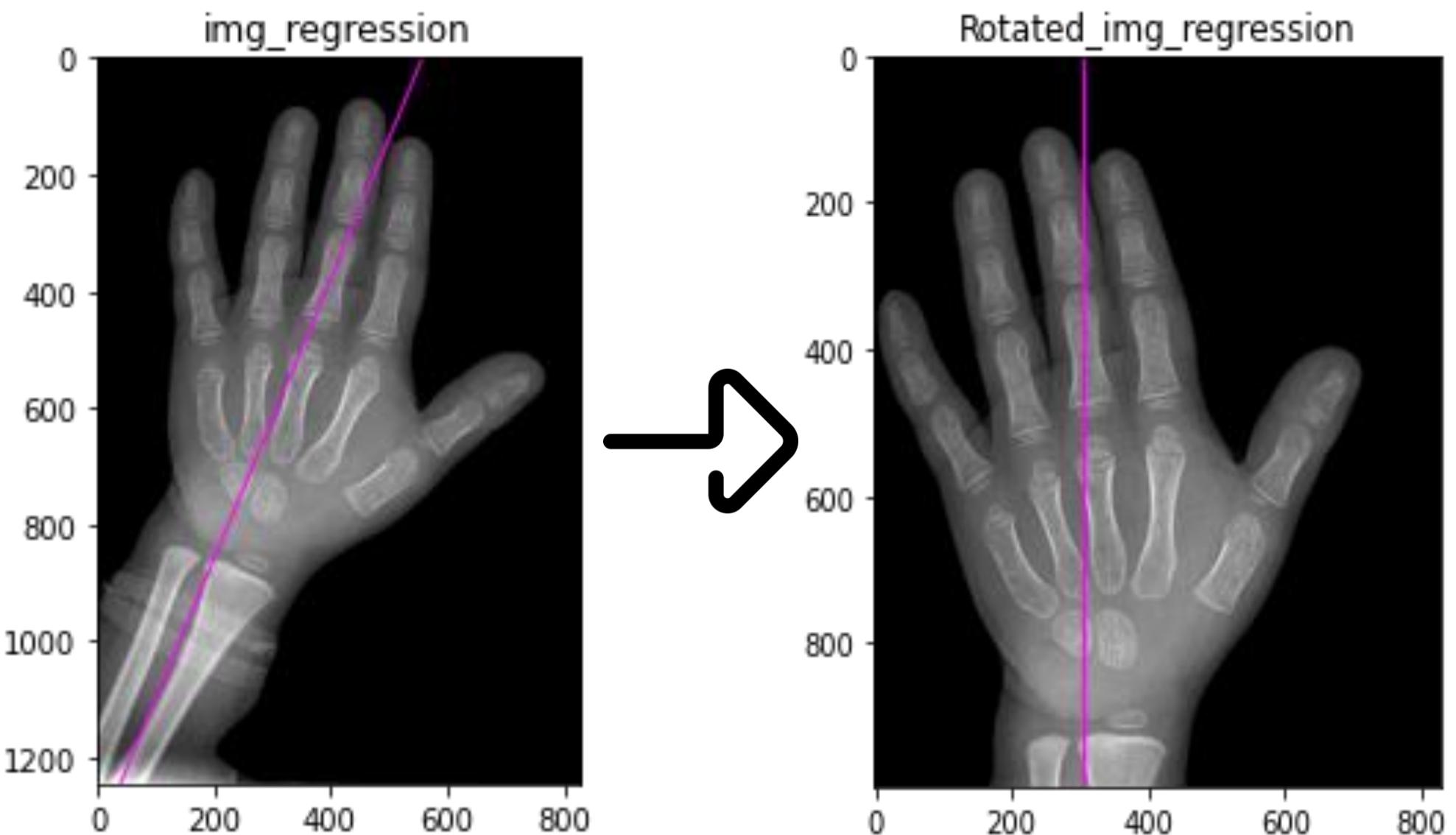
제거

\* 비트연산: 두 이미지에서 동일하게 값이 있는 부분만 출력



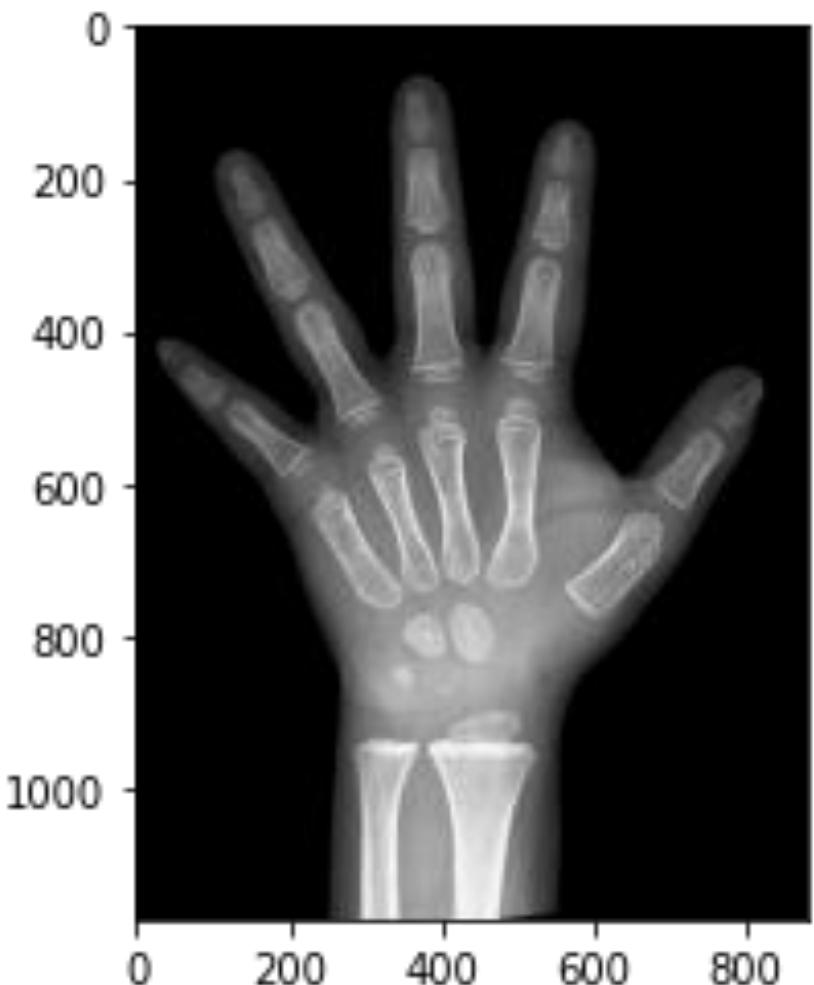
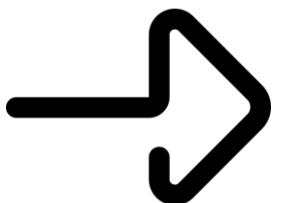
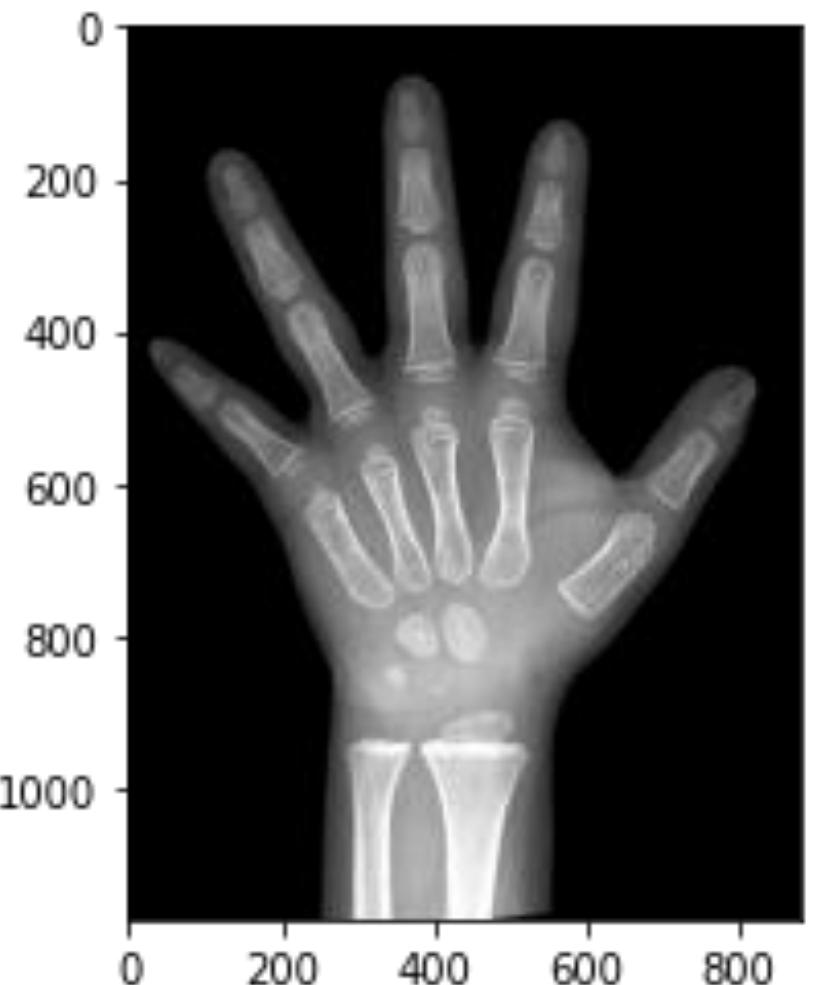
### 1-3) 손목 Rotation

- 손목 중간과 손목이 시작되는 지점을 x,y로 설정
- x,y로 선형회귀선을 그림
- 선형회귀선이 직각이 될 수 있도록 회전  
\* 회전할 각도:  
 $90 - (\text{atan2로 나온 각도})$



### 2-1) 뼈 강조 : 밝기 조정

- 이미지 밝기의 정도에 따라 가중치를 다르게하여 밝기값 조정

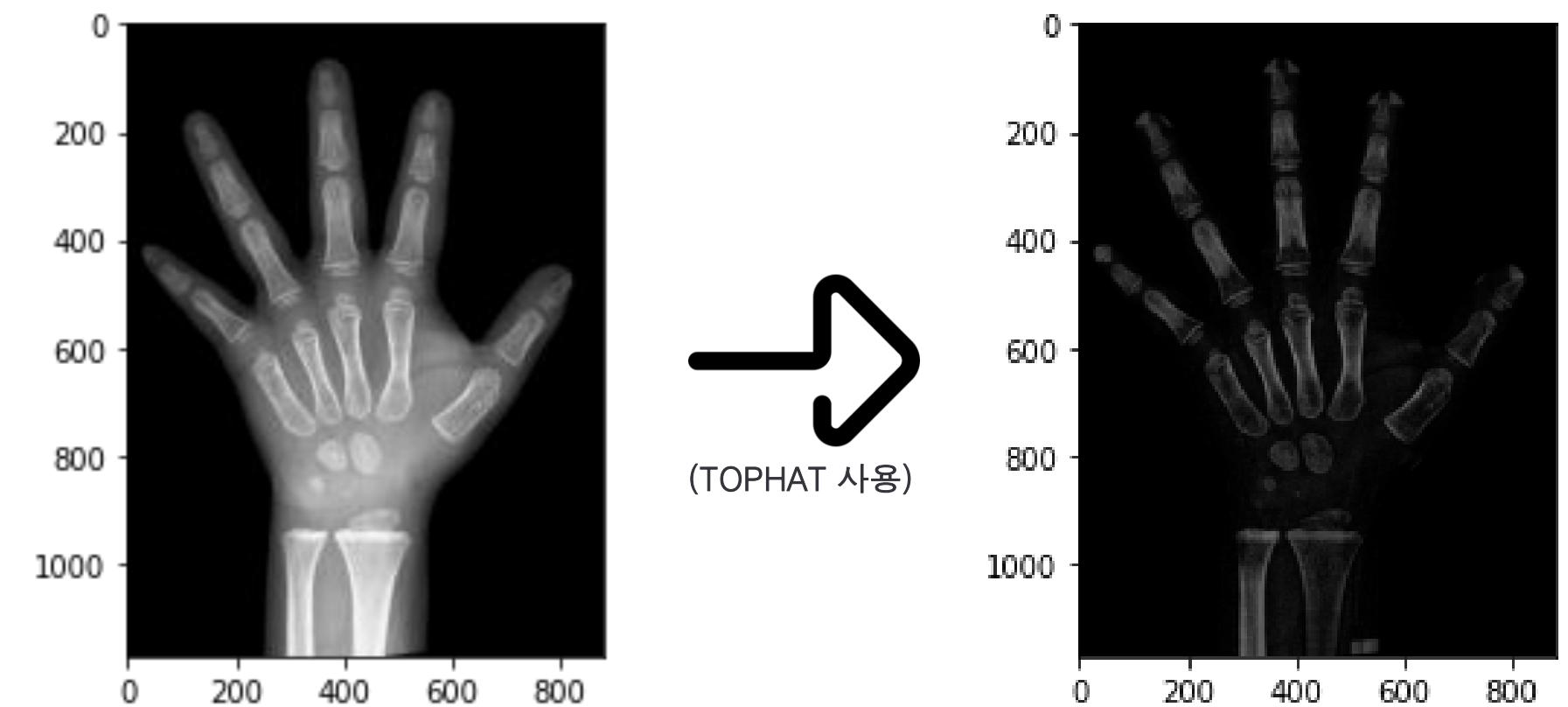


### 2-2) 뼈 강조 : 모폴로지 TOPHAT

- 모폴로지 연산 : 커널을 통하여 노이즈 제거, 구멍 채우기, 끊어진 선 이어 붙이기 등을 행하는 연산.
- 밝기 값이 크게 튀는 영역을 강조하기 위해 모폴로지 탑햇 사용

열림연산 = 침식 + 팽창

탑햇 = 원본 이미지 - 열림연산



### 2-2) 모폴로지 TOPHAT 추가 설명

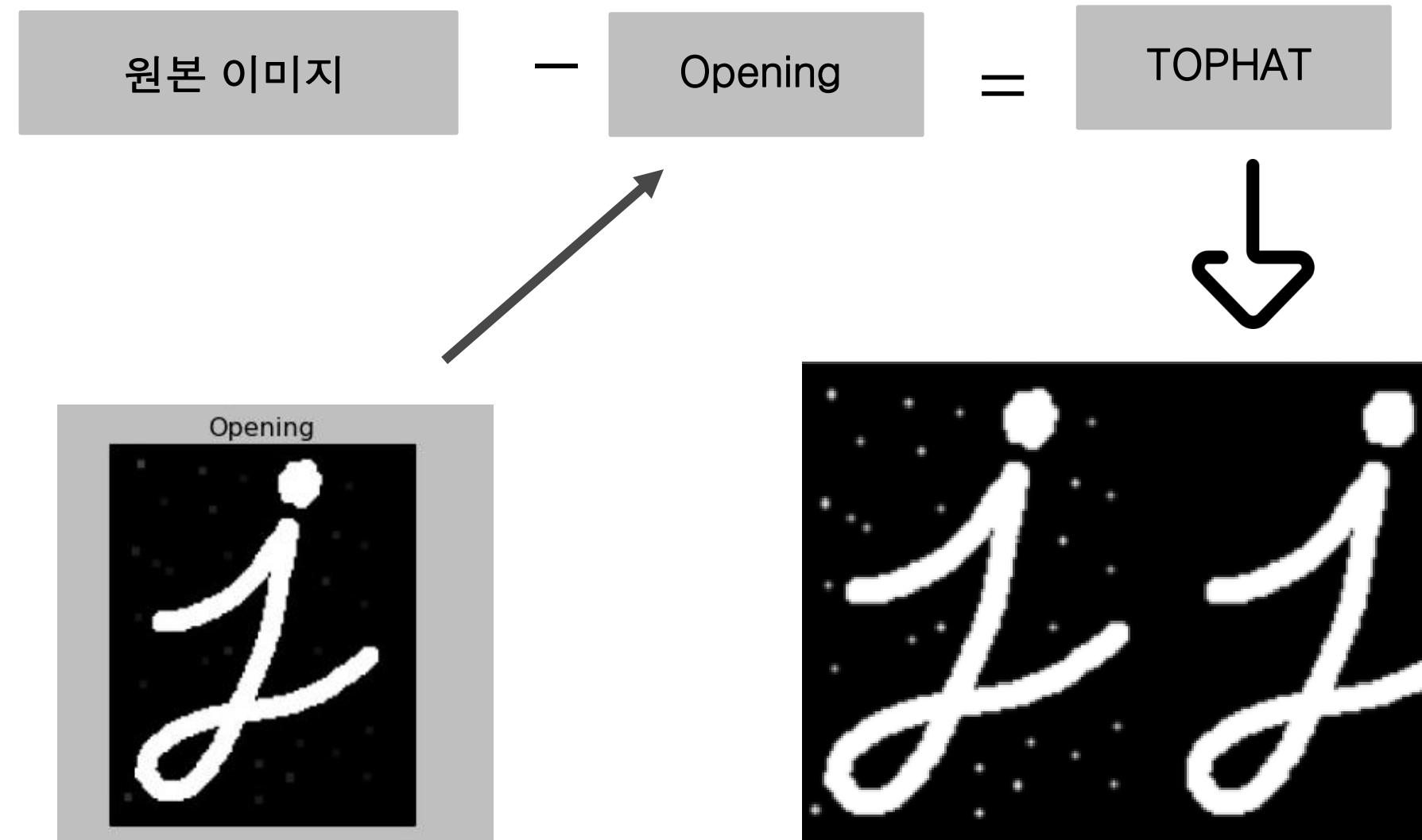
- 탑햇은 블랙햇과 같이 팽창 및 침식과 다르게 글씨의 굵기에는 변함이 없으면서 노이즈는 효과적으로 제거함.
- 탑햇은 밝기가 크게 튀는 부분이 강조됨.



