

 Team Leader: Eunji Jung  
 App Developer: EunSeong Lee  
 OnDevice Developer: HyenWoo Choi  
Validation Engineer: J. Hwan

조은지안조은지조 | **Feat.DPU**에 따른 **HW** 성능 비교

# Content

## 01 개요

- AI를 활용한 영양제 추천 시스템
- 주제 선정 배경
- 개발 환경

## 02 학습 모델 개발

- Introducing Dataset & Model
- Loss & Accuracy 성능 평가
- 웹캠 연동

## 03 모델 통합 및 결과

- Ollama Model
- S/W Architecture
- Flow Chart
- Main Features

## 04 가속기 w. FPGA

- FPGA DPU 구현 과정
- Performance comparison-1: ResNET50
- Performance comparison-2: YoloV3

## 05 Trouble Shooting

- Trained AI Model
- Integrated System
- Accelerator

## 06 시연 및 고찰

- 시연 영상
- 고찰
- Q&A



# 01 | 개요

# 서비스 개요



- 01 본 프로젝트는 카메라를 기반으로 안면인식을 하여 나이와 성별을 추론하여, 사용자의 적합한 맞춤형 영양제를 추천하는 인공지능 디지털 헬스케어 시스템을 제작한 Nutrfit 솔루션입니다.

## 서비스 필요 배

- 01 개인 맞춤형 건강 관리에 대한 수요가 급증함으로써 서비스 구축

- 02 과도한 마케팅으로 인한 불필요한 영양제 구매 및 섭취 문제

- 03 개인별 건강 상태 및 생활 습관에 대한 정확한 정보 부족

# 개발환경(Hardware)



×



×



×



Edge Device  
(Raspberry Pi)

FPGA  
(Ultra96-V2)

LLM  
(Ollama)

Webcam  
(Pleomax 300K)

# 개발환경(Software)



Python

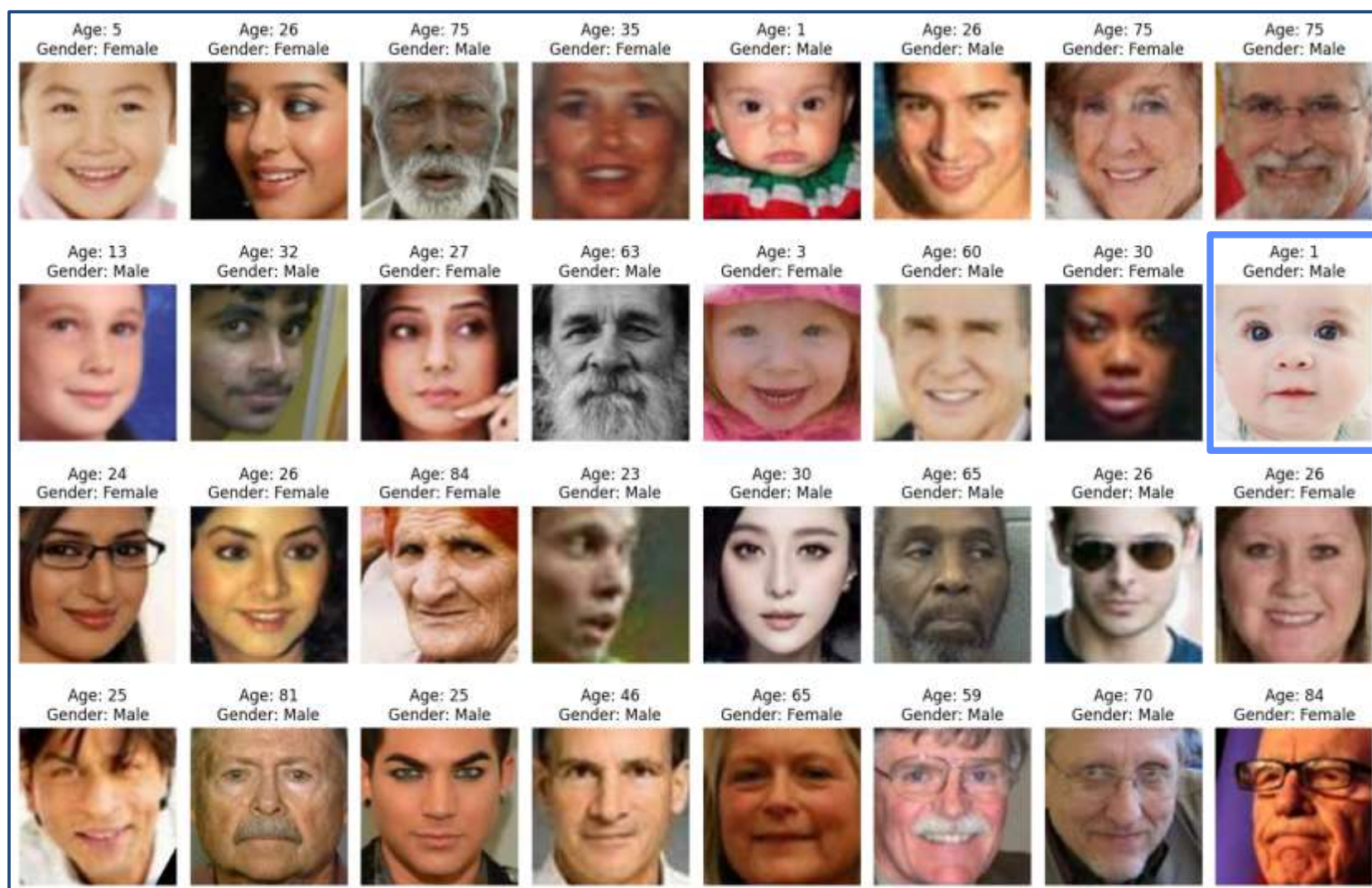


## 02 | 학습 모델 개발



# DataSet: UTKFace

UTKFace 데이터란? 얼굴 관련 모델을 훈련하고 평가하는 데 사용되는 대규모 공개 데이터셋



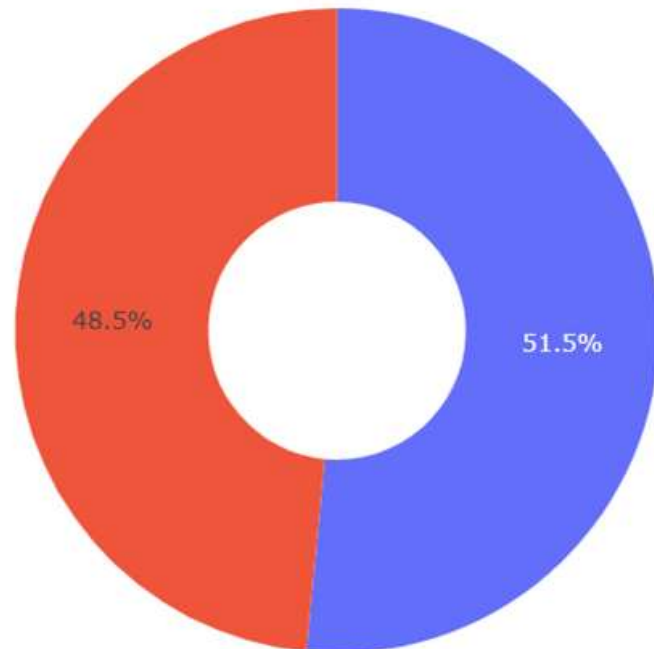
1\_0\_0\_20161219225952240.jpg

↓ ↓ ↓ ↓  
나이 성별 인종 수집 날짜 및 시

분류	문제 유형	예측값	출력층	Loss Func.
나이	회귀	실수	Dense(1)	MAE MSE
성별	이진 분류	확률 0 or 1	Dense(1, activation = 'sigmoid')	binary_crossentrop

# DataSet: UTKFace

UTKFace 데이터 출처: Keggel, roboflow

	Gender	Age	성비
데이터 개수	10,284	2,087	
Training Size	8227	1460	
Validation Size	1028	417	
Testing Size	1029	210	



# Backbone Model

Backbone이란? 이미지로부터 특징을 추출하는 역할을 하는 CNN Network

AgeNet

Layer (type)	Output Shape	Param #
VGG16(Fuctional)	(None, 7, 7, 512)	14,714,688
Slight Dropout (Dropout)	(None, 7, 7, 512)	0
FlattenEmbeddings(Flatten)	(None, 25088)	0
dense (Dense)	(None, 256)	6,422,784
AgeOutput (Dense)	(None, 1)	257

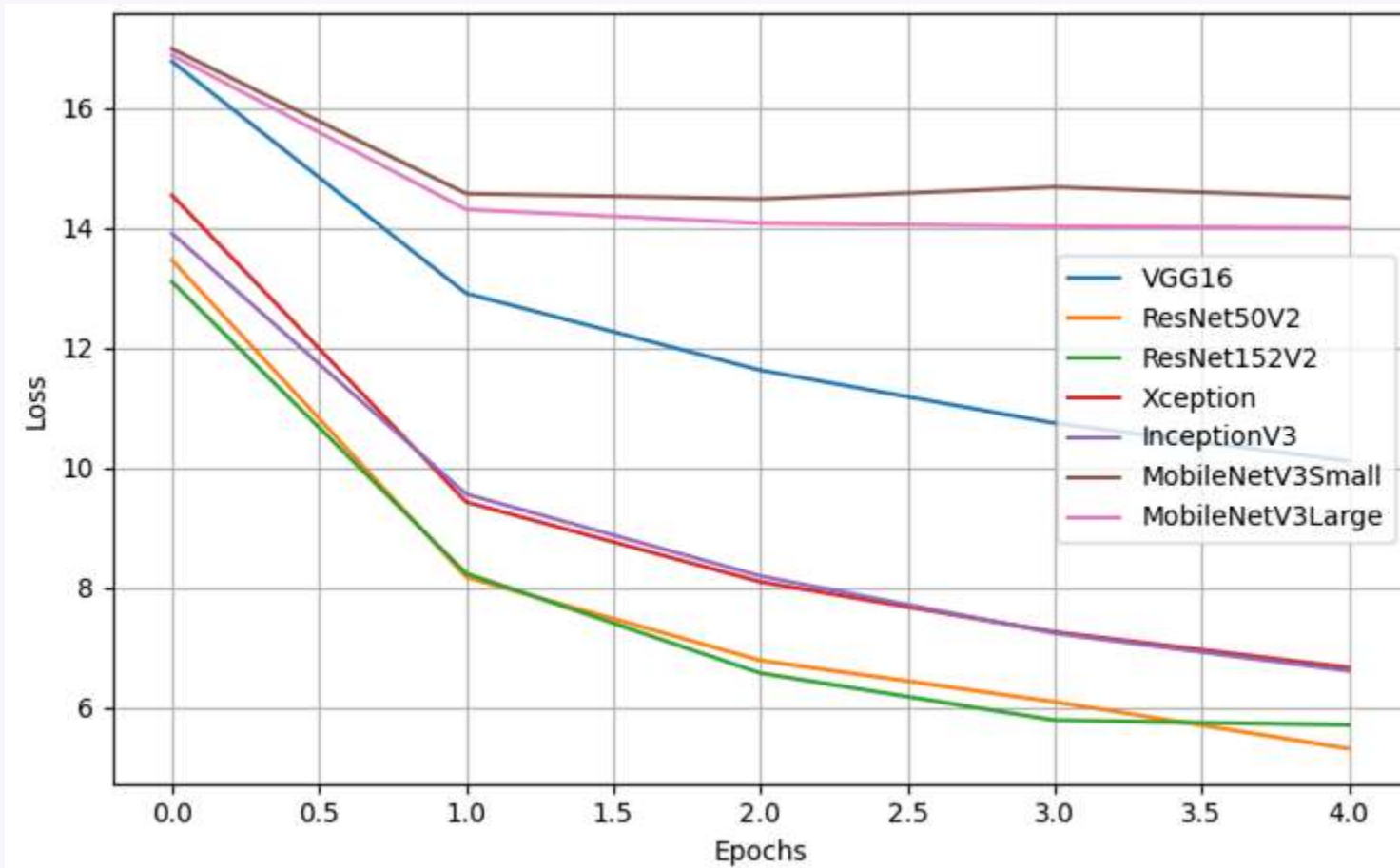
GenderNet

Layer (type)	Output Shape	Param #
RasNet15V2(Fuctional)	(None, 7, 7, 2048)	58,331,648
Slight Dropout (Dropout)	(None, 7, 7, 2048)	0
GlobalAvPooling (GlovalAveragePooling2D)	(None, 2048)	0
gender (Dense)	(None, 1)	2,049

