

시각장애인을 위한 영상 기반 충돌 회피 보행 보조 시스템

국립한밭대학교 서민경

Table of Contents

01 문제상황

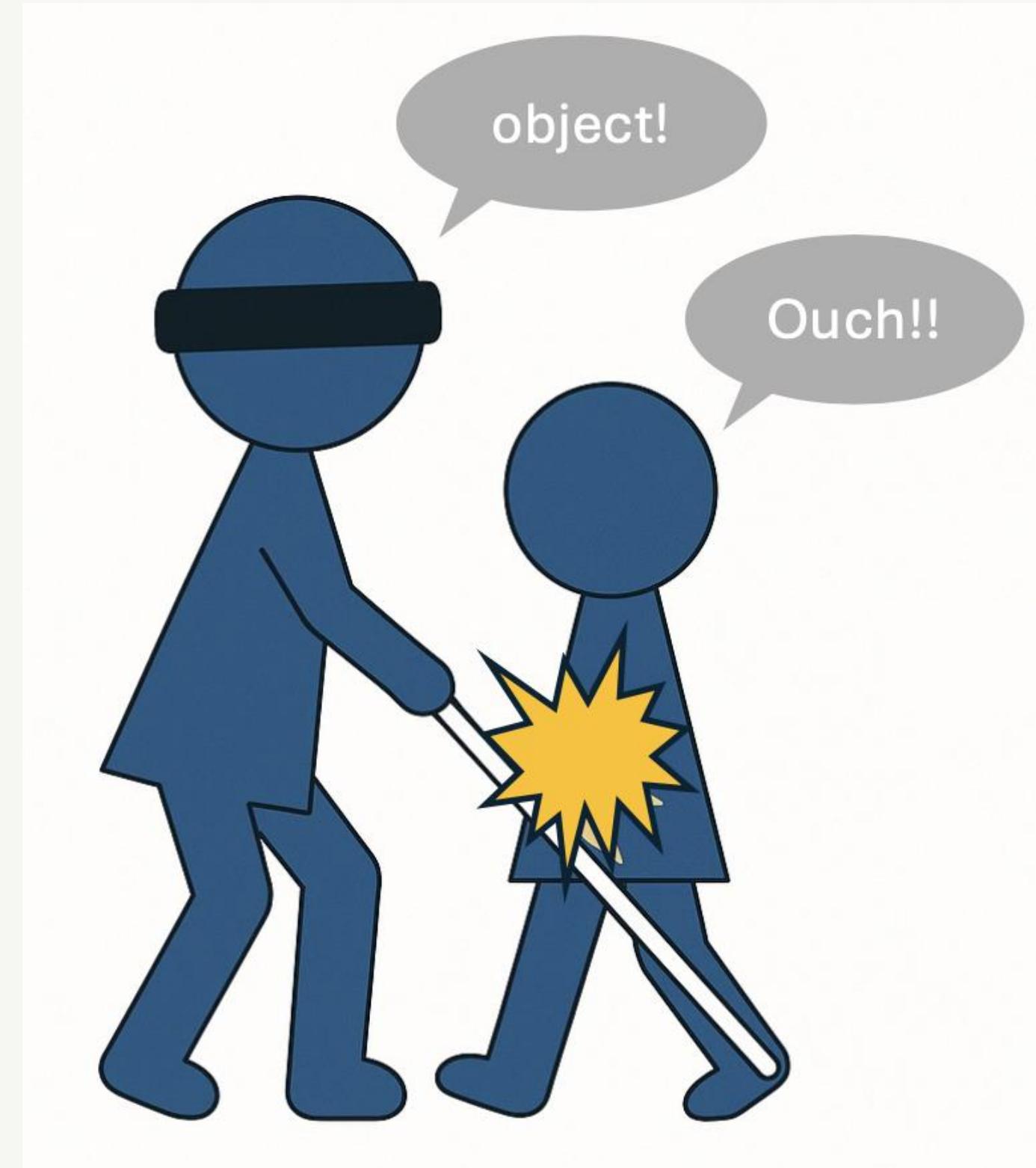
02 시스템 구조도

03 실험

04 결론

만약?
내가
시각장애인이라면

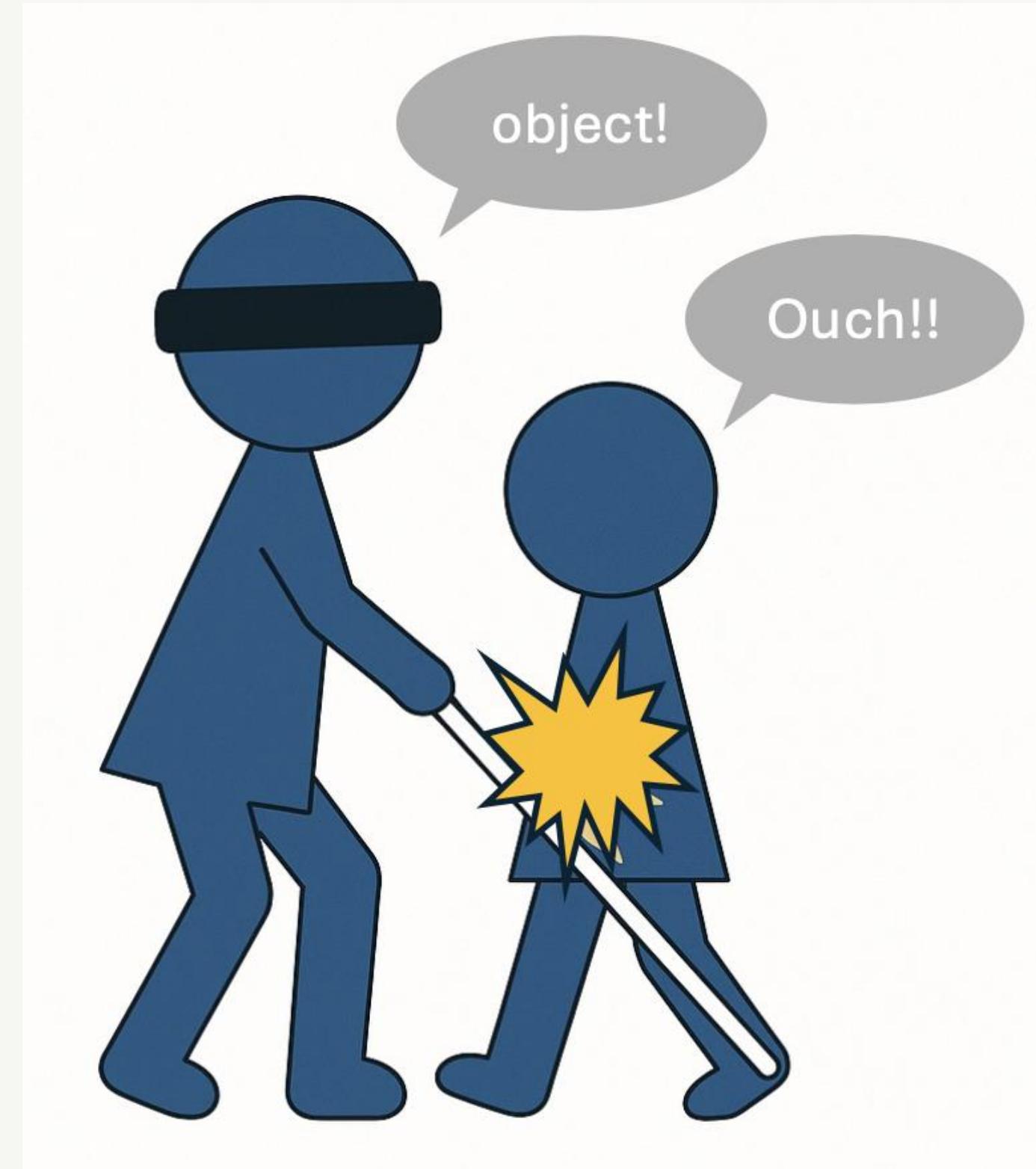
01 문제상황



문제 상황

- 물리적인 접촉 전에는 장애물 인지 X
- 보행환경에서 위험판단 X

01 문제상황

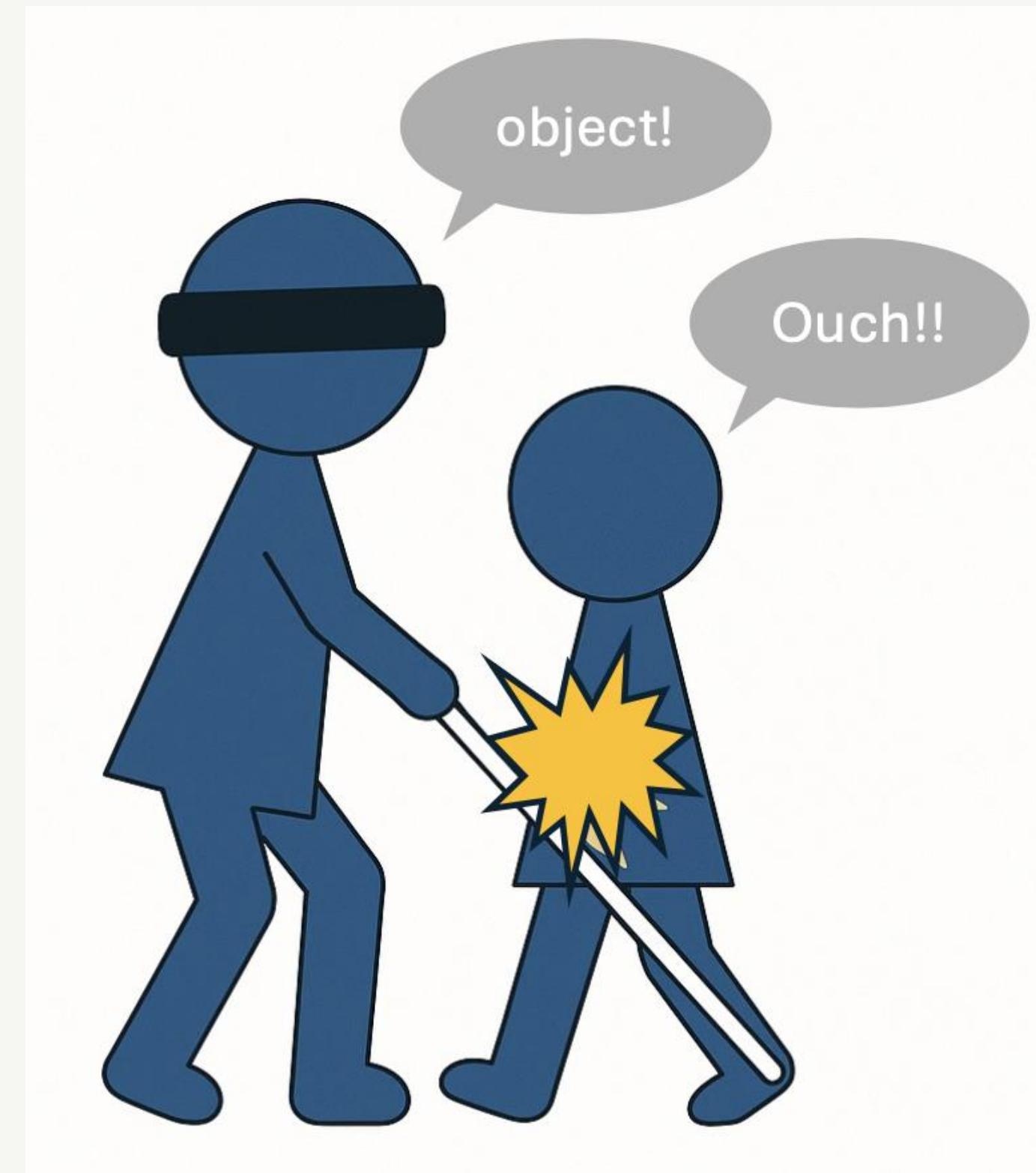


문제 상황

- 물리적인 접촉 전에는 장애물 인지 X
- 보행환경에서 위험판단 X



01 문제상황



문제 상황

- 물리적인 접촉 전에는 장애물 인지 X
- 보행환경에서 위험판단 X



해결 방법

- 물리적인 접촉 전에는 장애물 인지 O
- + 장애물의 위치와 종류 파악
- 보행환경에서 위험판단 가능!