침식



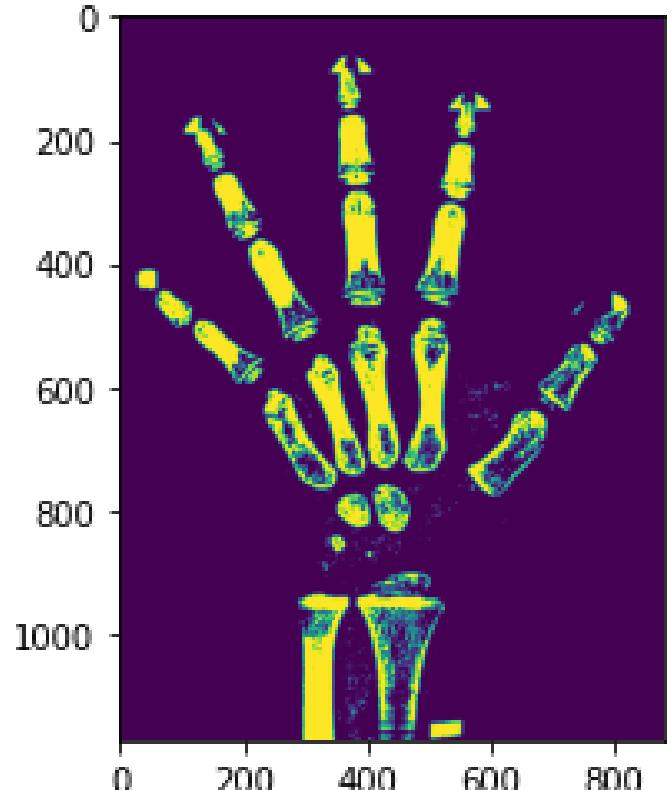
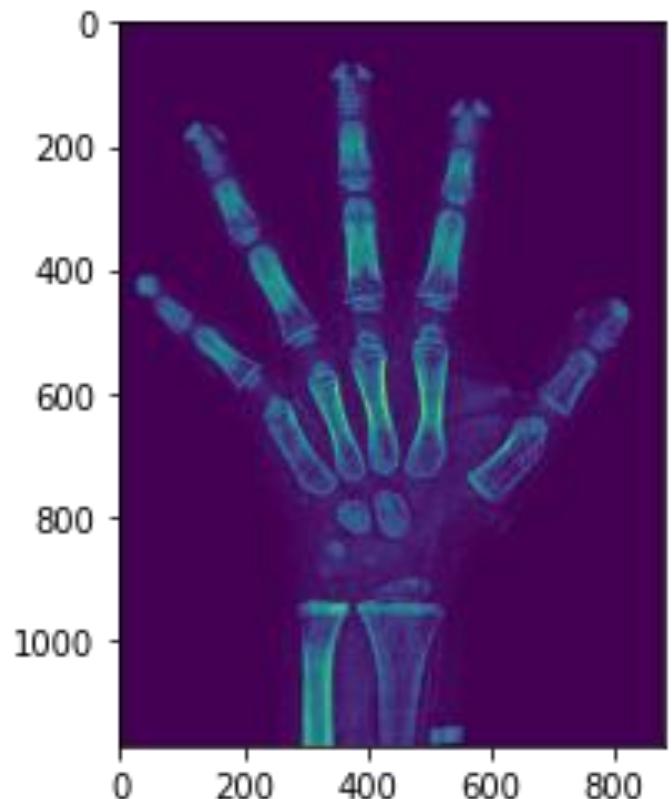
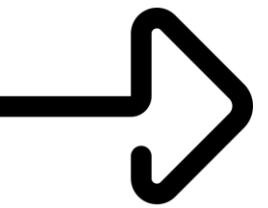
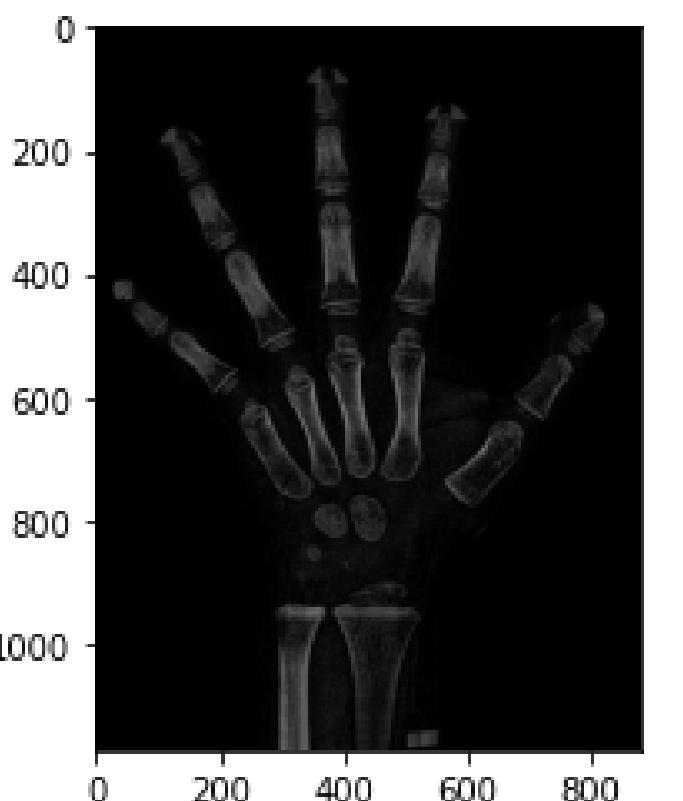
팽창



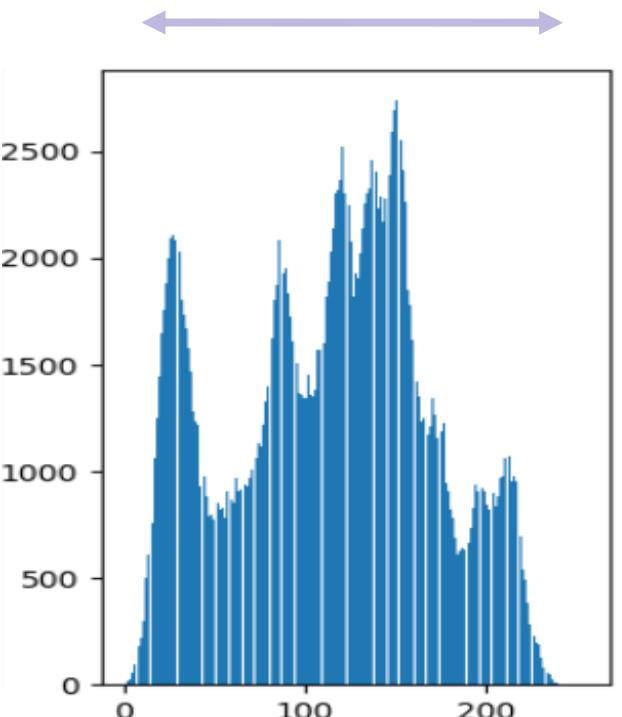
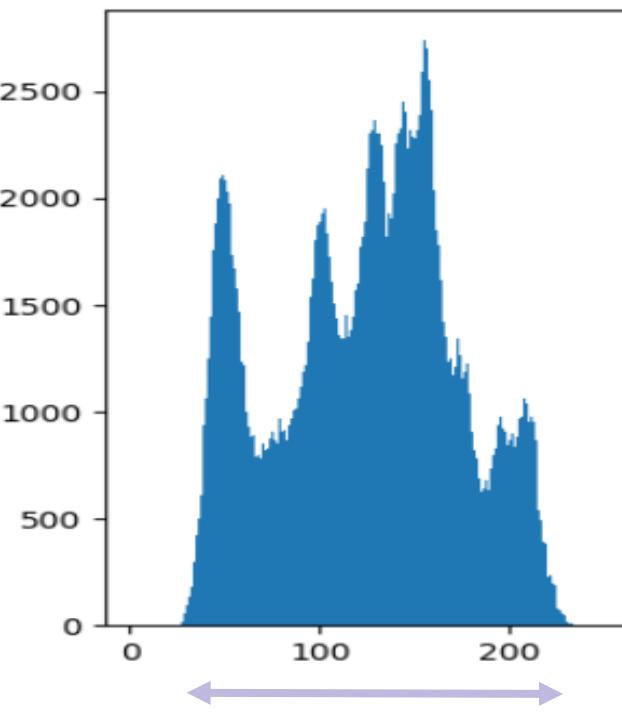
## Bone extraction

### 2-3) 뼈 강조 : 블러 / contrast

- 노이즈 최소화를 위해 블러처리
- 대비 조정 ( contrast 함수 사용 )  
\* 이미지 평균값에 따라 가중치를 다르게 설정



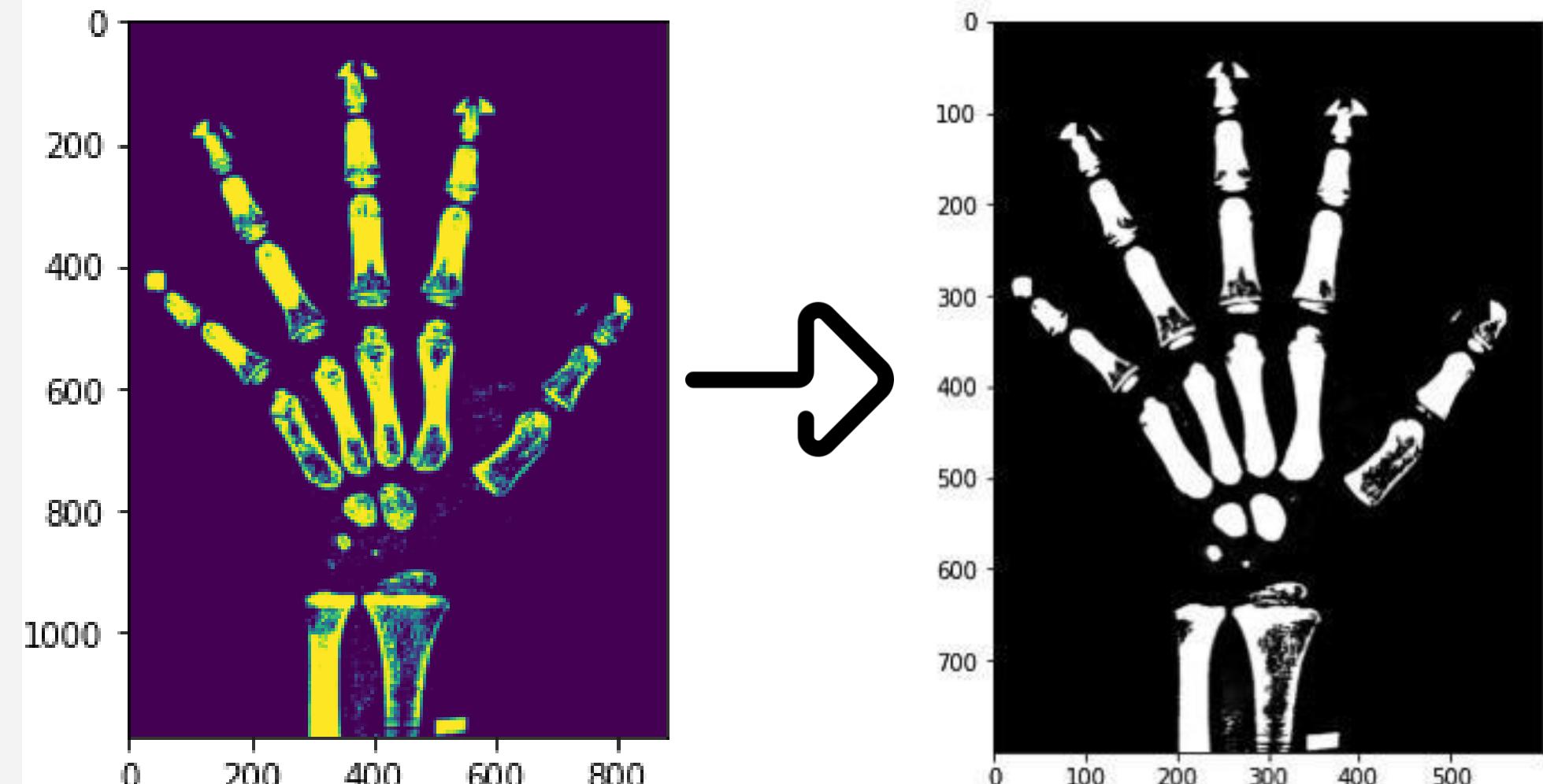
### 2-3) contrast 추가 설명



- contrast : 명암 대비 조정
- 흑백사진일 경우 밝기에 해당하는 명도값이 영상에 얼마나 들어있는지 표시
- 히스토그램이 고르게 분포해있으면 명암대비가 높은 것을 의미

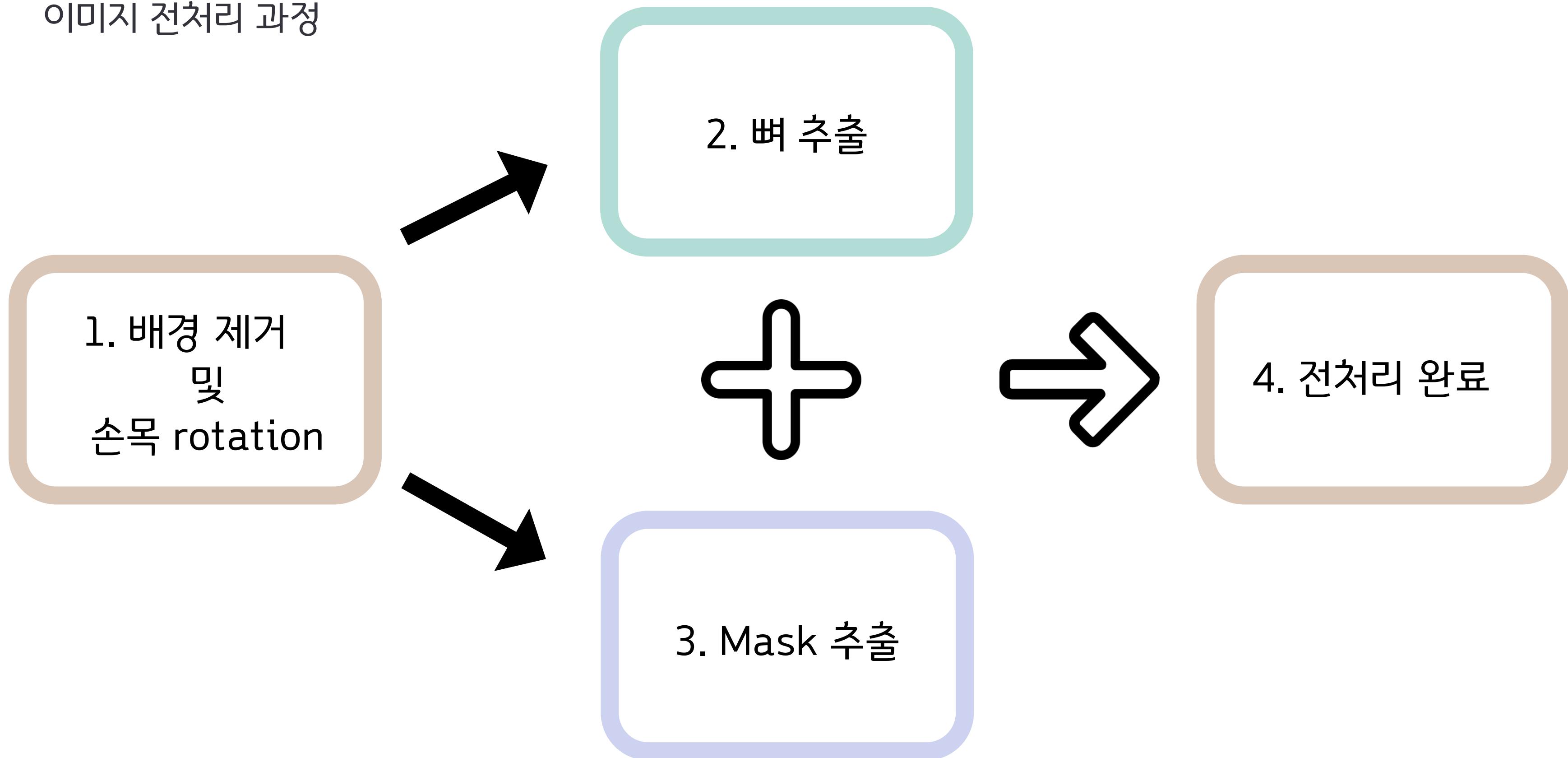
### 2-4) 뼈 강조: contours

- 뼈강조 전처리가 완료된 이미지에서 윤곽선을  
잡아주는 역할.
- object 검출을 위해 contours를 잡아줌.  
  