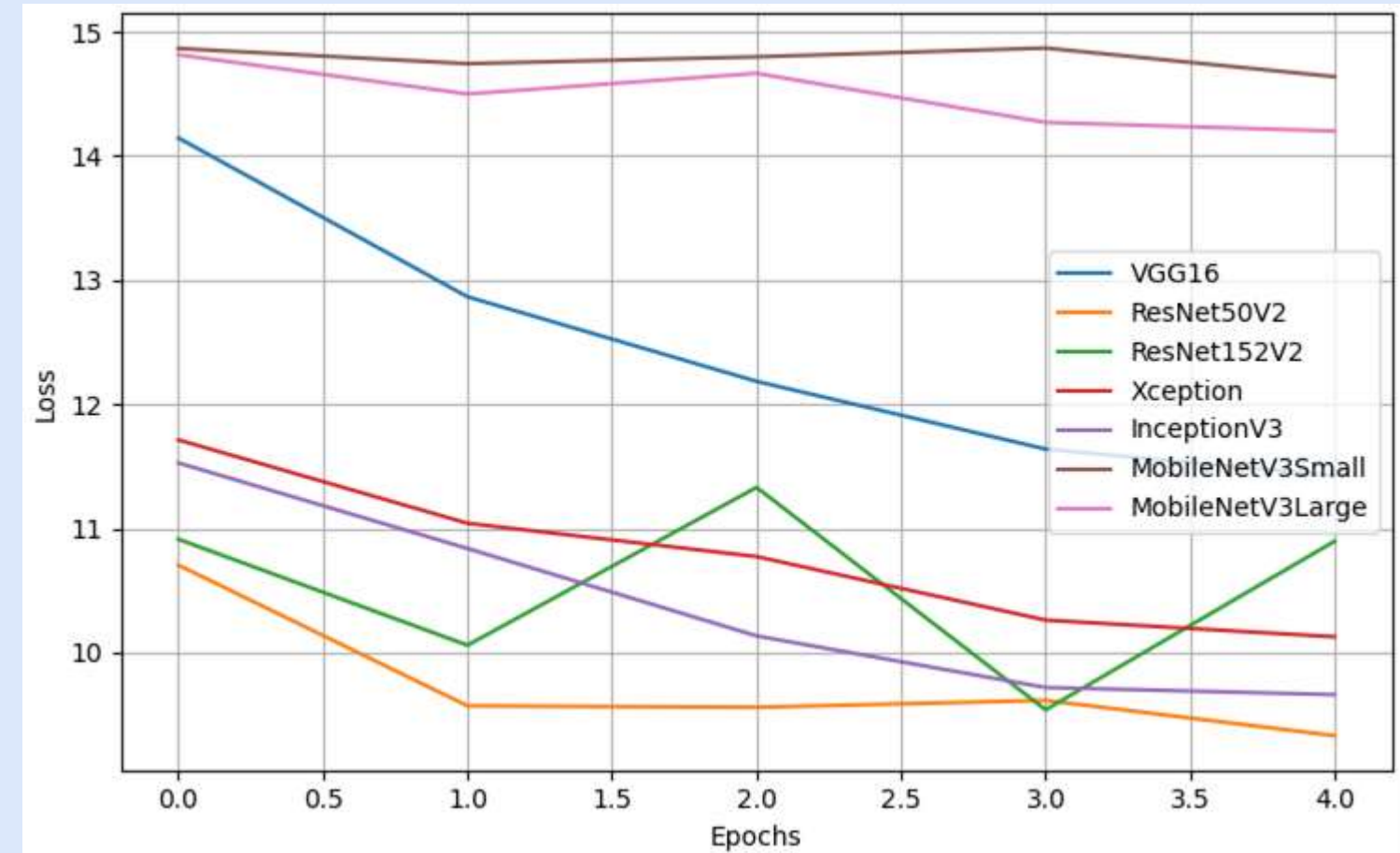
# Backbone Model

AGE Loss plot Backbone Model Comparison

## Train Loss Plot



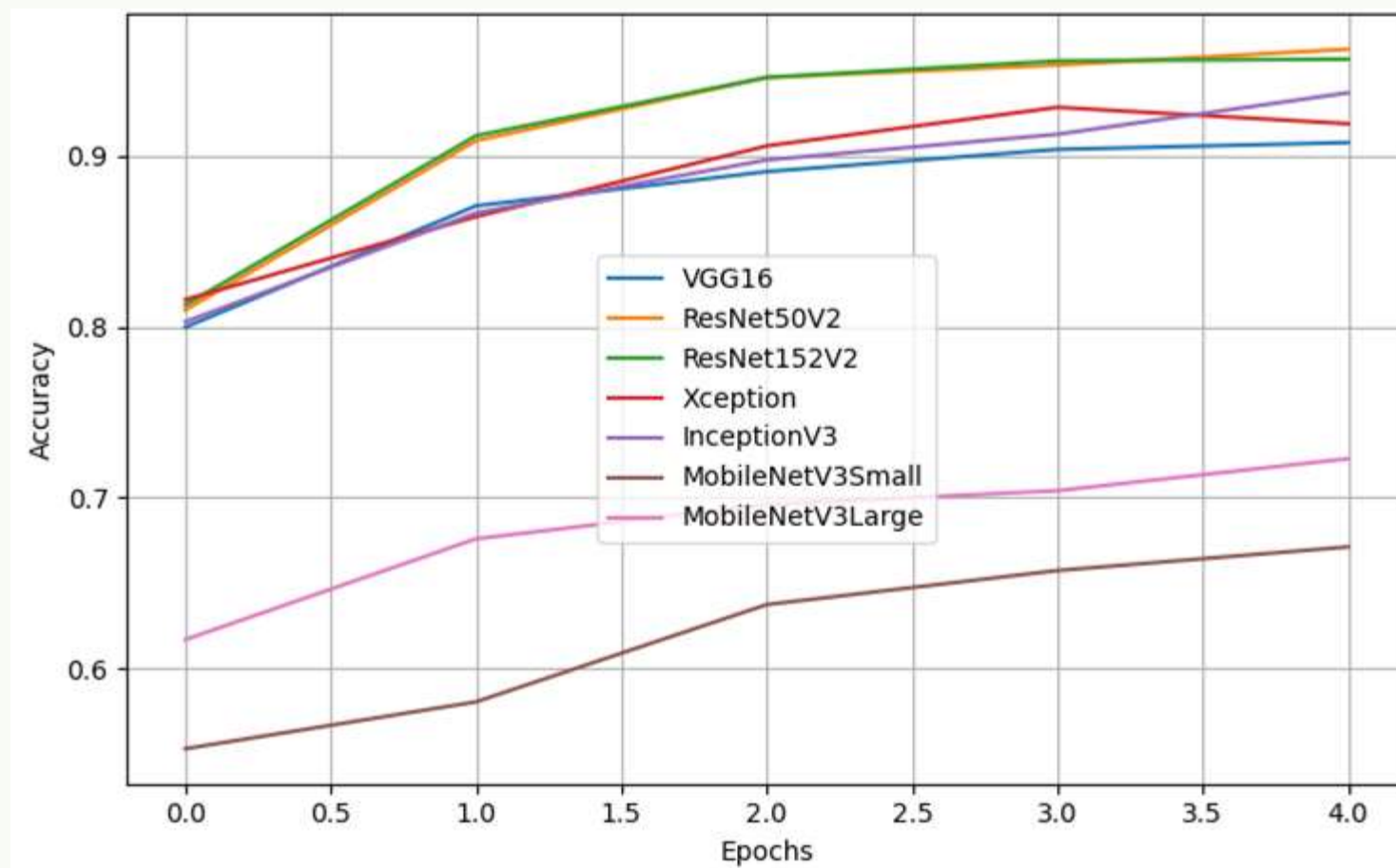
## Val Loss Plot



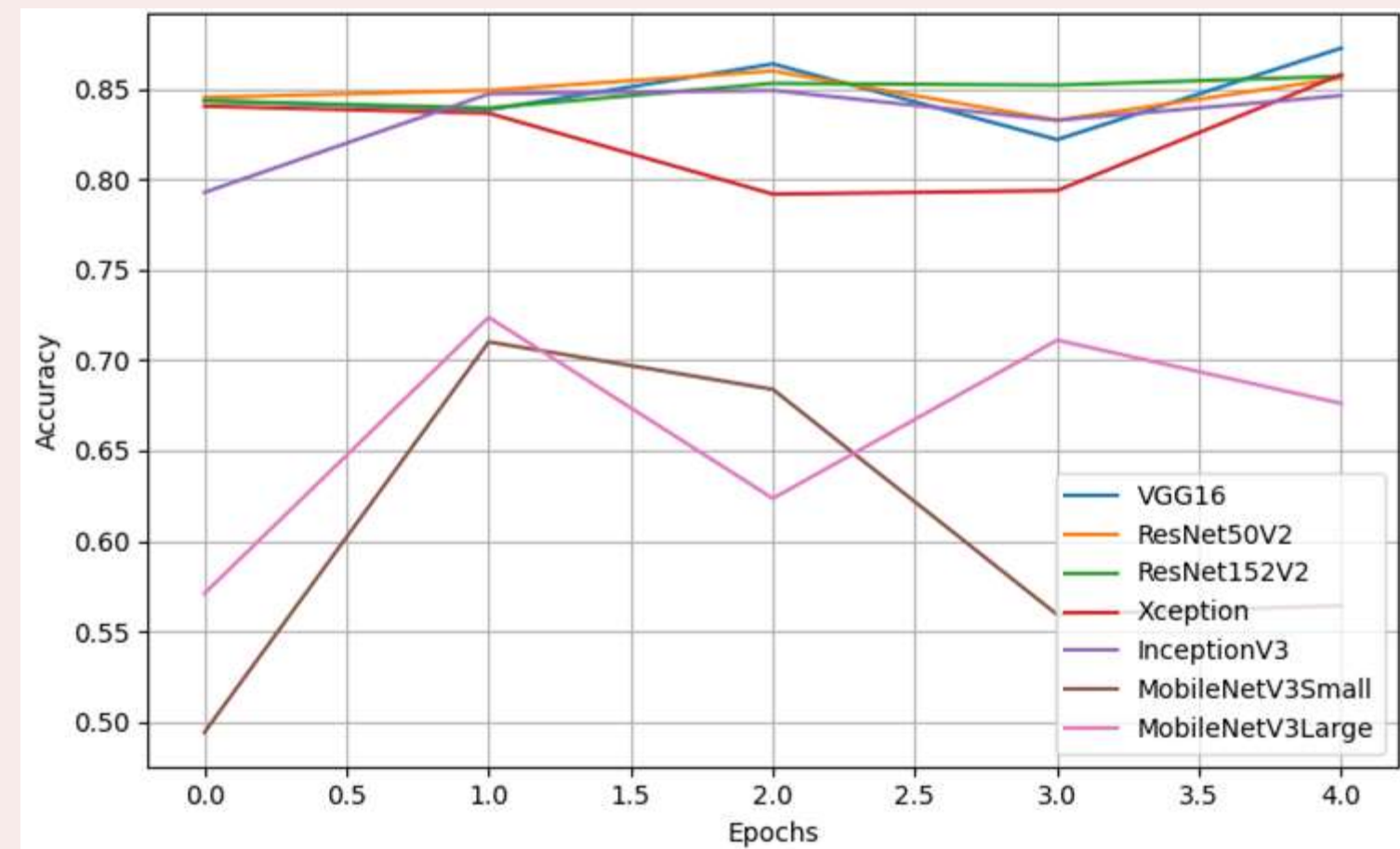
# Backbone Model

Gender Accuracy plot Backbone Model Comparison

## Train Accuracy Plot



## Val Accuracy Plot

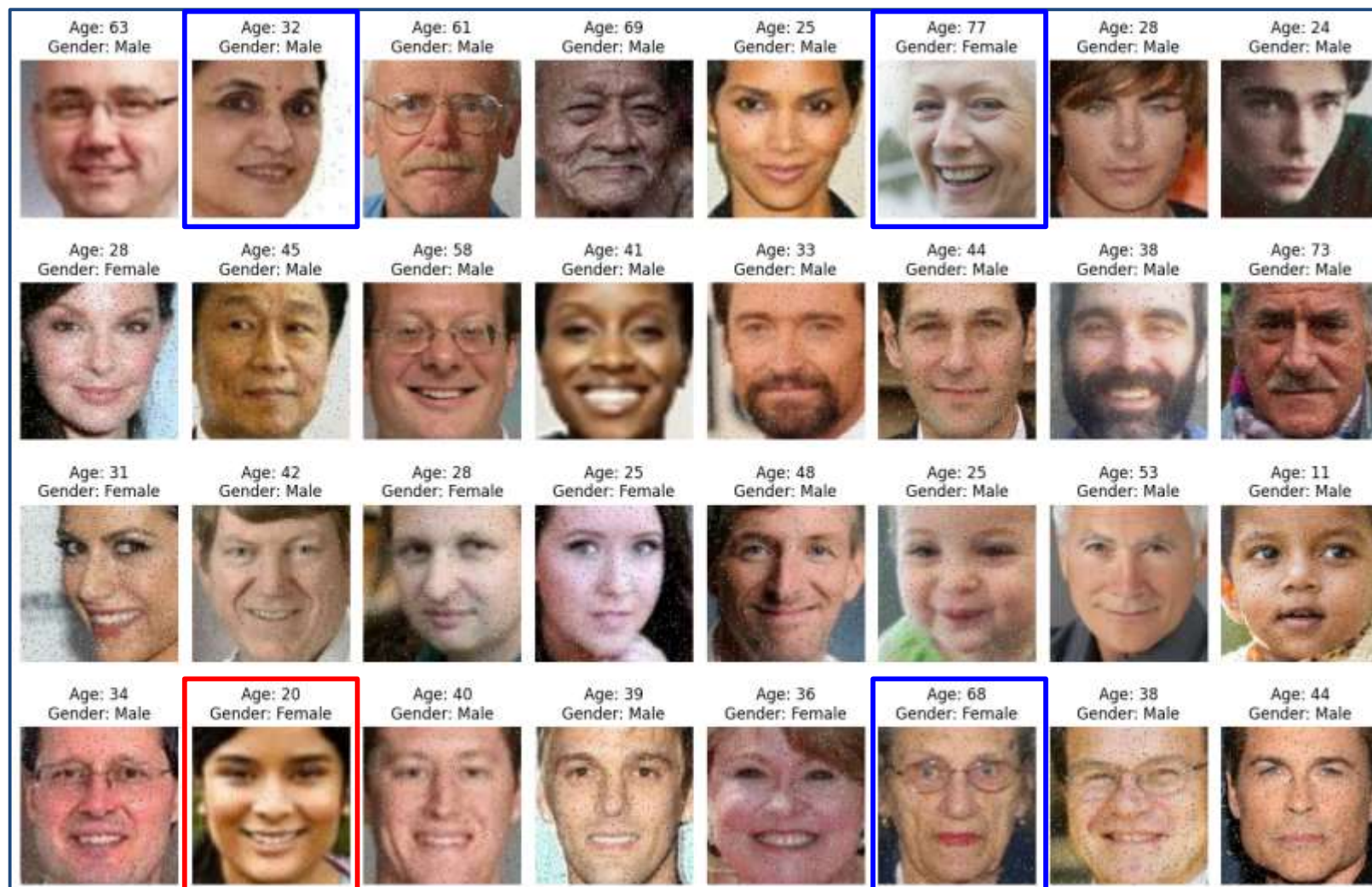




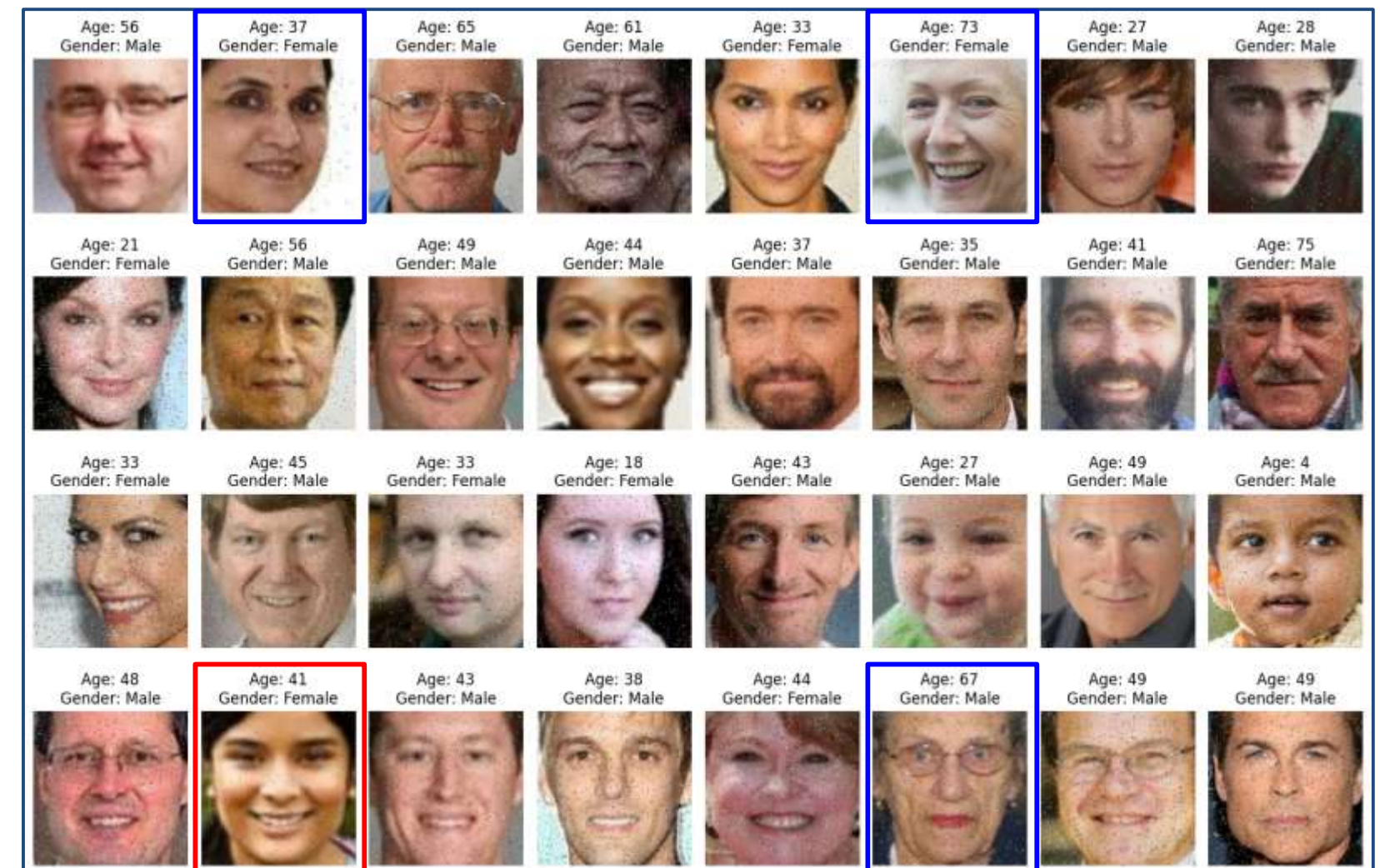