01 문제상황



문제 상황

- 문제 상황에 대한 장애물 인지 X

- 위험판단 X

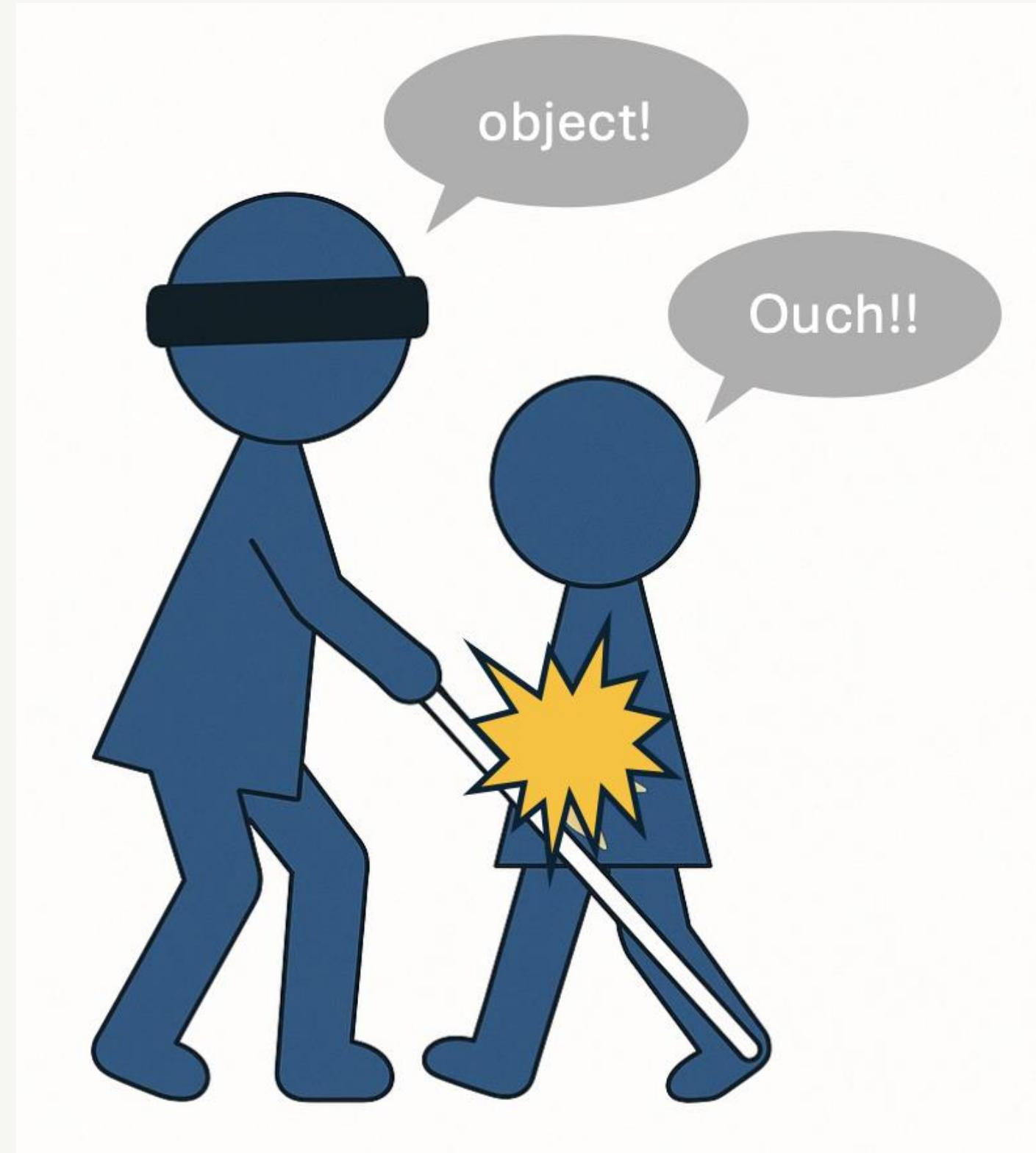
해결 방법

- 문제 상황에 대한 장애물 인지 O

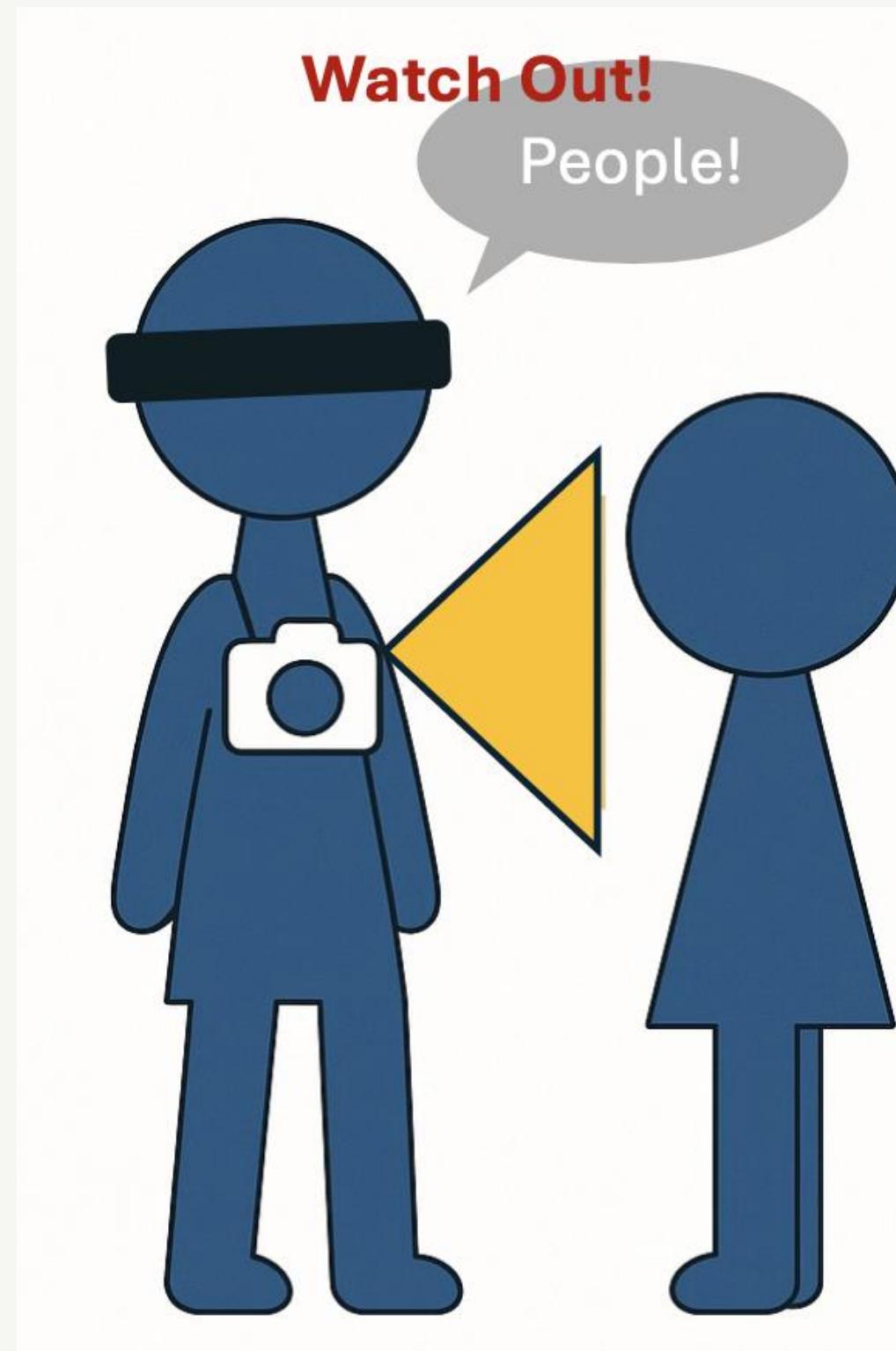
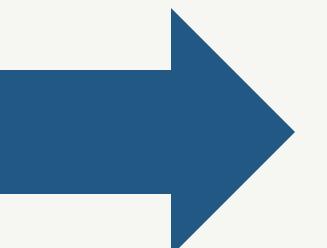
- 위험판단과 송류 파악

- 보행환경에서 위험판단 가능!

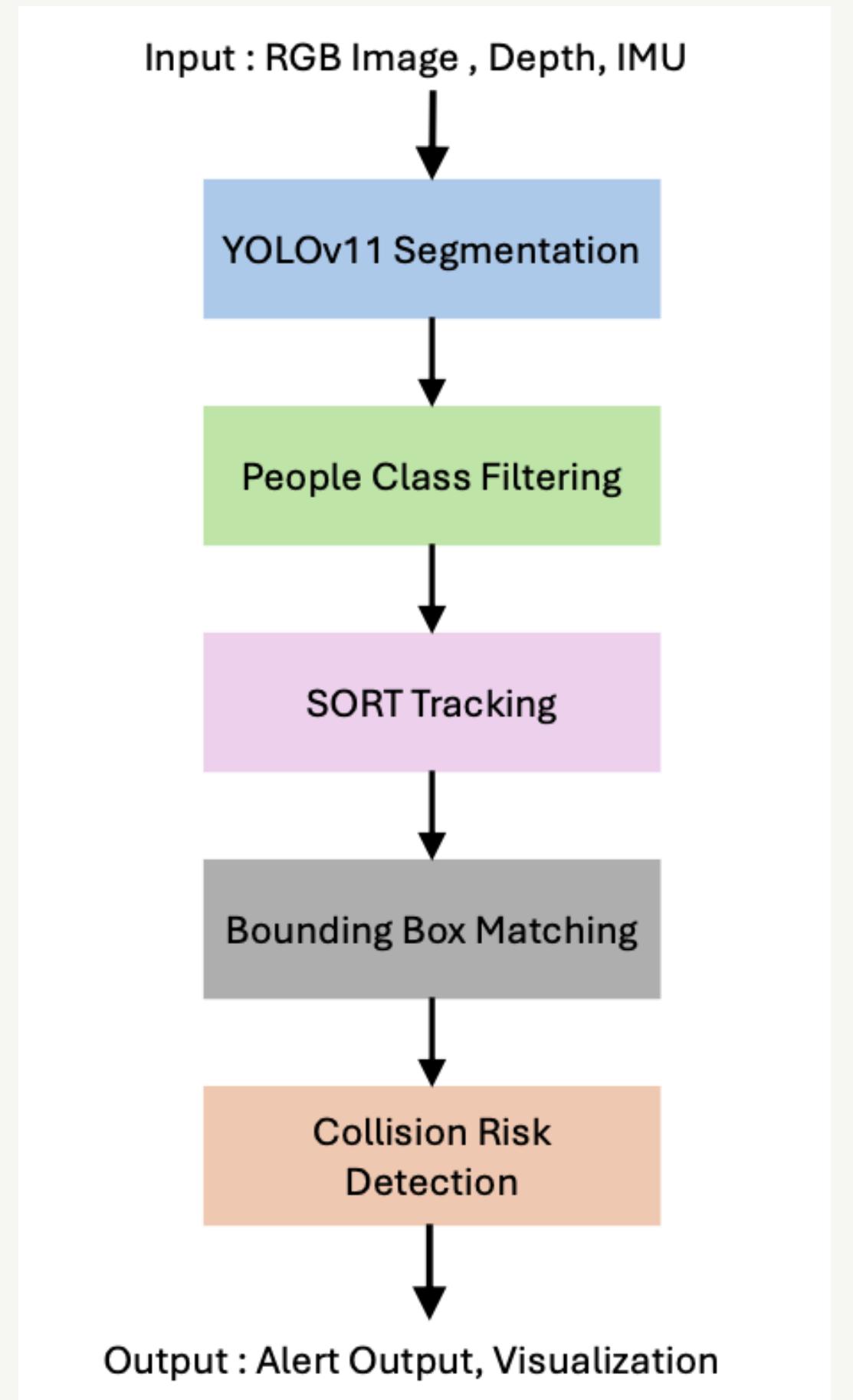
01 문제상황



01 문제상황



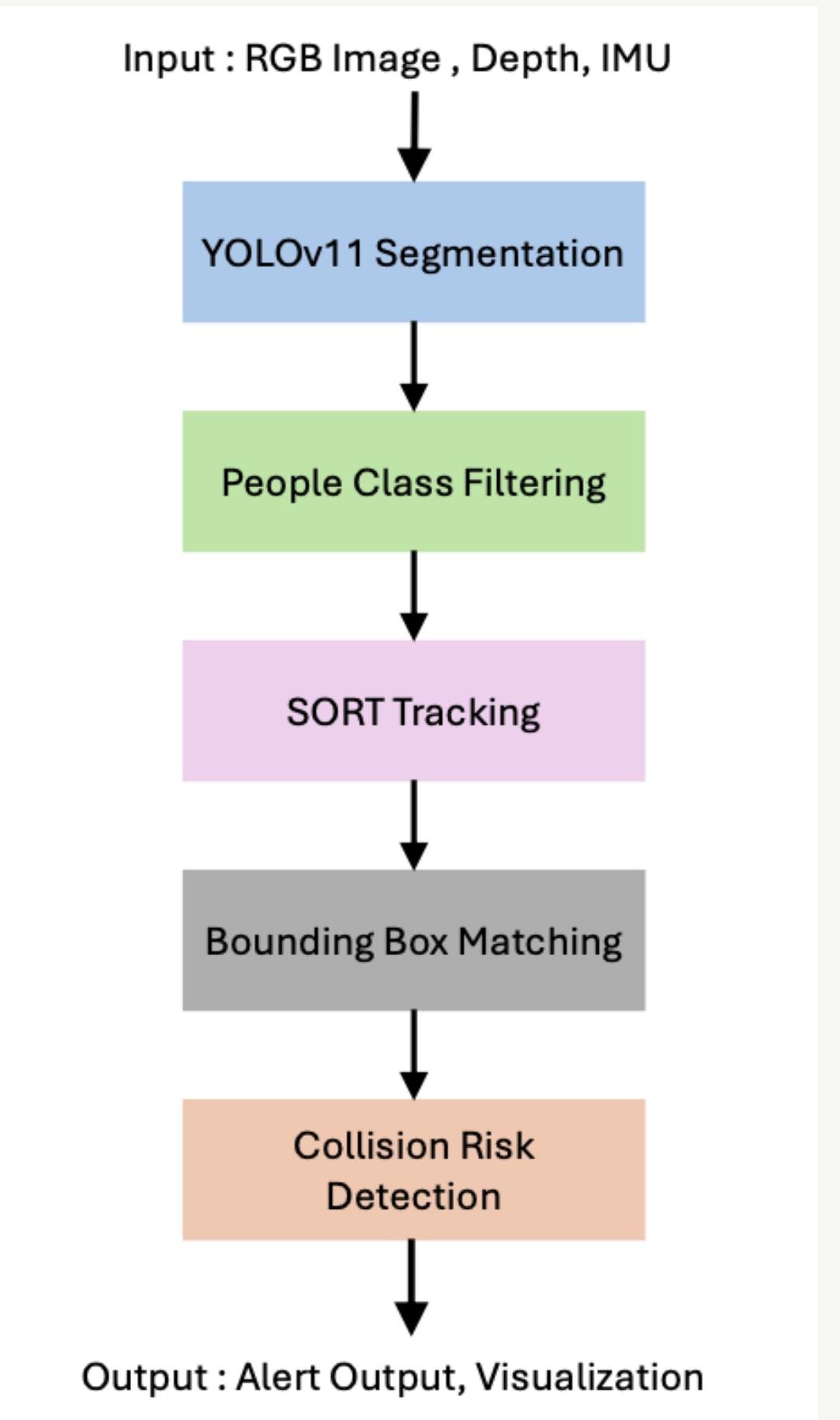
02 시스템 구조도



시스템 구조도 순서

1. 물체 종류 식별
2. 동적 물체 판단
3. 객체 추적
4. 결과 연결
5. 충돌 위험 감지

02 시스템 구조도

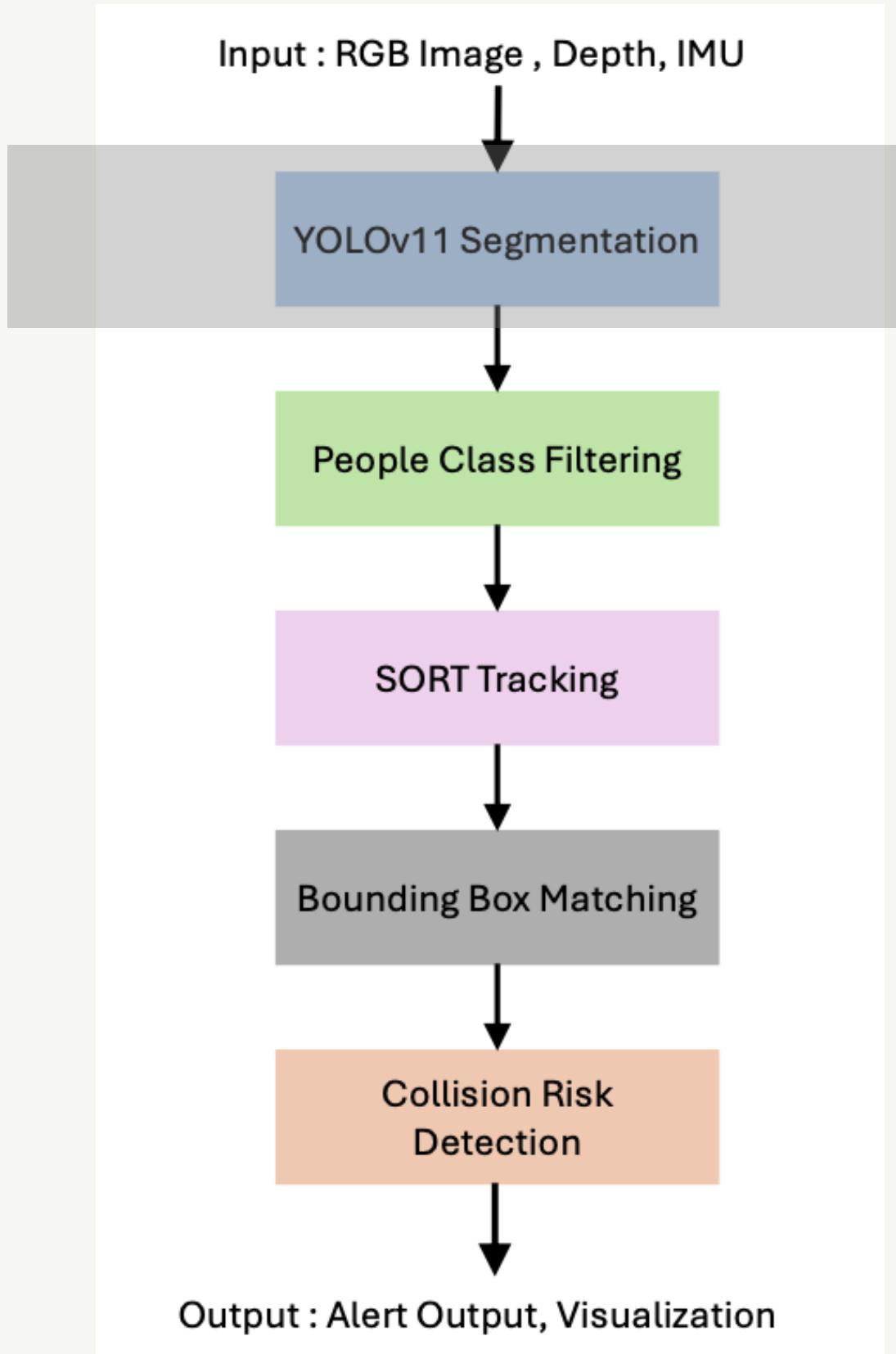


시스템 구조도 순서

1. 물체 종류 식별
2. 동적 물체 판단
3. 객체 추적
4. 결과 연결
5. 충돌 위험 감지

충돌 위험 !! 1.5m 이내 사람이 있습니다.