\*바깥쪽라인만 잡고 (흰색), 컨투어 채우기



## Bone extraction

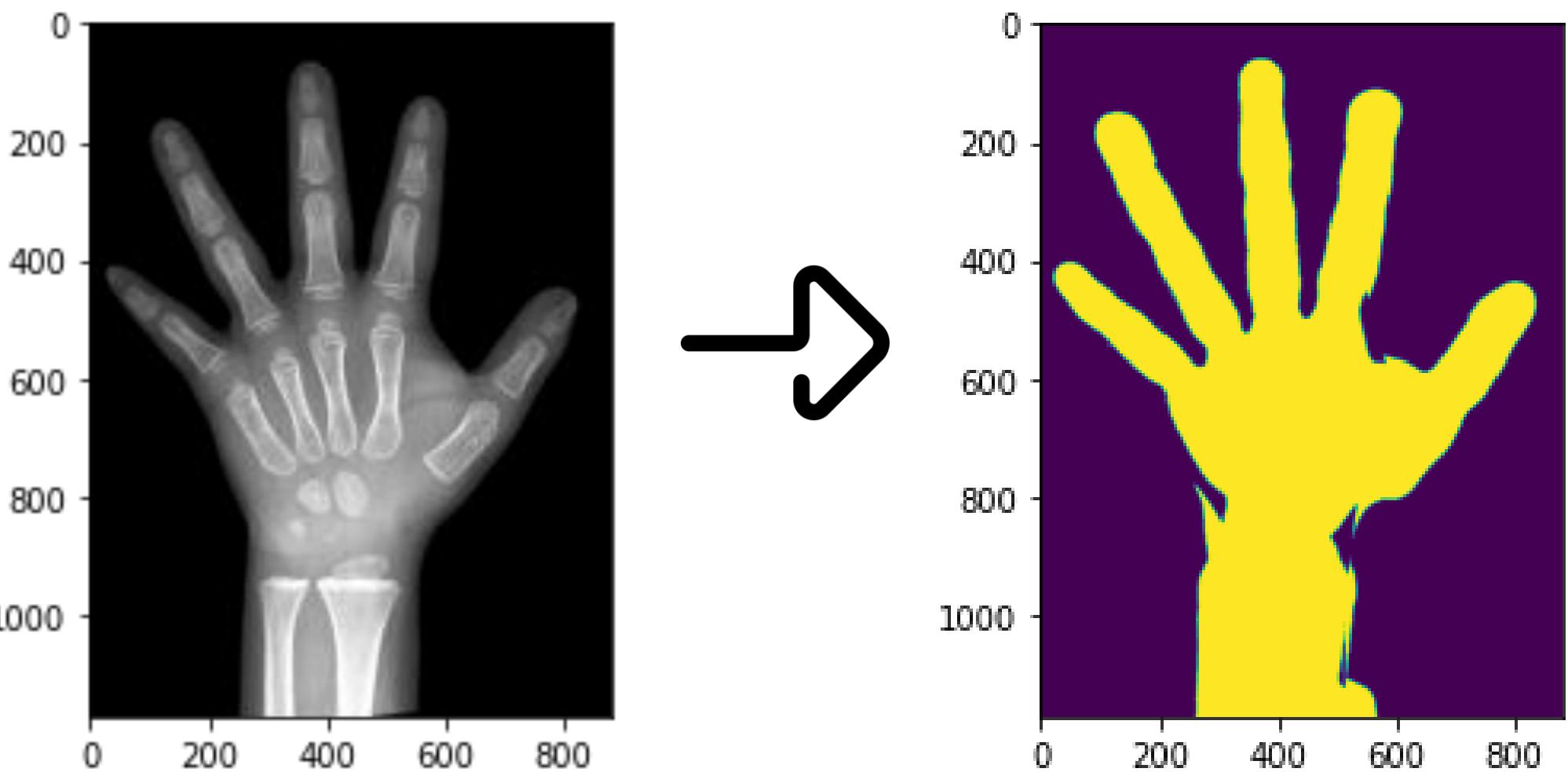
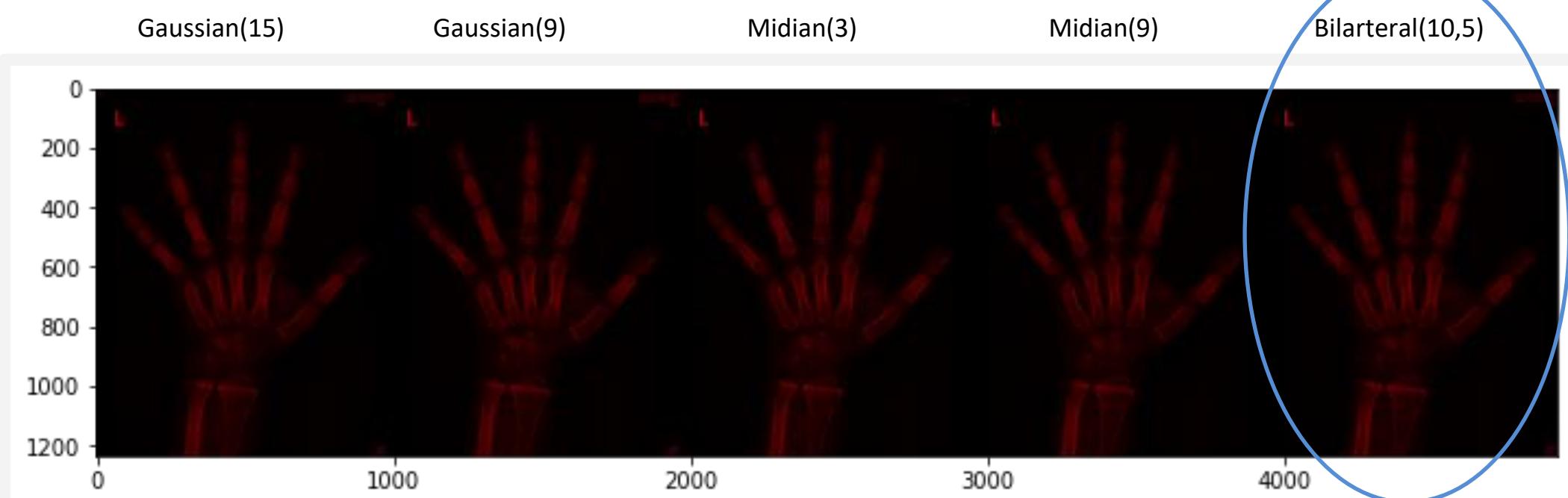
### 이미지 전처리 과정



## Bone extraction

### 3) Mask 추출

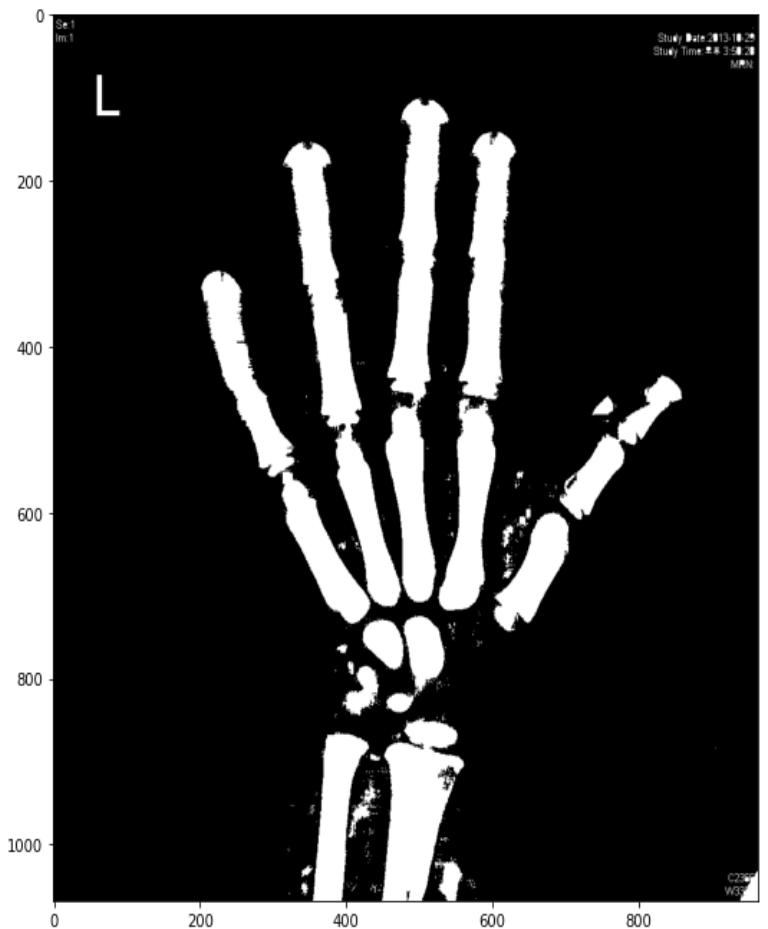
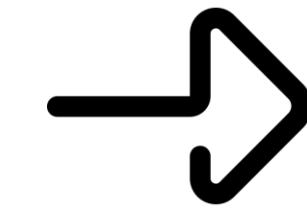
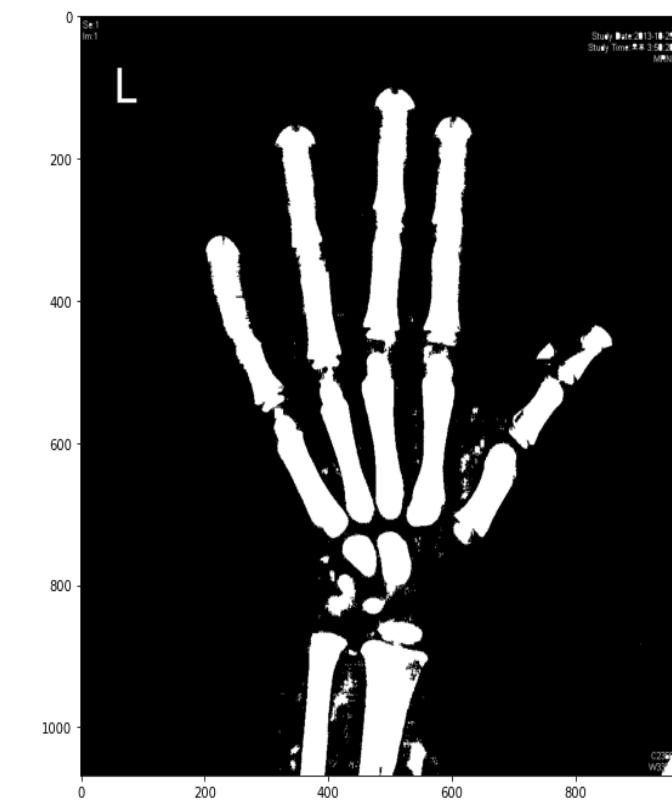
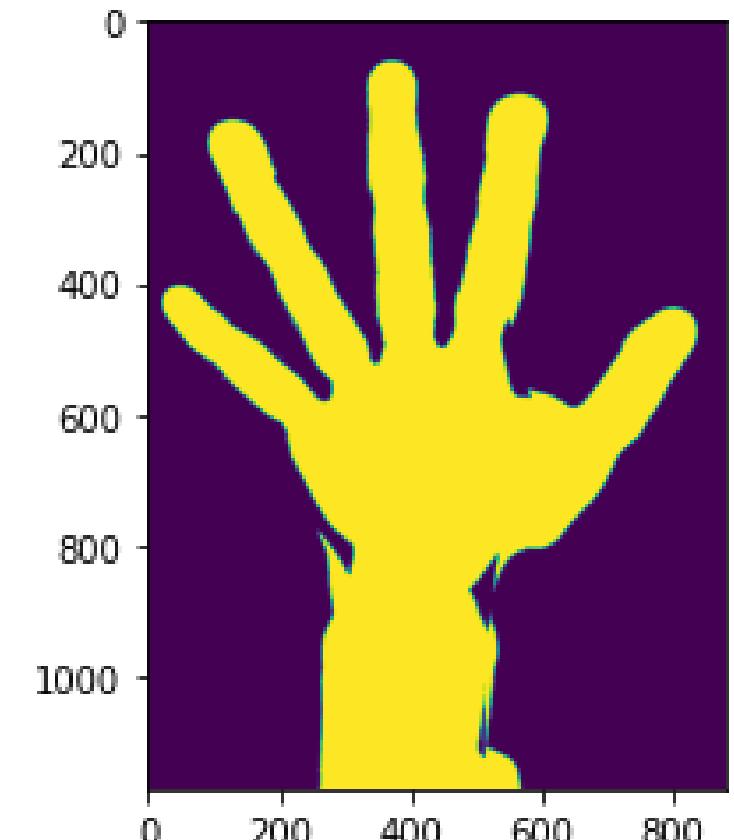
- 밝기, Lab, 모폴로지(탑햇), 블러필터, 정규화, 이진화, 컨투어를 통해 mask 생성



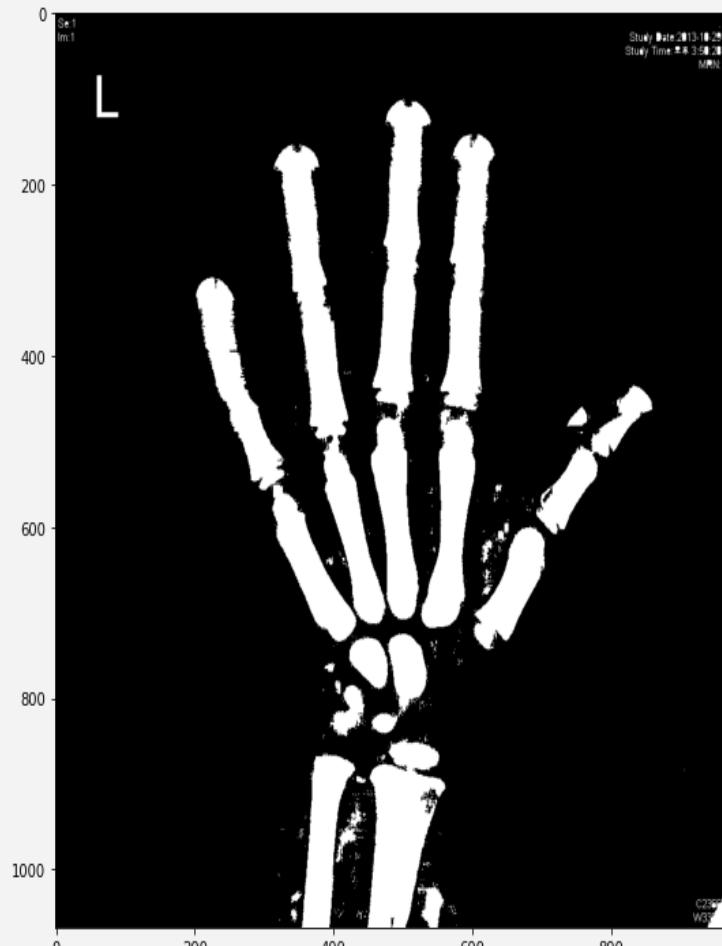
## Bone extraction

### 4) Mask & 뼈 강조 비트연산

- mask와 뼈강조 전처리 이미지를 비트연산  
\*비트연산: 값이 있는 영역만 추출

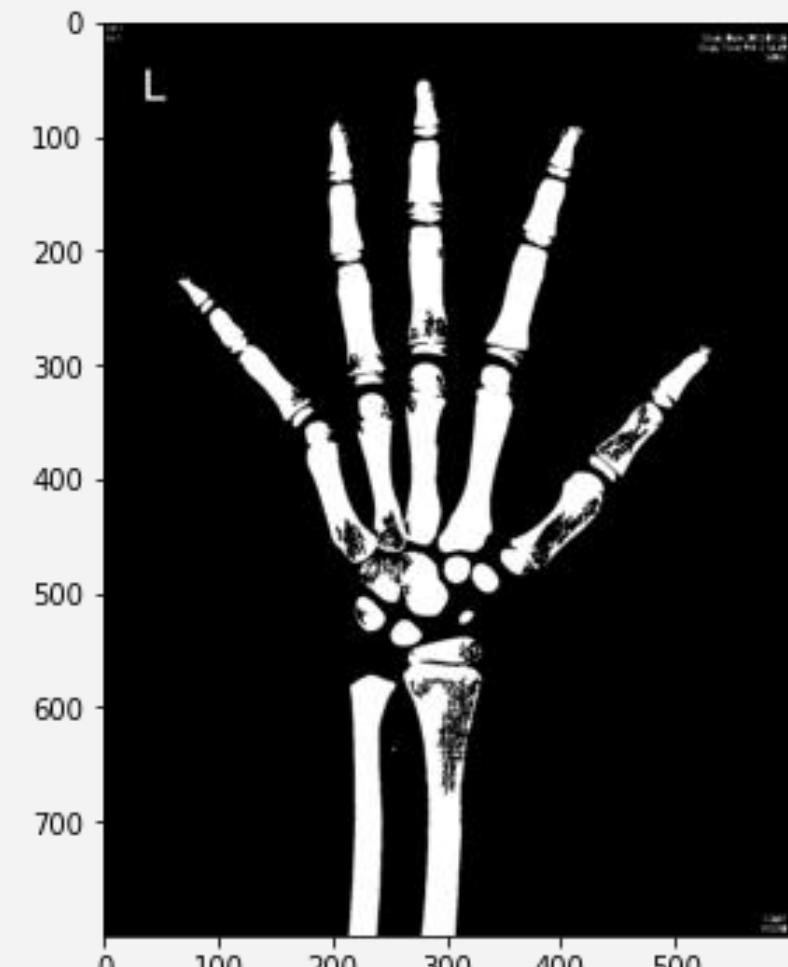
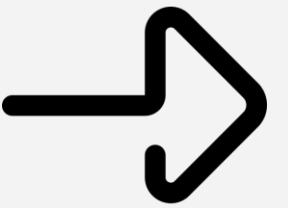