# Model Prediction

Age Backbone Model / Gender Backbone Model

VGG16 / ResNet152V2



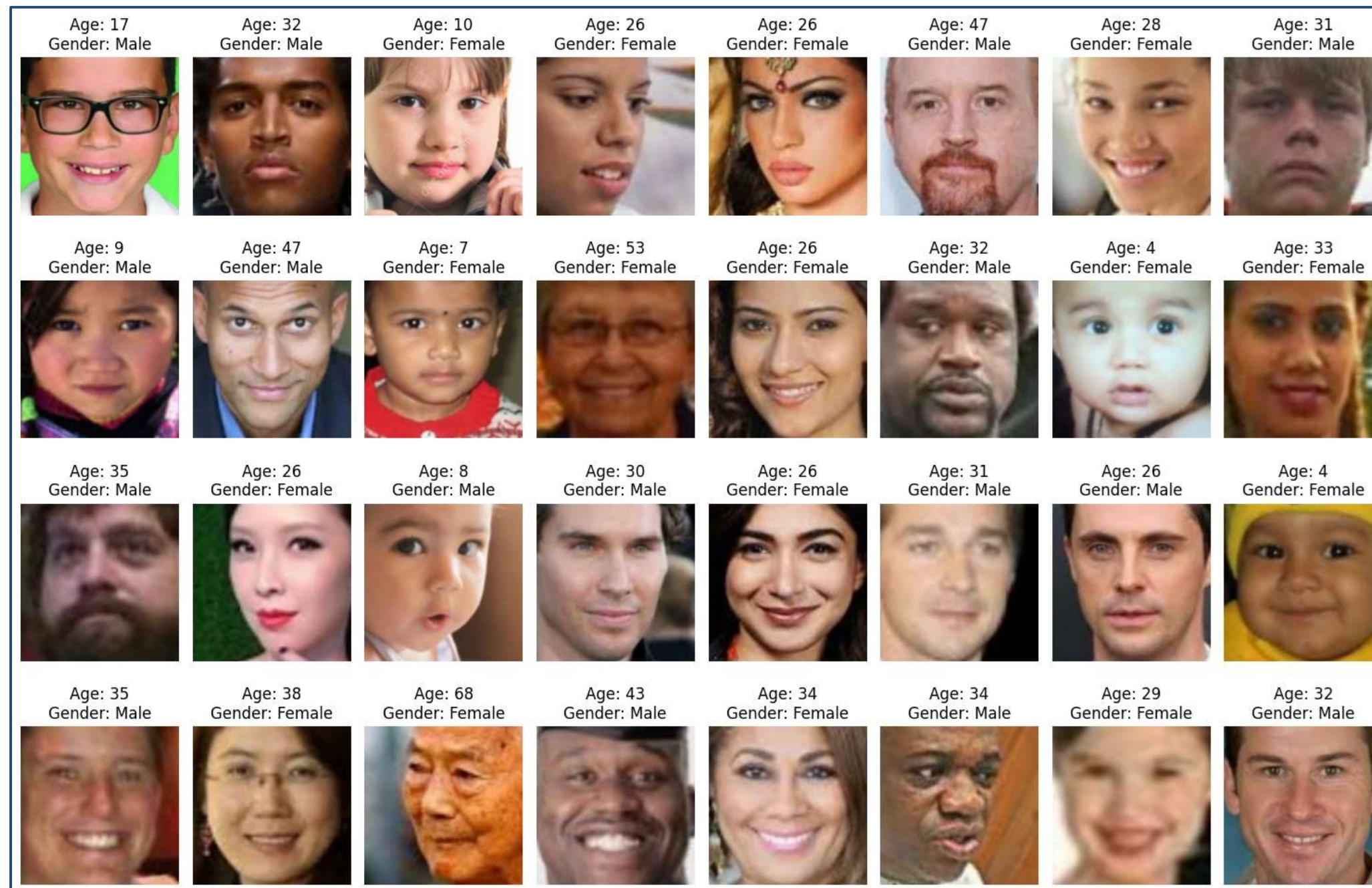
ResNet50V2/Xception





# Model Prediction

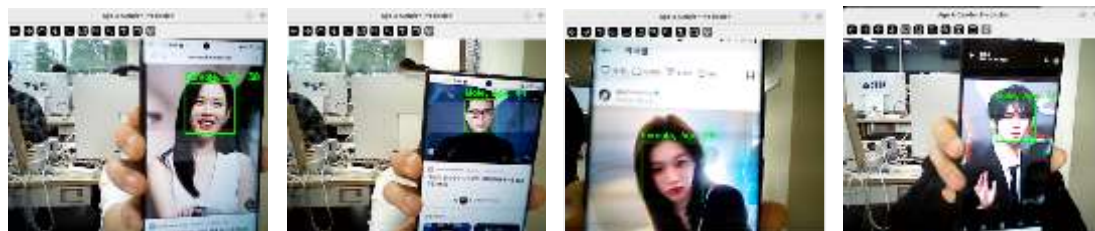
Data 개수 증가  $\Rightarrow$  성별 정확도 증가



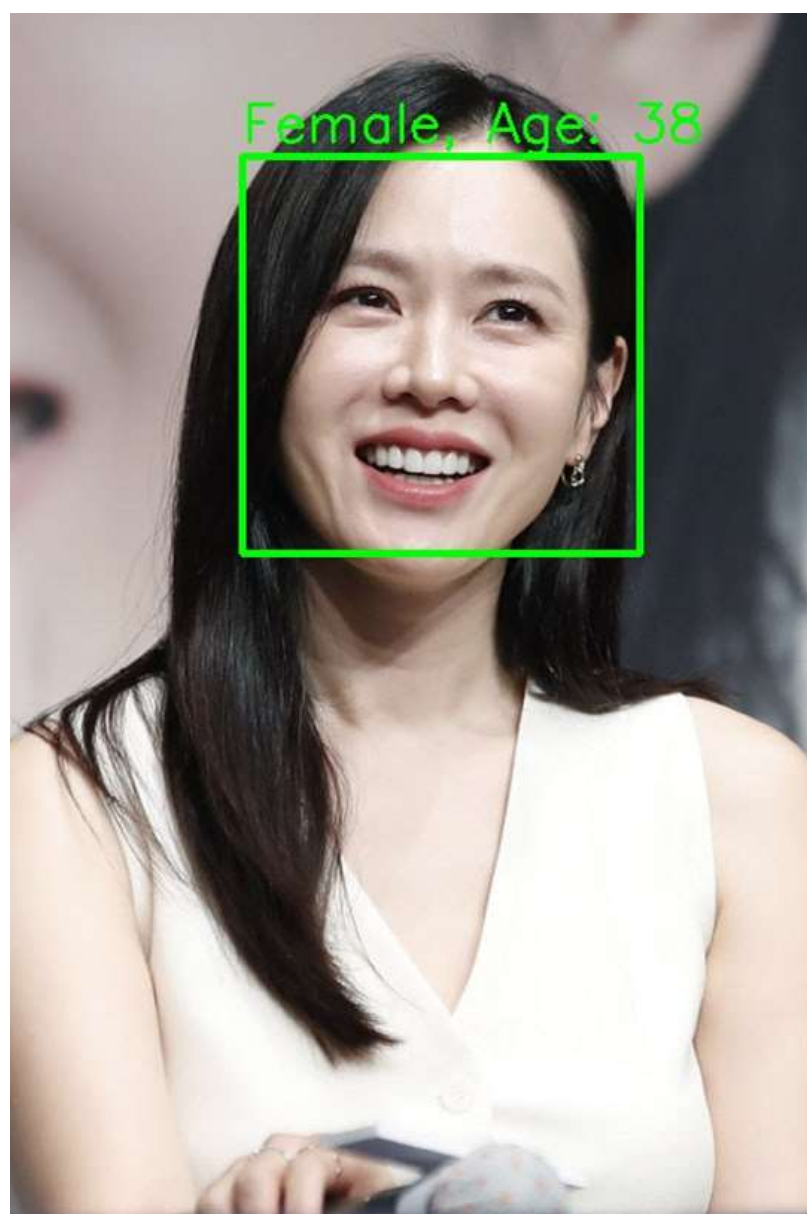


# Model Prediction

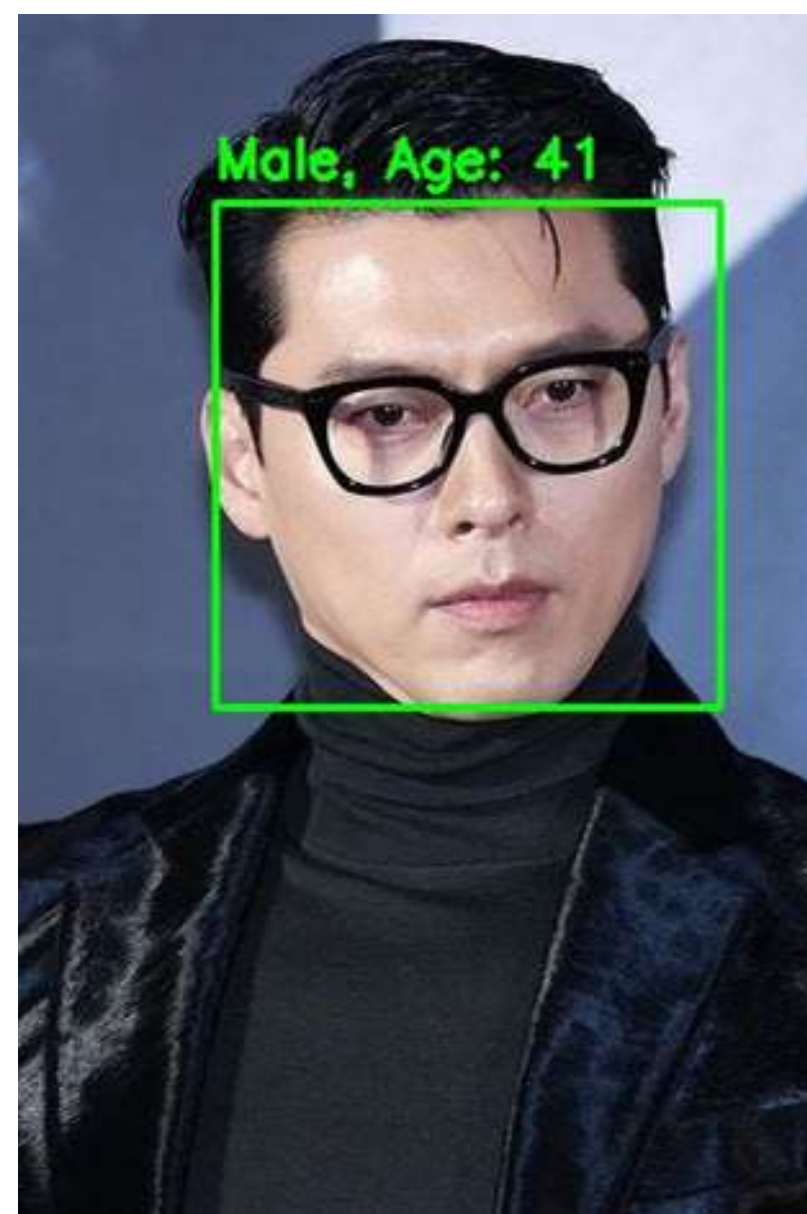
WebCam Prediction



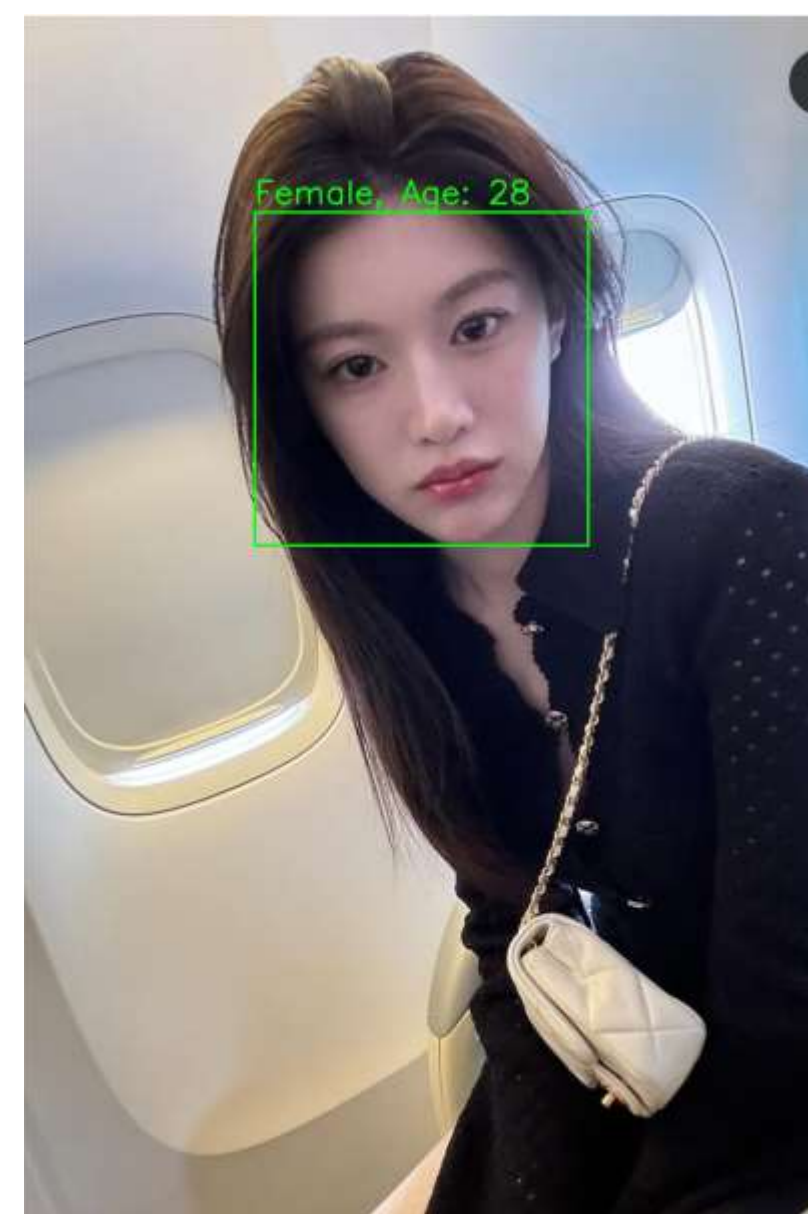
나이와 성별 특징의 비표준화!