02 시스템 구조도

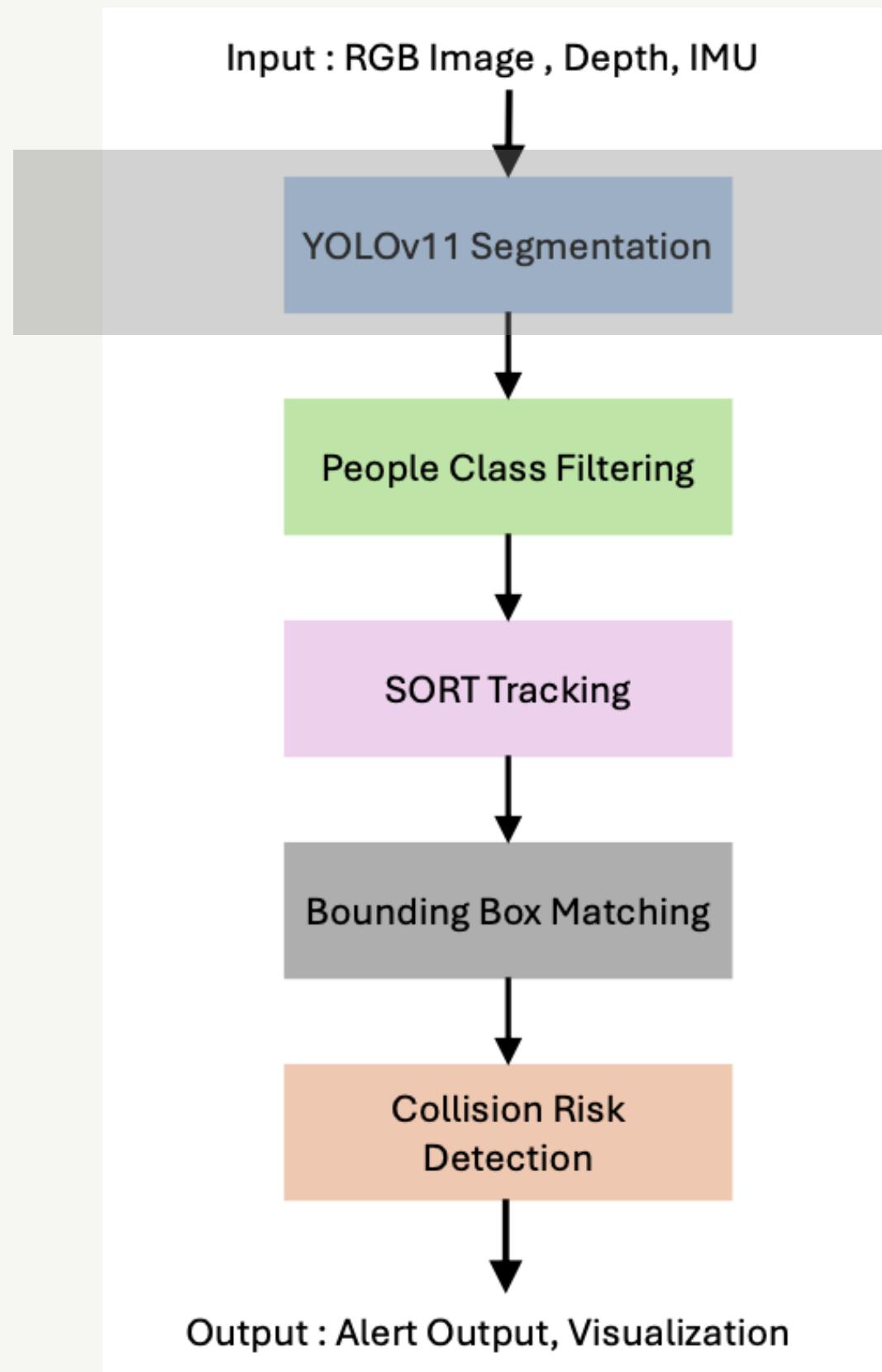


1. 물체 종류 식별

YOLO v11

: Instance Segmentation으로 픽셀 단위의 객체 추출 가능

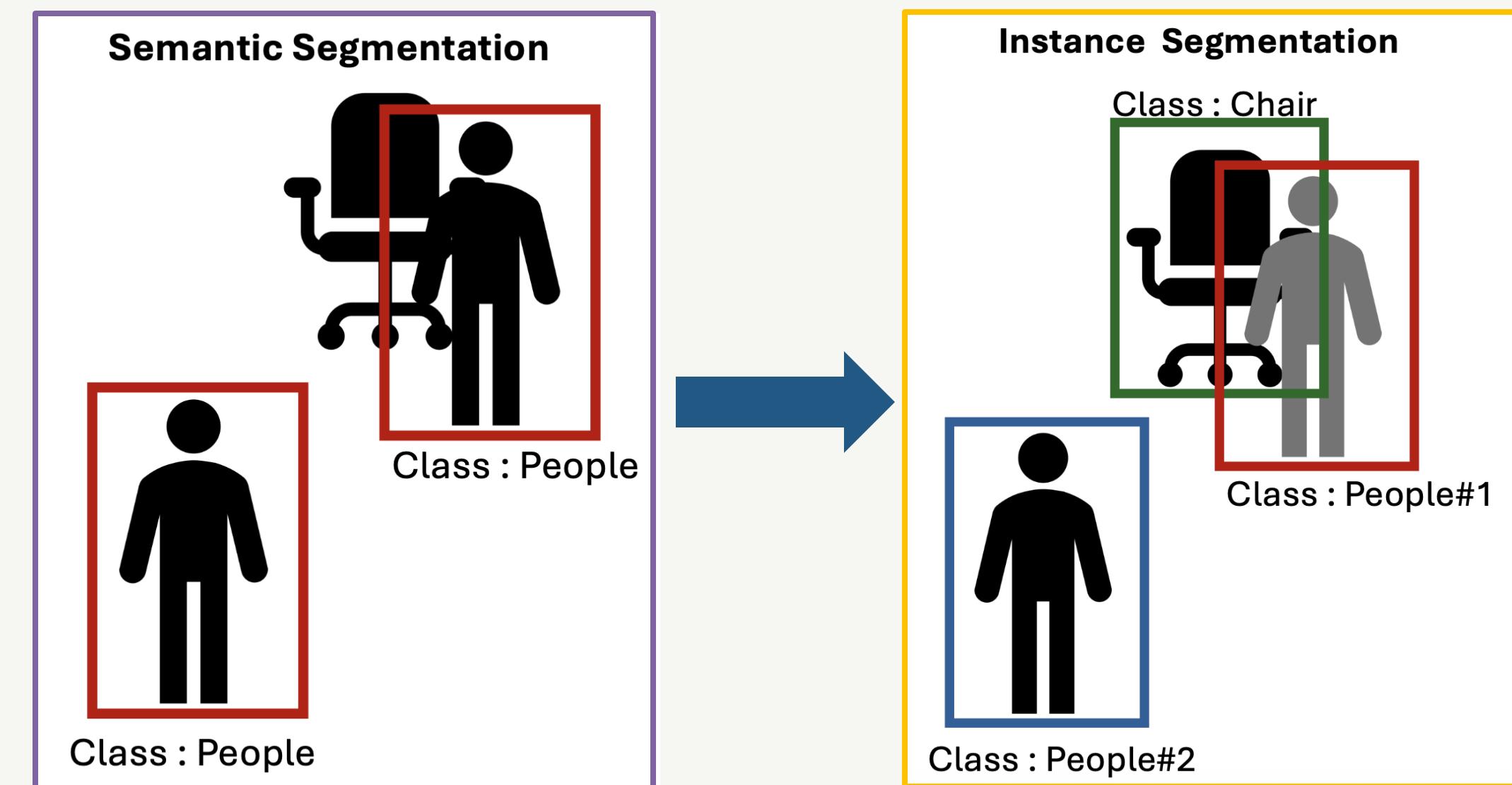
02 시스템 구조도



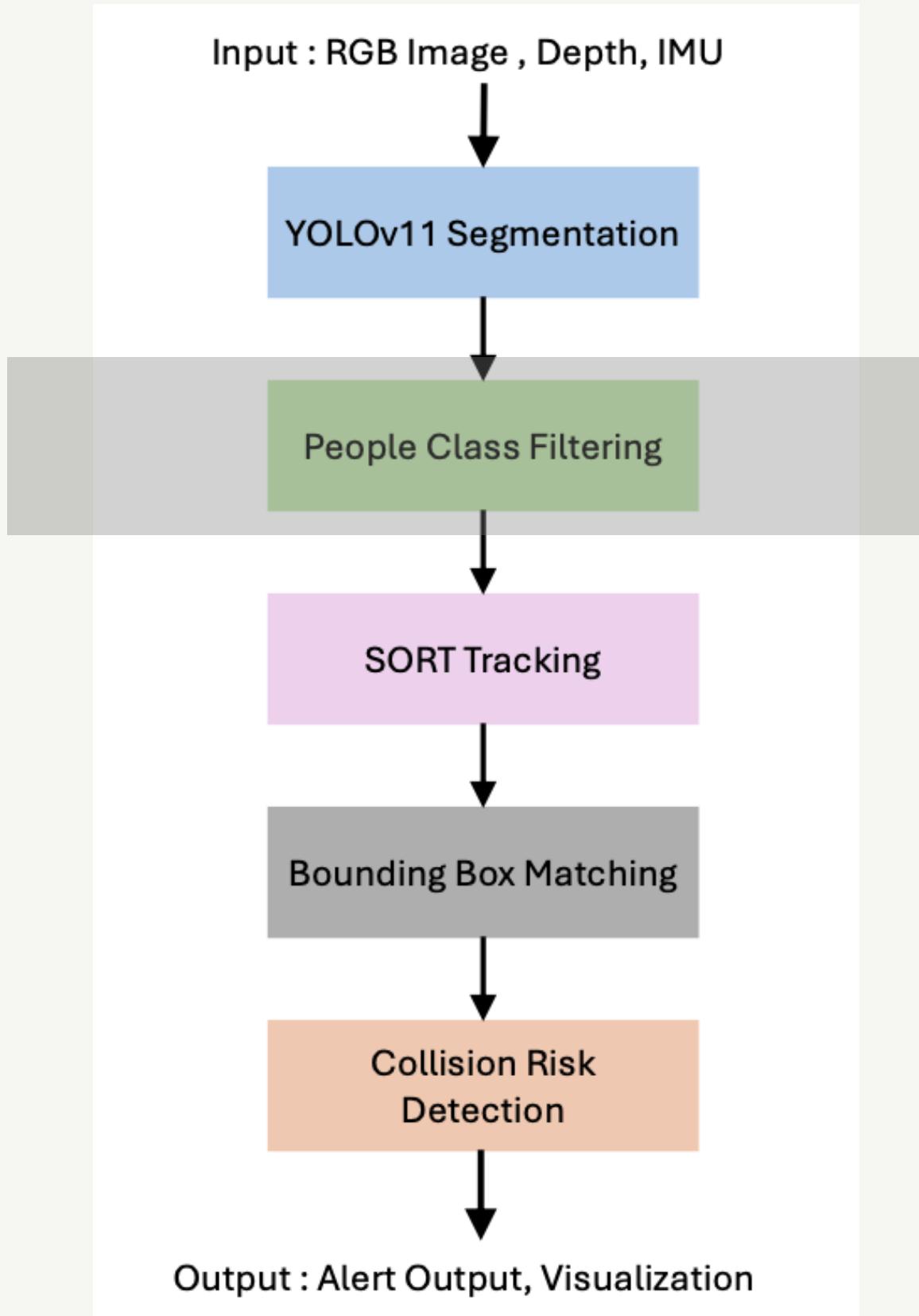
1. 물체 종류 식별

YOLO v11

: Instance Segmentation으로 픽셀 단위의 객체 추출 가능



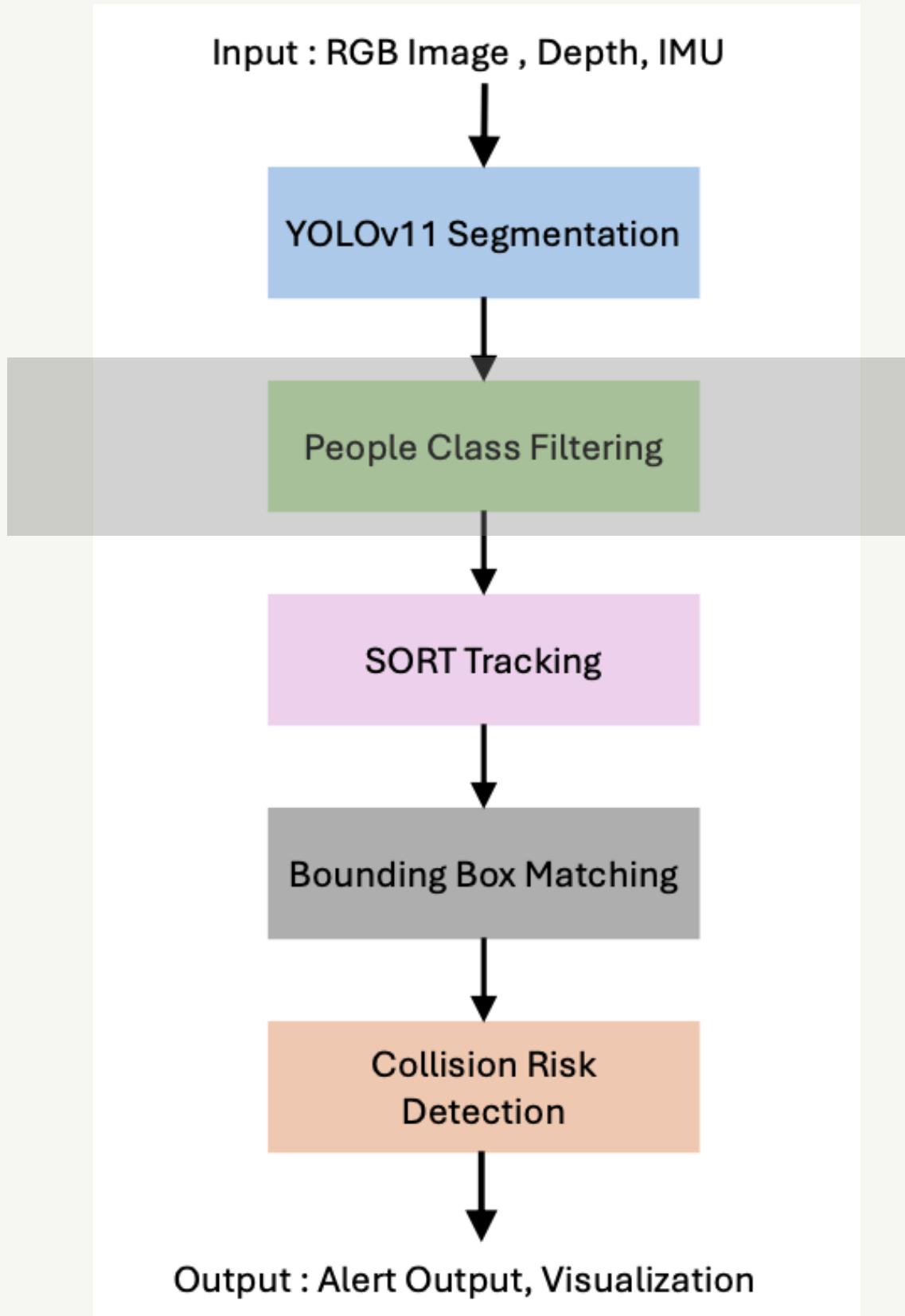
02 시스템 구조도



2. 동적 물체 판단

실내 환경에 초점을 맞춰 동적 물체 필터링
= 충돌 위험 대상

02 시스템 구조도



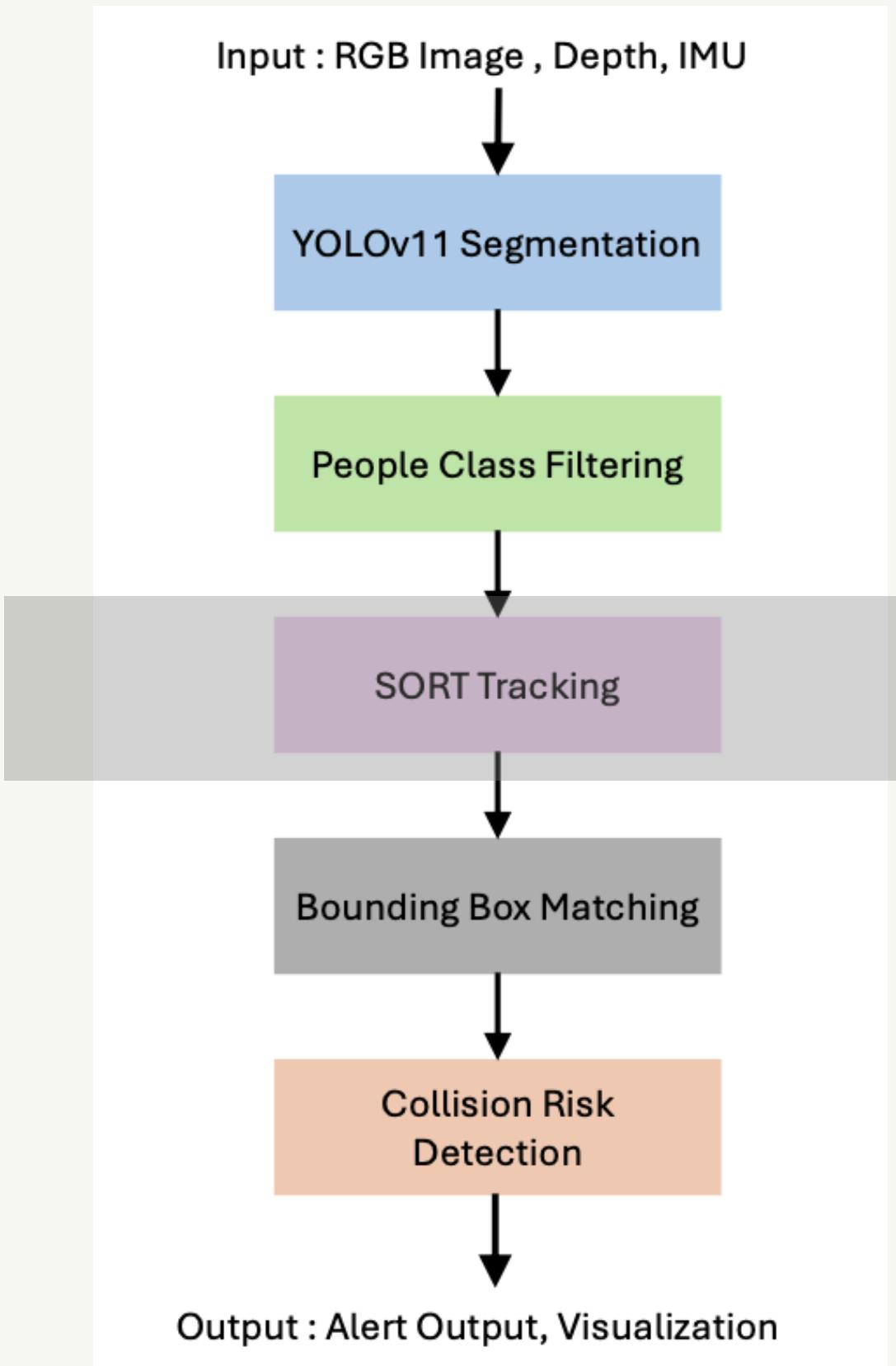
2. 동적 물체 판단

실내 환경에 초점을 맞춰 동적 물체 필터링
= 충돌 위험 대상

동적만 물체 필터링?