### 5) 전처리 과정의 값 설정



수치조정 전

- 뼈 강조를 위해 사용하는 모폴로지의 borderValue  
-> 55
  - mask를 잡는데 사용하는 모폴로지의 borderValue  
-> 60
  - bilateralFilter의 sigmacolor -> 50
  - bilateralFilter의 sigmaspace -> 25
- “ 경험적 수치 ” 조정으로 노이즈 최소화



수치조정 후

## ROI 추출, 객체 탐지 모델 탐색



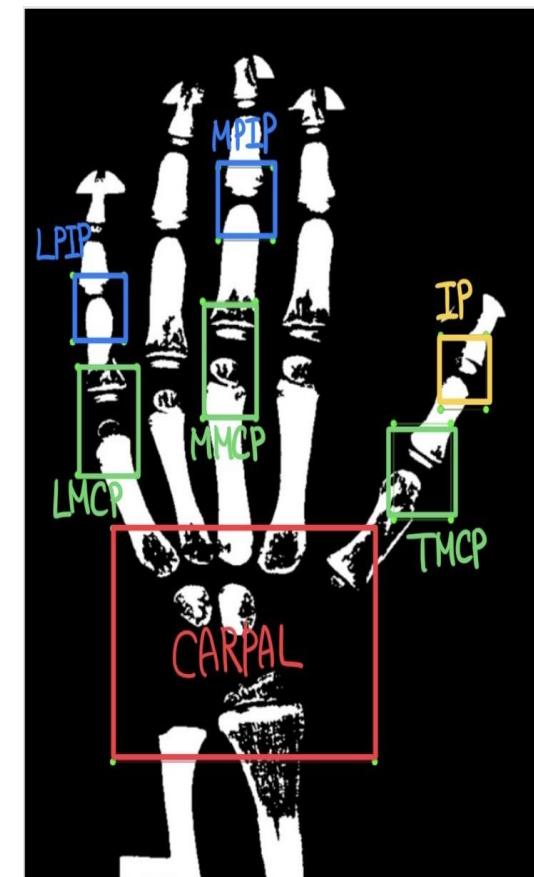
## YOLO 선택

- Mask R-CNN과 비슷한 성능을 보이는데, 접근 방식이 다름.  
Mask R-CNN의 경우 분류 모델을 사용하지만 Yolo의 경우 회귀 모델을 사용함.
- 율로는 처음 학습한 이후에 그에 맞는 객체를 전 이미지에서 예측하고 bounding box를 출력함.
- confidence score를 각각의 박스에 지정해서 쉽게 객체 탐지의 결과를 확인할 수 있음.

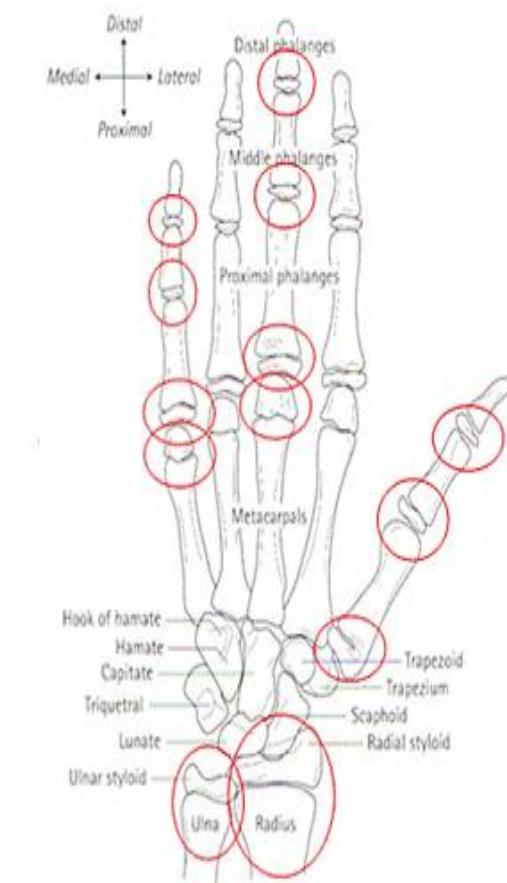
## 손가락 관절 Annotation

### 관절 7개를 뽑은 이유

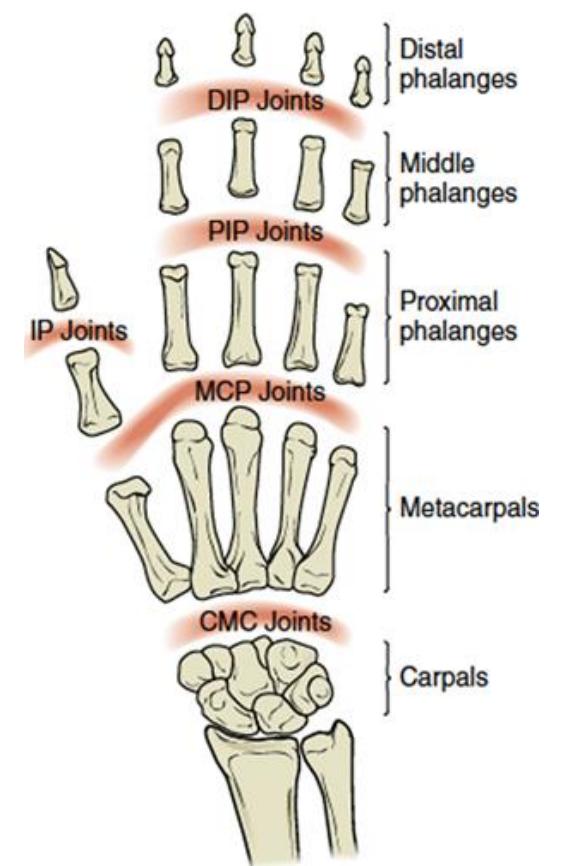
- TW3 기반
- 왼쪽 손과 손목사진을 찍은 후 7개의 수근골과 13개의 뼈 부위를 합쳐 뼈모양을 분석
- 제일 말단의 손가락 뼈와 중간 지골의 모양이 유사하여 하나로 인식
- 연구에서 따로 인식시킨 관절을 한번에 이미지로 학습



Annotation 7부분



참고 : TW3 판단 관절

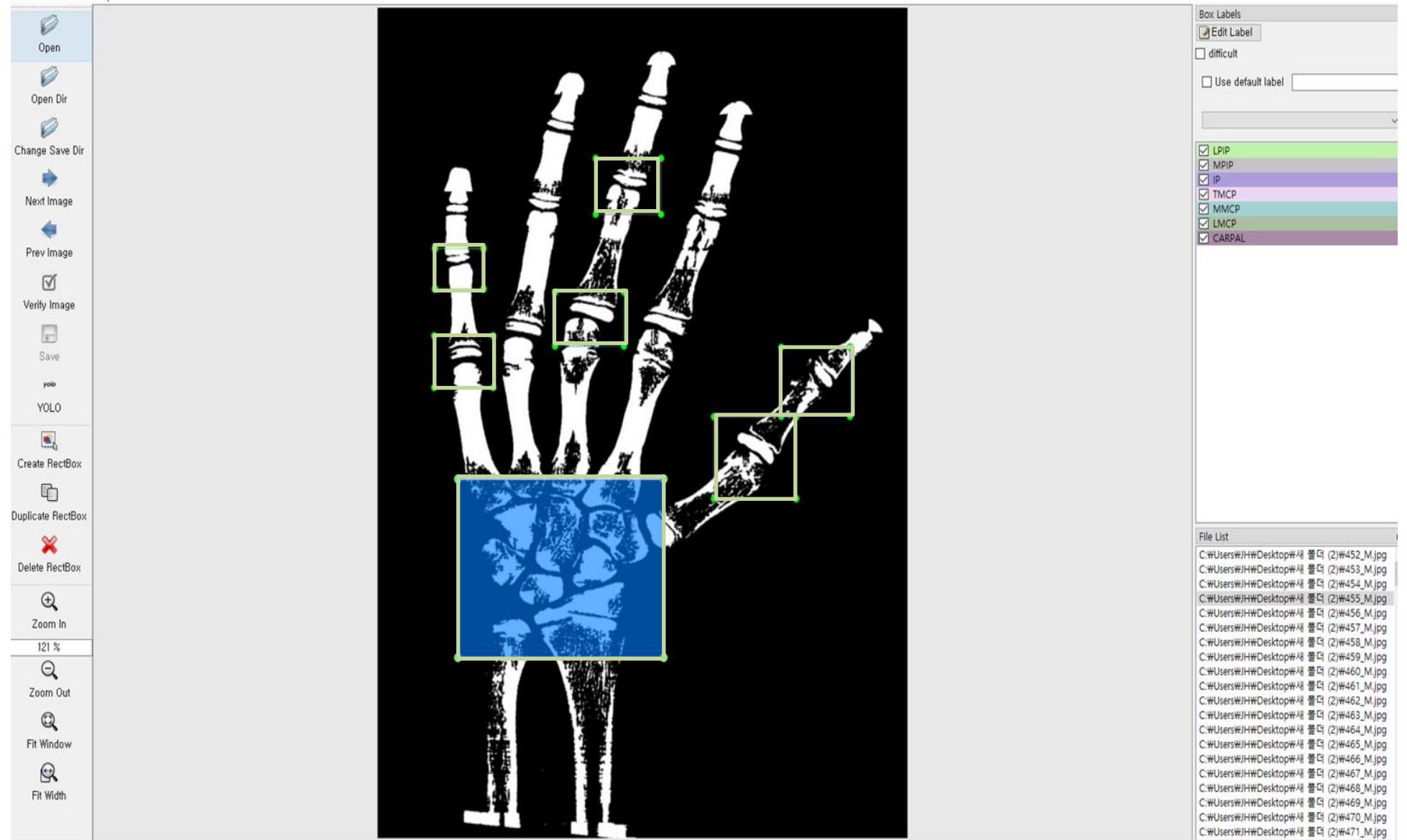
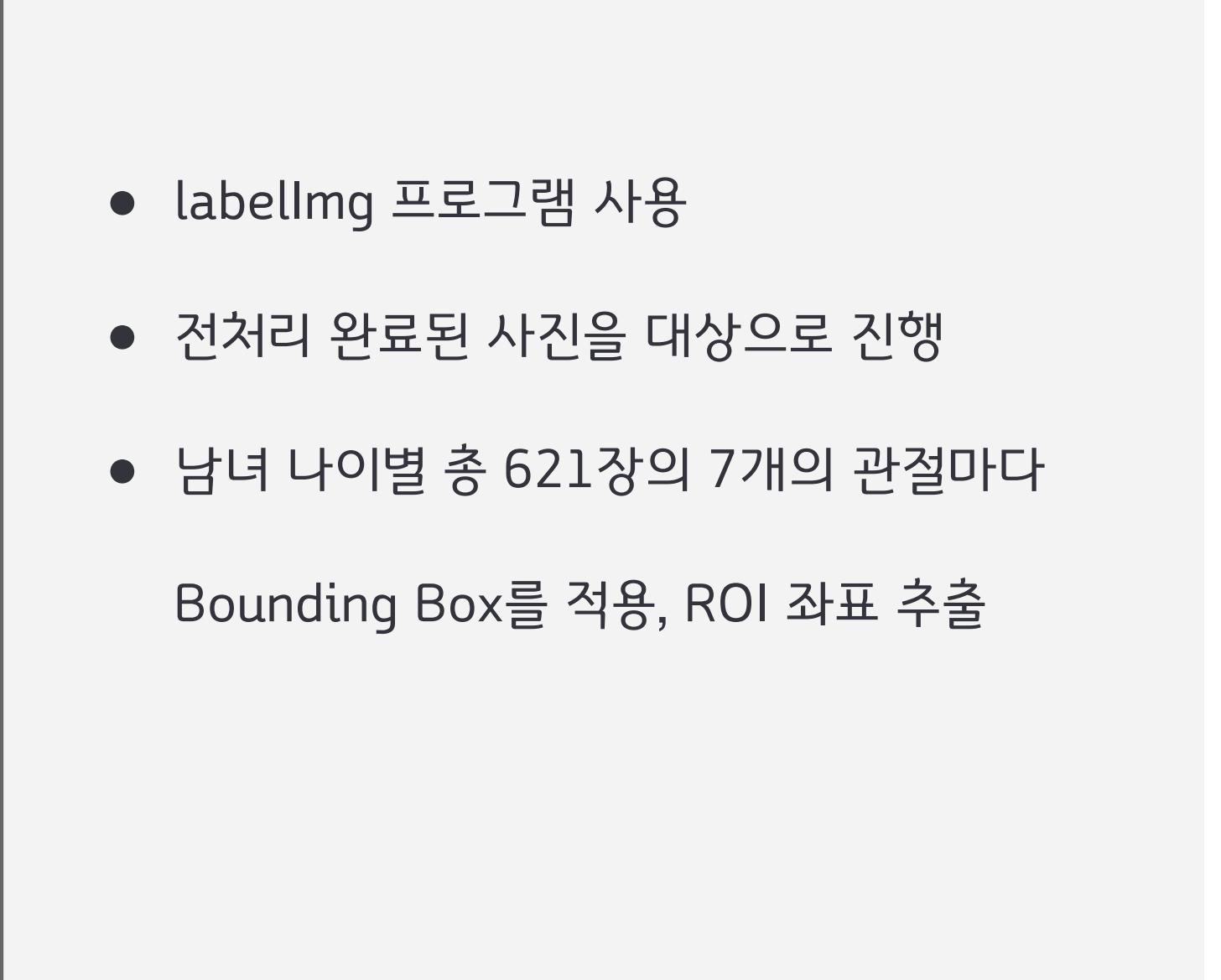


