PointAnything

Presented by Group 20

111550089 李宗諺
111550125 黃仁駿
111705069 劉冠言
111550101 游惠晴

Introduction

Overview:

從RGB照片中辨識人物指向的物體

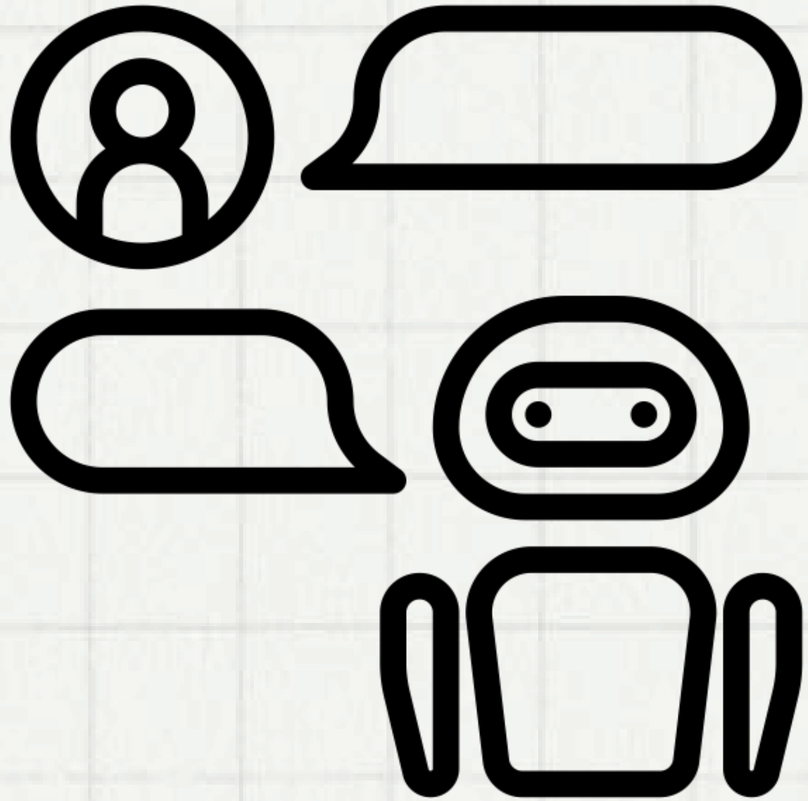
Importance:

人機互動

- 理解人類手勢與意圖
- 提升機器人的交互能力

Difficulties:

- 人物指向分析
- 物體辨識



Related Work

Paper:

Point Anywhere: Directed Object Estimation from Omnidirectional Images

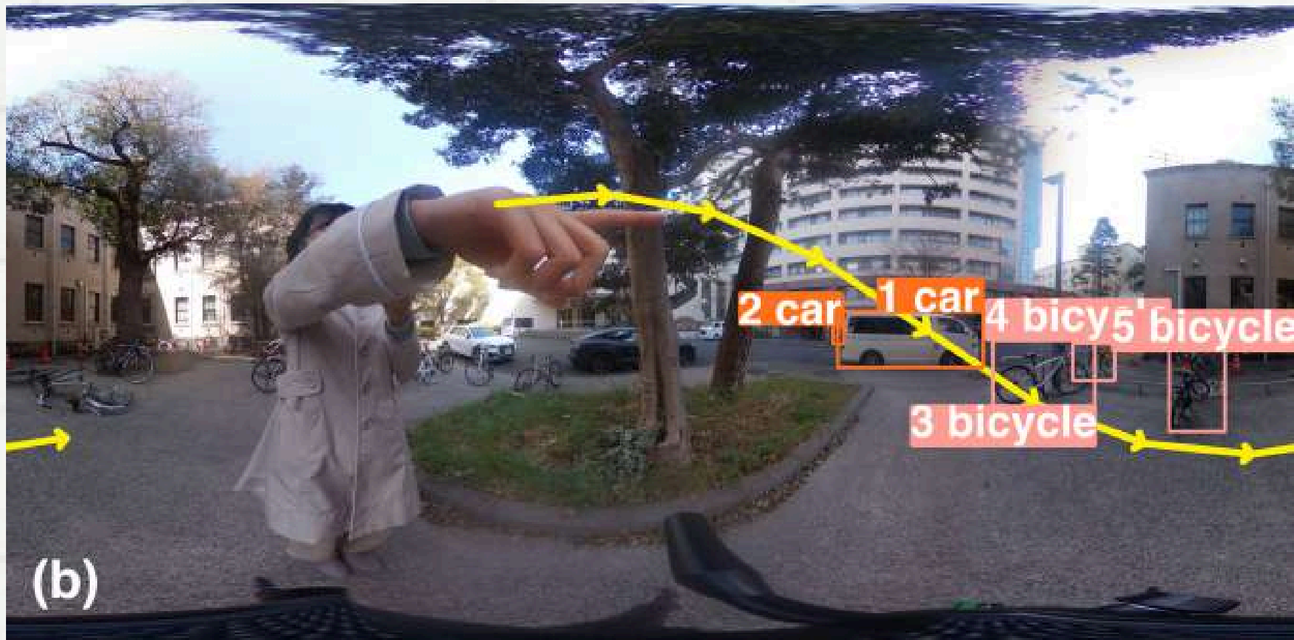
(Kotani, N., Kanezaki A., SIGGRAPH Poster, 2023)