--→ 시스템 사용자와 독립적인 움직임으로 **돌발 상황 발생** ↑

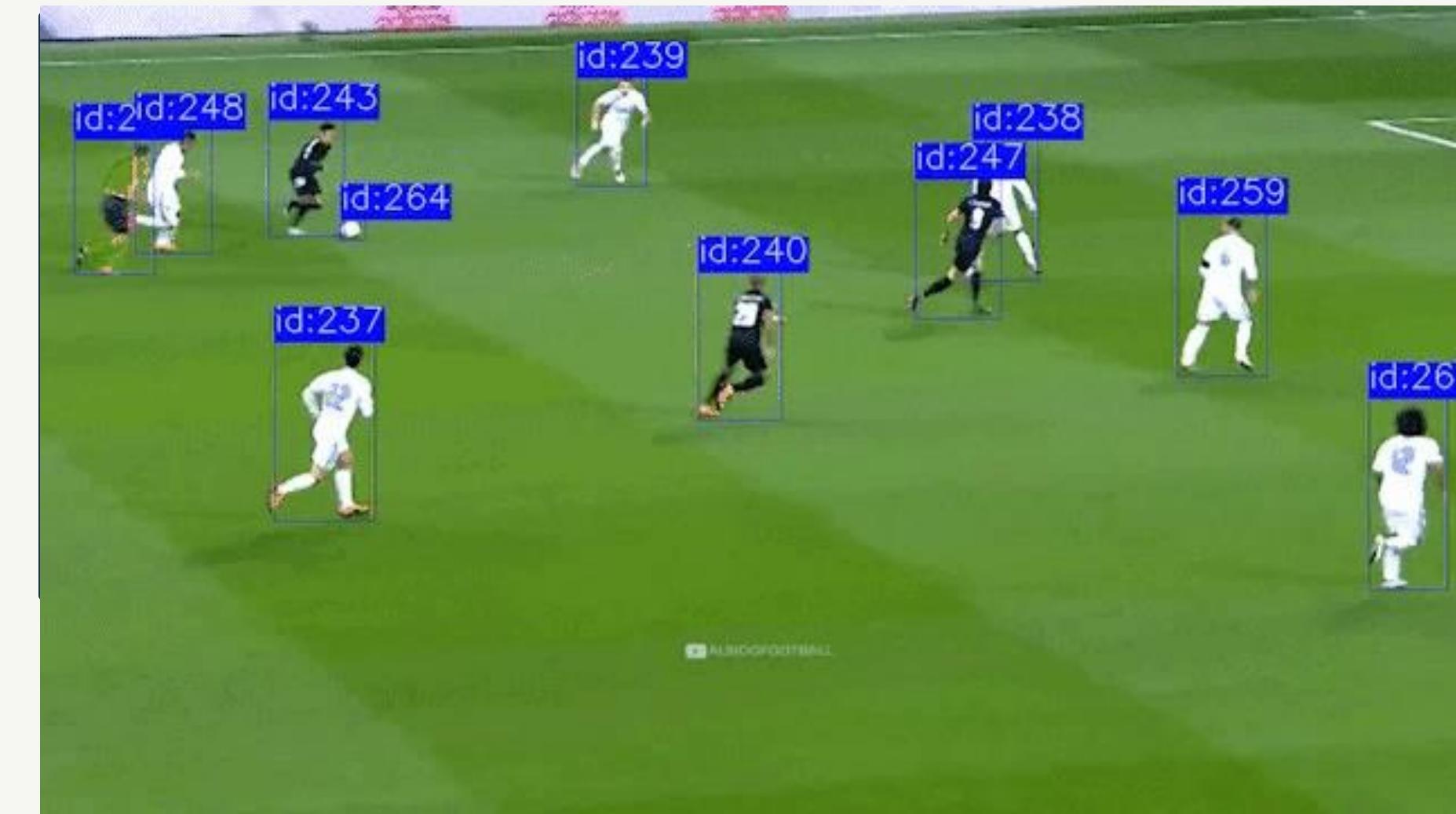
02 시스템 구조도



3. re-id (Re-identification ID) - 객체 추적

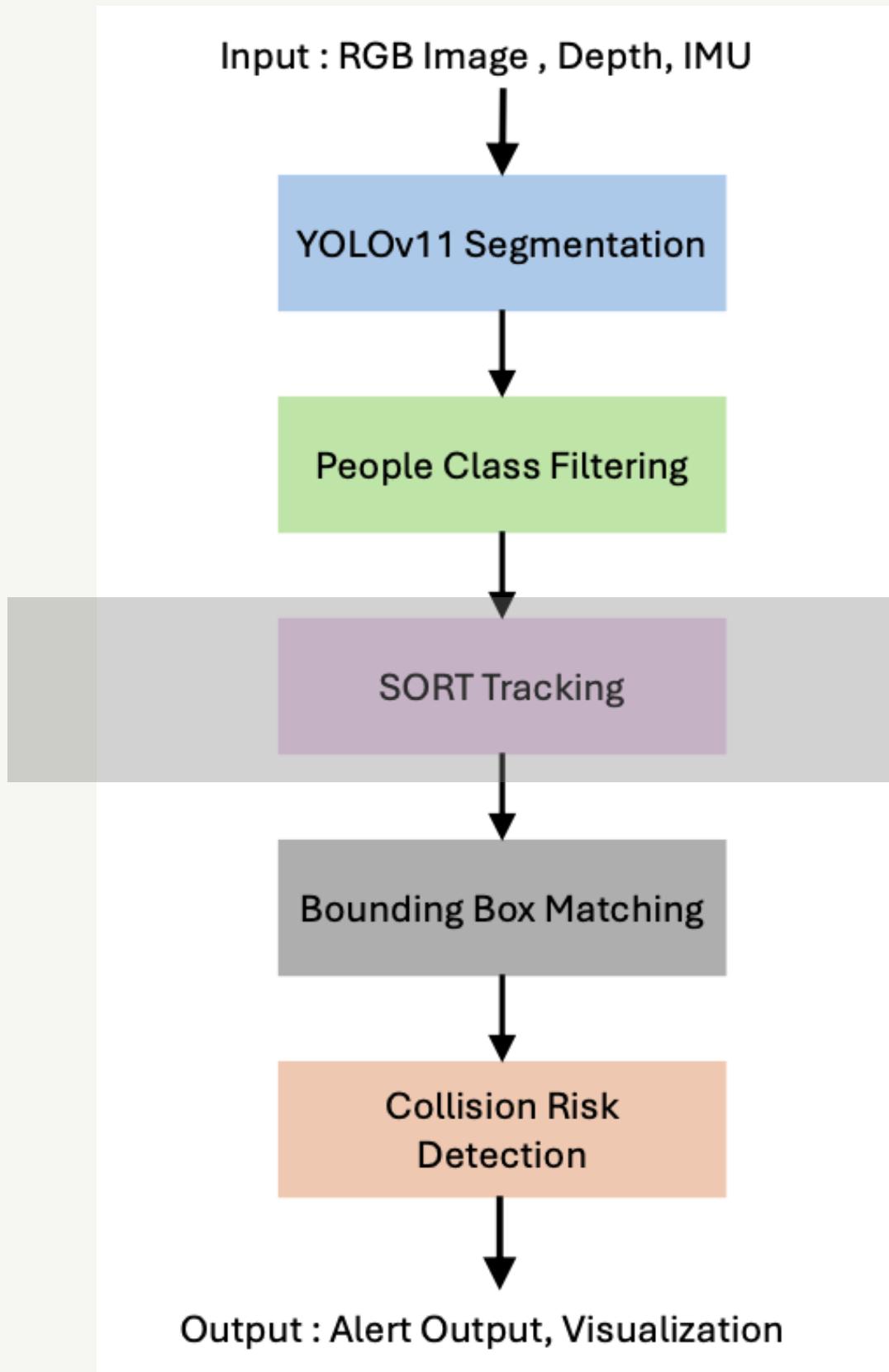
SORT Tracker

: 서로 다른 프레임에서 동일한 객체를 식별할 수 있음



출처 : <https://learnopencv.com/understanding-multiple-object-tracking-using-deepsort/>