참고 : 뼈의 명칭

## Bounding box

- labellmg 프로그램 사용
- 전처리 완료된 사진을 대상으로 진행
- 남녀 나이별 총 621장의 7개의 관절마다

Bounding Box를 적용, ROI 좌표 추출

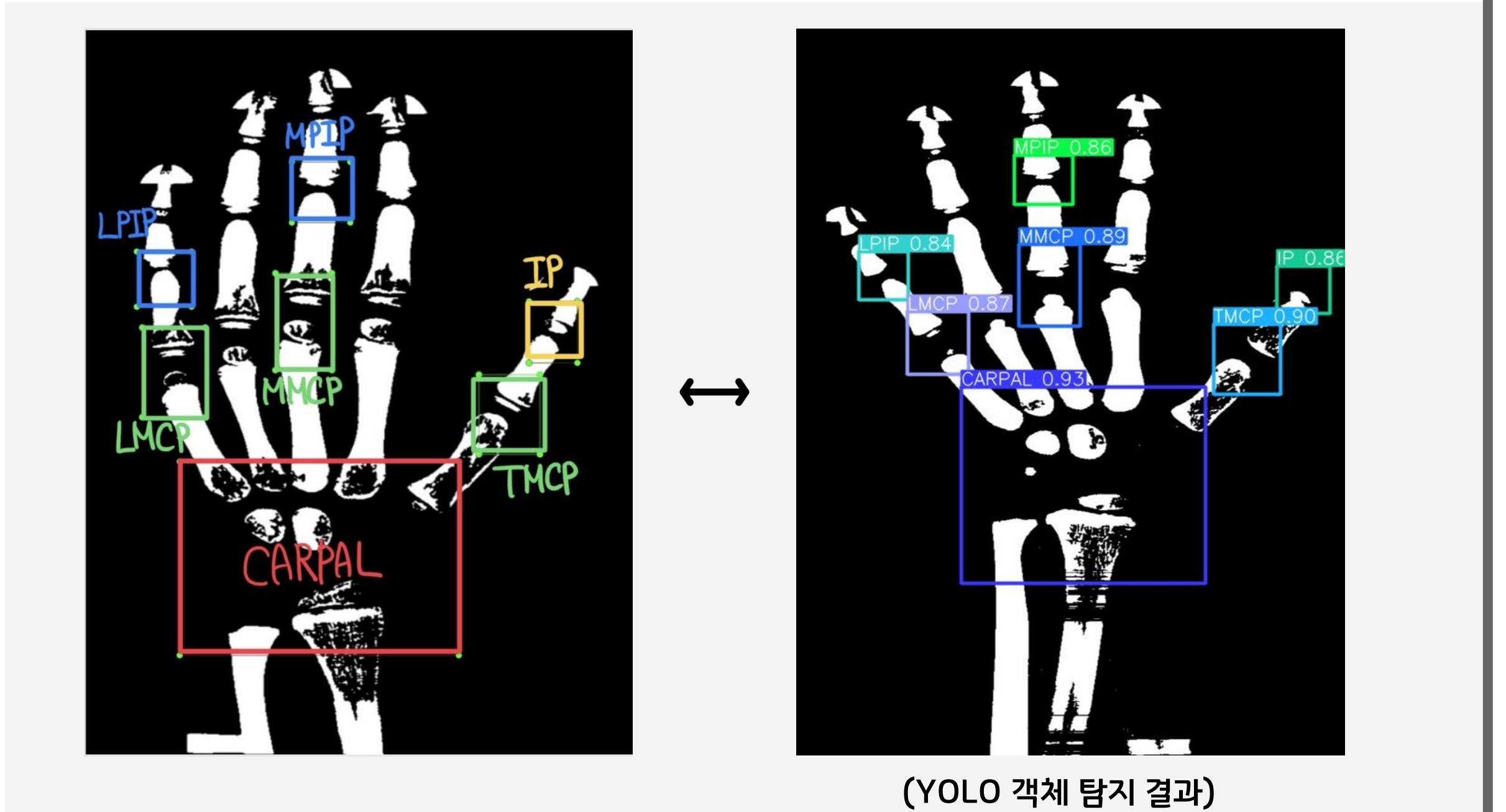


참고 : LabelImg

### YOLOv5 이용



- 총 621장의 사진
- Bounding box 라벨링한 사진을 학습
- 성별과 연령을 고려하여
- 8 : 2 비율로 학습과 검증 진행

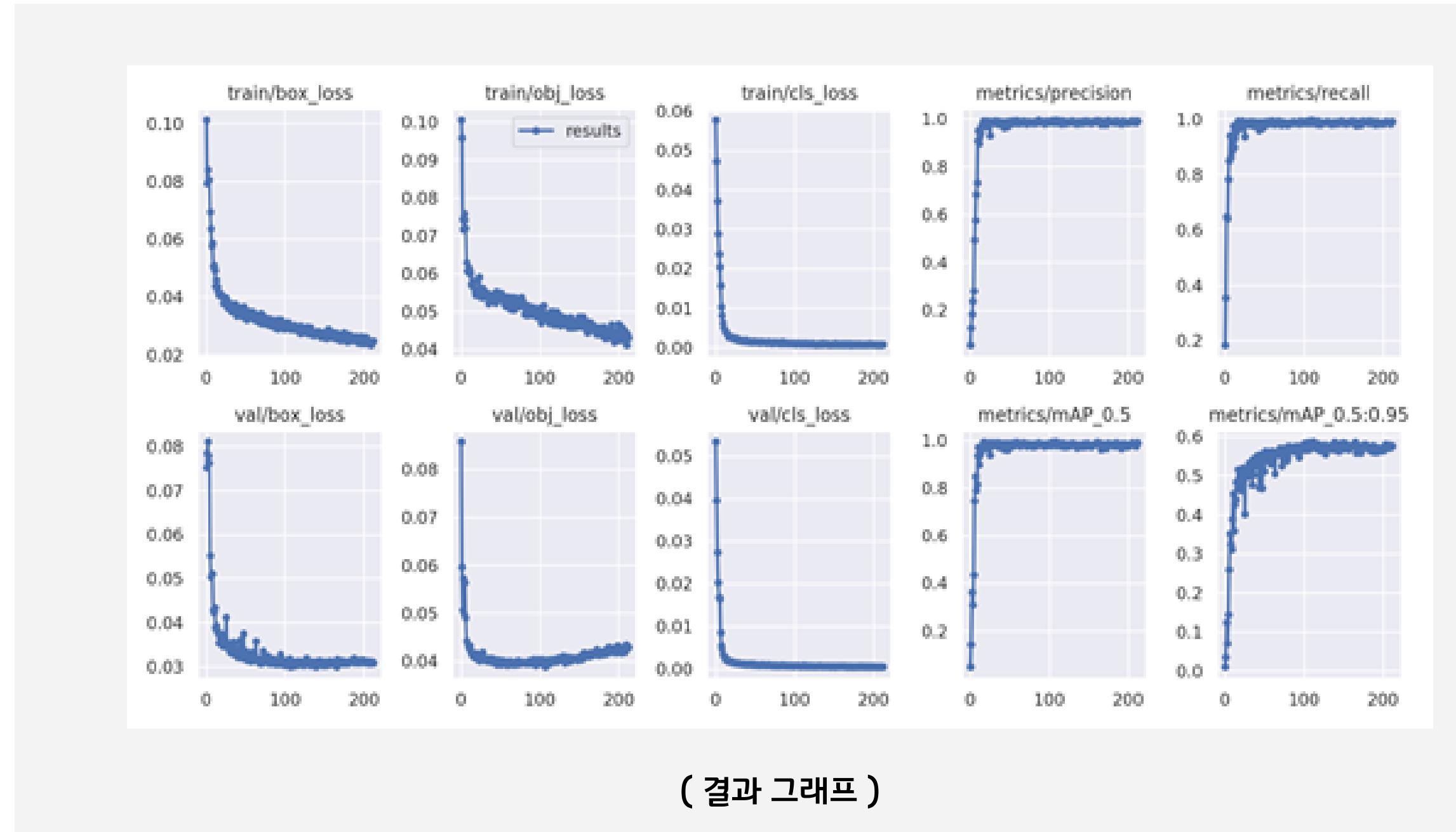


## YOLOv5 이용

## 최종 ROI 추출 모델 결과

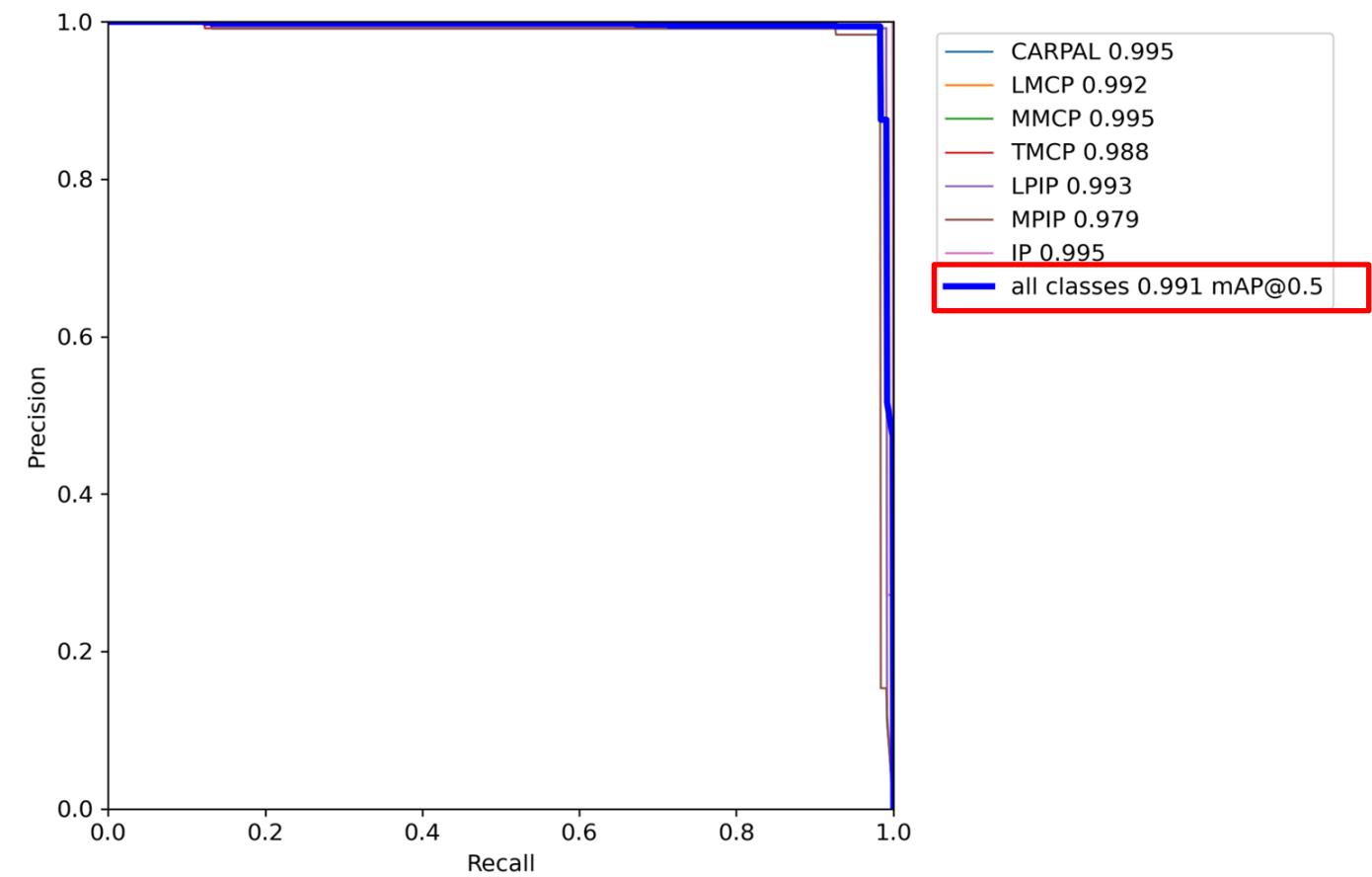


- 수행 결과 batch\_size=2, epoch=230에서 mAP@.5가 가장 높은 결과(0.991)가 도출함
- TJ-net모델에 적용하도록 데이터 구축에 이용



## YOLOv5 이용

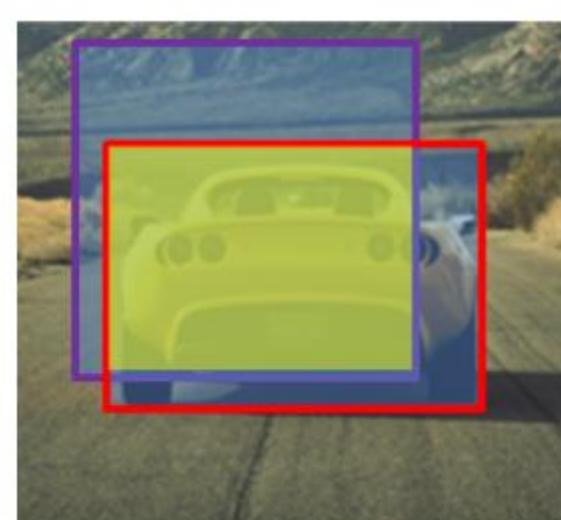
\*mAP : mean Average Precision



- mAP@.5 : Pascal VOC의 mAP 평가 방식.
- IoU > 0.5 인 detection을 True, 그 이하는 False로 평가함.

## IoU

- 검출하는 모델의 정확도를 측정하는 평가지표.
- 정답 바운딩 박스와 모델로부터 예측된 predicted bounding box 가 필요



Intersection over union (IoU)

$$= \frac{\text{size of intersection}}{\text{size of union}}$$

“Correct” if IoU  $\geq 0.5$

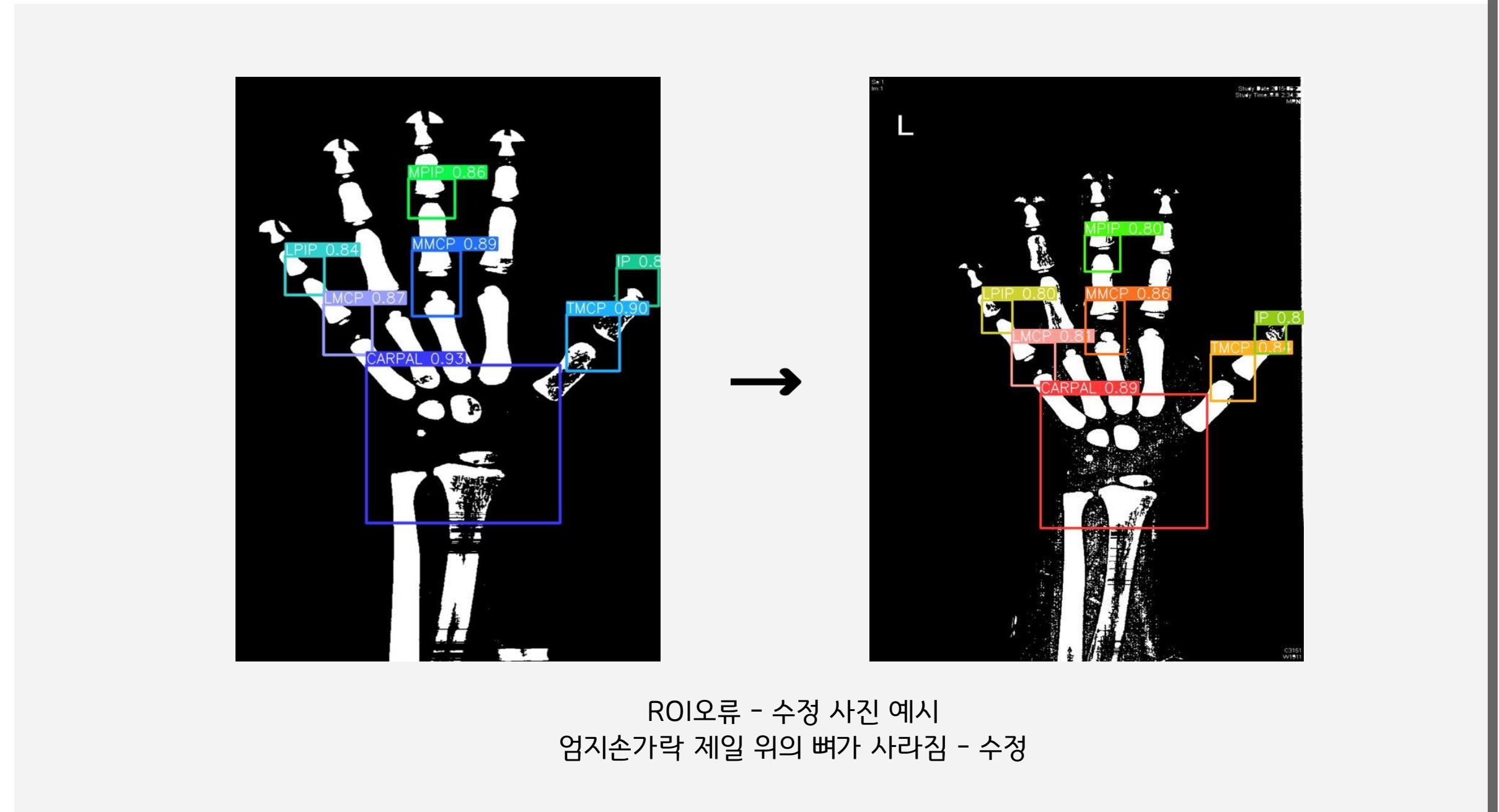
IoU 계산 방법

### 골연령 예측모델에 들어갈 데이터 검토



#### ROI 육안 검토

- 남자 3 IP부분 전처리 다시
- 남자 129 TMCP 노이즈로 추가 레이블링 됨
- 



## 훈련, 검증 데이터 분리

- 각 ROI마다 종속변수를 부여하기 위해 데이터프레임에 ROI의 파일 이름을 저장.
- ROI 이미지를 Input 가능한 크기로 바꾸고 나이, 성별별 이미지 수의 차이를 고려해 나이와 성별을 기준 7:3 비율로 계층적 샘플링하여 학습, 검증 데이터셋 구축.