손예진(여/만 43세)



현빈(남/만 42세)



고윤정 (여/만 28세)

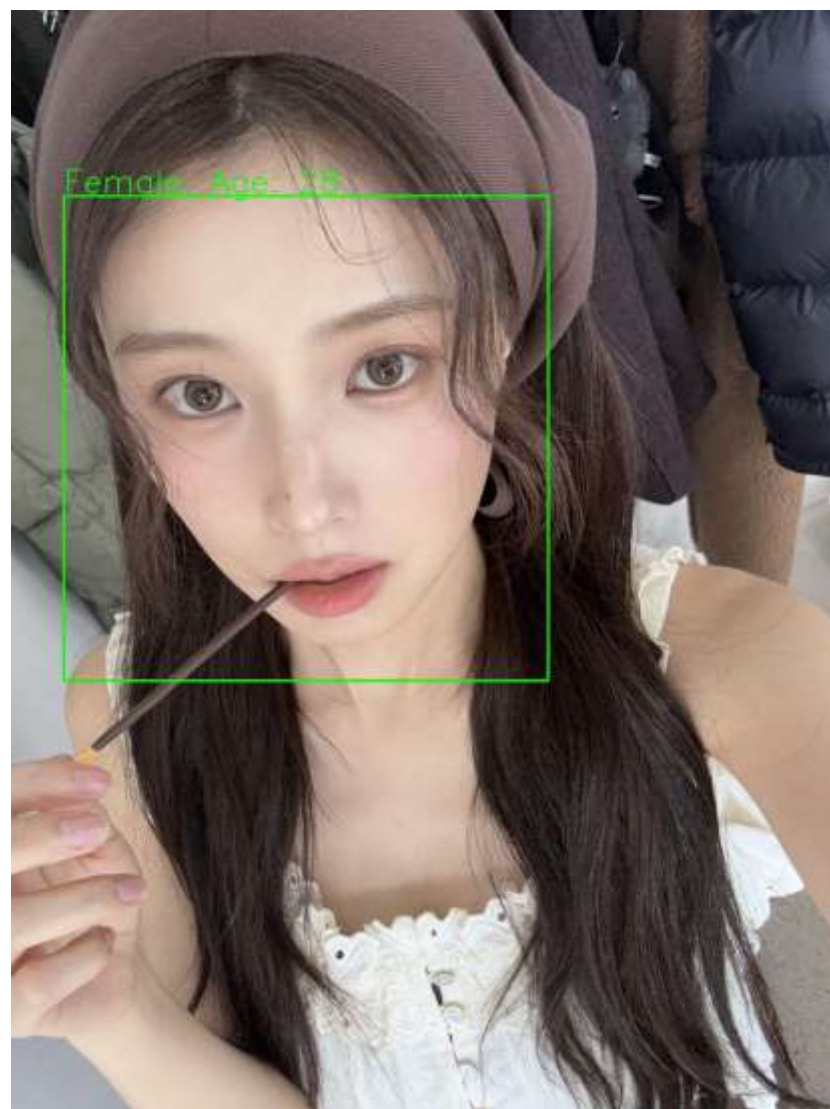


한유진 (남/만 19세)

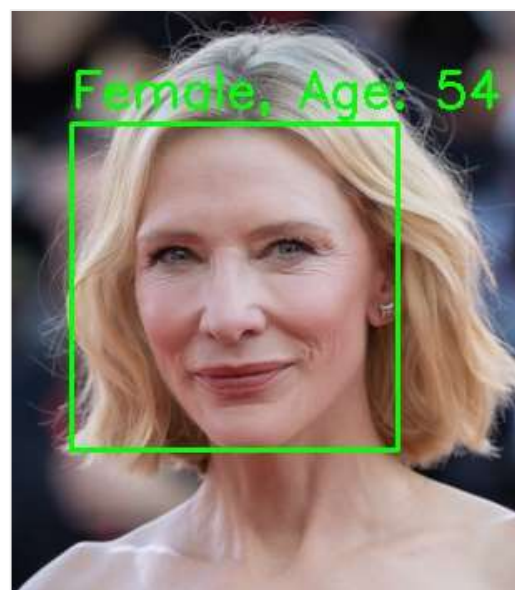


# Model Prediction

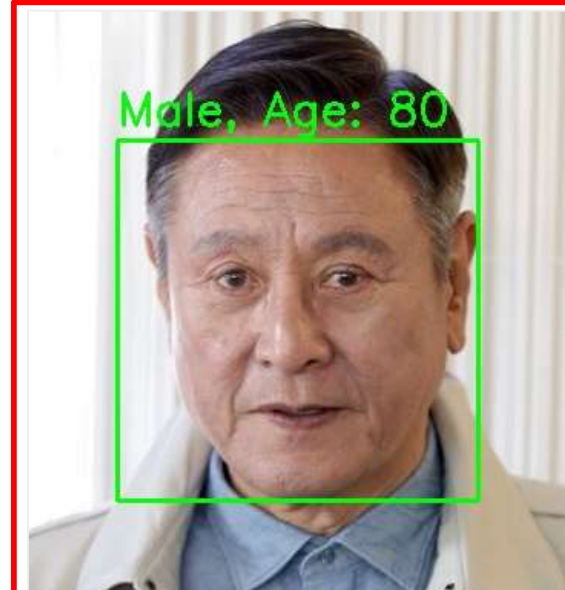
Image Prediction: 동서양 및 남녀 학습도 비교



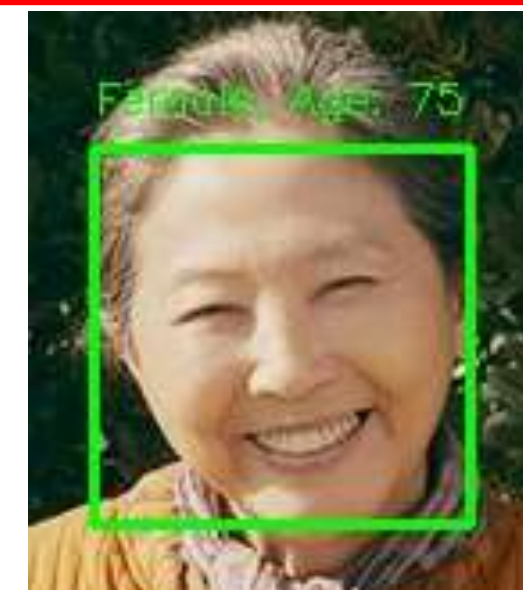
강혜원(여/만 27세)



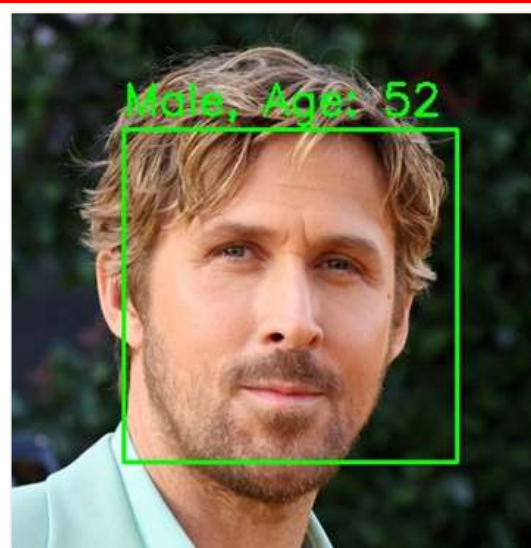
케이트 블란쳇(여/만 55세)



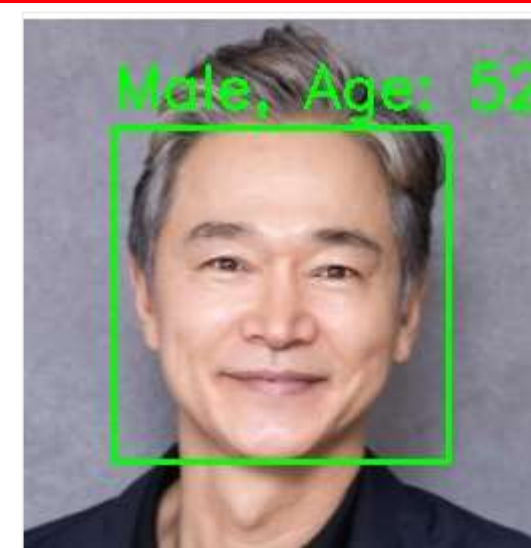
박근혜 (남/만 75세)



고두심 (여/만 71세)



라이언 고슬링(남/만 43세)



정보석 (남/만 62세)



톰 행크스 (남/만 62세)

\* 나이는 사진 기준

# Model Prediction

추론 코드 후보정 처리

## WebCam Prediction

- 후보정 적용 **X**
- 웹캠의 실시간 데이터 ⇒ 데이터 증강 효과
- **WebCam** 추론 성능 > Image 추론 성능

## Image Prediction

\* 후보정

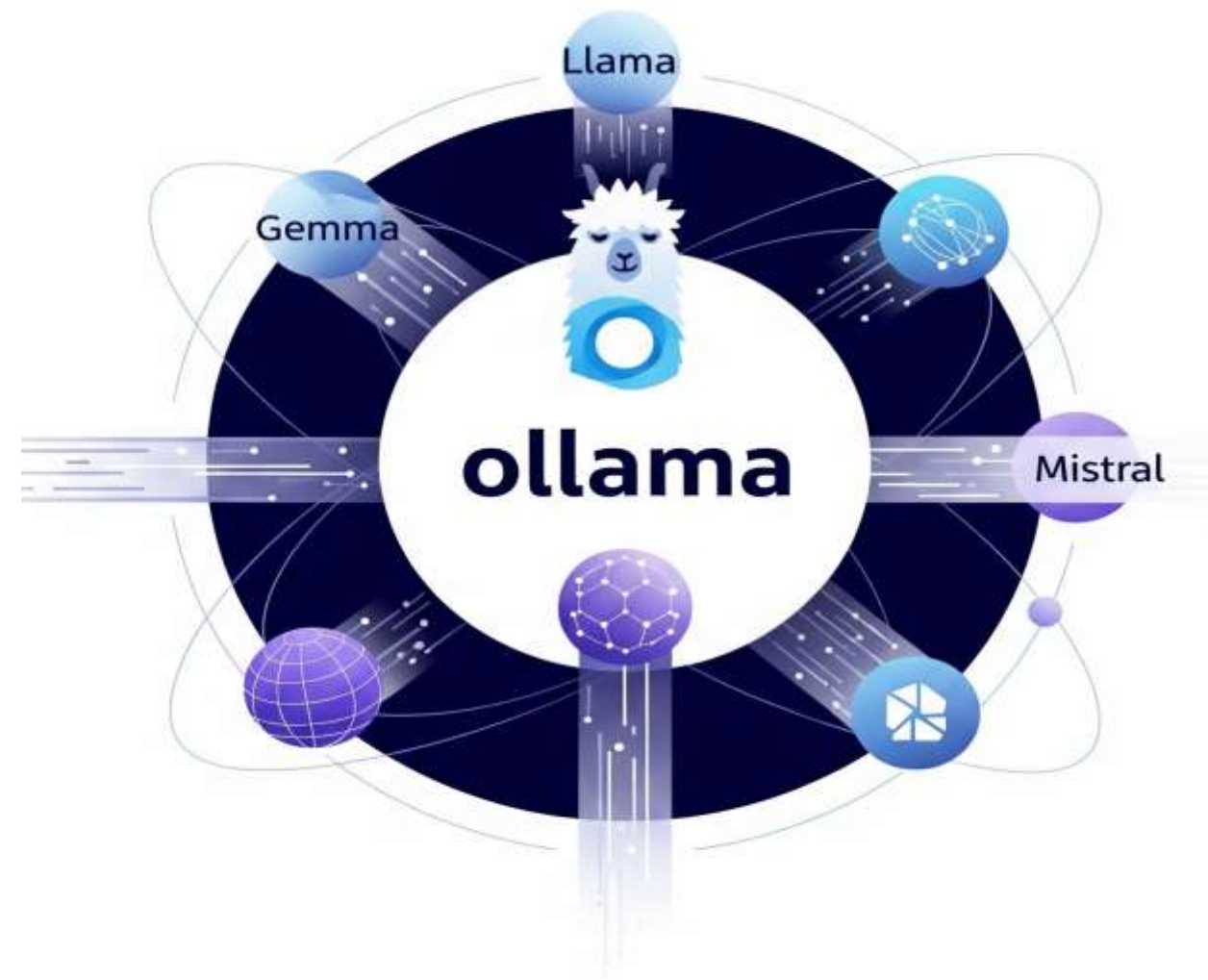
20-30	30-40	40-50	50-60	60-70	70-80
- 7	- 4	+ 5	+ 8	+ 13	+ 18

- 서양인 DataSet 多 ⇒ 서양인에 더 적합한 모델
- 이미지 추론 후보정 적용 ⇒ 기준 나이 대비 서양인: ▲ / 동양인: ▼
- WebCam, Image 추론 모두 여성 대비 남성의 나이 ▲ 판단



## 03 | 모델 통합 및 결과

# Ollama Model



## Ollama ?

→ 로컬 PC에서 다양한 대규모 언어 모델(LLM)을 쉽게 실행할 수 있도록 돕는 오픈소스 도구.

## USE ollama:1b

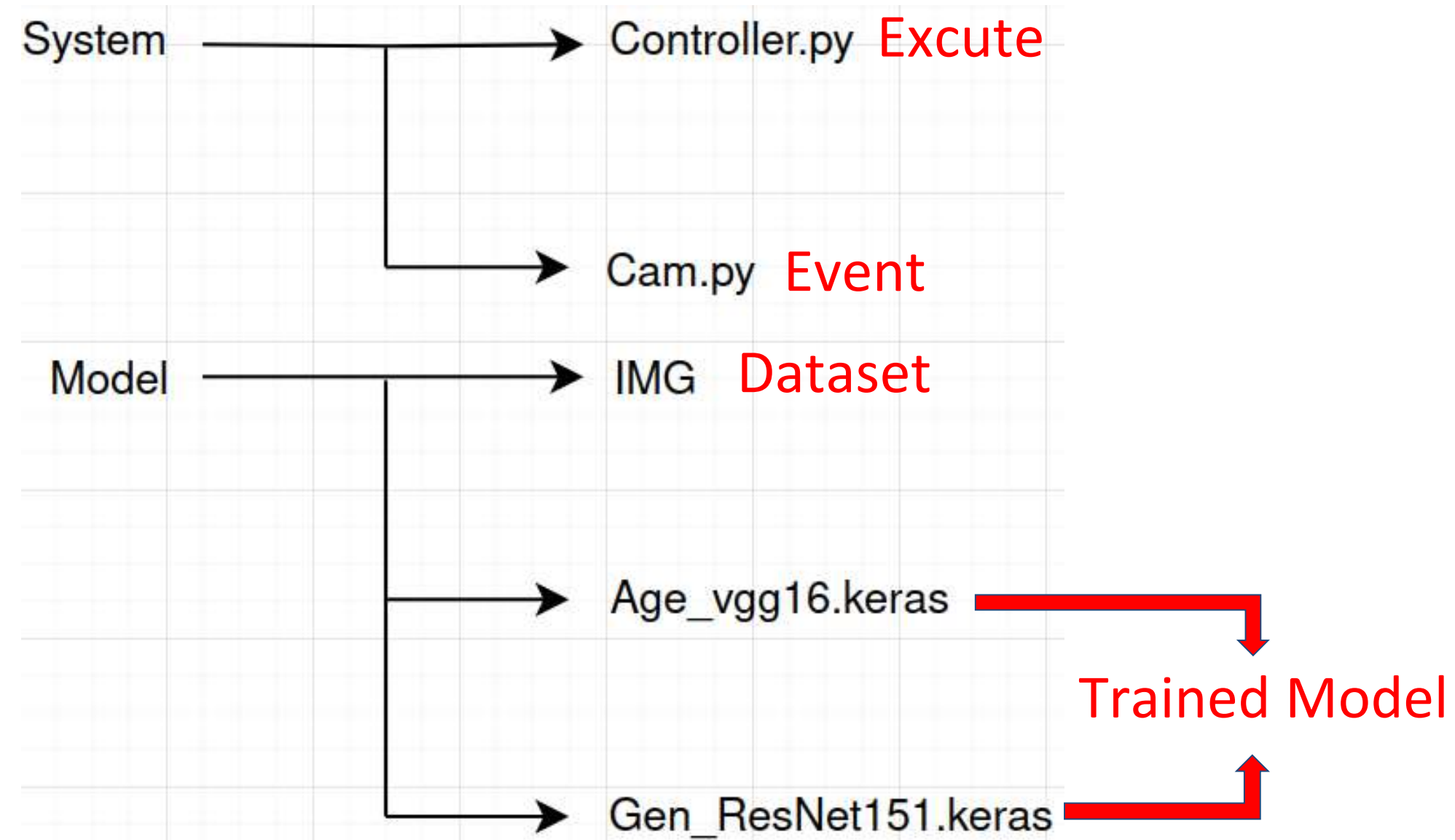
→ 라즈베리파이처럼 리소스가 제한된 환경에서도 구동 가능한 경량 모델에 주로 사용