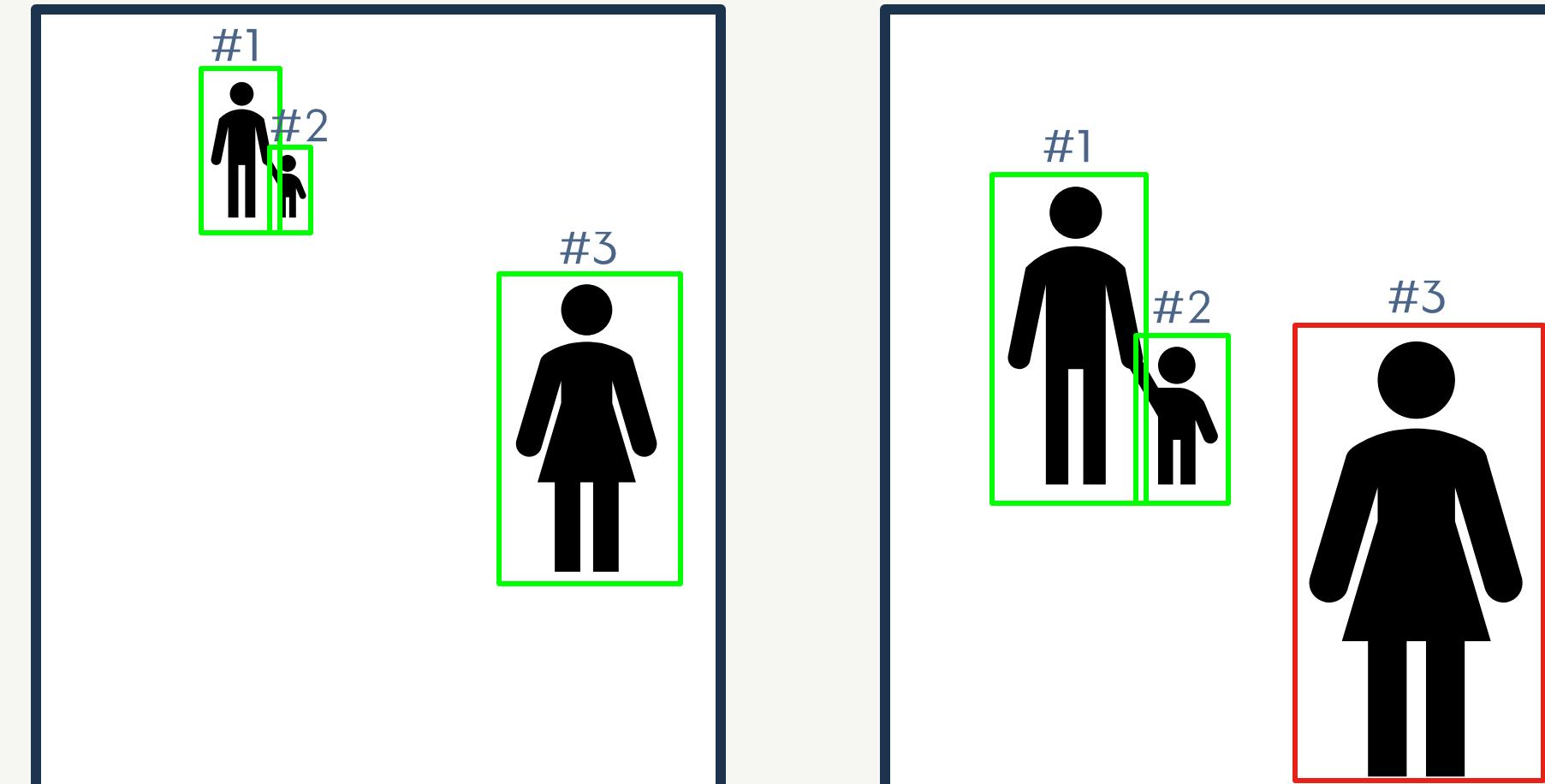
02 시스템 구조도



3. re-id (Re-identification ID) - 객체 추적

SORT Tracker

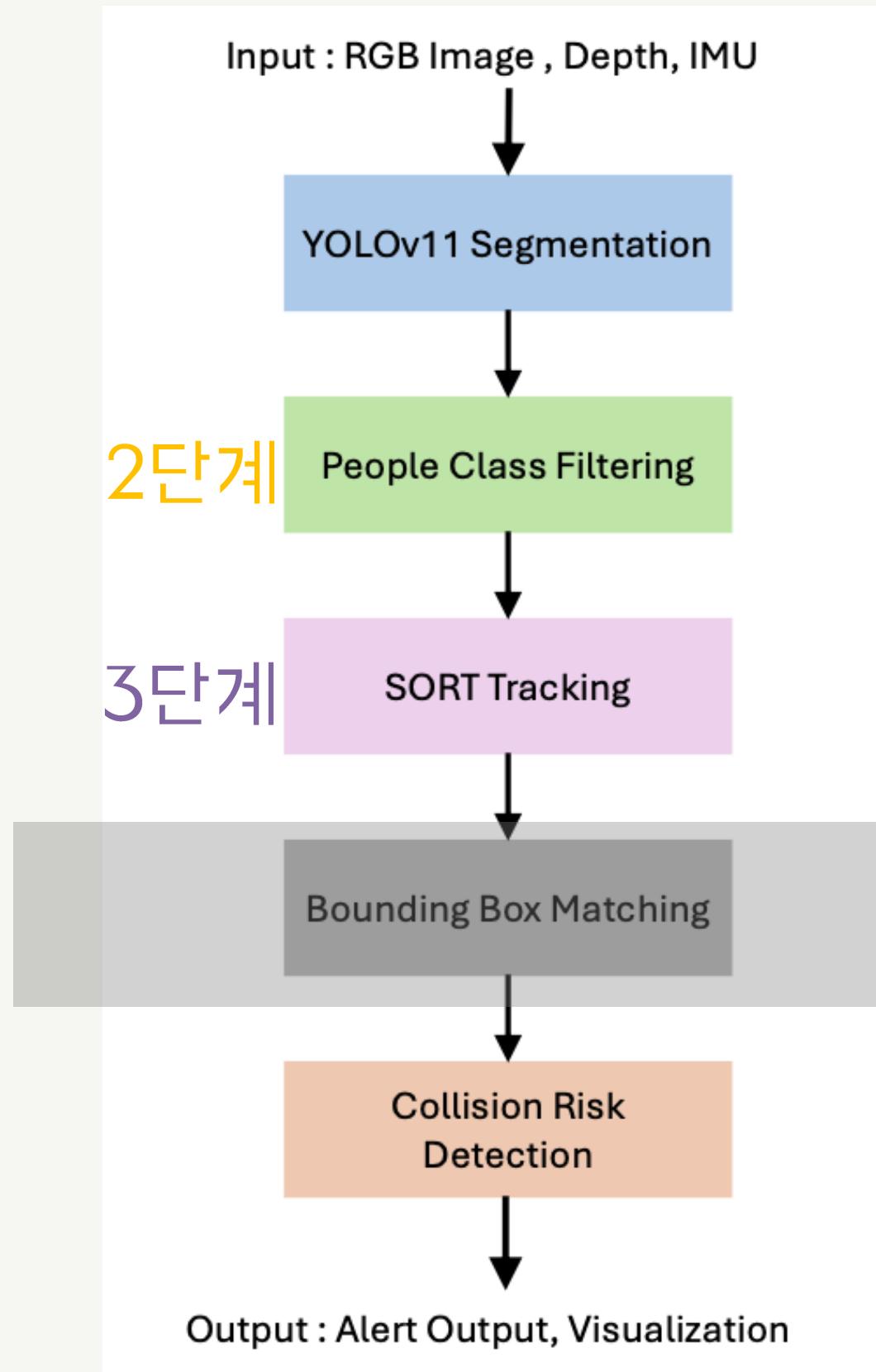
: 서로 다른 프레임에서 동일한 객체를 식별할 수 있음



Frame 4

Frame 9

02 시스템 구조도

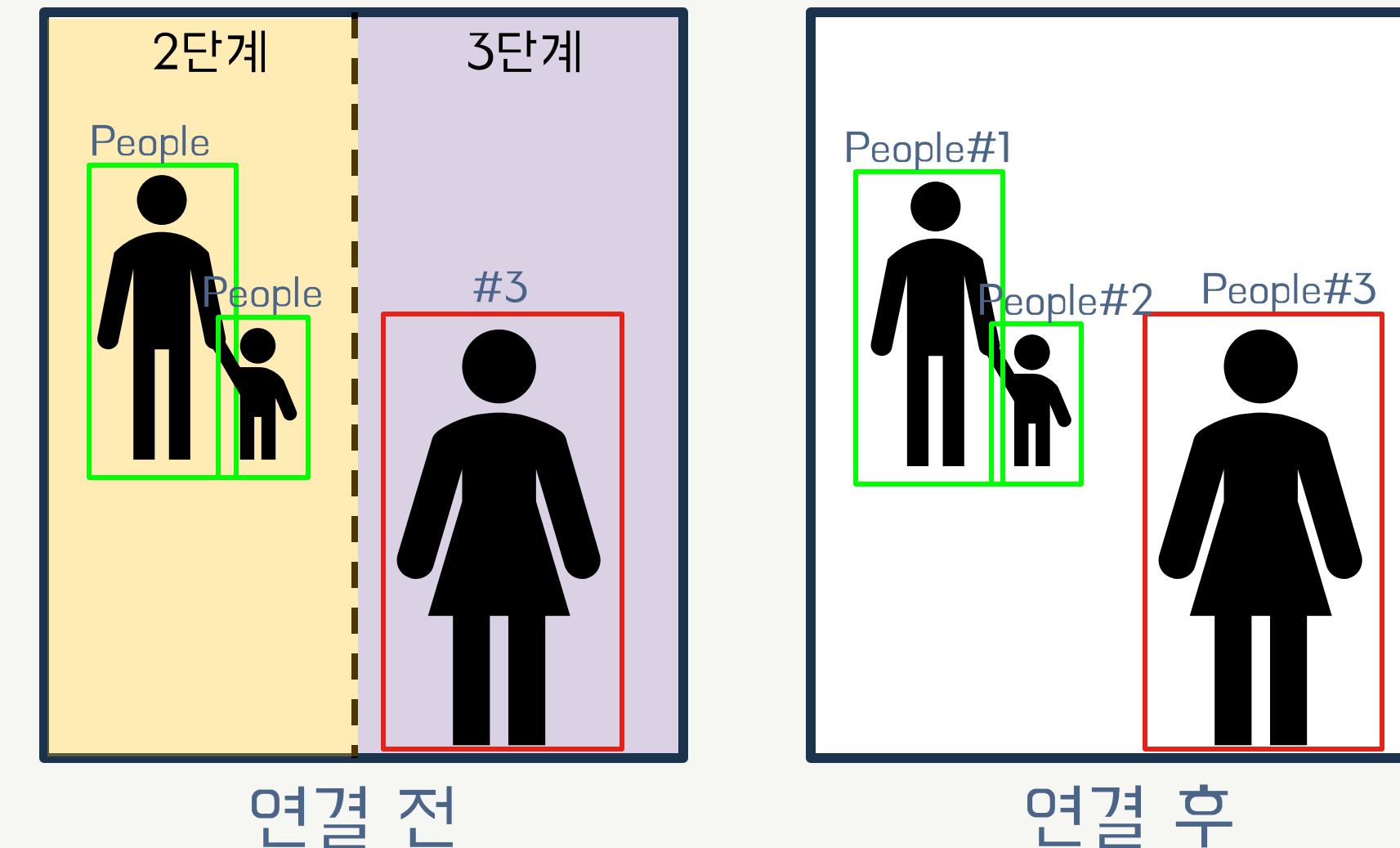


4. re-id (Re-identification ID) - 결과 연결

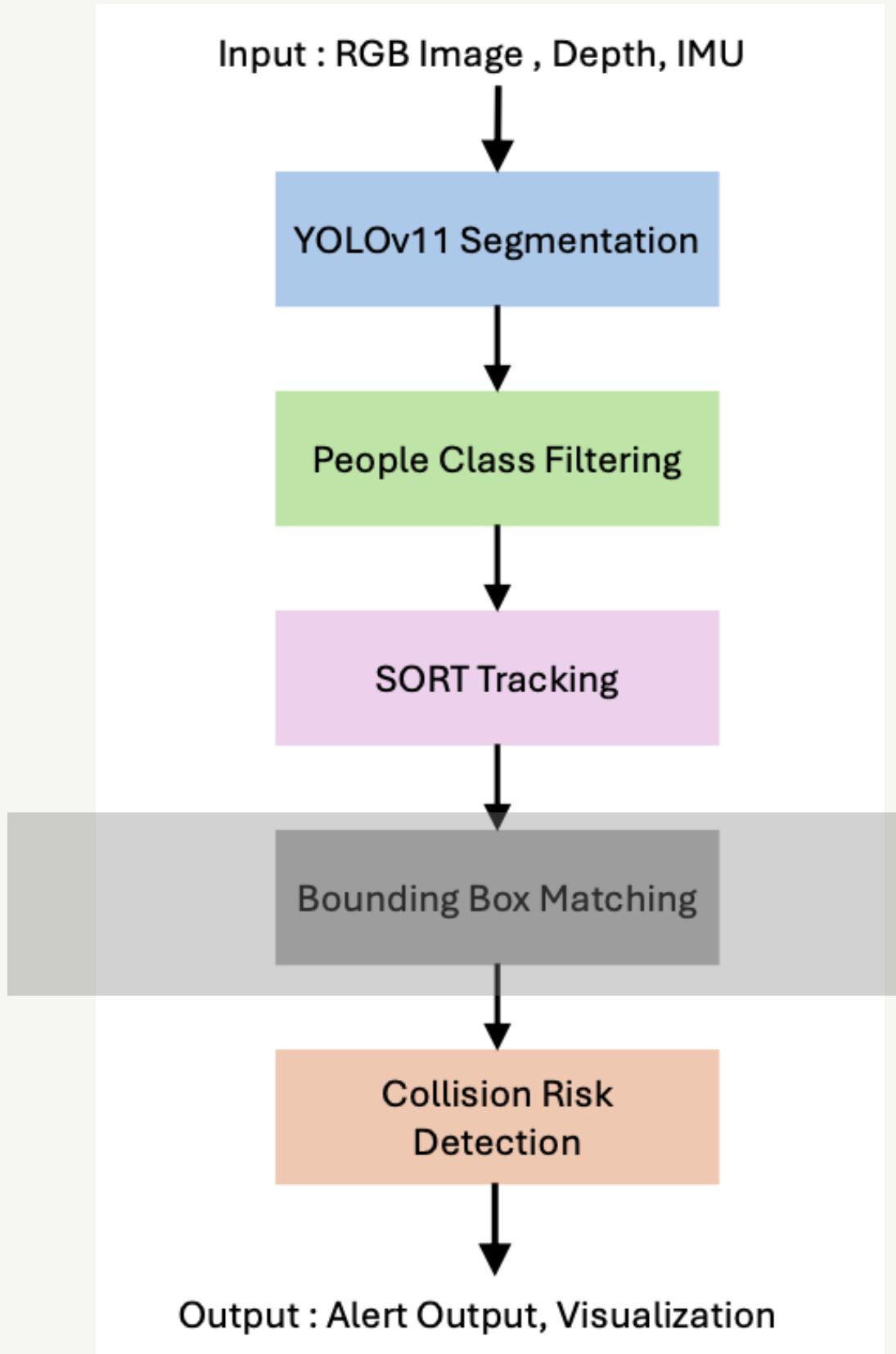
2단계 결과 : 각 Bounding Box의 Class

3단계 결과 : 각 Bounding Box의 고유한 ID

앞 선 두 단계의 결과를 연결

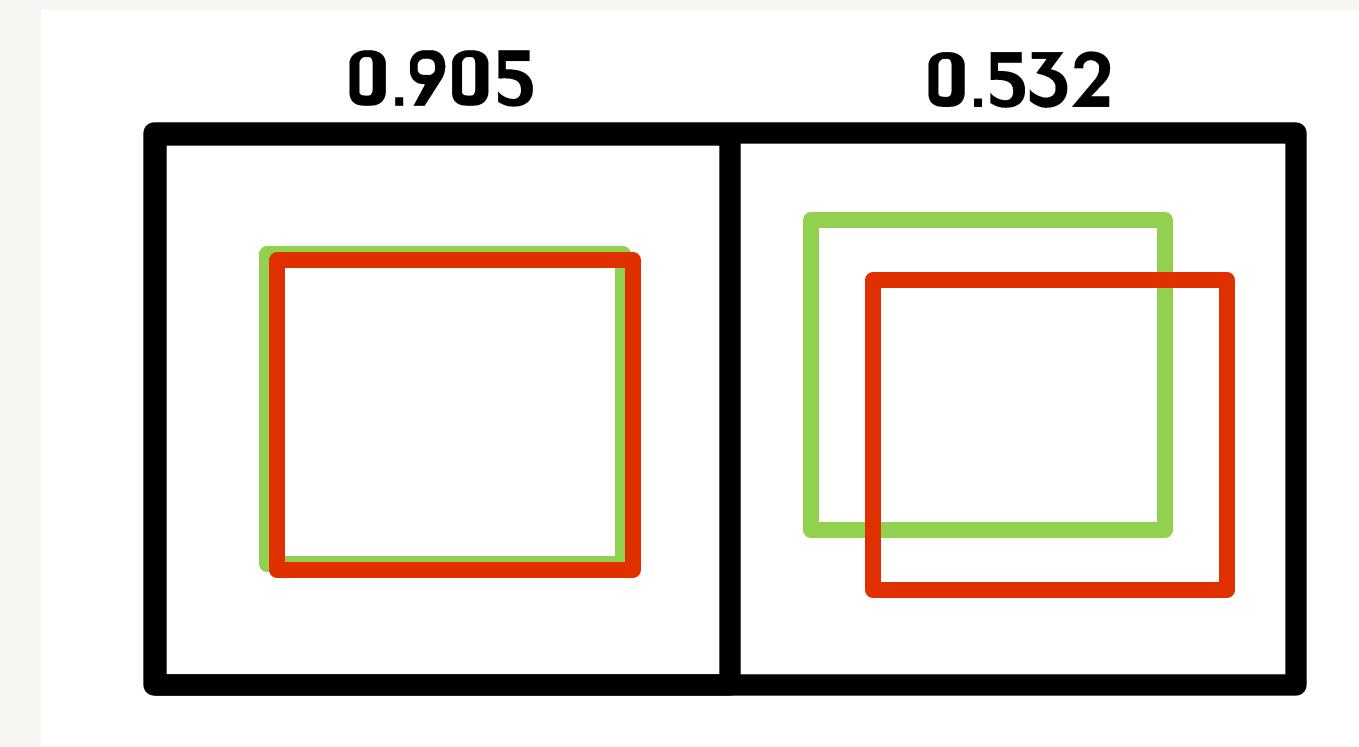


02 시스템 구조도



4. re-id (Re-identification ID) - 결과 연결

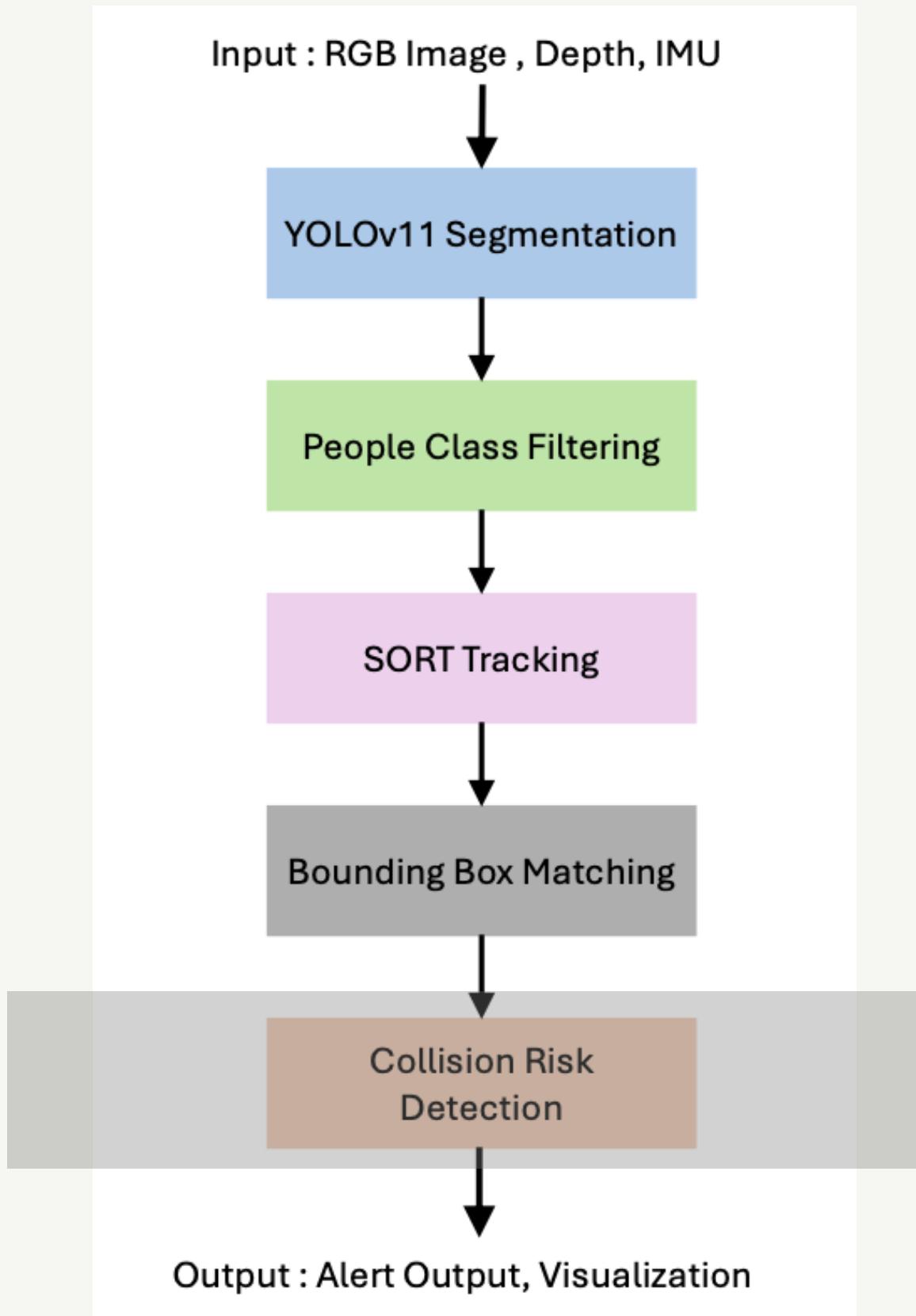
IOU 값을 사용하여 Depth, Bounding Box 면적, 객체 ID, Class 정보를 단일 정보로 구성