NO	gender	Group	REGI	DCT_CODE	BD	PD	OD	DIG_AGE	HEIGHT	WEIGT	BMI	BONE	BA_1	BA_2	BA_total	BA_zscore	CARPAL	LMCP	MMCP	TMCP	LPIP	MPIP	IP	ST
0	1	0	1568	8110925	1	2013-08-19	2016-08-01	2016-08-01	2.953425	88.9	11.3	14.3	39.0	3.25	3.25	3.250	-1.591672	1_F.jpg	1_F.jpg	1_F.jpg	1_F.jpg	1_F.jpg	1_F.jpg	3
1	2	0	440	5988227	2	2008-10-27	2011-10-25	2011-10-25	2.994521	95.9	13.5	14.7	21.0	1.75	1.75	1.750	-1.966593	2_F.jpg	2_F.jpg	2_F.jpg	2_F.jpg	2_F.jpg	2_F.jpg	3
2	3	0	1724	8292016	1	2013-08-08	2016-08-11	2016-08-11	3.010959	88.2	13.0	16.8	28.5	2.75	2.00	2.375	-1.810376	3_F.jpg	3_F.jpg	3_F.jpg	3_F.jpg	3_F.jpg	3_F.jpg	3

```
len(train_df), len(val_df)
(865, 371)
```

```
len(train_y), len(val_y)
(865, 371)
```

```
[<KerasTensor: shape=(None, 224, 224, 3) dtype=float32 (created by layer 'input_1')>,
<KerasTensor: shape=(None, 75, 75, 3) dtype=float32 (created by layer 'input_2')>,
<KerasTensor: shape=(None, 75, 75, 3) dtype=float32 (created by layer 'input_3')>,
<KerasTensor: shape=(None, 75, 75, 3) dtype=float32 (created by layer 'input_4')>,
<KerasTensor: shape=(None, 75, 75, 3) dtype=float32 (created by layer 'input_5')>,
<KerasTensor: shape=(None, 75, 75, 3) dtype=float32 (created by layer 'input_6')>,
<KerasTensor: shape=(None, 75, 75, 3) dtype=float32 (created by layer 'input_7')>,
<KerasTensor: shape=(None, 1) dtype=float32 (created by layer 'input_8')>]
```

## TJ-Net 모델

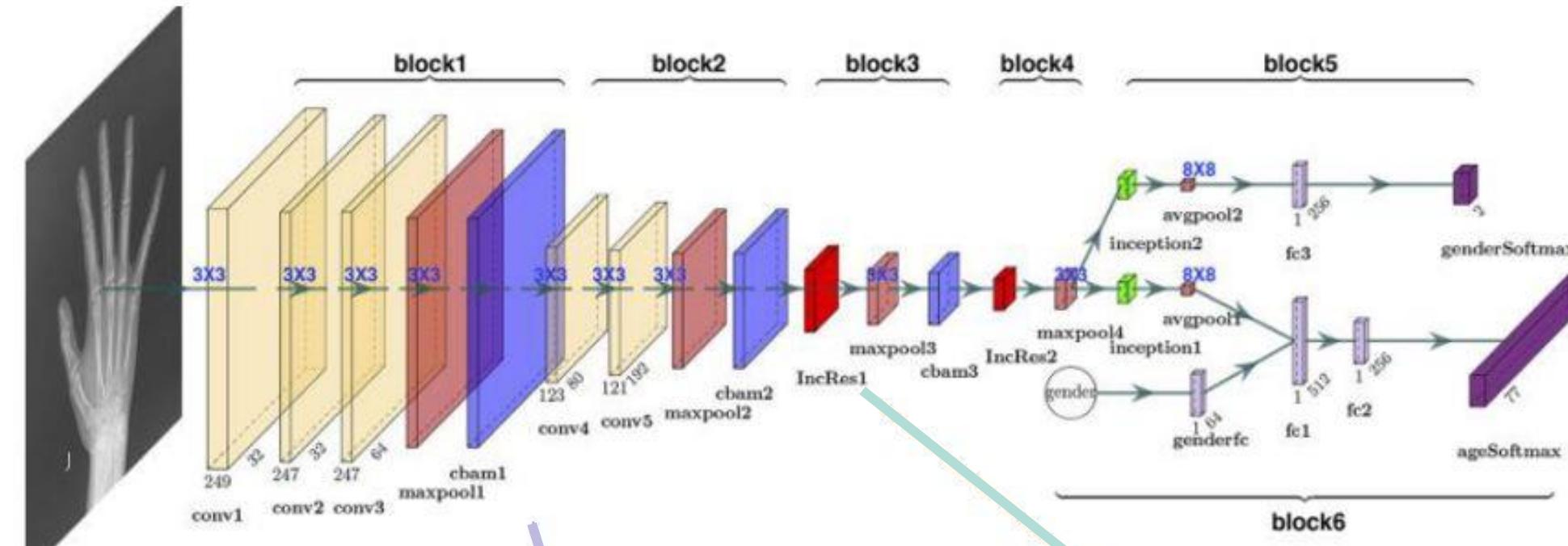


Figure 2. Architecture of TjNet.

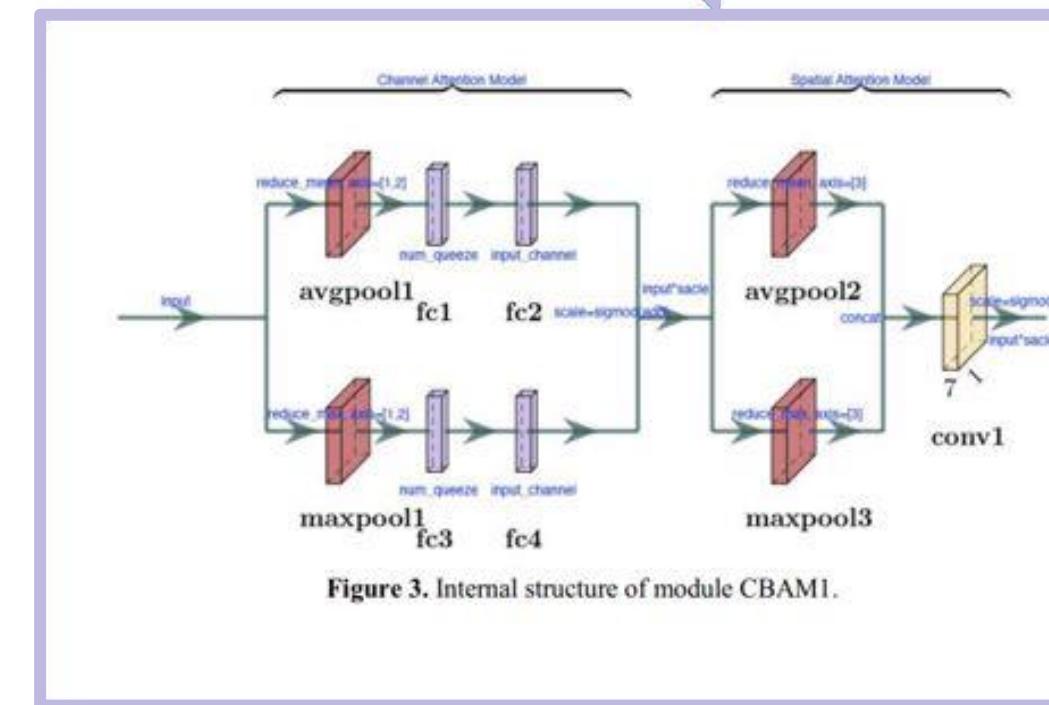


Figure 3. Internal structure of module CBAM1.

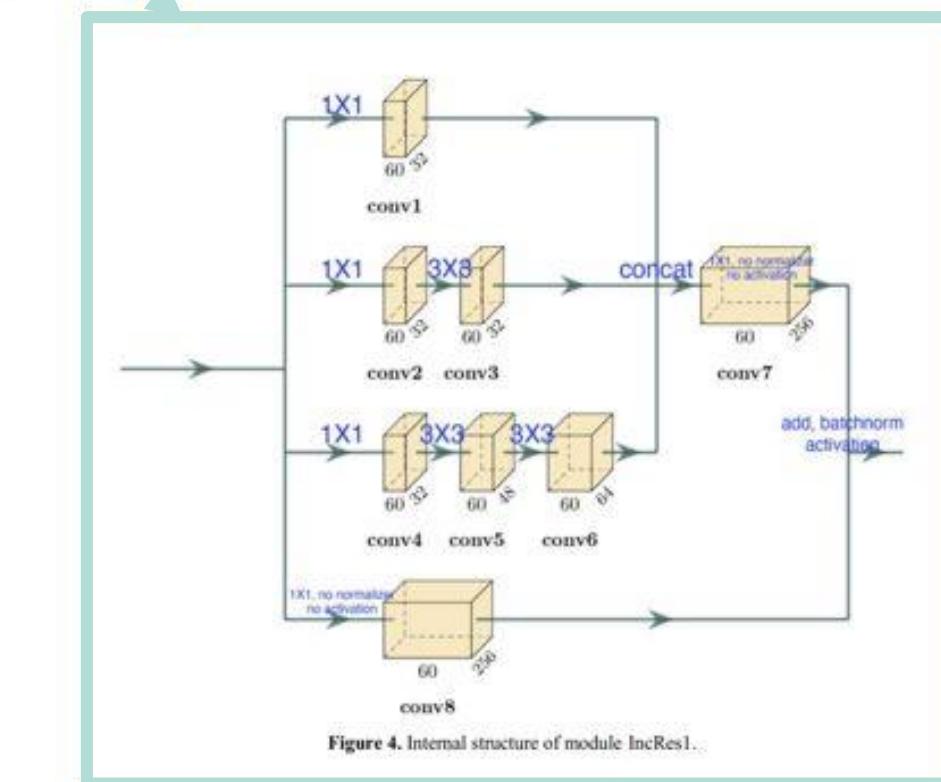
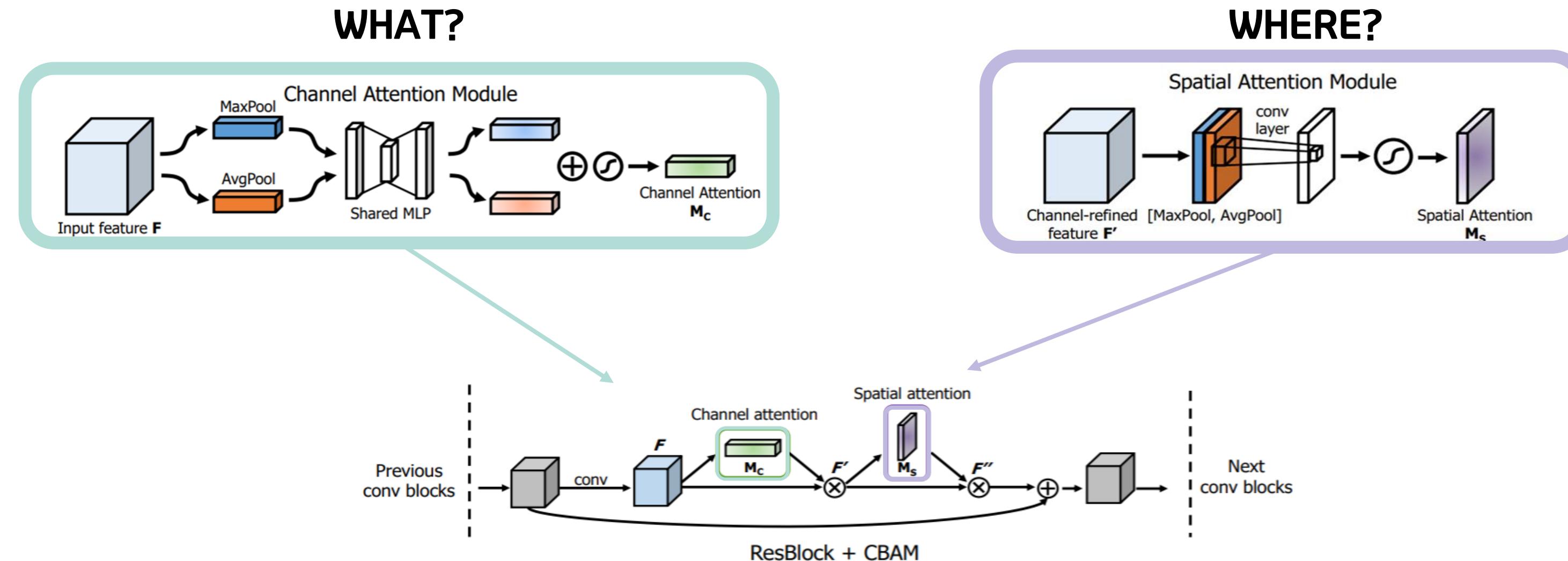


Figure 4. Internal structure of module IncRes1.

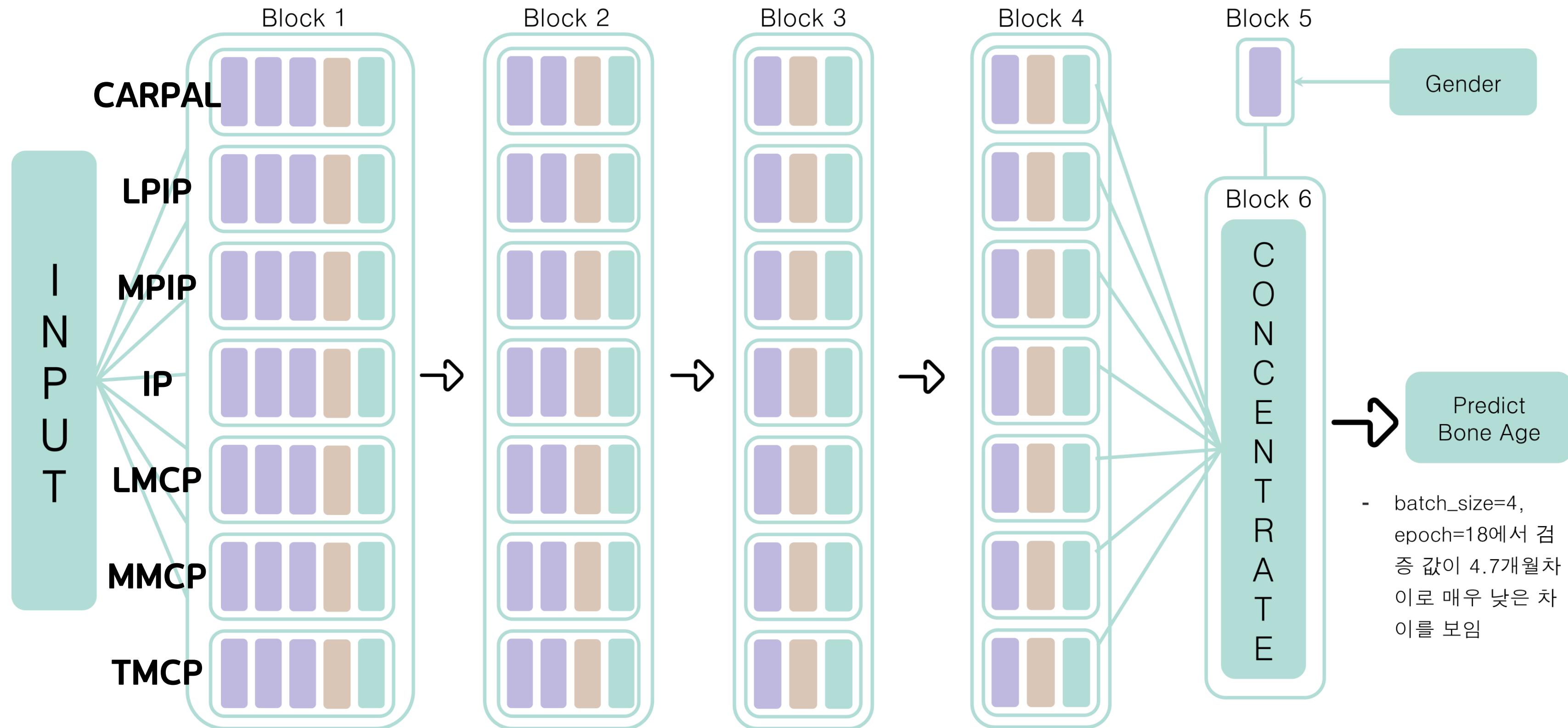
## CBAM (Convolutional Block Attention Module)



- CBAM은 합성곱층을 지나온 이미지를 어느 부분을 강조할지 정하는 Attention 기반 모듈입니다.