→ 모델 크기가 작아 메모리와 연산 부담이 적고, 추론 속도도 상대적으로 빠름

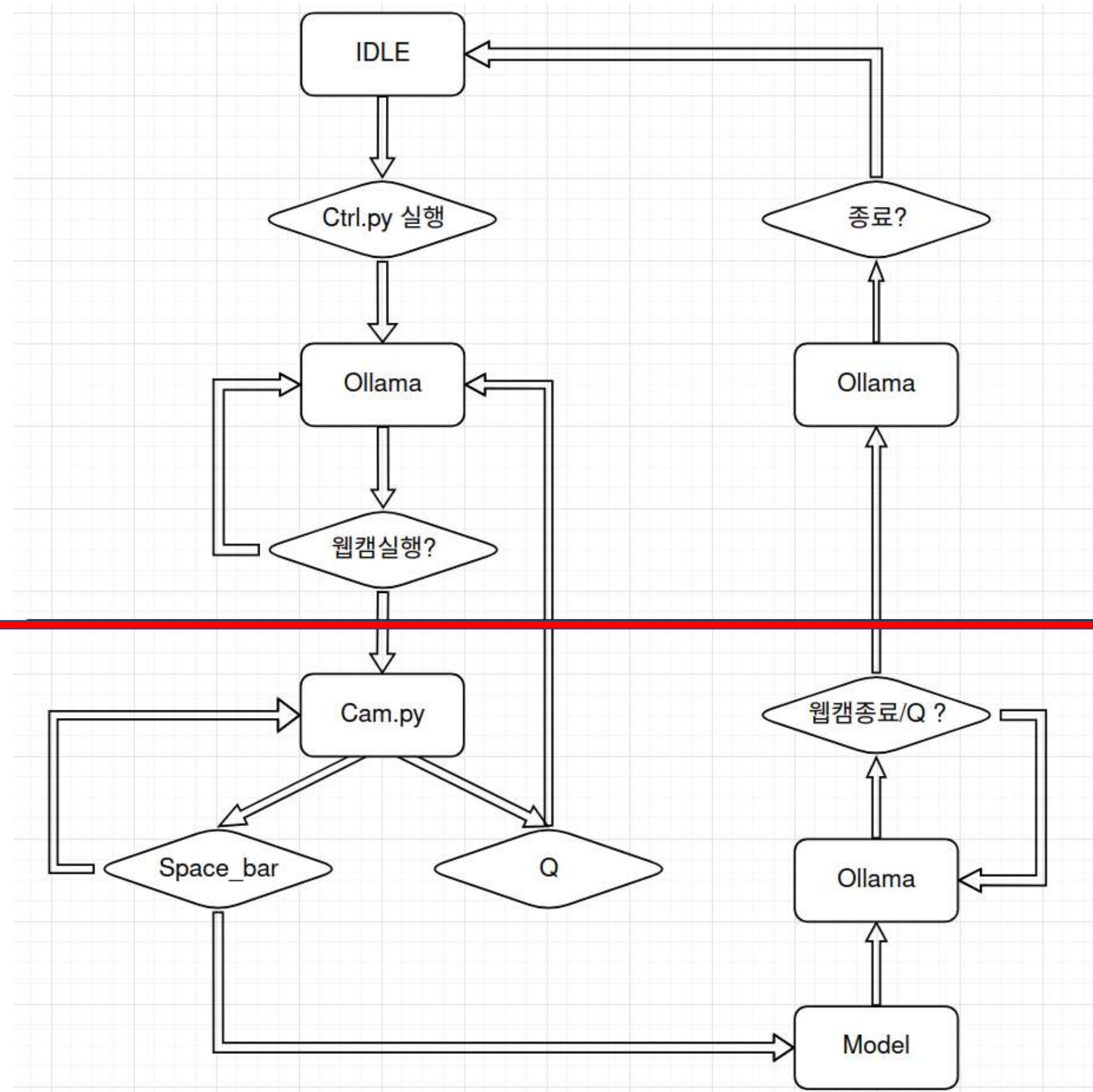


# S/W Architecture

---



# Flow Chart



**Ollama** Env

**CAM** Env



# Main Features

Vgg-16, ResNet-151 모델 사용하여 웹캠으로 사람의 나이, 성별을 거의 정확히 추정

```
~/final > python ctrl.py
```

---

Ollama 컨트롤러 시작!  
Ollama에게 물어보다가 '웹캠실행' 하면 카메라 ON  
'웹캠종료' 하면 카메라 OFF  
'q' 또는 '종료' 입력 시 프로그램 종료

---

나> 웹캠실행



# Main Features

추론한 나이와 성별 값은 자동 완성 프롬프트를 통해 Ollama로 전달 / 답변 하는 동안에는 웹캠 닫음.

[CAM] [LLM 요청] 생성 요청: 나는 34 살 Female 성별의 한국인이야.필요한 주요 영양제 3가지를 알려주고,각 영양제를 권장하는 이유를 5줄로 설명해 줘. 답변은 한국어로만 해줘.

[CAM] \*\*2. 오메가-3 지방산\*\*

[CAM]

[CAM] \* \*\*추천 이유:\*\* 오메가-3 지방산은 심혈관 건강 개선, 뇌 기능 향상, 염증 완화 등에 효과적입니다. 여성호르몬 수치 변화는 혈액 내 염증을 증가시키는데 기여하므로, 오메가-3 지방산 섭취는 혈관 건강을 개선하고 스트레스 완화에 도움을 줄 수 있습니다. 또한, 임신 및 출산 중 여성의 건강에도 중요한 영향을 미칩니다.

[CAM]

[CAM] \*\*3. 단백질\*\*

[CAM]

[CAM] \* \*\*추천 이유:\*\* 여성의 신체는 성장, 유지, 재생 과정에서 단백질이 가장 많이 필요합니다. 34세 여성은 나이가 들면서 단백질 섭취량이 감소하기 쉬우므로, 충분한 단백질 섭취는 근육 유지 및 강화, 면역력 증진, 체중 관리 등에 필수적입니다.

[CAM]

[CAM] \*\*추가적으로 고려할 사항:\*\*

[CAM]

[CAM] \* \*\*개인 맞춤 영양:\*\* 위에 추천한 영양제 외에도 개인의 건강 상태, 생활 습관, 목표에 따라 다른 영양제나 보충제를 고려해야 합니다.

[CAM] \* \*\*균형 잡힌 식단:\*\* 영양제는 보조적인 역할이며, 건강한 식단과 규칙적인 운동을 통해 전체적인 건강을 유지하는 것이 가장 중요합니다.

[CAM]

[CAM] \*\*주의:\*\* 영양제 복용 전에는 반드시 전문가와 상담하시기 바랍니다.

[CAM]

[CAM] 웹캠 재개.



# Main Features

---

```
[CAM] [INFO] 이미 추천을 생성 중입니다. 잠시만 기다려 주세요!  
[CAM] [INFO] 이미 추천을 생성 중입니다. 잠시만 기다려 주세요!  
[CAM] [INFO] 이미 추천을 생성 중입니다. 잠시만 기다려 주세요!
```

답변 준비 중 스페이스 바 입력 예외 처리

```
■ 웹캠 종료 중.  
● 웹캠 종료 완료.  
i 실행 중인 웹캠이 없습니다.  
나> |
```

Ollama에 웹캠 종료 명령

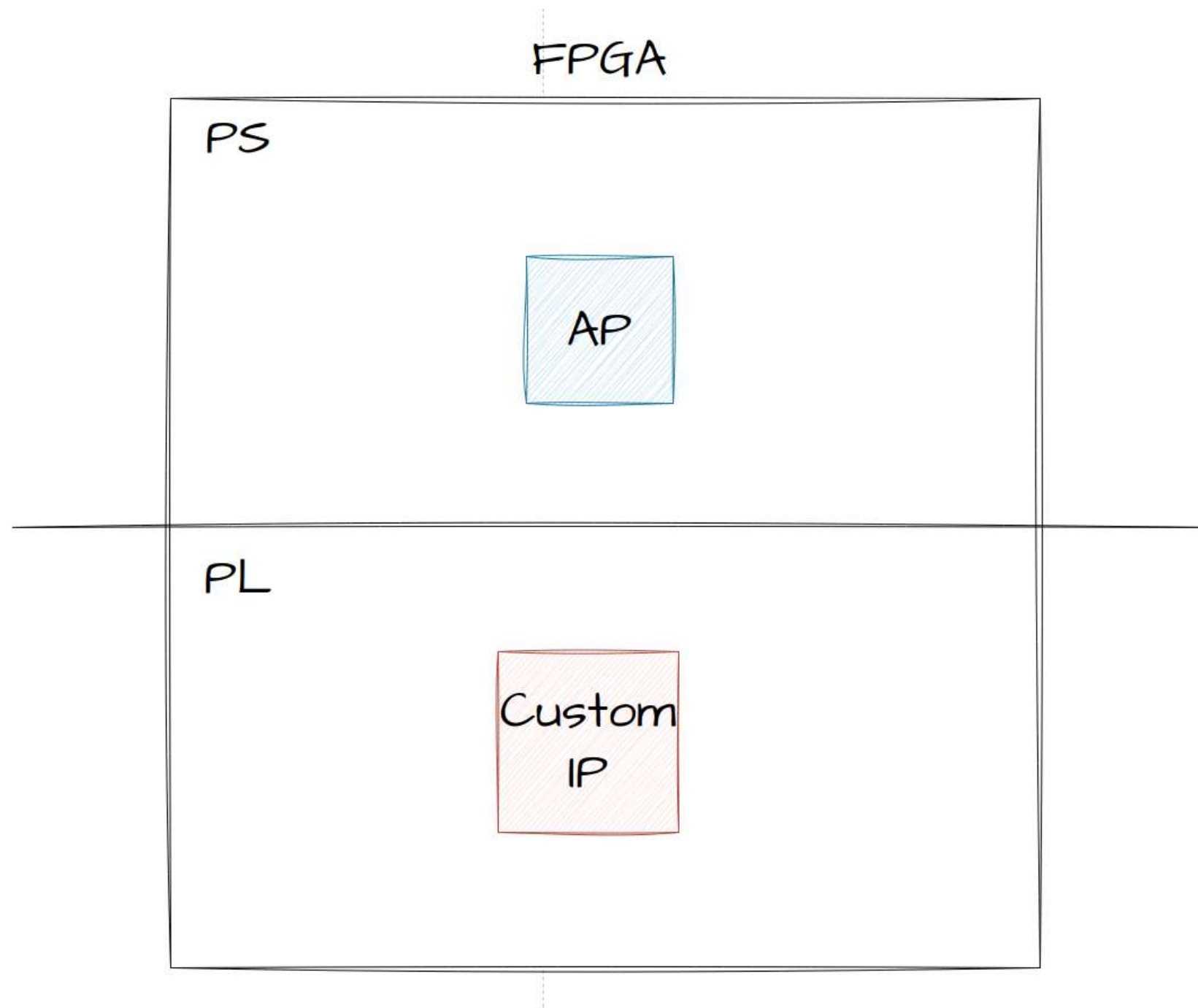
```
나> 종료  
i 실행 중인 웹캠이 없습니다.  
컨트롤러 종료.
```