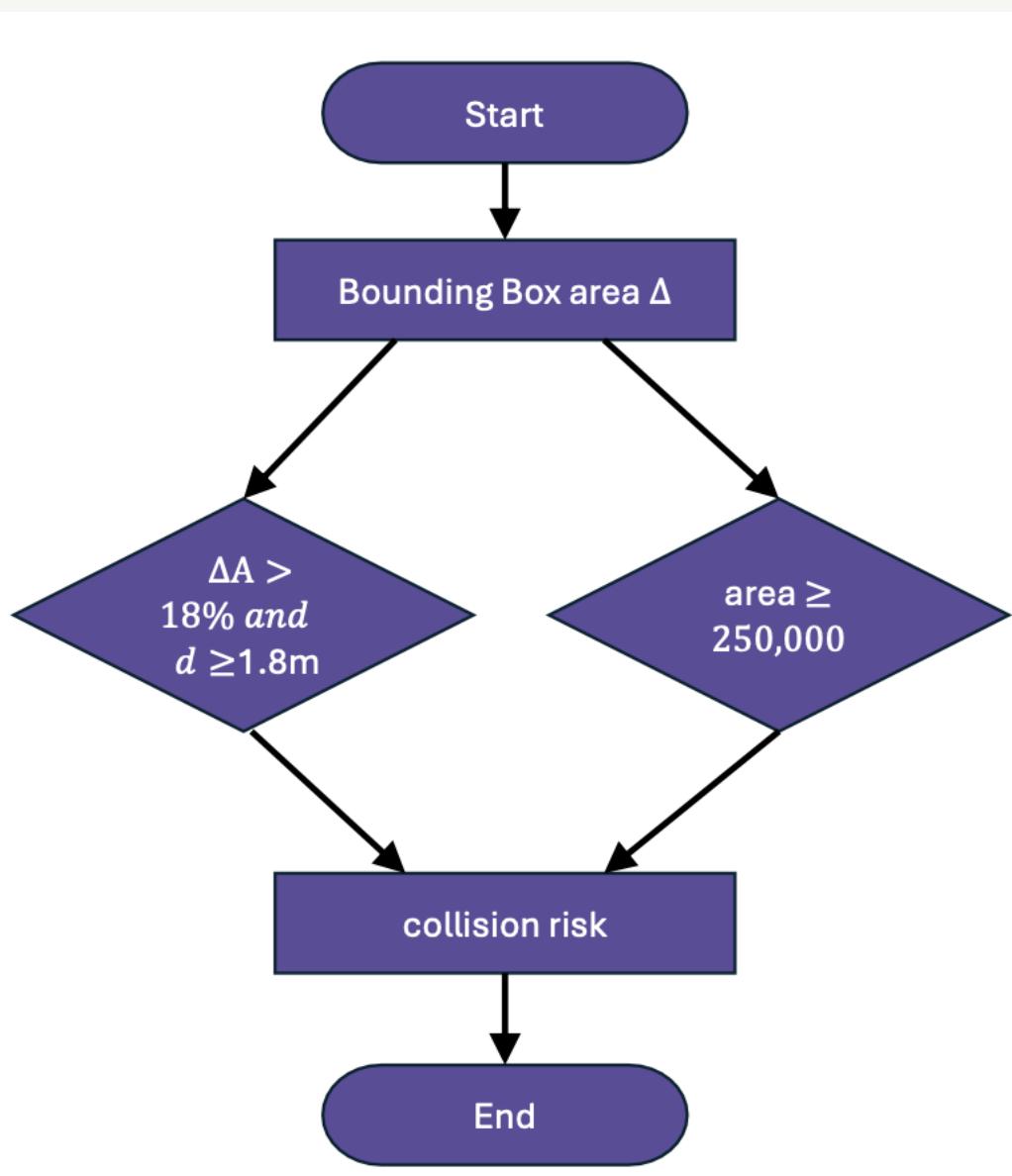
$$IOU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

IOU(Intersection over Union) : 예측 상자와 실제 정답 상자 간의 겹치는 영역을 측정하는 지표

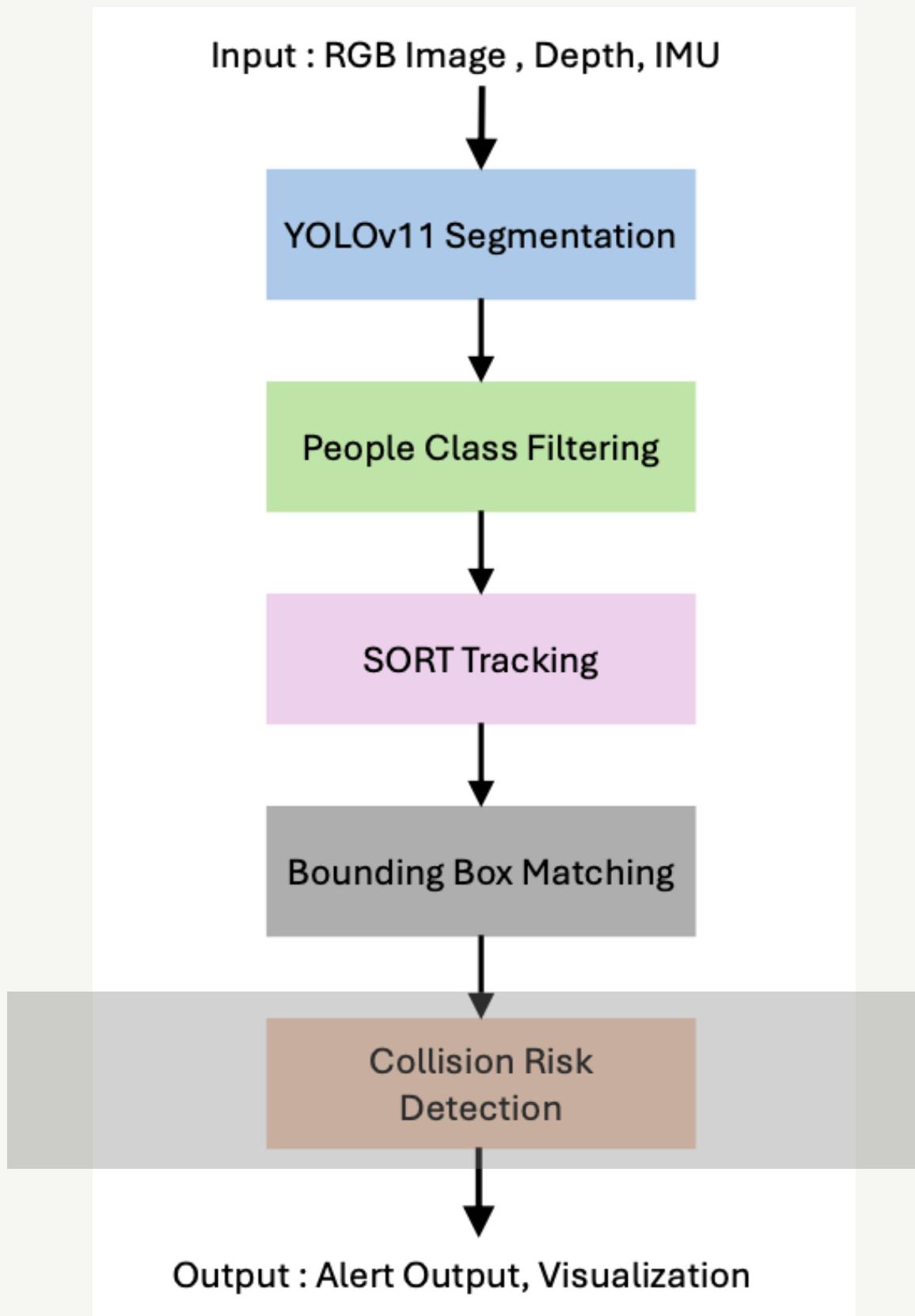
02 시스템 구조도



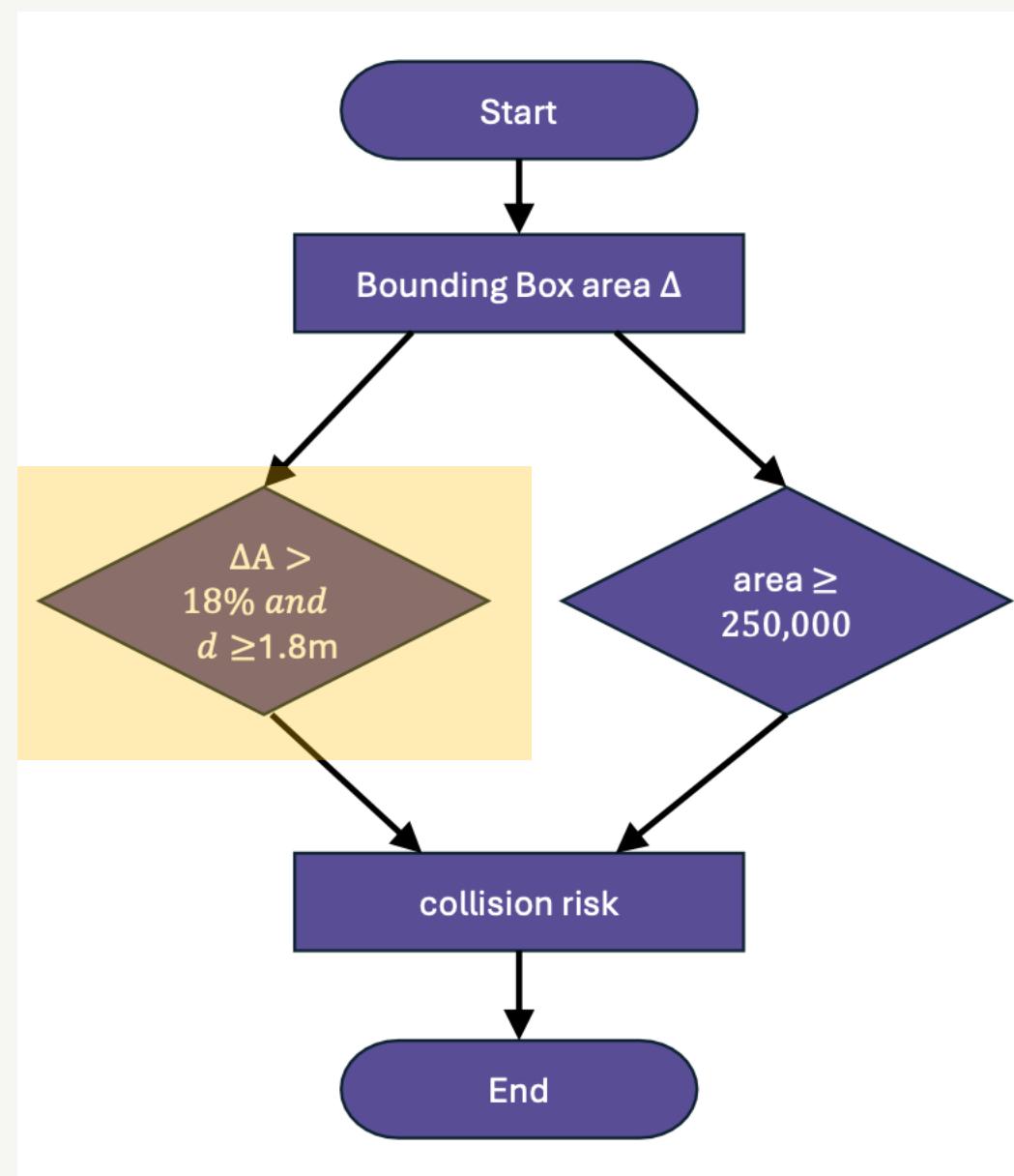
5. 충돌 위험 감지



02 시스템 구조도

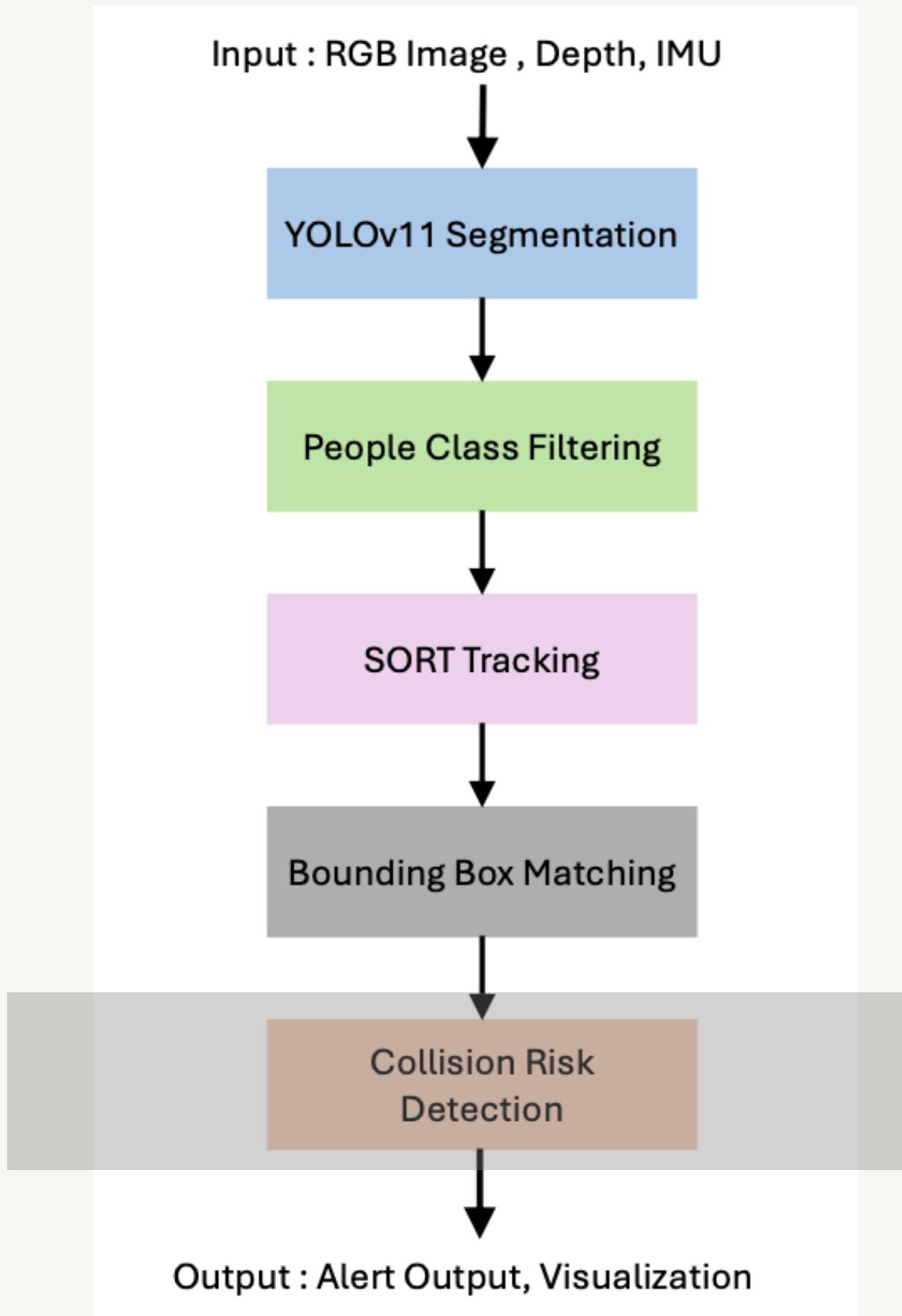


5. 충돌 위험 감지

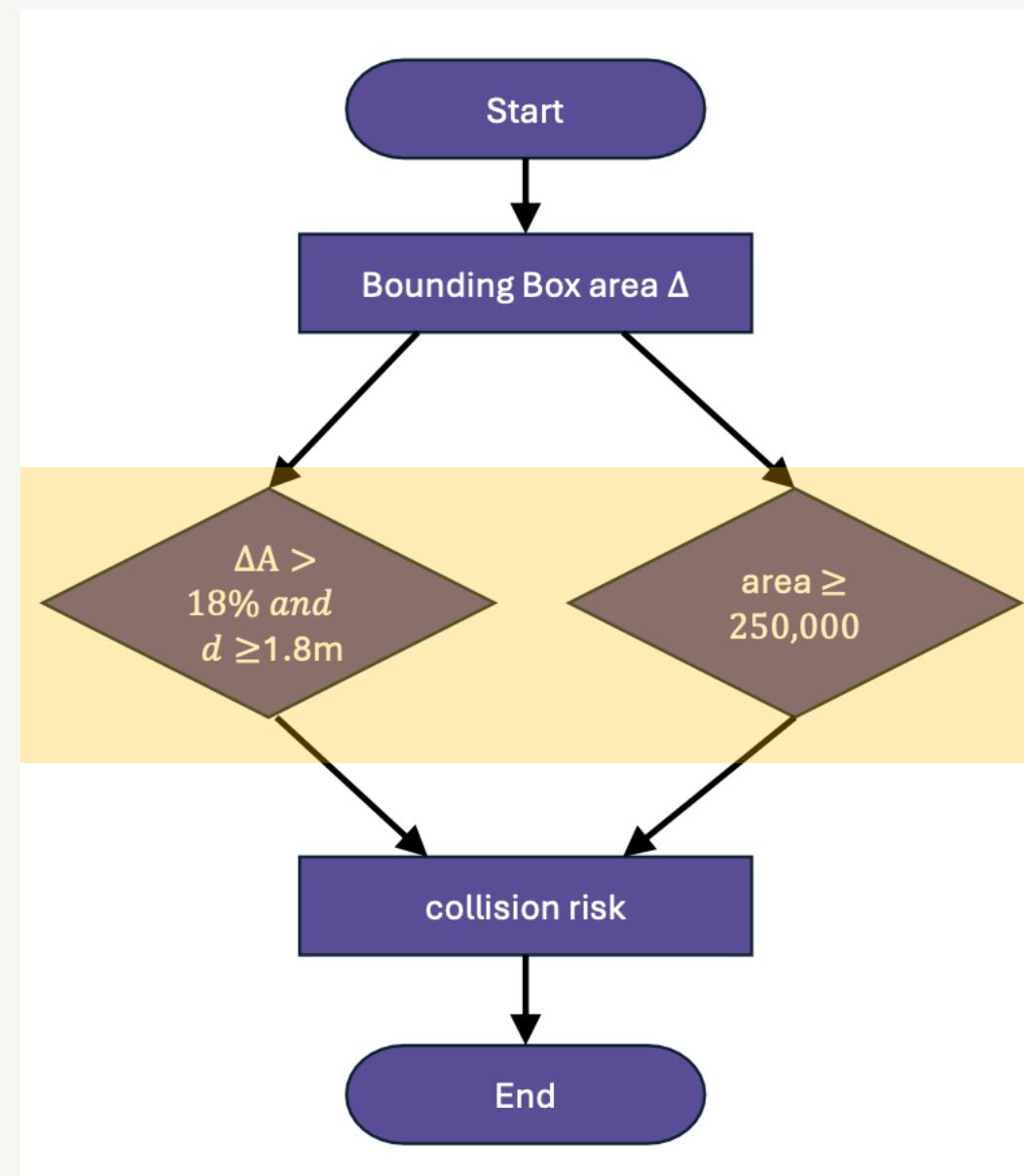


조건 1. 사용자와 일정 거리 이내에
있고 가까워지고 있을 경우

02 시스템 구조도



5. 충돌 위험 감지



조건 1. 사용자와 일정 거리 이내에
있고 가까워지고 있을 경우

조건 2. 사용자와 매우 근접할
경우

충돌 위험 !! 1.5m 이내 사람이 있습니다.