# TJ-Net 모델

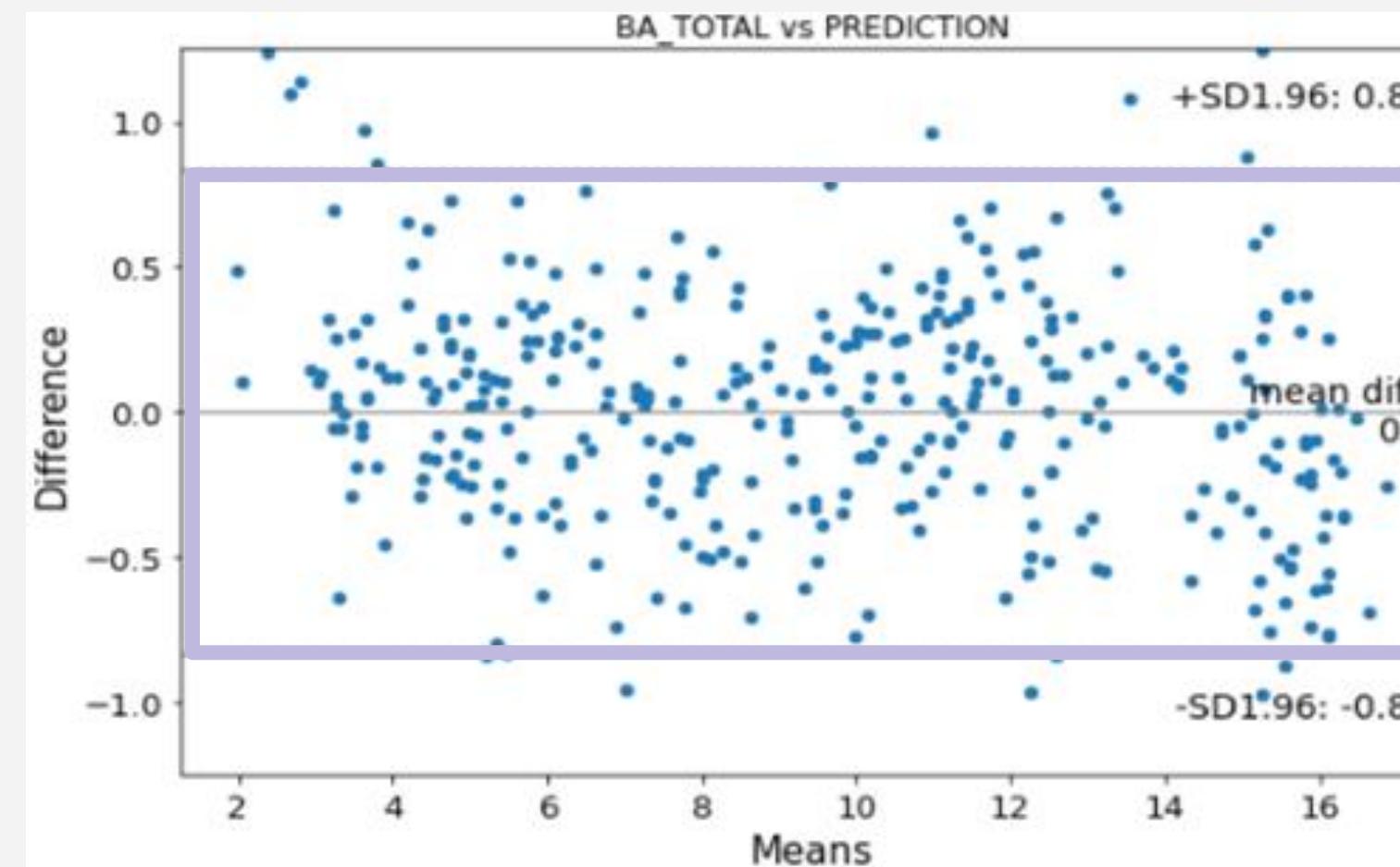
: Conv2D  
: CBAM  
: MaxPool2D



- batch\_size=4,  
epoch=18에서 검  
증 값이 4.7개월차  
이로 매우 낮은 차  
이를 보임

## 모델 적용 확인

## 의사 1,2 진단 평균과 비교



- Bland-Altman 그림은 두 가지 측정 기법(의사의 예측 vs TJ-Net 의 예측)을 비교하는 시각화 방법.
- 이 그래프에서는 두 기법의 평균에 대해 두 기법의 차이를 표시함.
- 수평선은 평균 차이와 차이의 표준 편차를 더하고 빼는 1.96배로 그려짐.

Bland-Altman Plot을 사용하여 모델 예측 결과와 실제 의사의 진단 차이를 확인

## LMS: 신장예측방법

### 골연령 기반 만 18세 신장 예측법

- $L, M, S, x$ 를 이용한 함수를 사용, 표준정규분포(Z-score) 도출  
\* $L, M, S$  = 측정된 골연령에 해당,  $x$  = 현재 신장

$$f(x) = \begin{cases} ((x/M)^L - 1)/LS & (L \neq 0) \\ [\ln(x/M)]/S & (L = 0) \end{cases}$$

- $L, M, S, Z$ -score를 이용한 함수를 사용해 예측 신장 도출  
\* $L, M, S$  = 18세에 해당하는 신장,  $Z$  = Z-score

$$g(z) = \begin{cases} M(1 + LSz)^{1/L} & (L \neq 0) \\ M \exp(Sz) & (L = 0) \end{cases}$$

- 신장 예측 공식 :  $M_{18} * (1 + (L_{18} * s_{18} * Z))^{**} (1 / L_{18})$

2017 소아 청소년 성장도표

성별	개월	L	M	S	신장(cm) 백분위수	
					1st	3rd
남	0	1.0000	49.8842	0.0380	45.5	46.3
남	1	1.0000	54.7244	0.0356	50.2	51.1
남	2	1.0000	58.4249	0.0342	53.8	54.7
남	3	1.0000	61.4292	0.0333	56.7	57.6
남	4	1.0000	63.8860	0.0326	59.0	60.0
남	5	1.0000	65.9026	0.0320	61.0	61.9
남	6	1.0000	67.6236	0.0317	62.6	63.6
남	7	1.0000	69.1645	0.0314	64.1	65.1
남	8	1.0000	70.5994	0.0312	65.5	66.5
남	9	1.0000	71.9687	0.0312	66.8	67.7
남	10	1.0000	73.2812	0.0312	68.0	69.0
남	11	1.0000	74.5388	0.0313	69.1	70.2
남	12	1.0000	75.7488	0.0314	70.2	71.3
남	13	1.0000	76.9186	0.0315	71.3	72.4
남	14	1.0000	78.0497	0.0317	72.3	73.4
남	15	1.0000	79.1458	0.0320	73.3	74.4
남	16	1.0000	80.2113	0.0322	74.2	75.4
남	17	1.0000	81.2487	0.0325	75.1	76.3
남	18	1.0000	82.2587	0.0328	76.0	77.2
남	19	1.0000	83.2418	0.0331	76.8	78.1
남	20	1.0000	84.1996	0.0334	77.7	78.9

출처 : 질병관리청 - 소아 청소년 성장도표

## LMS: 신장예측방법

```

def Height_prediction ( gender, BA, current_H ) :
    import pandas as pd
    lms_df = pd.read_csv('/content/drive/MyDrive/2차 프로젝트 원본 데이터/growth/height_df.csv')
    month_age = round(BA * 12)
    if gender == 1:
        lms_index = month_age - 36
        L_18, M_18, S_18 = lms_df.iloc[191,2], lms_df.iloc[191,3], lms_df.iloc[191,4]
    elif gender == 0:
        lms_index = month_age - 36
        L_18, M_18, S_18 = lms_df.iloc[383,2], lms_df.iloc[383,3], lms_df.iloc[383,4]

    L,M,S = lms_df.iloc[lms_index,2], lms_df.iloc[lms_index,3],lms_df.iloc[lms_index,4]
    x = current_H

    Z = (((x/M)**L)-1)/(L*S)
    Z = round(Z,4)

    pred_height = M_18 * (1 + (L_18 * S_18 * Z)) ** (1 / L_18)
    pred_height = round(pred_height, 1)
    return pred_height

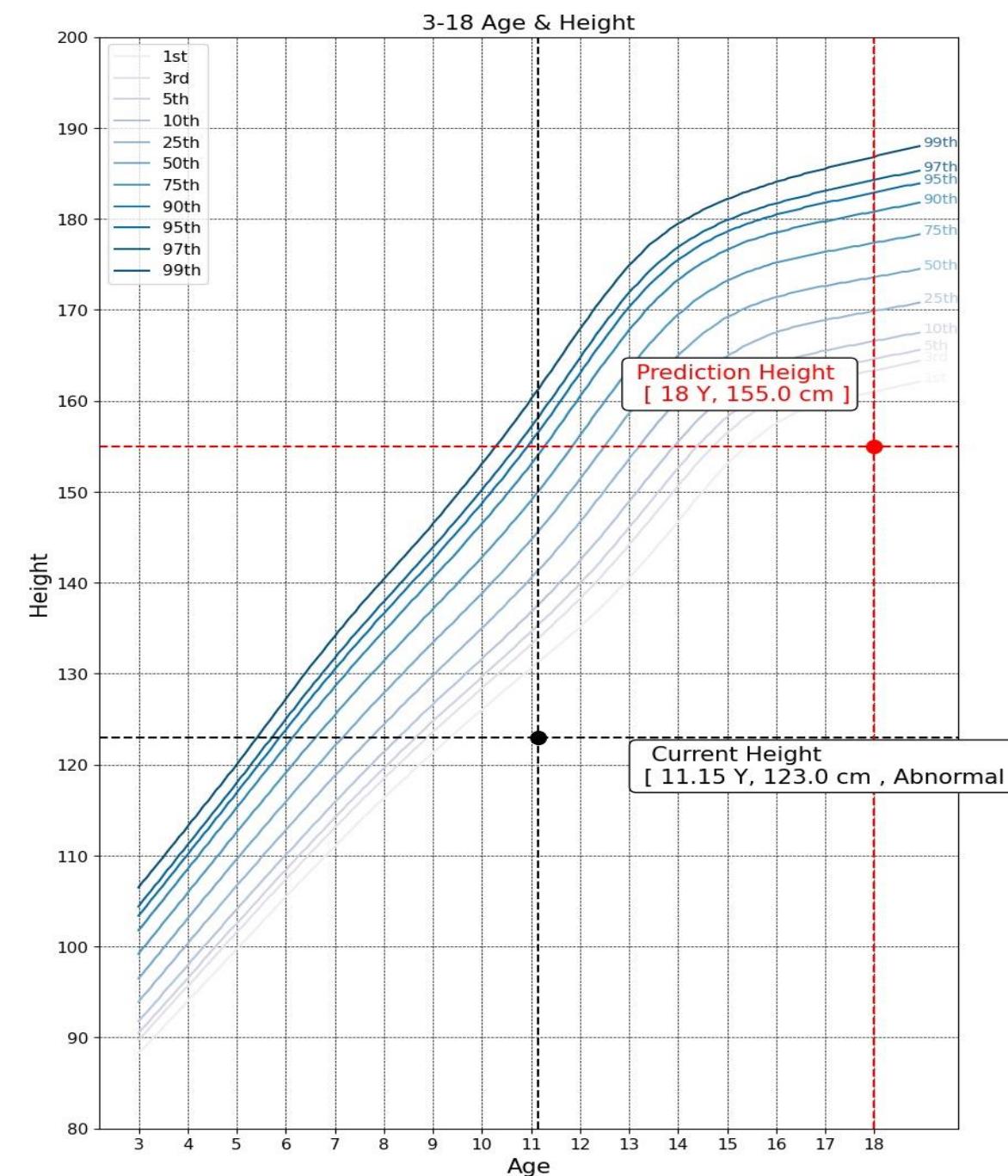
def find_th(df,BA,Height):
    df = df.reset_index()
    find_df = df[(df['AGE'] >= BA) & (df['MONTH'] >= BA*12)] >= Height
    cdf = df[(df['AGE'] >= BA) & (df['MONTH'] >= BA*12)]
    find_df = find_df.drop('MONTH',axis=1)
    find_df = find_df.iloc[0]
    find_df = find_df[find_df==True].index
    try:
        find_th = list(find_df)[0]

        if cdf.loc[(cdf.index[0]) , find_th] - 1.5 > Height :
            find_th = 'Abnormal'

        else:
            pass
    except:
        if cdf.loc[(cdf.index[0]) , '99th'] + 1.5 >= Height :
            find_th = '99th'
        else:
            find_th='Abnormal'

    return find_th

```



- ▶ 예측된 골연령값 BA, 입력된 현재 신장, 성장도표에 위치한 분위수는 해당 그래프에 검은색 점으로 출력됨.
- ▶ (18세 기준) 예측된 신장 값은 빨간점으로 출력됨.

## LMS: 신장예측방법

	AGE	MONTH	1st	3rd	5th	10th	15th	25th	50th	75th	85th	90th	95th	97th	99th
0	3.00	36	88.3	89.7	90.5	91.8	92.6	93.9	96.5	99.2	100.7	101.8	103.4	104.4	106.5
1	3.08	37	88.7	90.2	91.0	92.3	93.2	94.5	97.0	99.8	101.3	102.3	103.9	105.0	107.1
2	3.17	38	89.2	90.7	91.5	92.8	93.7	95.0	97.6	100.3	101.8	102.9	104.5	105.6	107.6
3	3.25	39	89.7	91.2	92.0	93.3	94.2	95.5	98.1	100.9	102.4	103.5	105.1	106.1	108.2
4	3.33	40	90.2	91.7	92.5	93.8	94.7	96.1	98.7	101.4	103.0	104.0	105.6	106.7	108.7
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
187	18.58	223	161.7	164.0	165.2	167.2	168.5	170.4	174.2	177.9	180.0	181.4	183.6	184.9	187.6
188	18.67	224	161.8	164.1	165.3	167.3	168.6	170.5	174.2	178.0	180.1	181.5	183.7	185.0	187.7
189	18.75	225	161.9	164.2	165.4	167.3	168.6	170.6	174.3	178.1	180.2	181.6	183.7	185.1	187.8
190	18.83	226	162.0	164.3	165.5	167.4	168.7	170.7	174.4	178.2	180.3	181.7	183.8	185.2	187.9
191	18.92	227	162.1	164.4	165.6	167.5	168.8	170.8	174.5	178.3	180.4	181.8	183.9	185.3	188.0

192 rows × 15 columns

```

1 gender =1
2 BA = 3.0
3 current_H = 96.5
4
5 df_m = pd.read_csv('/content/drive/MyDrive/2차 프로젝트 원본 데이터/growth/male_year.csv',index_col='AGE')
6 df_fm = pd.read_csv('/content/drive/MyDrive/2차 프로젝트 원본 데이터/growth/female_year.csv',index_col='AGE')
7
8 if gender == 0:
9     df = df_fm
10 elif gender == 1:
11     df = df_m
12
13 def Prediction(gender, BA, current_H, df) :
14     prediction = Height_prediction(gender,BA,current_H)
15     result_th = find_th(df,BA,current_H)
16     print(f'[Info] Bone Age > {BA} // Current_Height > {current_H} // {result_th} ')
17     print(f'Prediction Height > {prediction} // {result_th}')
18
19 Prediction(gender,BA,current_H,df)

[Info] Bone Age > 3.0 // Current Height > 96.5 //50th
Prediction Height > 174.5 // 50th

```

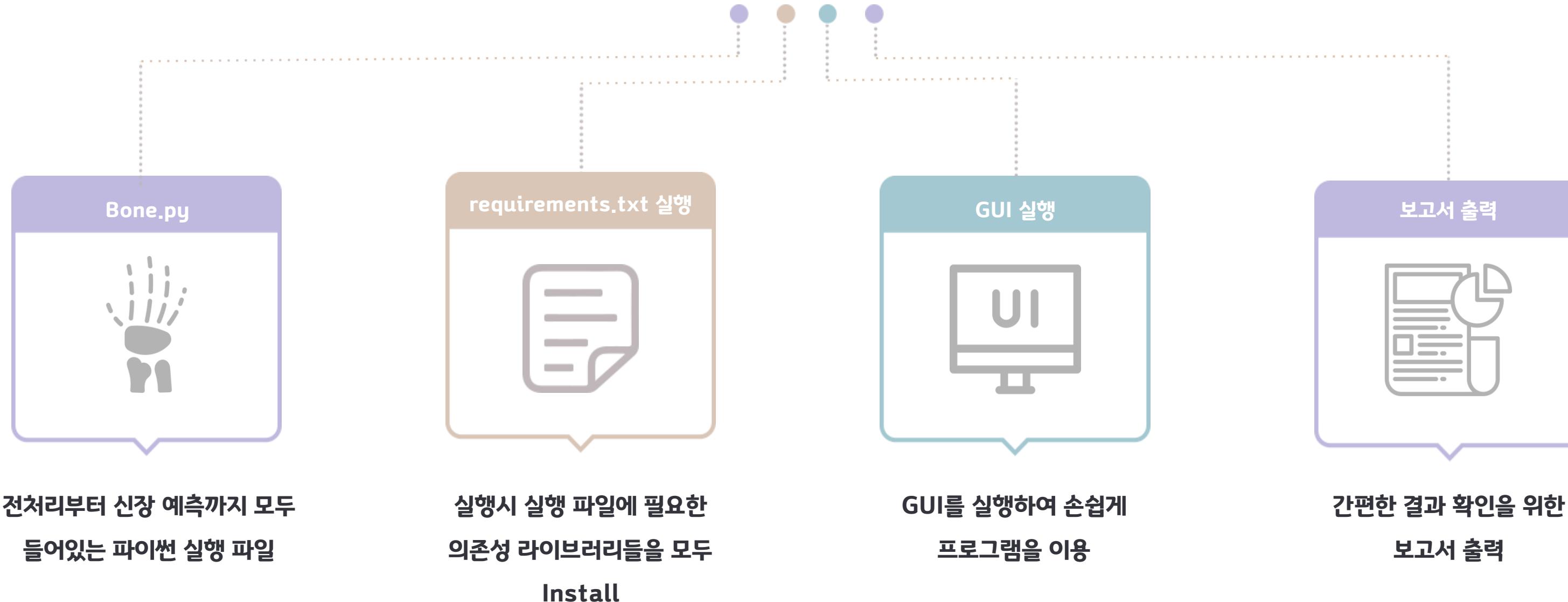
- 검증을 위해서 코드 결과입니다.
- 예측된 골연령은 3.0세, 현재 신장 96.5cm, 50분위수에 위치하여 있는 검사자는 예측값 174.5cm 50분위수에 위치하는 것을 확인할 수 있었습니다

03.