시스템 종료 명령



## 04 | 가속기 w FPGA

# AI Accelerator: DPU



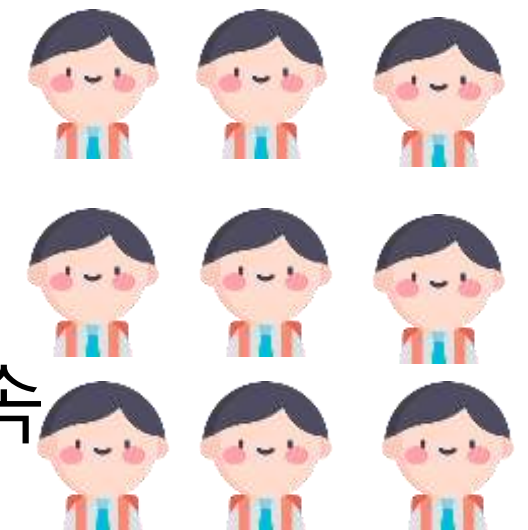
## PS

- Processor
- ARM-Core(ARM Cortex - A53)
- 순차 연산 특화



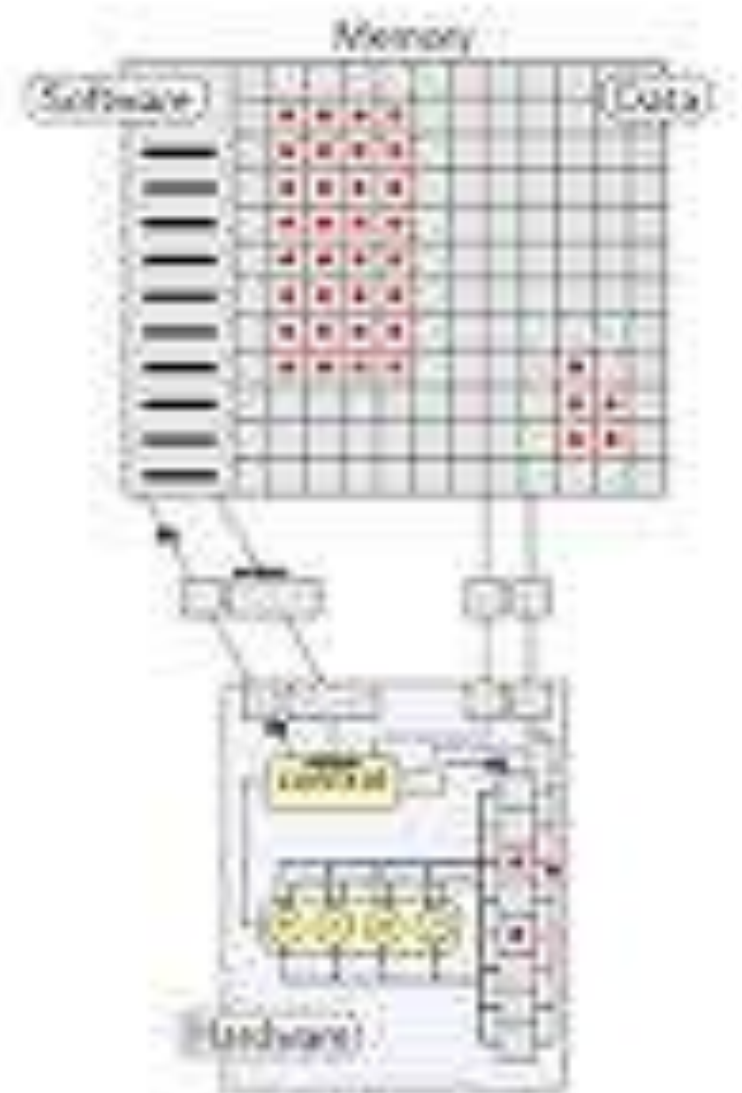
## PL

- Programmable Logic
- 병렬 데이터 처리 특화
- 특정 알고리즘 연산 가속

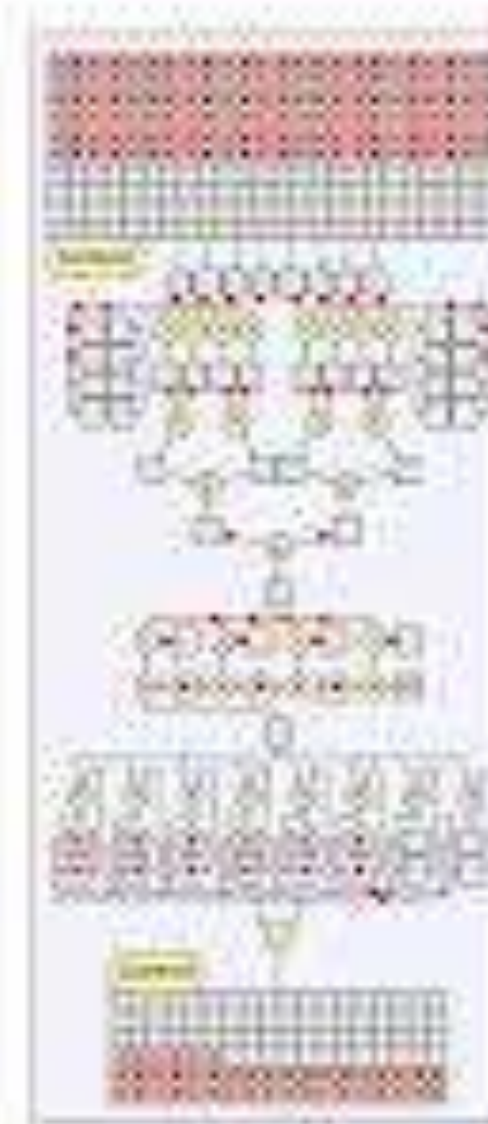




# AI Accelerator: DPU



**Normal processor**  
Software controls computation process



**FPGA**  
Computation integrated in hardware



# How To?



## Vivado & Vitis-AI Platform 사용

> HW Level부터 설계

### Pros)

- HW level부터 설계 → 모델별 최적화 가능
- 특정 알고리즘을 위한 가속기 설계 가능

### Cons)

- 설계 난이도 급증
- FPGA linux 이미지를 직접 제작  
→ 검증 X



## PYNQ OS

> 미리 구성된 HW 제공

### Pros)

- 가속기(DPU)가 포함된 리눅스 이미지 제공
- SW 상위수준에서 HW 제어 가능
- 다양한 python 라이브러리 제공

### Cons)

- HW Level에서 최적화 불가능
- Python 라이브러리의 경우, 시스템 OS에 의해 보호됨  
→ 사용자 Customizing에 제약

04

# BenchMark: Performance Test

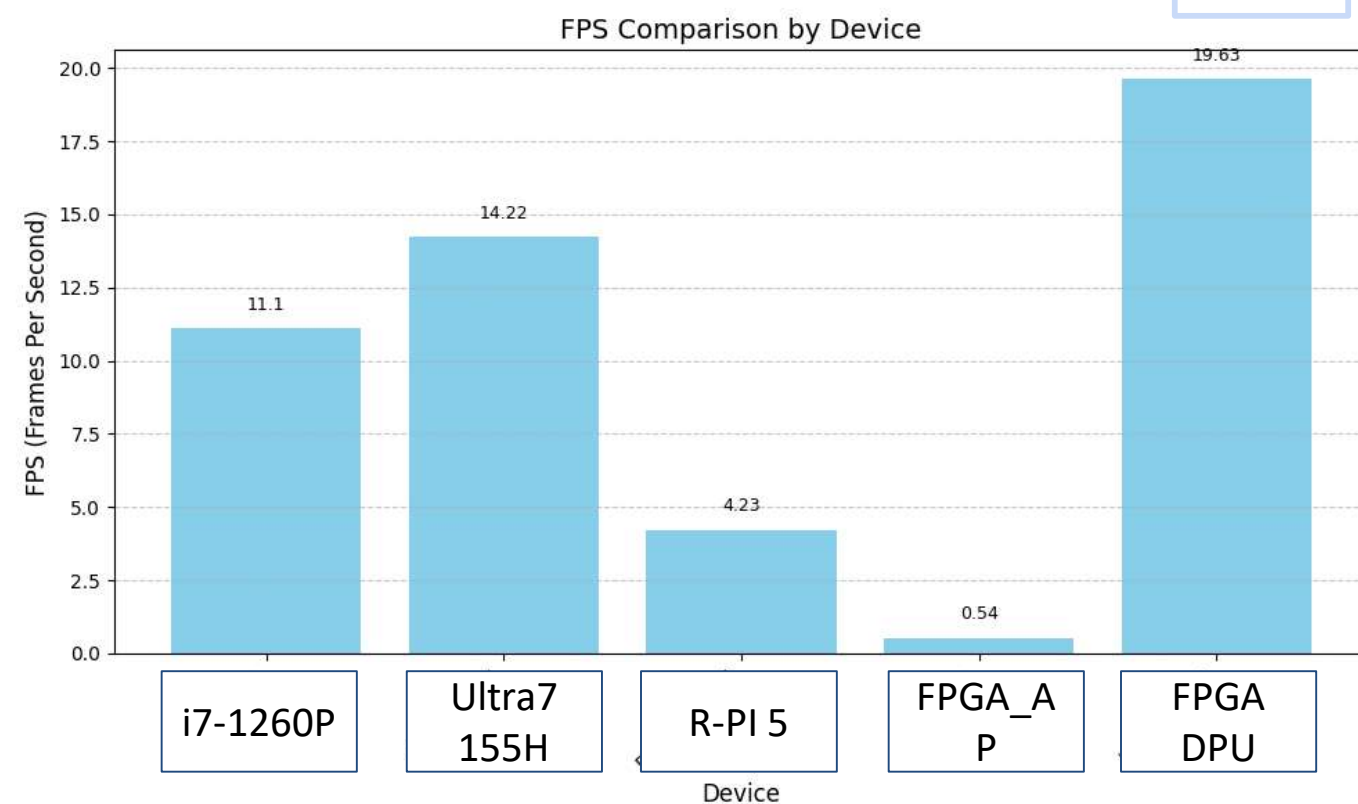


	i7-1260P	Ultra7 155H	RaspberryPi 5 (ARM Cortex A76)	Ultra96V2 (ARM Cortex A53)
Max CLK Speed	4.7GHz	4.8GHz	2.4GHz	1.5GHz

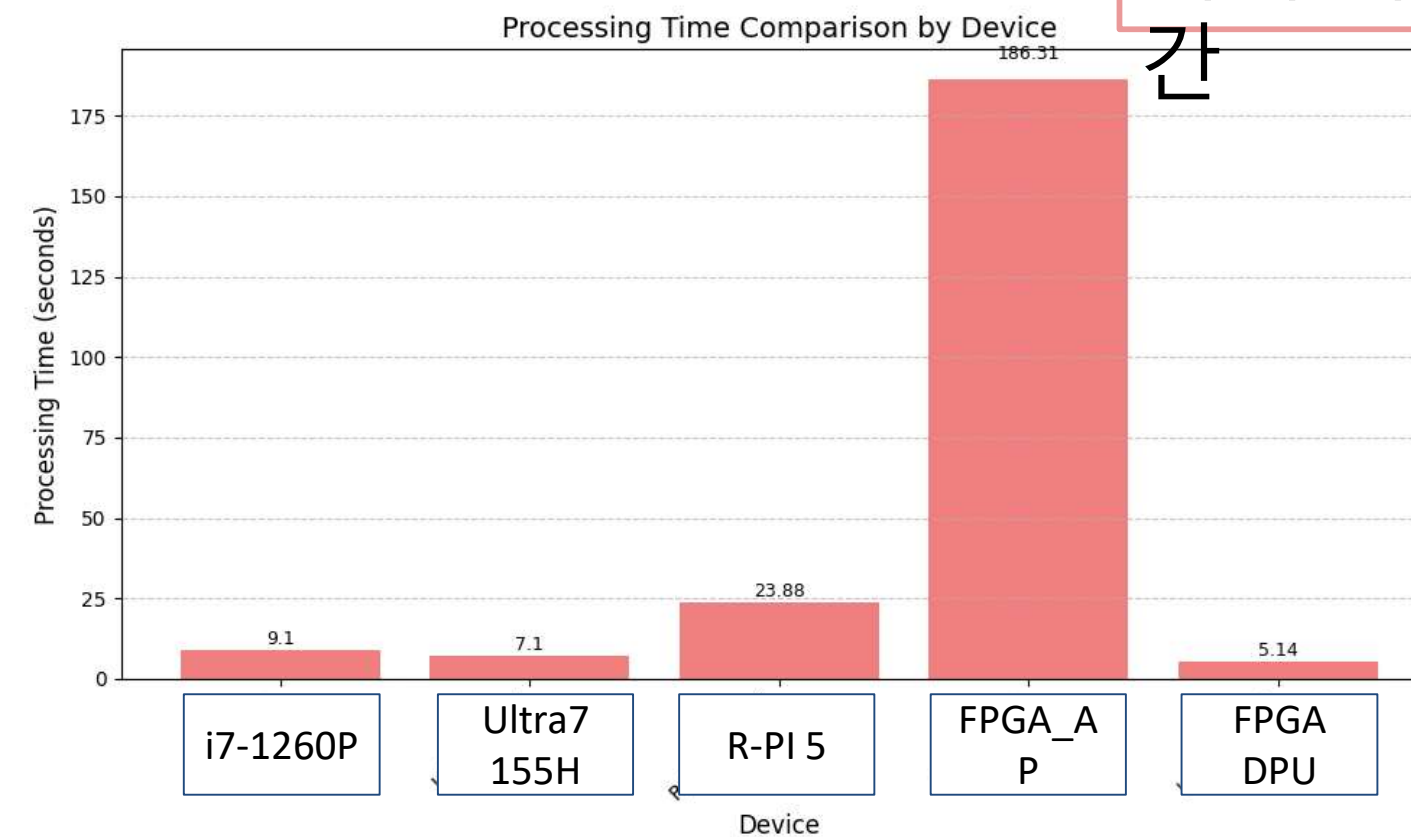
05

# ResNET50

FPS



처리 시간

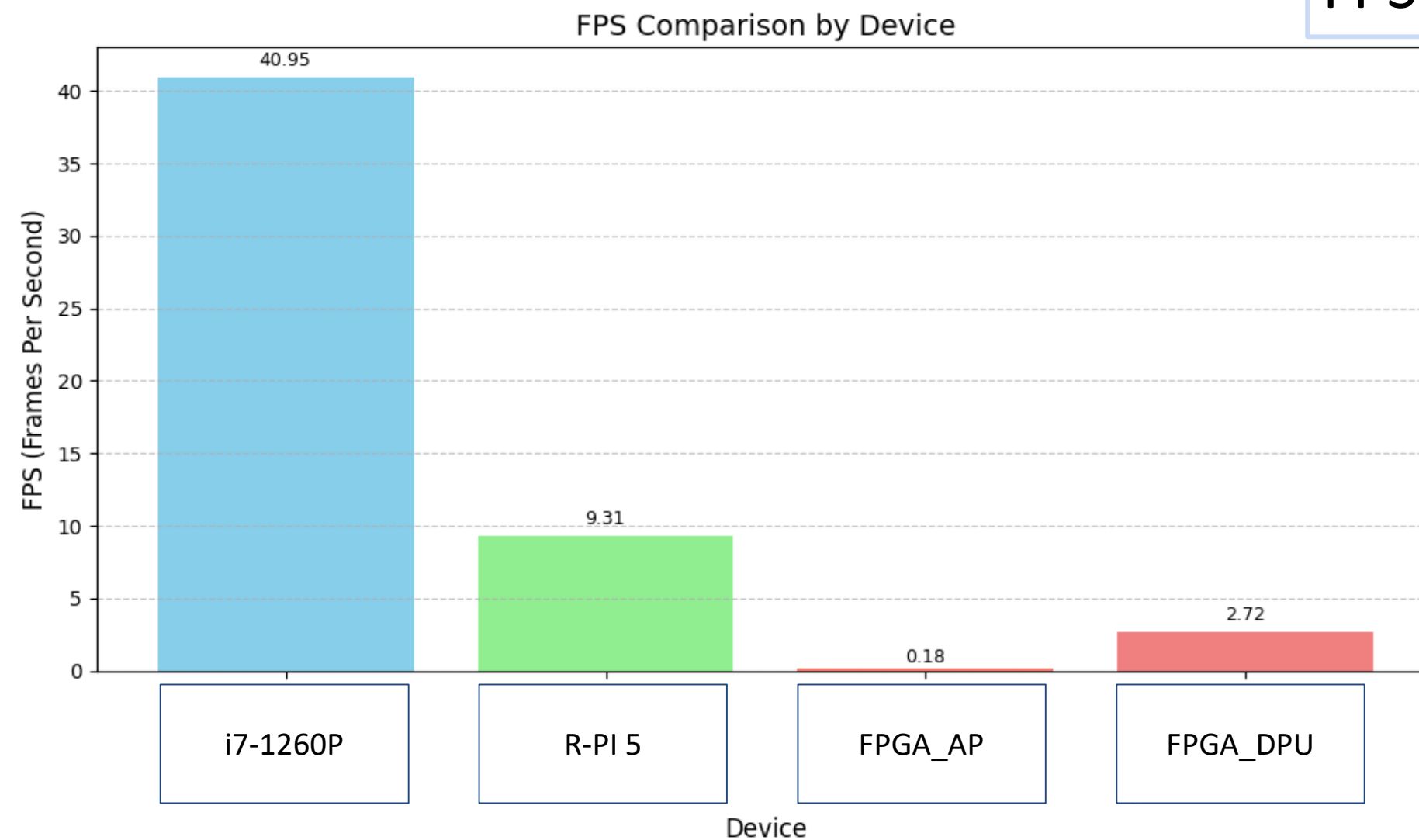


	i7-1260P	Ultra7 155H	RaspberryPi 5	Ultra96_AP	Ultra96_DPU
FPS	11.1	14.22	4.23	0.54	19.63
처리 시간(sec)	9.10	7.1	23.88	186.31	5.14

Performance X36 !!

06

# YoloV3



FPS

	i7-1260P	RaspberryPi - 5	Ultra96V2_AP	Ultra96V2_DPU
FPS	40.95	9.31	0.18	2.72

Performance X15 !!