03 실험

Bounding Box Matching 방법 (Abs vs IOU)

실험 장비 : Intel RealSense D435i

실험 장소 : 국립한밭대학교 세종공동캠퍼스 건물

실험 데이터 : 350 프레임 영상

만약, Bounding Box Matching이 안된다면?

→ 충돌 위험 대상에서 **제외됨**

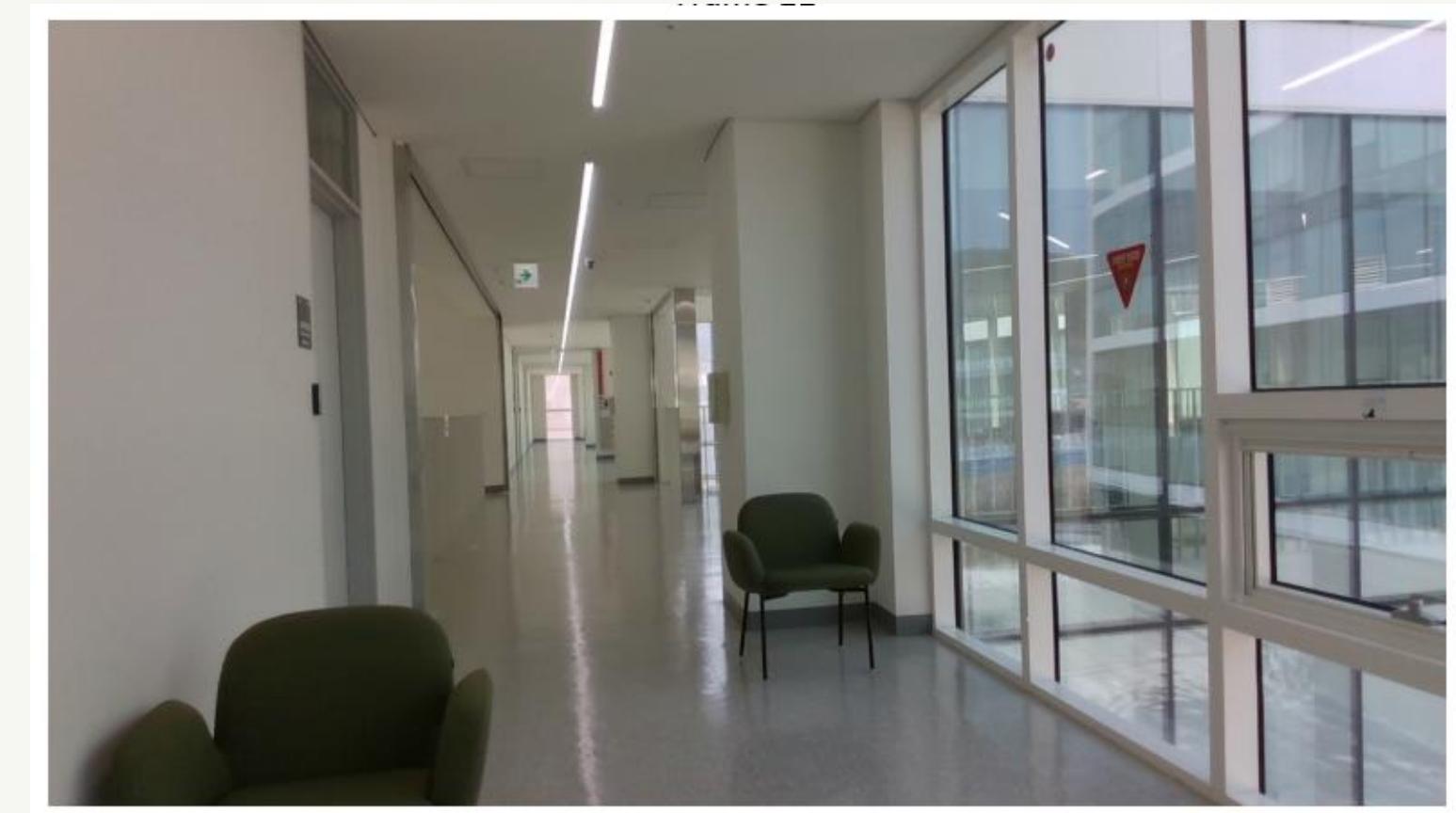
03 실험

Bounding Box Matching 방법 (Abs vs IOU)

실험 장비 : Intel RealSense D435i

실험 장소 : 국립한밭대학교 세종공동캠퍼스 건물

실험 데이터 : 350 프레임 영상



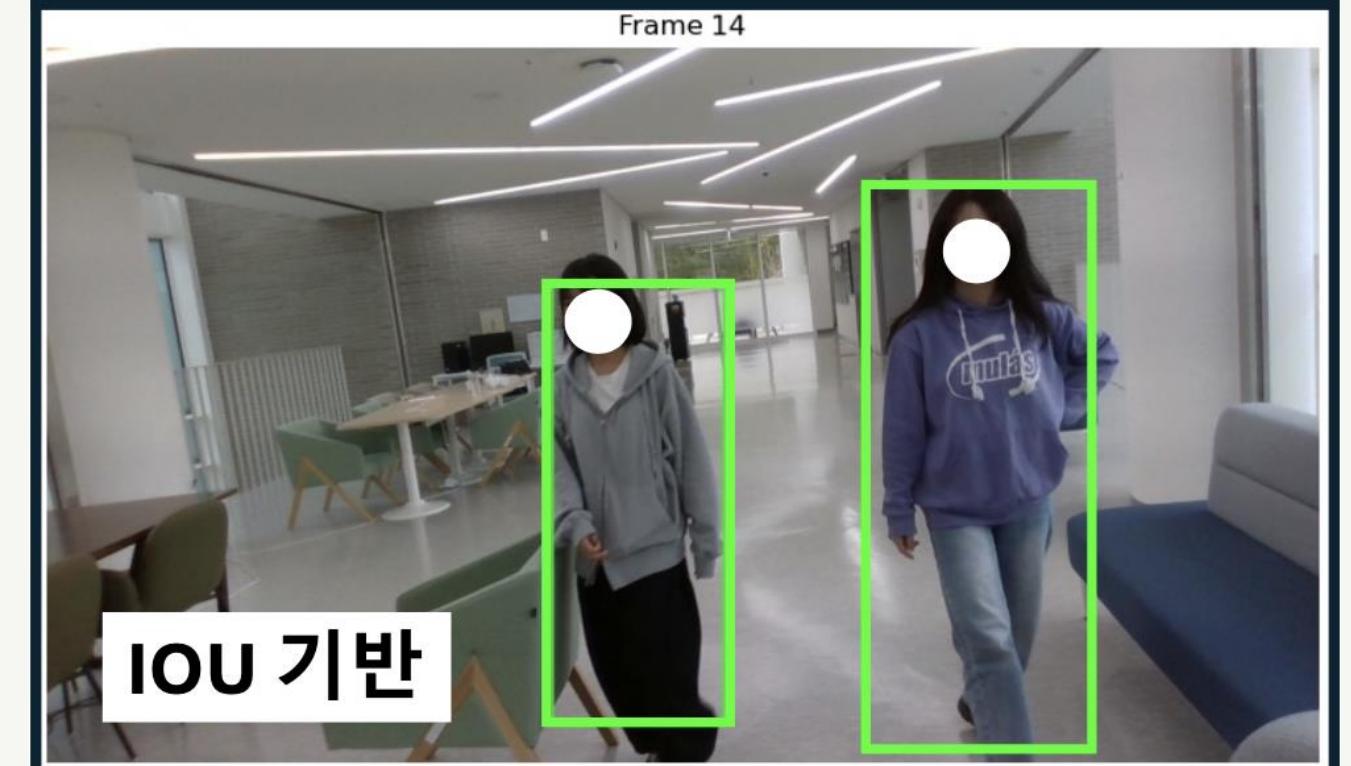
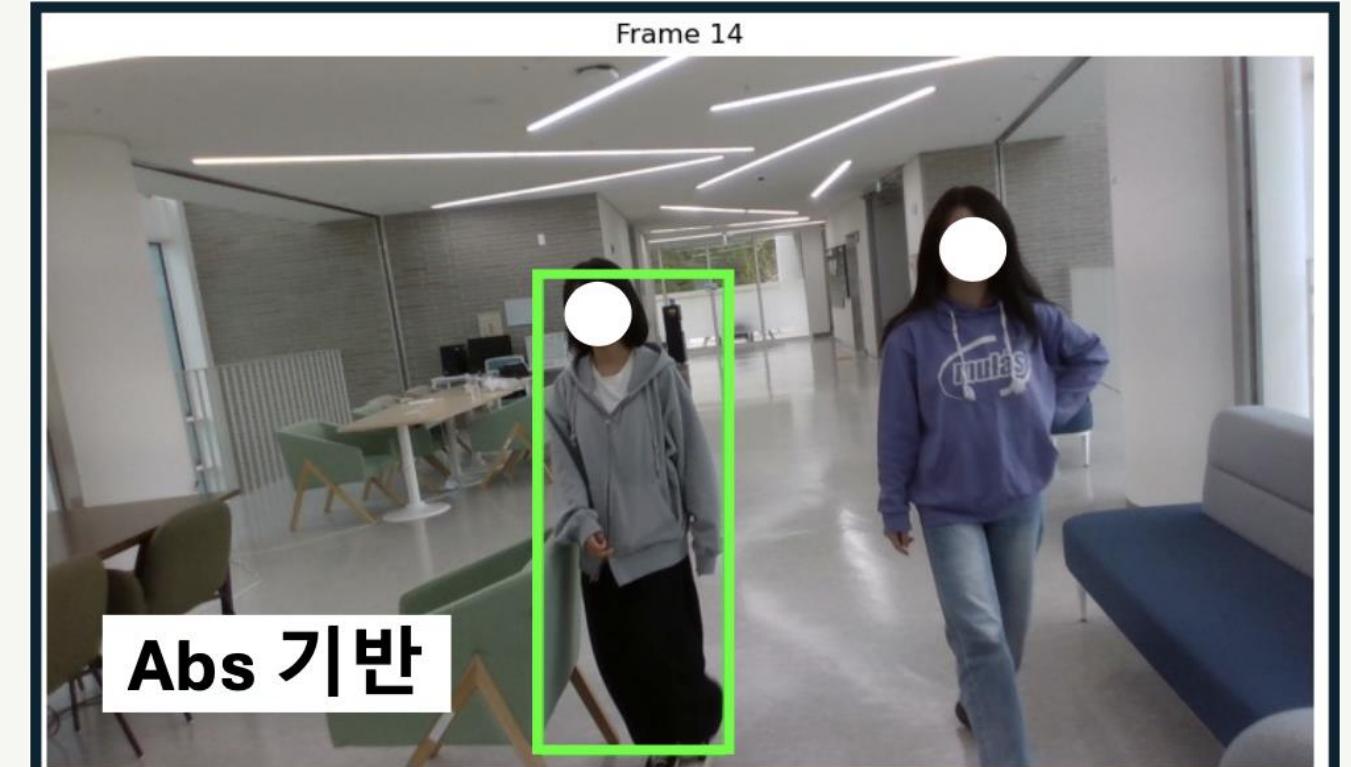
03 실험

Bounding Box Matching 방법 (Abs vs IOU)