Objective:

- Omnidirectional images as input

Naive Approach

Poor Performance (Accuracy:27%)



The Problem of Naive Approach

The importance of depth information

Output: Sofa

Answer: Teddy bear

Related Work

Paper:

Foundations of Visual Linear Human–Robot Interaction via Pointing Gesture Navigation

(Tölgyessy, M. et al.,Int J of Soc Robotics 9,2017)

Objective:

- Dog-like robot
- Identify the area being pointed at
- The robot move to the place being pointed at

Related Work

Paper:

Recognition and Estimation of Human Finger Pointing with an RGB Camera for Robot Directive

(Bamani, E. et al., ArXiv, 2023)

Objective:

- Dog-like robot
- Identify the area being pointed at

Dataset

為何不使用已有的資料集?
網路上並沒有適合的已標註資料集

我們的資料集:
在交大校園與家中各處拍照，
標註圖片中正在指向的物體

共計 80 張

Example:

Ans: Bench

Ans: Bus

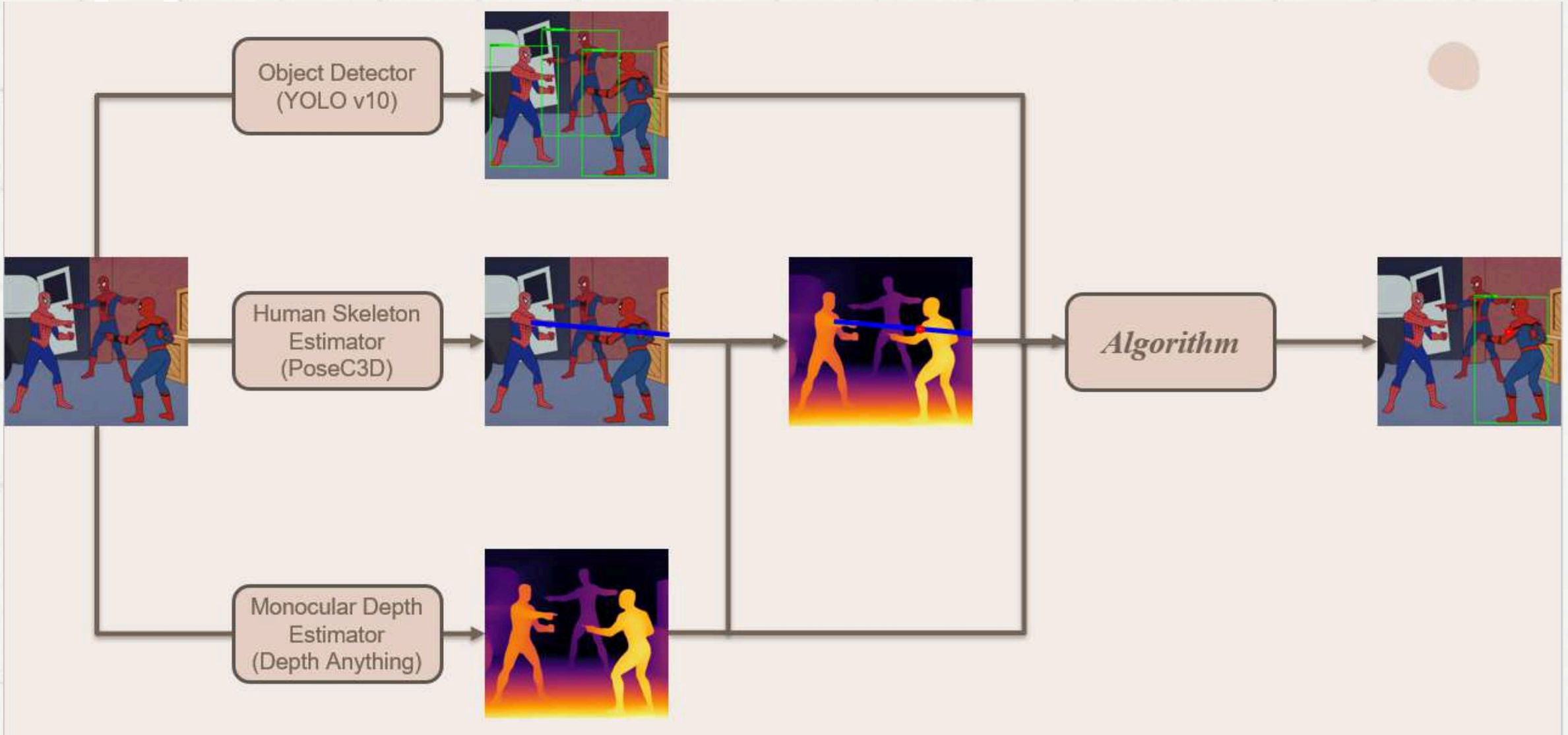