07

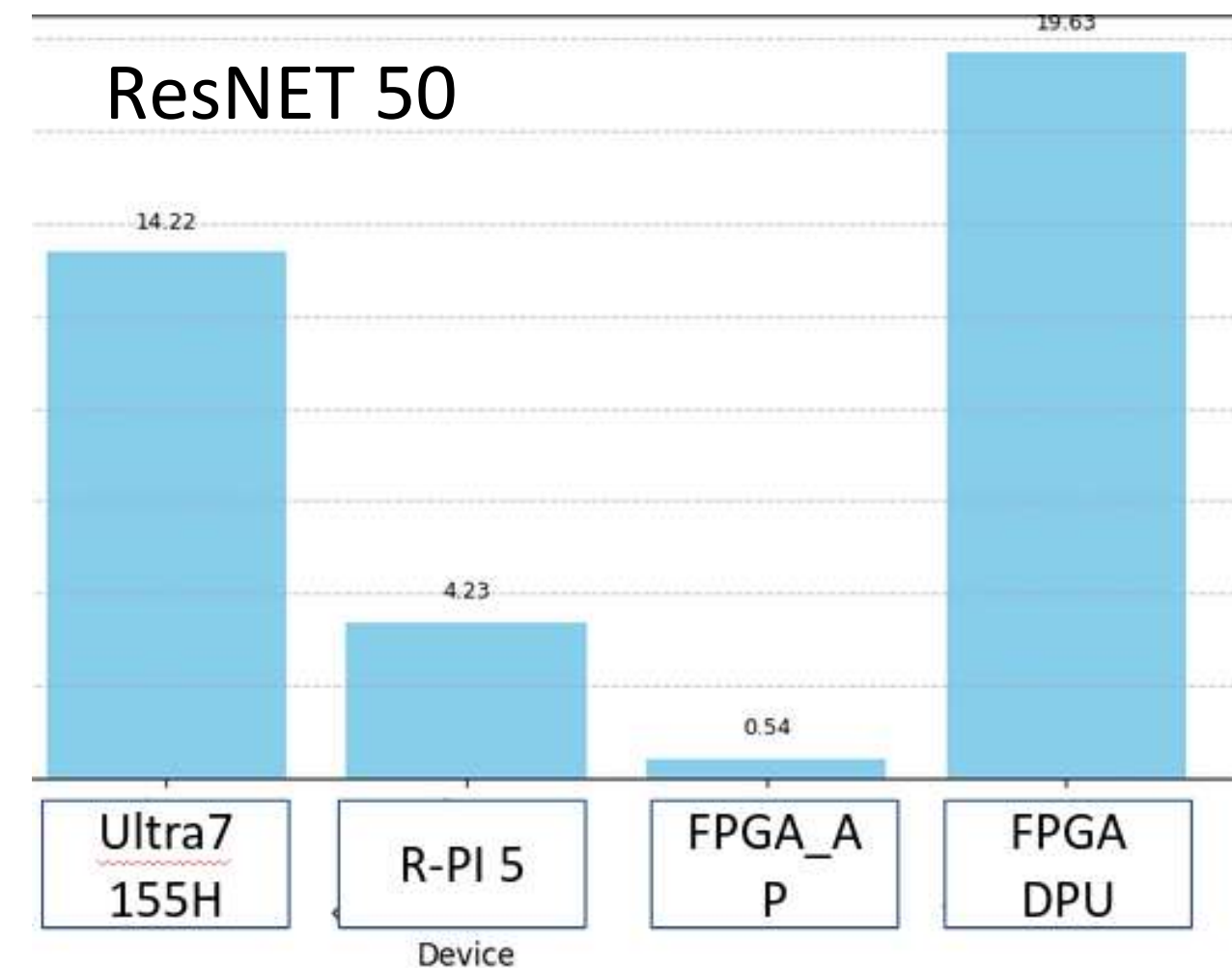
# DPU: ResNet Outperforms YOLO

YOLO\_V3

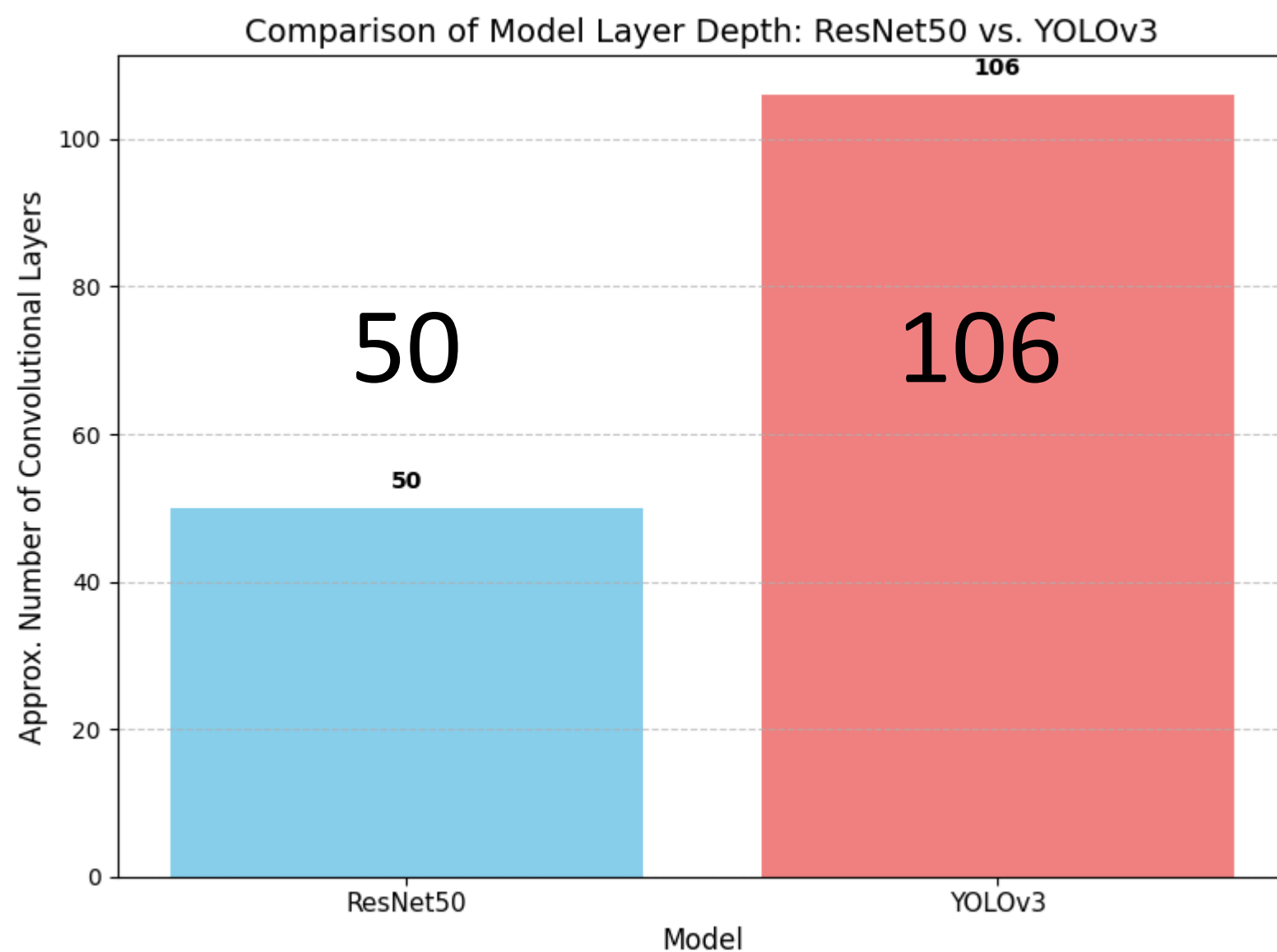
Ultra96V2_AP	Ultra96V2_DPU
0.18	2.72

**Performance X15 !!**

ResNET 50



# DPU: ResNet Outperforms YOLO



## Layer 구성

- ResNET50: 50 Conv Layer
- YoloV3: 53 BackBone(Conv) Layer + 추가 탐지 레이어
  - 후처리 로직(특히 NMS)
  - 순차 처리 + 조건 분기 → CPU가 효율적

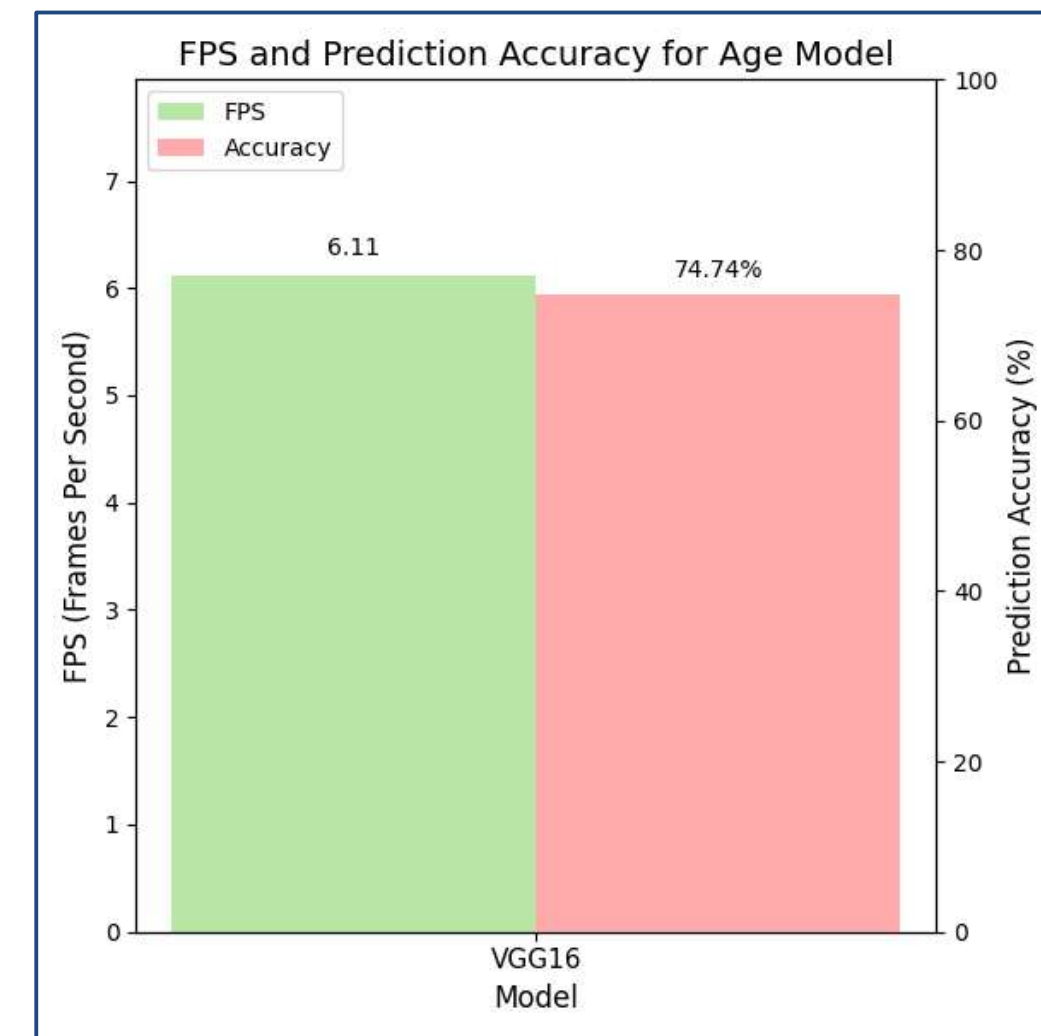
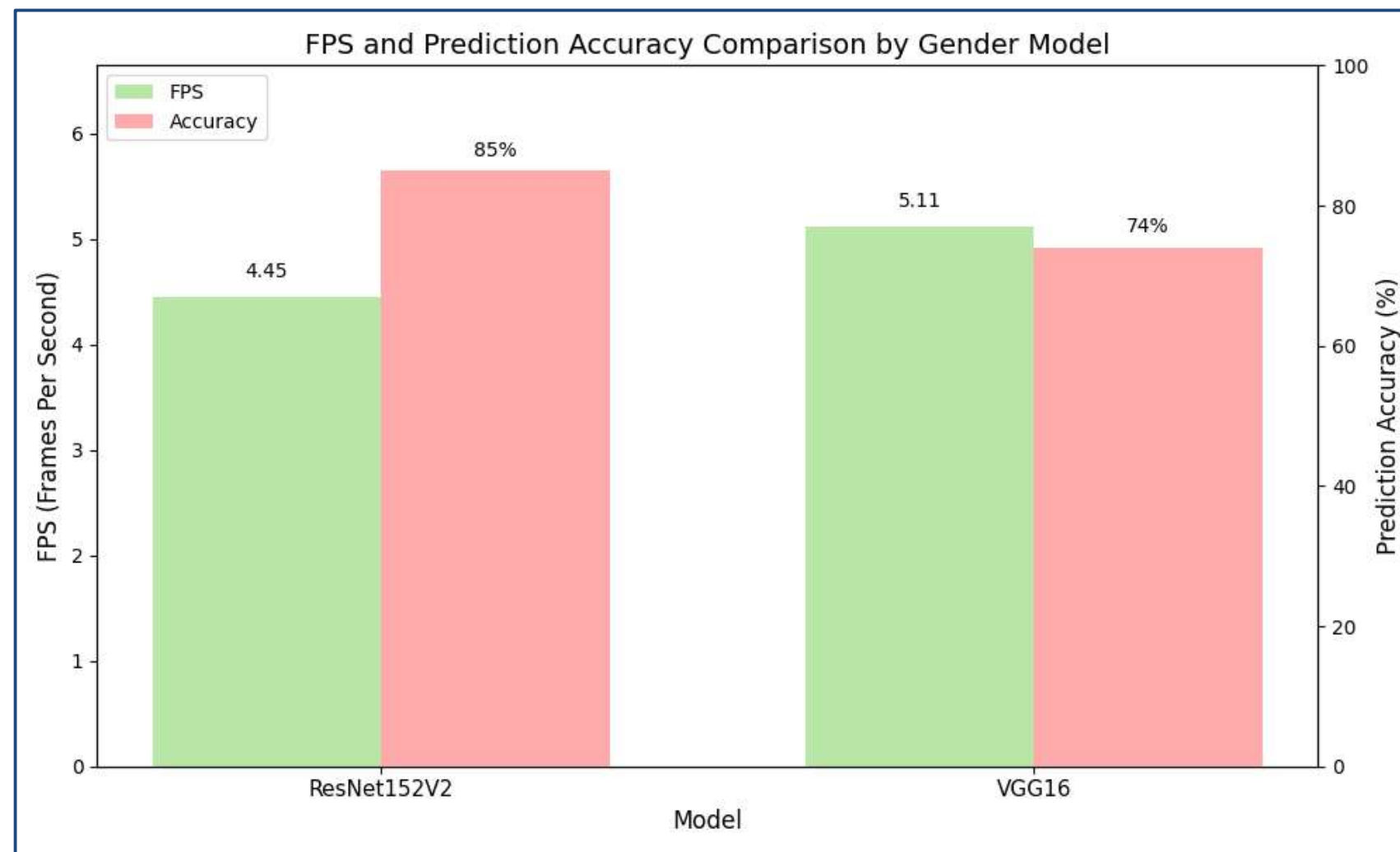
## DPU

- 합성곱 연산에 최적화
- 모든 딥러닝 연산에 가속 불가능



# VGG16

- VGG16 모델로 성능 비교 시도 ⇒ 리소스 부족으로 Ultra96-V2에 업로드 실패
- 일반 데이터가 아닌 UTKFace Data로 예측 정확도 추론 성능 비교





## 05 | Trouble Shooting

# 05 Trouble Shooting: Trained AI Model



## Problem

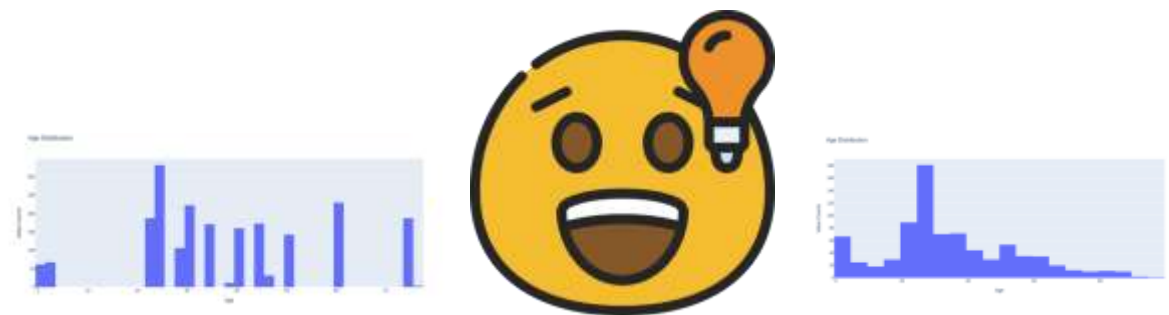
### [DataSet 불균형 문제]

나이 분포가 정규분포가 아닌 비대칭적인 구조

특히, **20대** 데이터가 전체의 대부분을 차지  
→모델이 **특정 연령대에 과도하게 최적화**되는  
편향(Bias)을 유발 가능성 UP:  
실제로 **28살만** 출력됨

### [라벨링 오류]

다른 연령대 데이터가 잘못 포함됨



## Solution

**[DataSet 교체 ⇒ 중년층 Data 확보]**

**[학습 모델 변경: 코드 교체]**

전이학습:

Feature Extraction or Fine-tuning 가능 (선택)  
→ Feature Extraction only

**Backbone Model:**

<EfficientNetB3> →  
<VGG16, ResNet50V2/152V2, Xception,  
InceptionV3, MobileNetV3Small/Large>

# 05 Trouble Shooting: Trained AI Model



## Problem

### [낮은 학습 성능]

**WebCam** 추론 진행 시,  
실시간으로 나이 추정값이 변화 → 속도 ▼  
나이 추정 변동성이 큼 → 정확도 ▼

**Image** 추론 진행 시,  
WebCam 대비 → 정확도가 크게 ▼  
얼굴 인식을 못 하는 문제 ▲  
→ 얼굴 각도, 크기, 10년 전 데이터에 기반한 얼굴



## Solution

### [학습 데이터 교체 및 개수 변화]

20,000(D1) → 900(D2) → 10,000(D2) → 2,000(D3)

**Age:** 데이터 품질 즉, 나이 분포도 → 성능

**Gender:** 학습 데이터 개수 → 성능

얼굴을 인식하고 나이를 추정할 충분한 시간 제공  
(3s)

후 고정 값 출력 → 정확도 및 속도 개선

### [추론 모델 후보정 처리]

UTKFace Data가 아닌 일반 사진으로 평가  
→ 경향성을 평가 및 후보정에 반영

# 05 Trouble Shooting : Integrated System



## Problem

웹캠 프레임에서 얼굴 검출과 추론을 수행하면서 영상이 버벅이고 느려지는 현상 발생

- 매 프레임마다 얼굴 검출 및 추론
- CPU 과도 사용
- 웹캠 영상 끊기거나 응답 지연 발생



## Solution

예측 빈도를 낮춰 불필요한 추론 연산을 감소시켜 시스템 부하를 감소시킴

- 초당 1번만 추론 되게 진행
- CPU 사용률 감소, 상대적으로 영상 부드럽게 출력
- 정확도의 큰 차이는 없음!



# 05 Trouble Shooting : Integrated System



## Problem

### [Raspberry PI 에서 모델 추론 지연]

→ 초당 1회만 추론하도록 제한했음에도 불구하고, 웹캠과 Ollama를 동시에 실행하면 모든 CPU 코어가 사용되어 성능 저하 및 시스템 불안정 문제가 발생함.

-> 최적화 필요!



## Solution

### [리소스 관리 중심의 알고리즘 최적화]

→ Ollama가 답변을 준비하는 동안 웹캠을 닫고, 답변이 완료되면 웹캠을 다시 재개하는 방식

추론 중 발생하던 CPU 과부하와 영상 지연을 해결하고, 웹캠과 LLM 추론을 안정적으로 병행함.