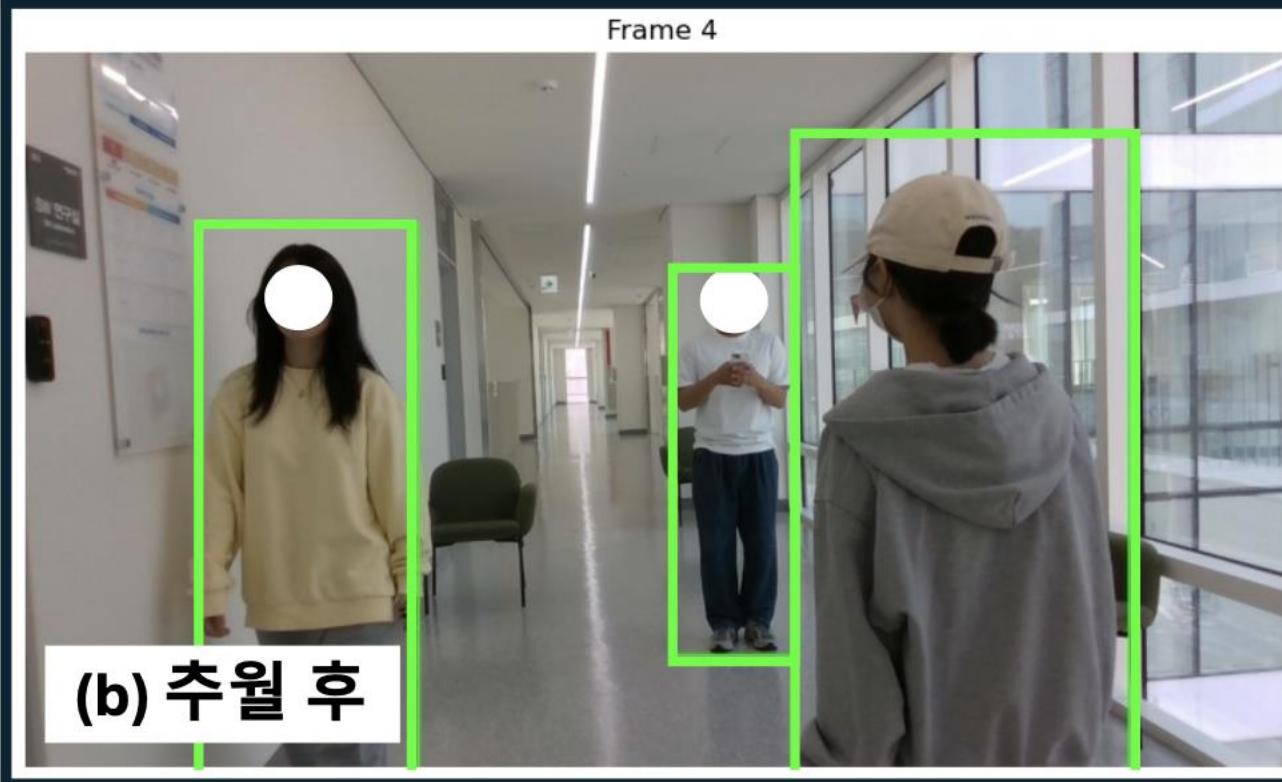
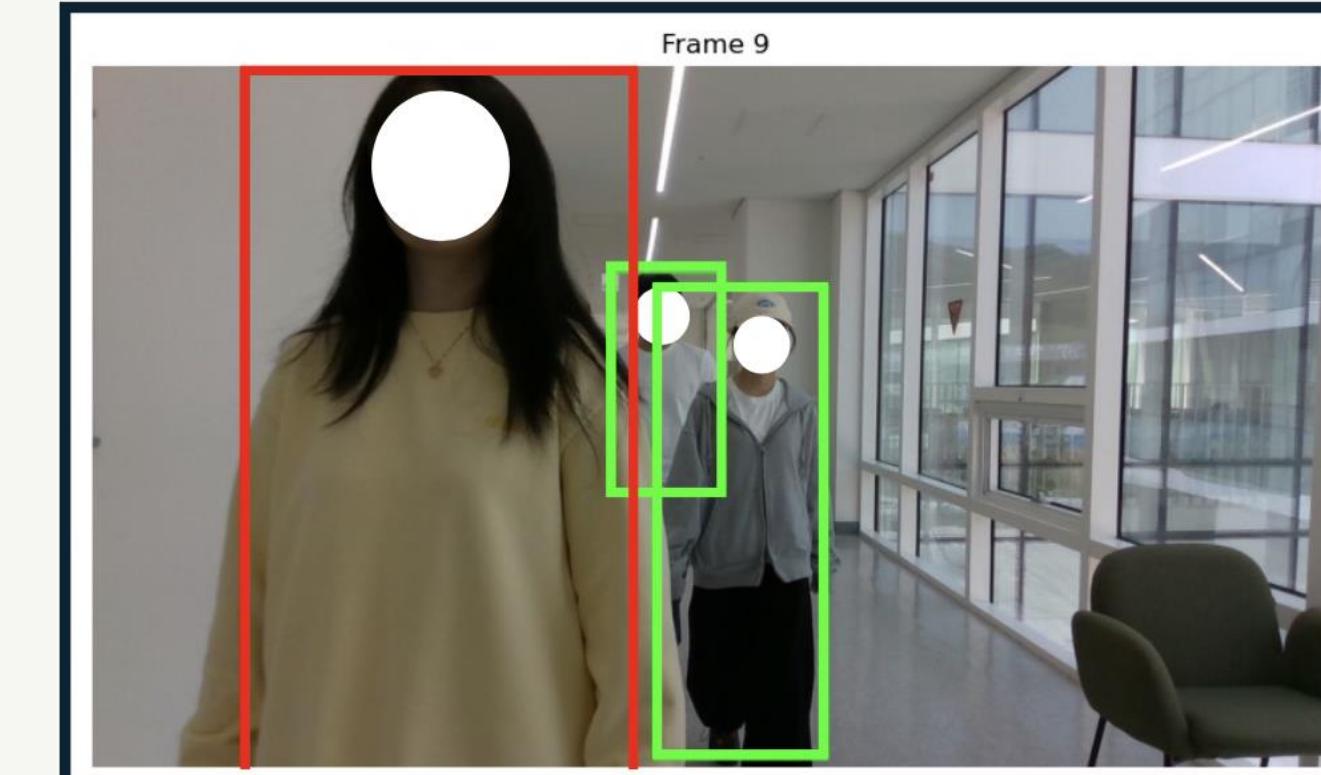
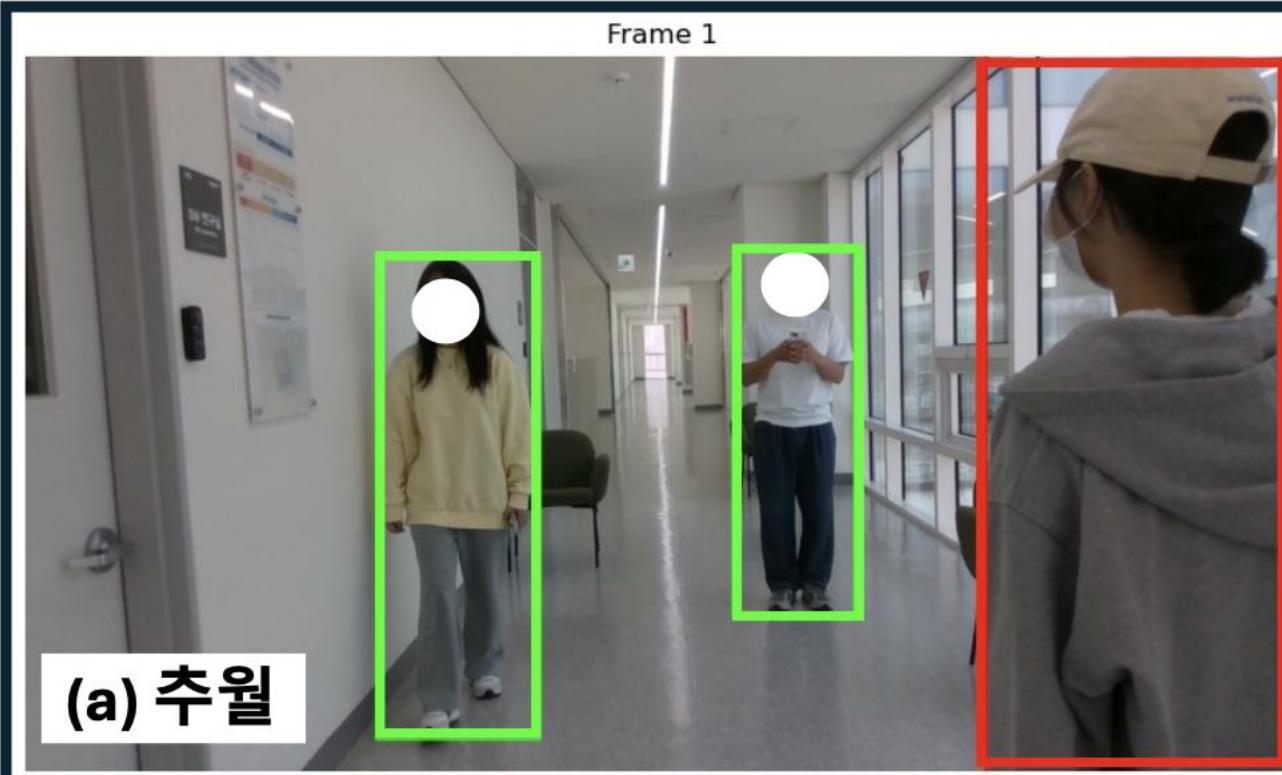
Abs 기반 : 네 개의 꼭지점 좌표 값으로 연결
----→ 작은 변화에도 오차 값이 큼

IOU 기반 : Bounding Box가 겹치는 영역으로 연결
----→ 안정적인 연결 가능

Accuracy	
Abs 기반	0.9143
IOU 기반	0.9857



04 결론

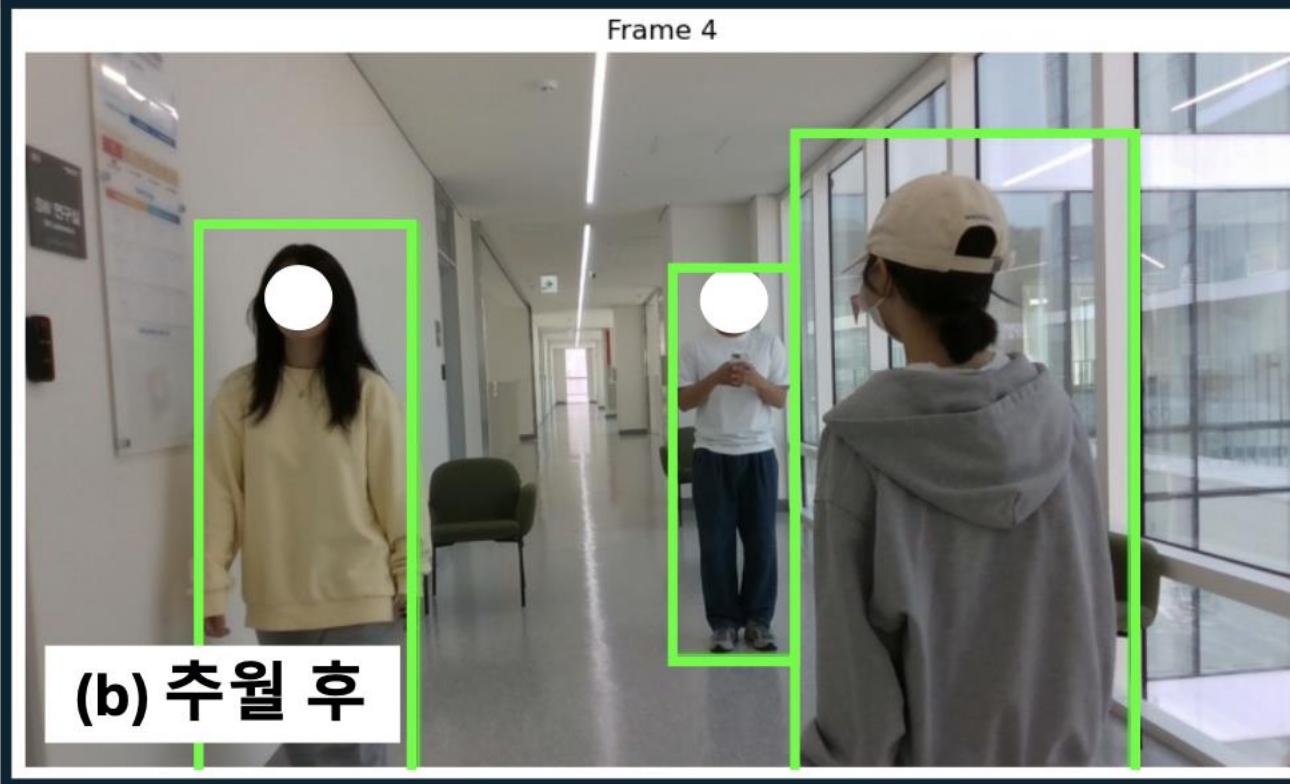
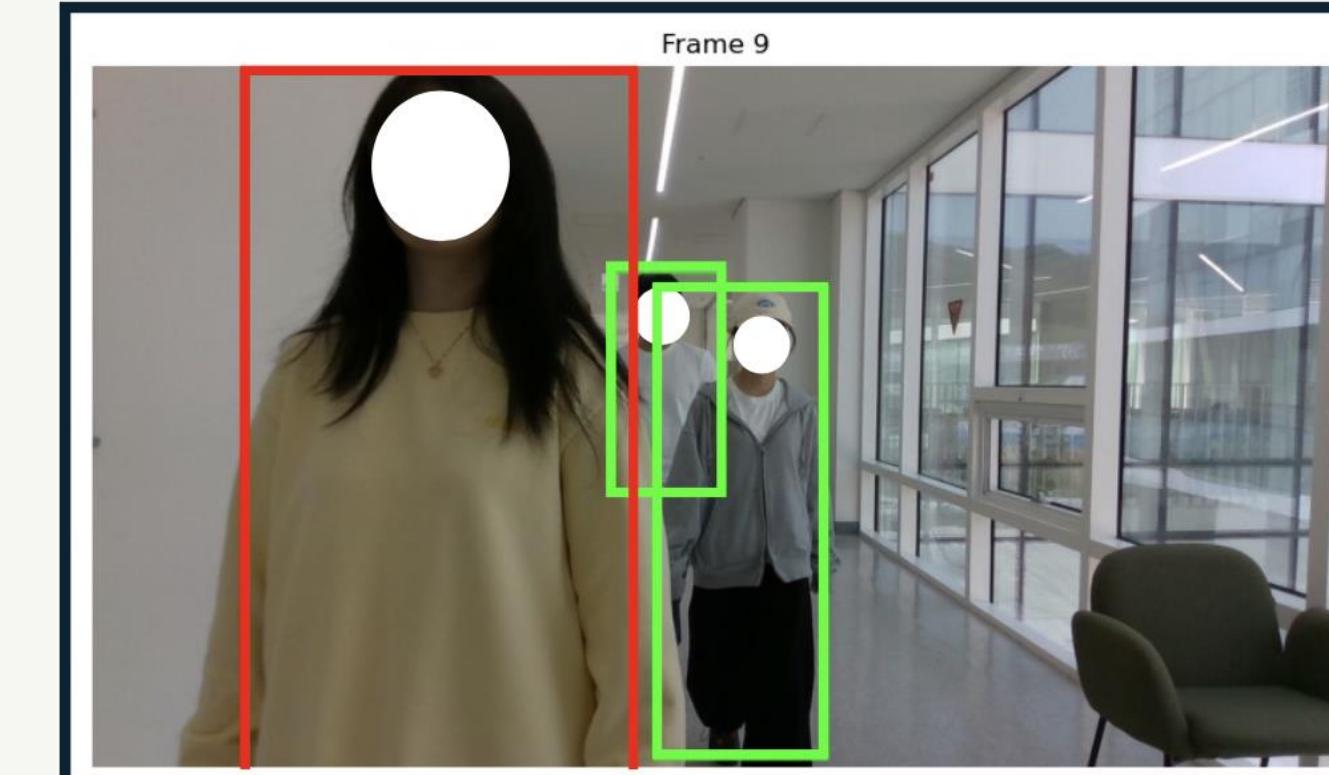
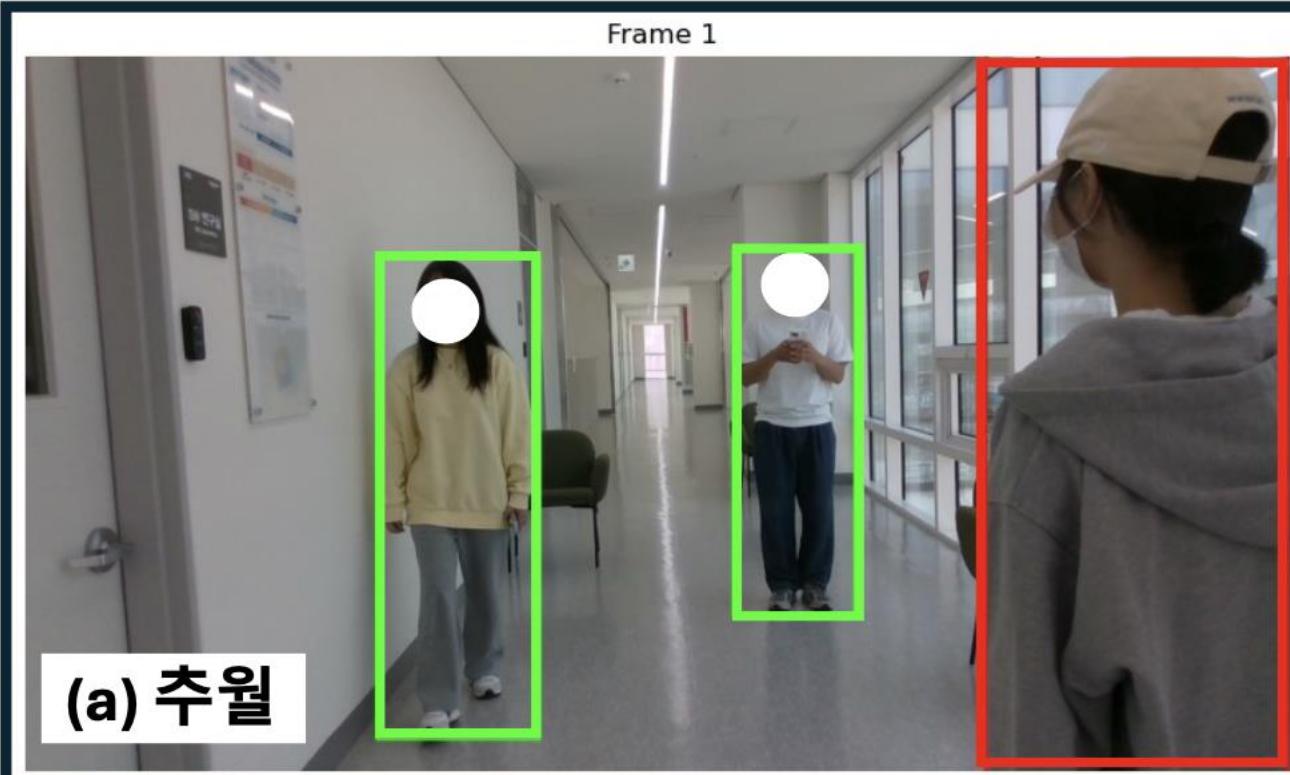


추월 상황

충돌 위험 : 빨간색 Bounding Box
동적 장애물 : 초록색 Bounding Box

평균 처리 시간: 0.045초
평균 FPS: 22.27

04 결론



추월 상황

- 약 98%의 높은 정확도와 다양한 상황에서의 충돌 위험 판단 가능
- 기존 지팡이의 형태를 벗어나 사용자가 보다 자유롭고 안정적인 보행이 가능해짐

감사합니다