Baseline

Why we choose Large Multimodality Models?

沒有找到現存的與我們題目相同的研究。而我們預想的應用場景是人機互動，我們認為多模態模型在理解和處理多模態數據方面具有顯著優勢。這些模型能夠有效地辨識出RGB照片中人物指向的物體。因此，我們選擇了幾個先進的多模態模型作為對照組。

- LLaVA-v1.5-13b-4-bit
  Release:2023/10/5
- LLaVA-NeXT-72b
  (LLaVA v1.6)
  Release:2024/5/10
- GPT-4o
  Release:2024/5/13

Approach


Object Detector
(YOLO v10)
Human Skeleton
Estimator
(PoseC3D)
Monocular Depth
Estimator
(Depth Anything)
Algorithm

Algorithm

1.利用YOLOv10偵測到的bounding box(包含物品種類和範圍)，與Depth Map的深度資訊，找出代表該物體的點的座標(利用Normalized的Depth Map，使用眾數作為代表該box的點的深度，統計Normalized的Depth Map在該深度的所有點的座標平均)

2.利用PoseC3D的前臂骨架座標與Depth Map的深度資訊，算出手臂射線的參數式

3.找到與Bounding Box中代表點與射線最近的，即是所辨識到的物體

Validation

For Large Multimodality Models:

輸入Dataset中的圖片，使用以下Prompt:

“Please describe the item that the person in the image is pointing at. Focusing on the type of the item”