# 05 Trouble Shooting : Accelerator



## Problem

### [VITIS AI 플랫폼 호환 문제]

Vitis-AI 플랫폼 양식 미준수 시

→ 파일 읽기 혹은 변환과정 손상 발생

플랫폼 시스템이 보호하는 라이브러리

→ 보호 라이브러리 외 다른 버전 설치 시  
시스템 에러 발생



## Solution

### [가상 환경 생성]

플랫폼 내에 가상환경 구축

→ 플랫폼의 프로그램 이용 시 컨테이너 로컬에  
서 수행

→ 컨테이너 내 다른 버전 사용 필요 시 가상환  
경 생성

→ 그 때 필요한 라이브러리만 가상환경에 구축

# 05 Trouble Shooting : Accelerator



## Problem

[웹캠 송출 및 os 이미지 충돌]

평상 시

PYNQ OS의 Xwindow 화면 송출

웹캠 구동 시

Xwindow와 웹캠 출력이 DP Port에서 충돌



## Solution

[스위칭 옵션 추가]

`"print(overlay.ip_dict)"`

→ DP 포트에 해당하는 IP 및 구조 파악

→ DisplayPort라는 Pynq 라이브러리 제어

코드 실행 시: Xwindow OFF/ Webcam ON

코드 종료 시: 원상복구

# 05 Trouble Shooting : Accelerator



## Problem

### [.xmodel 파일 생성]

PYNQ-DPU를 효율적으로 사용 위해선

AI 모델 → xmodel파일 변환 필요

### [Problem]

1. Vitis-AI 미지원 모델 존재
2. .xmodel 파일 사이즈 大 → 커널 Down



## Solution

### [지원 버전]

Vitis-AI 컨테이너 제공 xmodel 리스트 확인

→ 해당 리스트에 존재하는 모델 이용

### [경량화]

Keras → tflite 파일 변환 후 xmodel 변환





## 06 | 시연 및 고찰

"웹캠 실행" 입력

01

# 고찰

병렬 데이터 多  
(Conv)

DPU

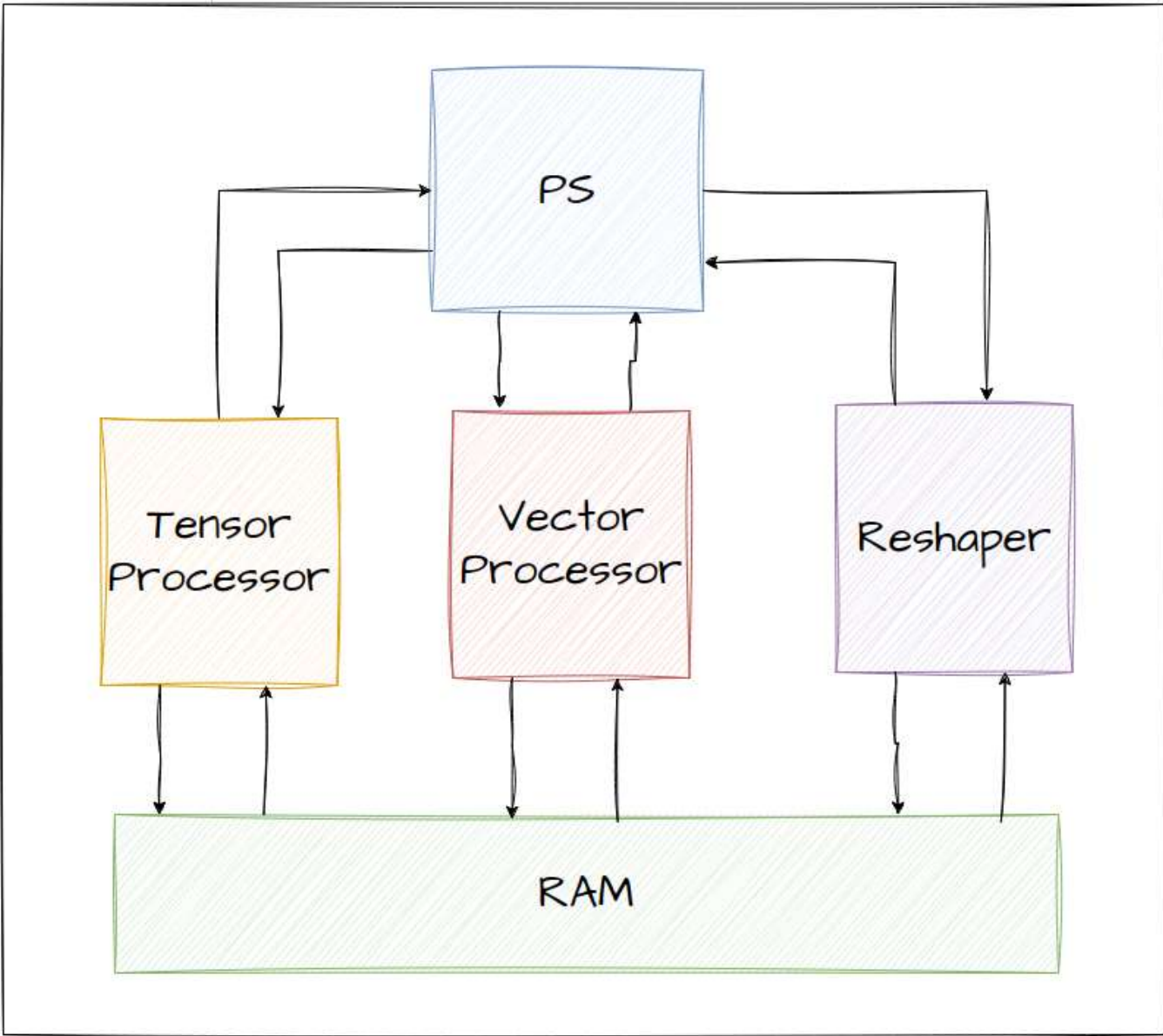


Performance ?

Sometimes  
CPU << PS w DPU

복잡한 후처리 연산  
과 단순 요소 연산은  
?

합성곱 연산을 제외한 나머지 연산을 가  
속하는  
보조 Processor



	Tensor Executor	Vector Executor
주요 연산	Matrix Multiplication Convololution	벡터 요소별 연산 활성화 함수 정규화
병렬 처리	대규모 병렬연산 처리 (여러 행렬 곱셈 동시 처리)	SIMD 병렬 연산 (하나의 명령어로 여러 벡터 요소 처리)
연산 분야	신경망 주요 레이어 (Conv, FC)	Pooling Batch Activation Fn
연산 복잡도	대량의 병렬 데이터 연산 특화	간단한 반복작업 특화
주요 목적	AI 모델의 핵심 연산 가속	데이터 전처리/후처리 보조 연산



# 기대 효과

01

## 데이터 I/O 효율성 향상

Tensor Executor가 일하는 동안 Vector Executor가 전/ 후 처리 연산 수행 → **Pipelining** 성능 향상

02

## 범용성 및 유연성 증대

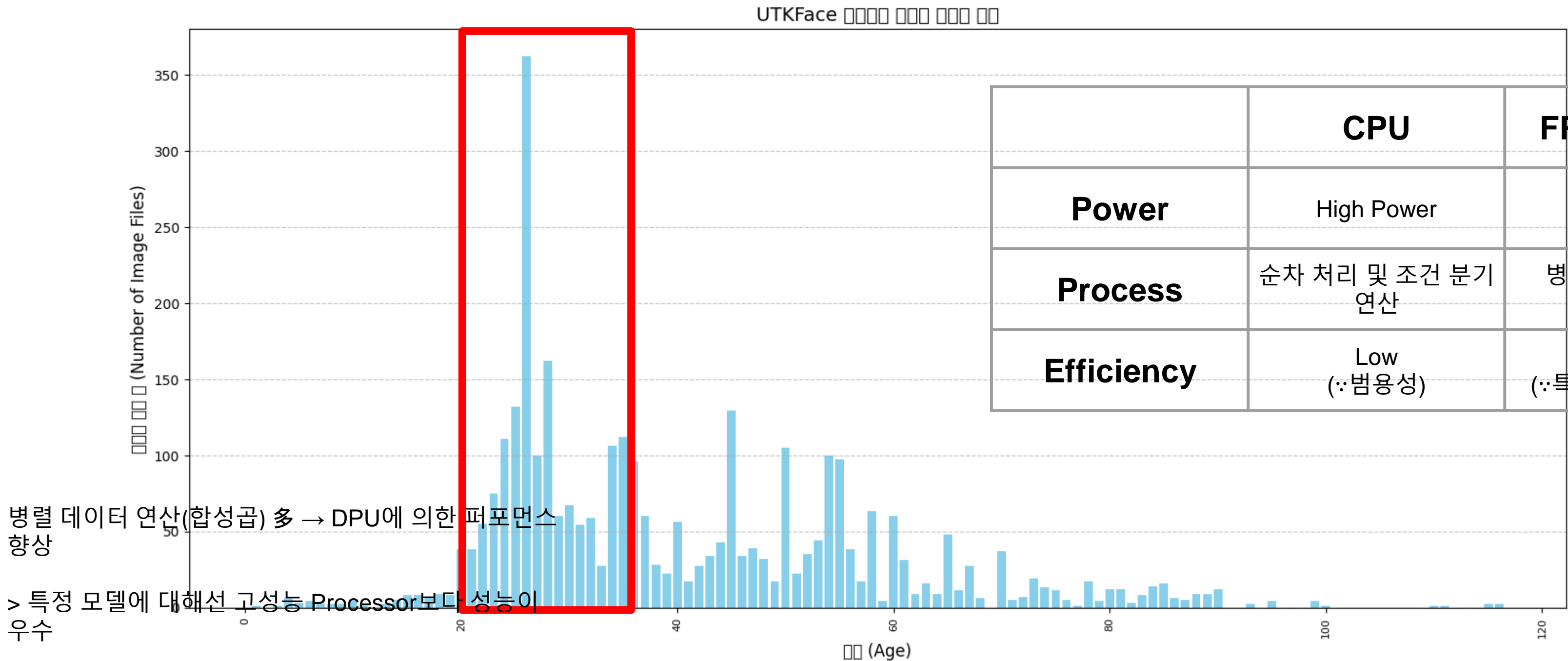
합성곱 혹은 병렬 연산에 특화된 Tensor를 Vector가 보조해줌 → **다양한 딥러닝 모델**을 가속 가능

03

## 전력 효율성 향상

기존: 병렬 데이터 처리 이외 연산 CPU 부담 → Vector Executor 도입으로 **CPU 부하를 감소**

Q&A



	CPU	FPGA(ASIC)
Power	High Power	Low Power
Process	순차 처리 및 조건 분기 연산	병렬 데이터 연산 (PIPO)
Efficiency	Low (∴범용성)	High (∴특정 작업 최적화)

기존: 합성곱 연산에 특화된 DPU만 사용

→ YoloV3와 같이 후처리 로직 및 벡터 요소 연산이 多 경우

→ DPU가 효율적으로 처리할 수 없음

→ AI 연산 가속화 성능이 저하됨

## 28살 데이터셋 분포 이미지

구분	명령어 (Instruction)	데이터 (Data)	설명
SIS D	1개	1개	순차적 실행 방식 (전통적인 CPU)
SIM D	1개	여러 개	같은 연산을 여러 데이터에 동시에 적용 (GPU, 벡터 연산)
MIS D	여러 개	1개	특수한 연산 방식 (에러 검출 시스템 등)
MIM D	여러 개	여러 개	각 프로세서가 독립적으로 다른 연산 수행 (멀티코어 CPU, 클러스터 컴퓨팅)

✓ 병렬 컴퓨팅에서 가장 널리 사용되는 방식은 SIMD(GPU)와 MIMD(멀티코어 CPU)

알고리즘 파트:

- 데이터셋, 학습에 활용된 모델 소개(3분 내로), 성별 나이에 따른 회귀, 이진 모델 설명(짧)
- 결과: 학습된 모델별 Loss, Accuracy 그래프 -> 최적 모델
- 웹캠을 통한 성능 테스트(사진으로 보여주기)

H/W 비교 파트

VGG16을 통한 UTKFace 나이, 성별 모델 성능 체크(미완이지만 약간 여백의 미처럼....)

전체 순서

개요 → 학습과정 → 모델 개발 → 모델 최적화 →올라마 → 가속기에 따른 성능 비교 → 고찰(trouble shooting)

학습과정: 후보 모델들의 정확도 그래프 보여주고 → 선정 이유

ResNet-50, vgg16 기본 모델구조, 추론하는데 사용한 모델 구조

학습된 데이터와 올라마와의 연동, 인터페이스

H/W 비교 ResNet-50 모델, 동물 이미지 데이터셋 라즈베리파이

개발 한계점, 개선할 점



- 가속기 여부에 따른 추론 성능 비교
- 보드에 가속기 포팅한 과정
- HW 레벨에서 DPU 포함하는 방법
- 미리 제공되는 OS 사용(PYNQ)
- ResNET50에 대한 HW별 추론 성능 비교
- YoloV3에 대한 HW별 추론 성능 비교
- 리소스 사용량 줄인 방법(htop)
- 웹캠 속도 향상 방법



# Trouble Shooting

- 1. 데이터의 노후화로 중장년층의 정확도 감소, 데이터 불균형 등,서양의 데이터 비율 차이

1차 시도 이슈  
-> 1. 나이 데이터 라벨링 자료에 다른 Class의 자료가 들어가 있었다.  
-> 2. 데이터에 20대가 몰려 있어 28살만 출력되는 이슈

2차 시도 이슈 (데이터셋과 코드 교체)  
-> 1. 데이터 개수 다양화: 900, 10,000, 2,000  
-> 2. 데이터셋 2차 교체 (중장년층 데이터 확보)  
-> 3. 여러 학습 모델 시도  
- 1 시도와의 학습 모델 비교  
- 2. 코드 교체 이후 학습시킨 7종의 모델 비교  
-> 그래프와 현실 성능 간극

3차 시도- 나이 후보정 시도: 아직 구현 X just idea  
(근데 발표까지 시간이 남아서 진짜로 보정할지도)

- 1. 문제-웹캠의 속도  
해결-매 프레임마다 인물 나이, 성별 추정하는 것보다 일정 단위를 설정해 그 단위마다 추정하도록 수정

1. Ollama 연동 시 자원 고갈 문제  
-> 웹캠 제어 및 리소스 관리의 필요성:  
-> Ollama 중복 호출, Ollama 답변 지연/비동기 문제

## 신세한탄

## CPU 9시간 돌리다 날아감

5. 일반 사진을 모델을 통해 나이를 추론하면 웹캠 사용 시보다 오차가 크게 나는 문제가 있었음 -> 나이는 TEST 데이터셋의 퀄리티(연령층 조절)에 따라 성능을 개선시킬 수 있었고, UFKFace 자료를 사용해 얼굴 인식 성능을 높일 수 있었음. 성별은 이진 분류라 큰 문제 x

6. 올라마랑 연동 전에 웹캠 테스트 시  
초창기 모델의 성능이 낮아 얼굴이 인지되고 적절한 나이를 추론하기 까지 시간이 걸렸다.  
-> 인식 후 3초 후에 나이를 출력하고 5초 후에 사라지는 후보정을 통해 정확도 및 속도 개선

- 1. Vitis-AI가 제공하는 플랫폼 양식에 맞춰야 함  
-> 안 맞추면 파일을 읽지 못하거나 변환과정에서 깨짐  
-> 가상환경의 중요성을 깨달음
- 2. 웹캠 속도 개선
- 3. FPGA 보드에서 웹캠 구동시 출력 모니터 화면 깨짐  
-> OS 이미지와 웹캠 출력이 겹침  
-> 웹캠 출력에는 xwindow를 종료
- 4. 학습할 때 GPU로 돌리는데 메모리 누수 나면 튕김  
-> batch size를 줄여서 해결
- 5. .xmodel 파일 변환시 문제  
-> segmentation fault 발생  
-> docker container로 제공되는 사전 변환된 xmodel 파일 다운
- 6. yolov3-tiny  
-> weights, cfg 파일 불일치  
-> yolov3로 방향 전환