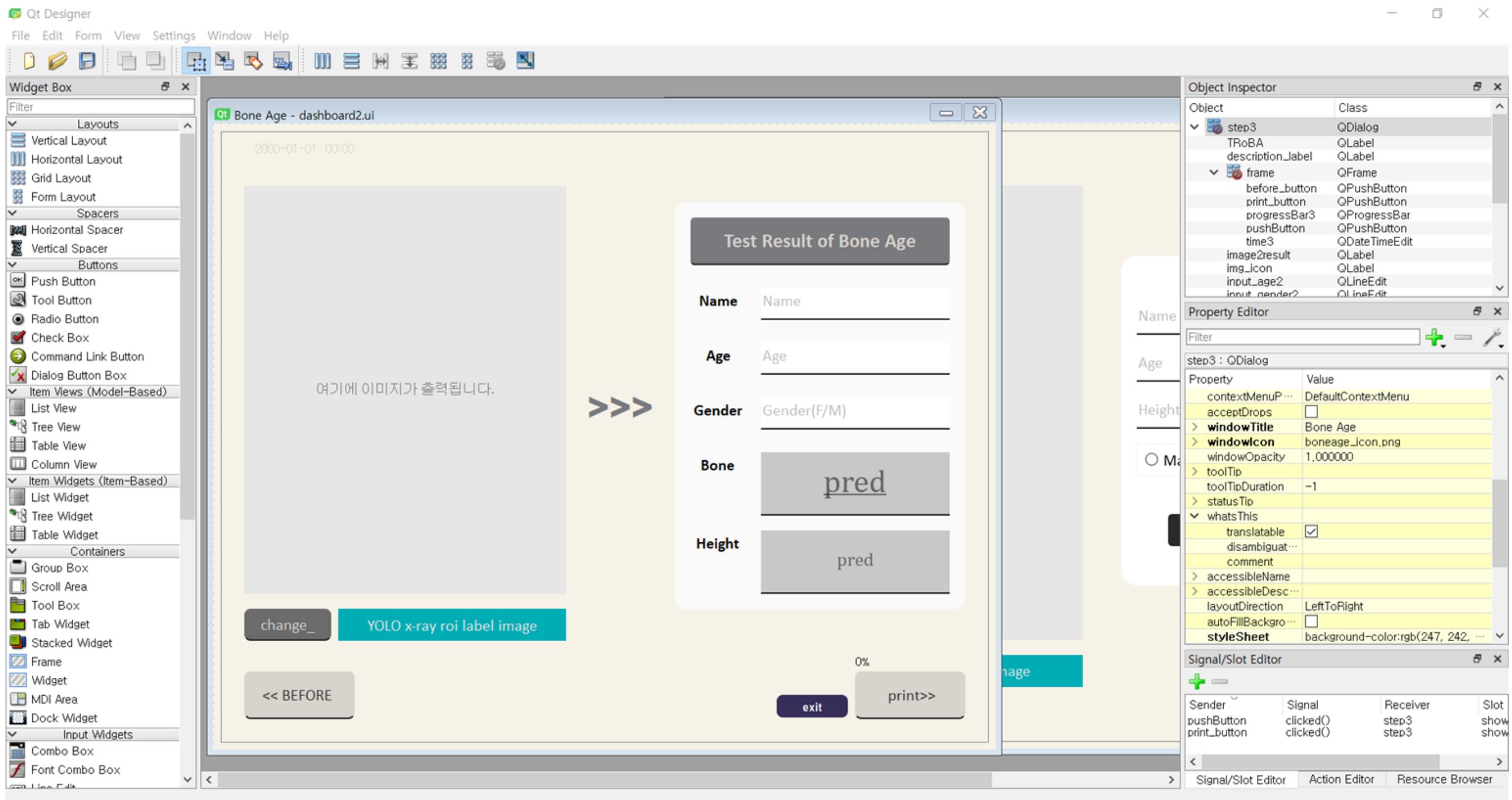
## 함수 자동화 및 GUI

# 함수 자동화 및 GUI

사용된 함수들을 한곳으로 모아 이미지 경로와 성별을 입력하고  
이미지 전처리부터 골연령 예측까지 순서대로 처리하는 연결 후  
PyQt를 이용한 GUI에 출력 되도록 작성



## PyQt5, QT Designer



Anaconda Prompt (anaconda3)

(env1) C:\Users\JH\Desktop\boneage>python bone\_age.py

아나콘다 프롬프트로 실행합니다



기본 정보 입력



- X-Ray 원본이미지
- 이름
- 신장
- 성별

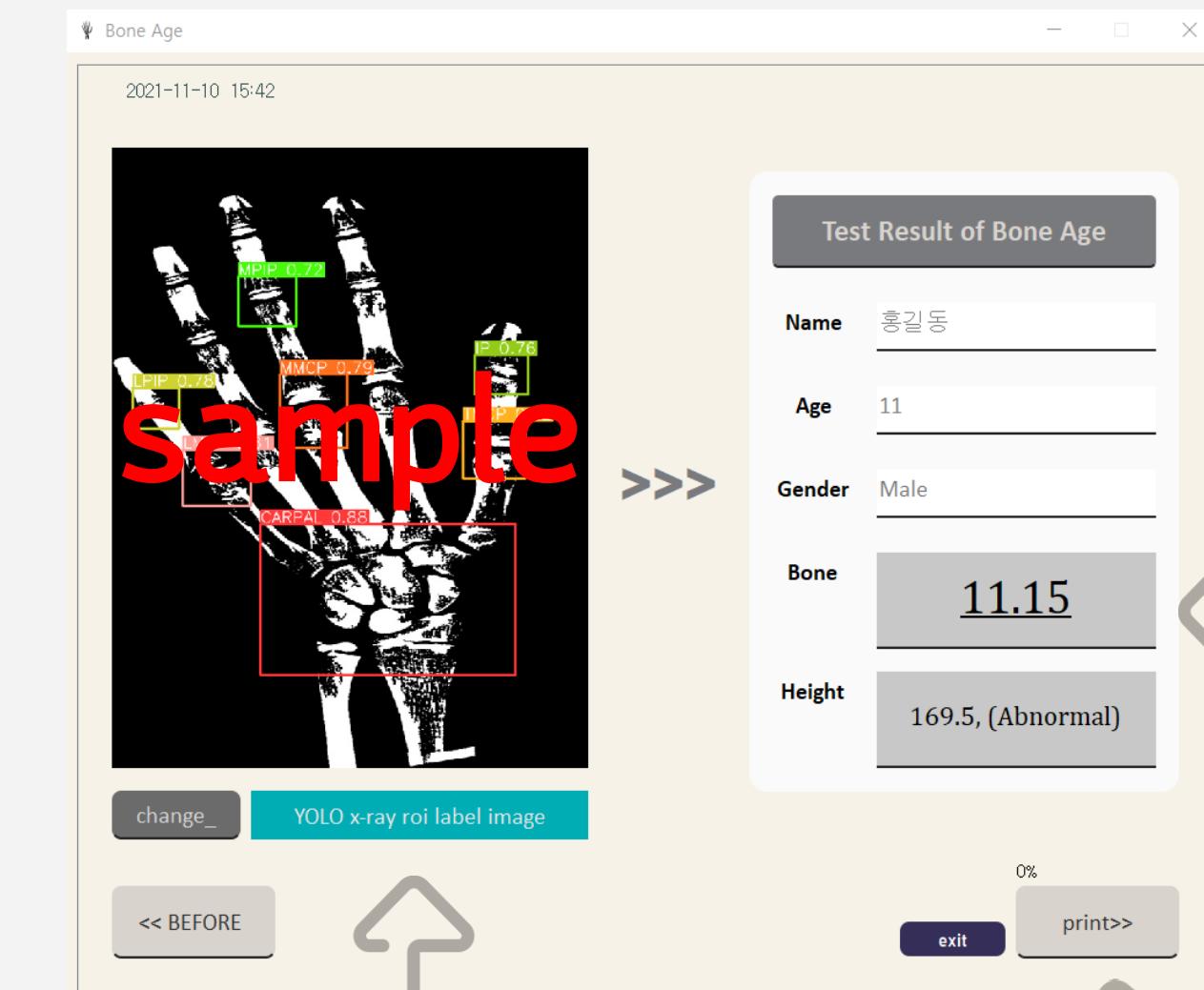
## 프로그램



### Input information



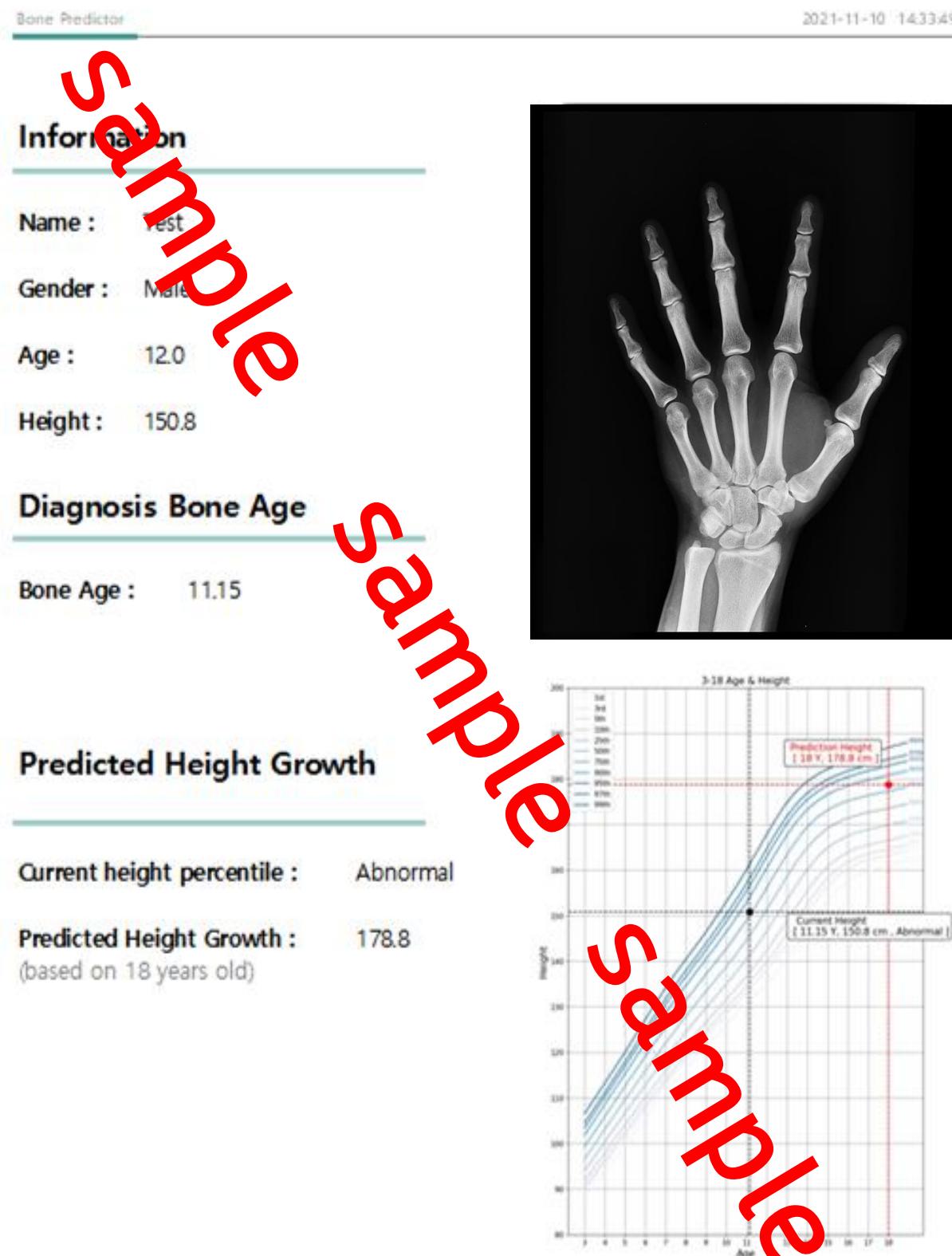
Bone extractions



YOLO crop img

Predict  
Bone age

Print  
report



## 보고서 출력 예시



- 환자정보
  - 환자성명
  - 환자성별
  - 환자 현재 나이
  - 환자 현재 신장
- 골연령 진단
  - 환자 골연령 예측값
- 키 성장 진단
  - 현재 신장 백분위수
  - 만 18세 기준 예측 신장

## 기대효과

X-ray 사진만 있으면 개인도 사용 가능

골연령 평가시 시간, 비용 절약

전문의들의 골연령 평가값을 일반화

골연령 평가 결과의 오차가 적음

## 접수증

신청인 : 허진경 님이 신청한 접수내역입니다.

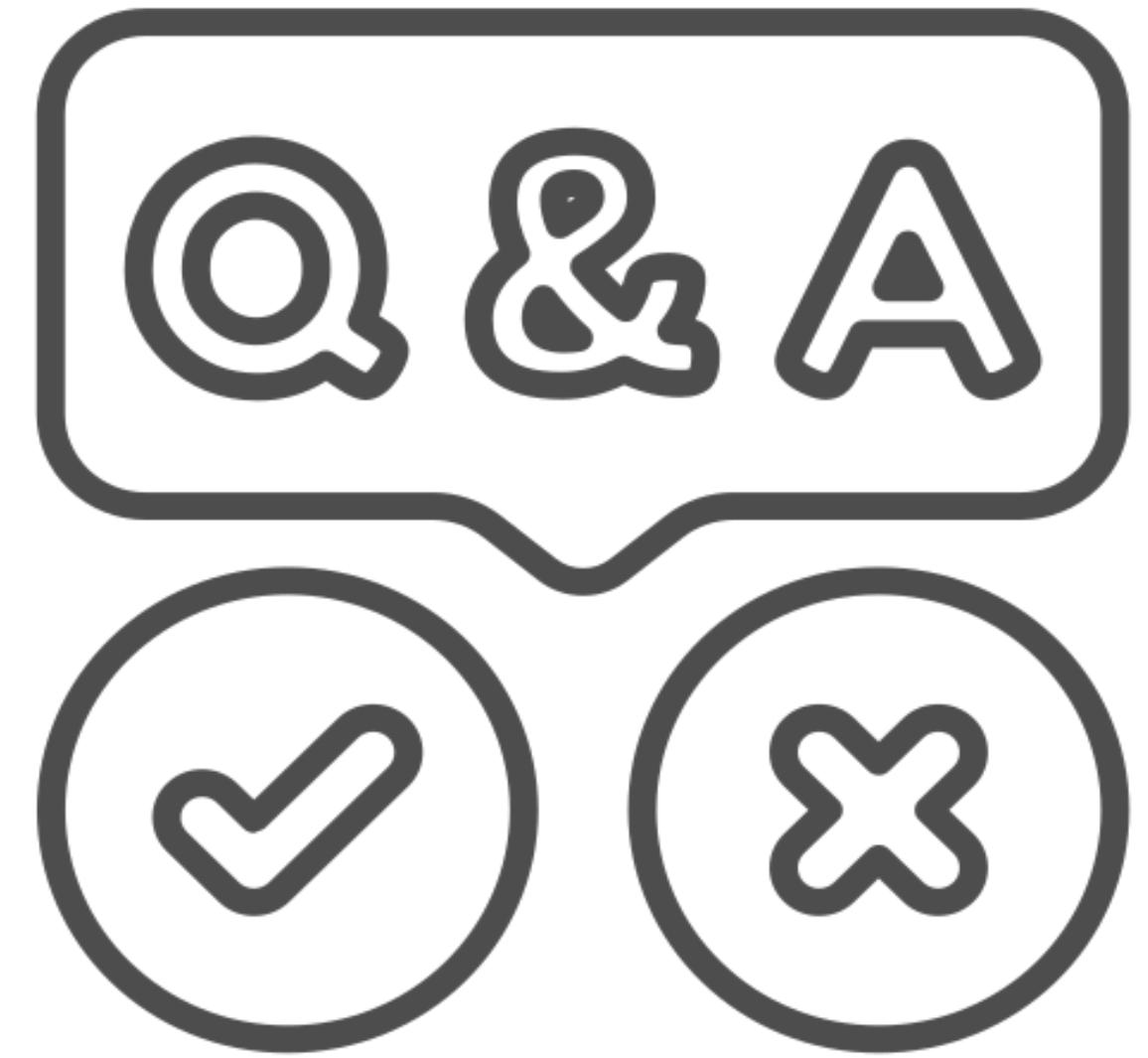
접수일자 : 2021년 11월 10일

접수번호	신청내용	신청제호
2021-051631	프로그램 등록	손목과 손가락 관절을 이용한 소아/청소년의 뼈 나이 예측 프로그램

2021년 11월 10일

한국저작권위원회 위원장







THANK YOU