得到回覆後人工批改，檢查是否與Dataset中的答案相同

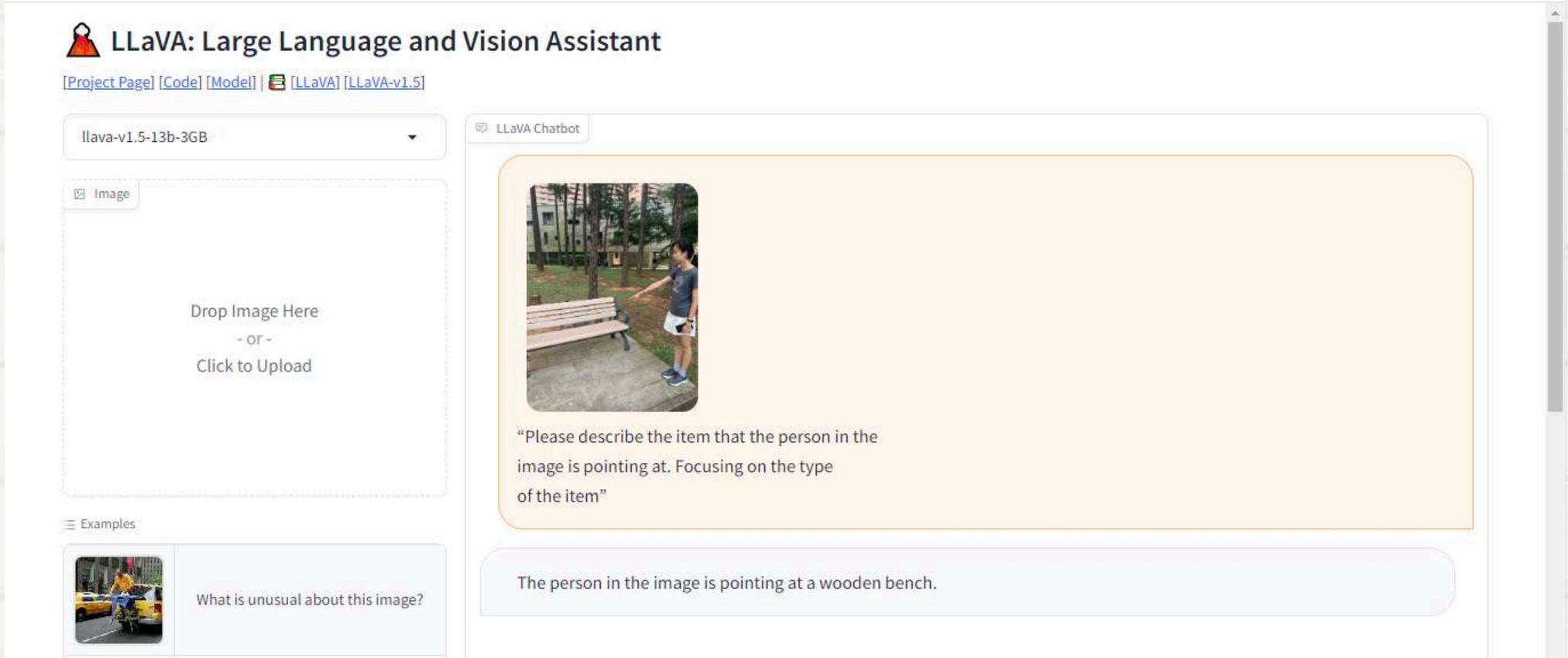
For Our Method:

將Dataset中的照片，經過我們設計的pipeline，得到兩隻手分別指的物體的照片，人工批改(包含檢查bounding box範圍是否正確與物品種類是否正確)

Validation

How we test our baseline?

- LLaVA-v1.5-13b-4-bit
  Release:2023/10/5



Validation

How we test our baseline?

- LLaVA-NeXT-72b
  (LLaVA v1.6)
  Release:2024/5/10

https://llava-next.lmms-lab.com/

Validation

How we test our baseline?

- GPT-4o

  Release:2024/5/13

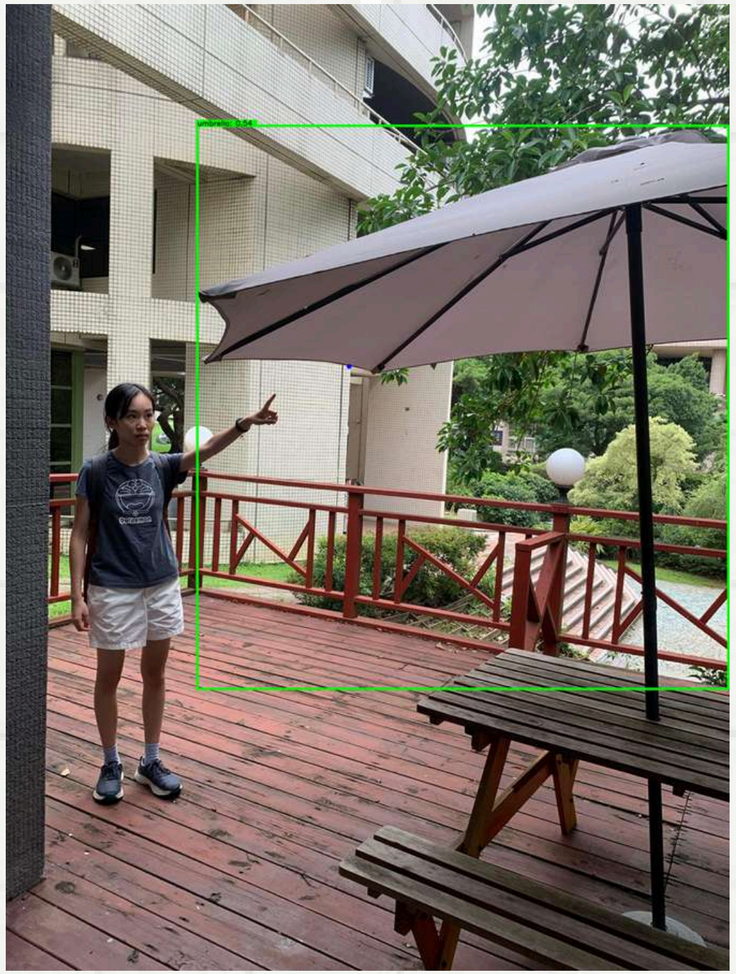
Results

	Ours	LLaVA-v1.5-13b-4-bit Release:2023/10/5	LLaVA-NeXT-72b (LLaVA v1.6) Release:2024/5/10	GPT-4o Release:2024/5/13
Accuracy	18/80=22.5%	53/80=66.25%	53/80=66.25%	64/80=80%

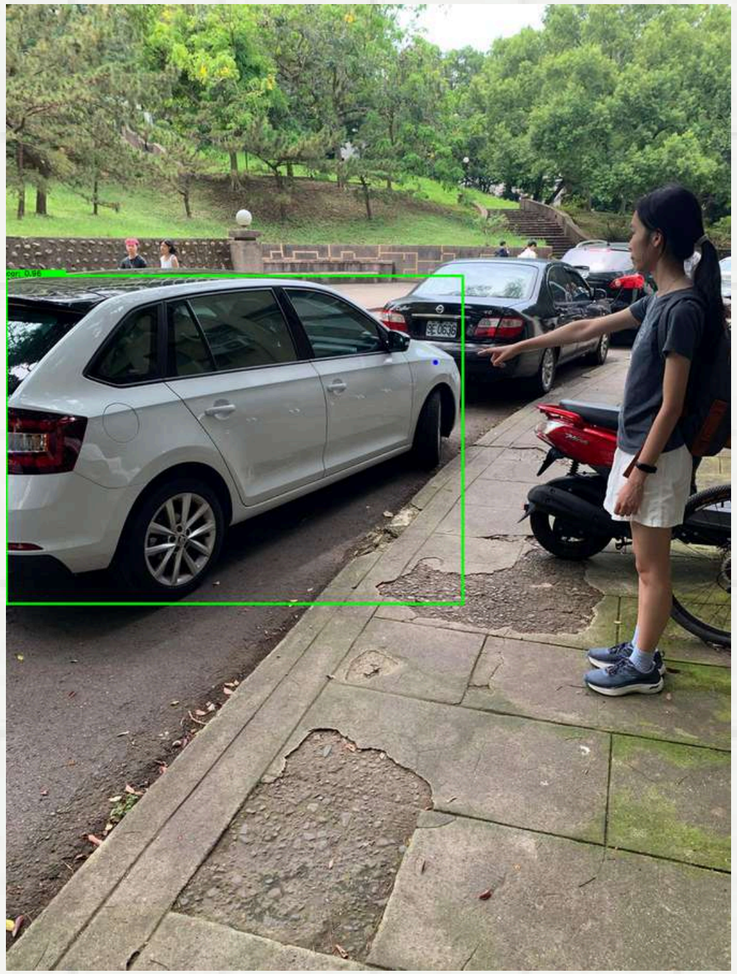
Link of the Results:

Success Examples

Ans: Bench



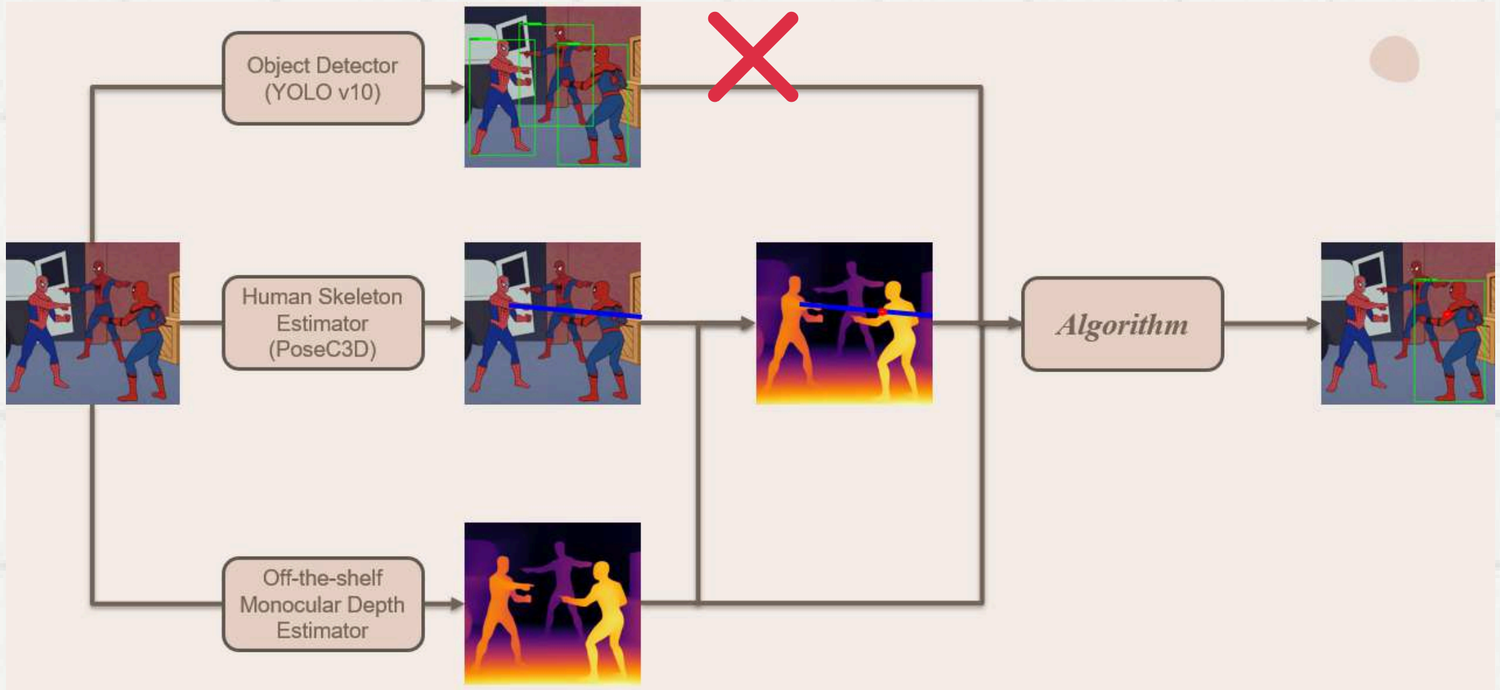
Ans: Umbrella



Ans: Car

Problems We Encountered

Case 1. Object Detector Fail

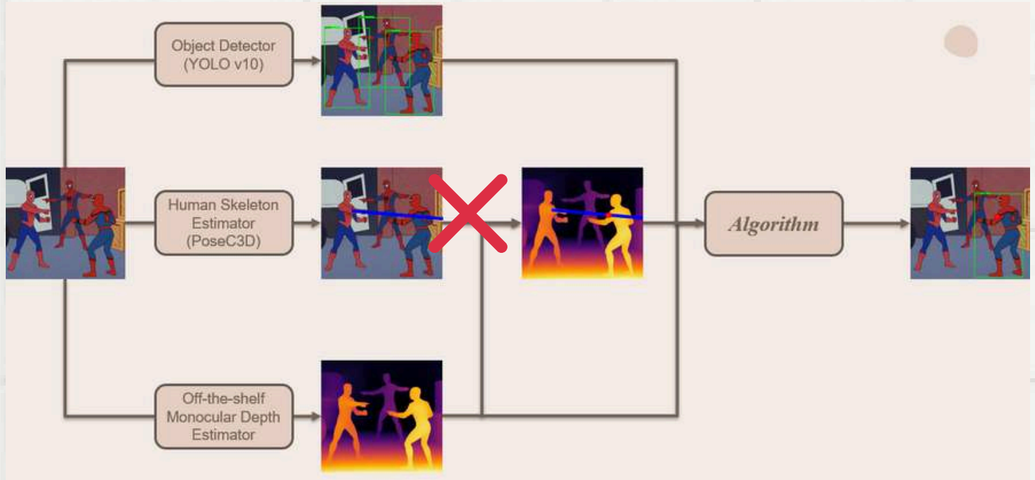


Example

Problems We Encountered

Case 2.

Hands Pointing inaccurate

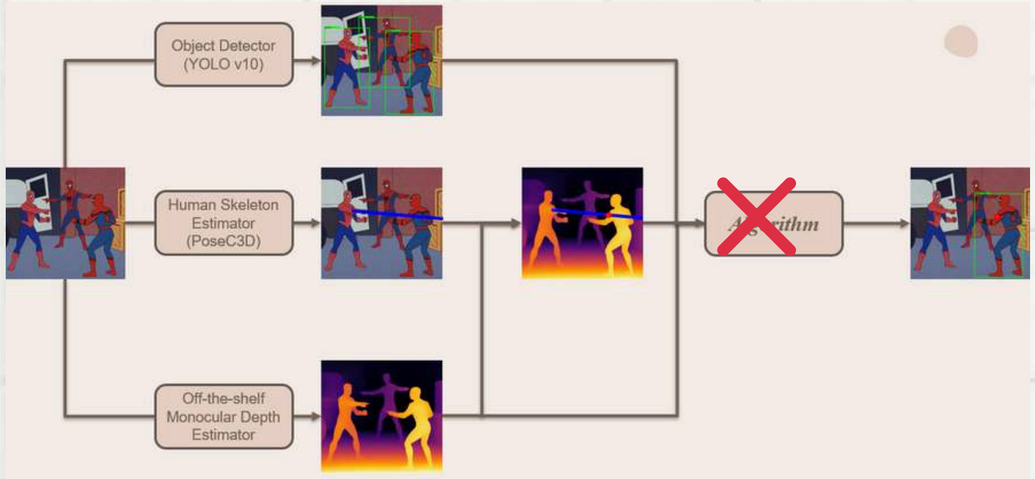


Example

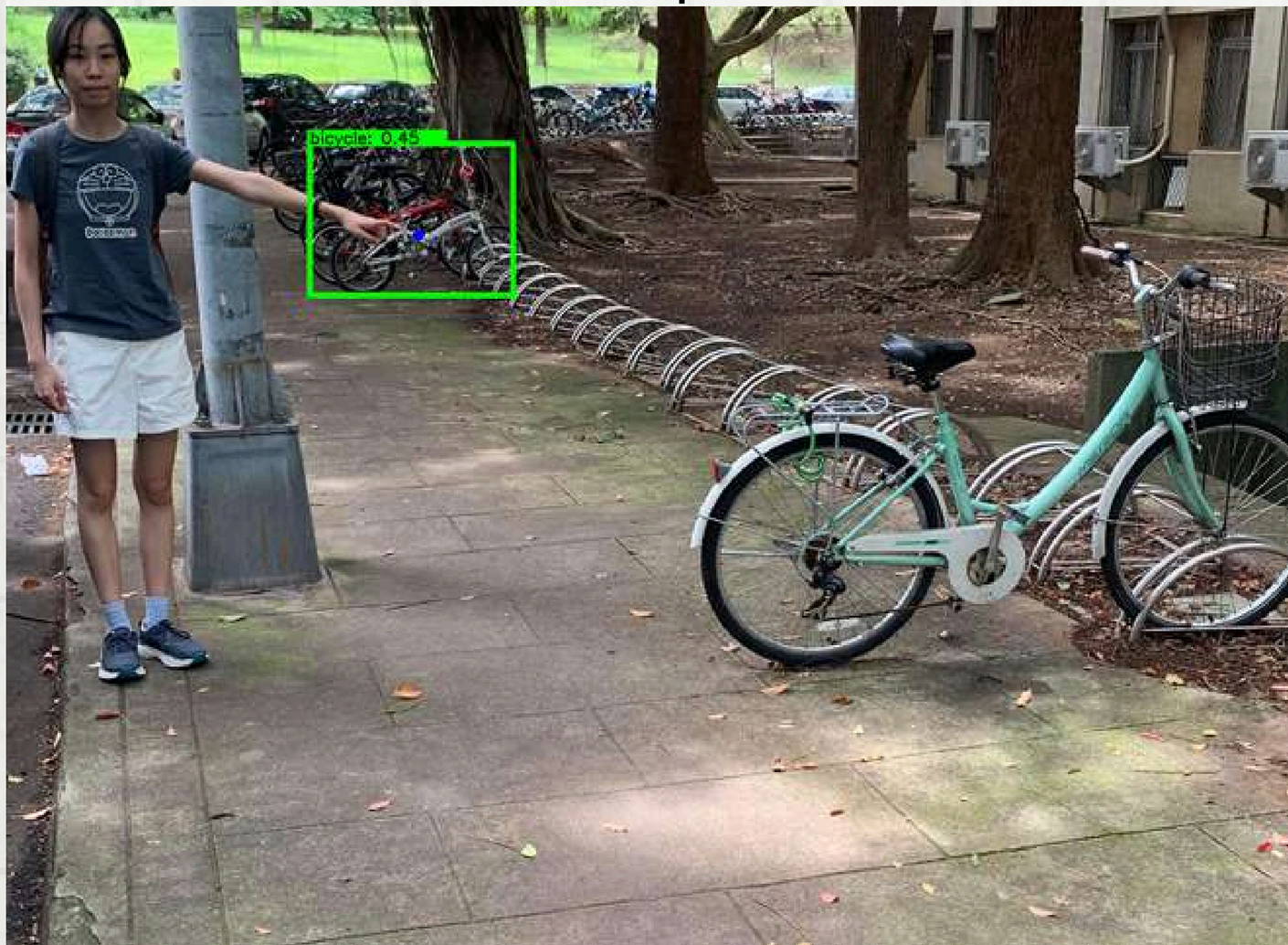
Problems We Encountered

Case 3.

Depth map Inaccuarte



Example



Analysis

Case 1 : Object Detector Fail

我們認為可以在獲得更好的物體辨識模型後解決(YOLO v10用的COCO只有80種物品)

Case 2 : Hands Pointing inaccurate，我們認為Physic-based的模型較難解決，可能要針對指向的射線作更精確地估計與調整

Case 3 : Bounding Box Depth Estimate Failed

我們觀察到分成以下情形

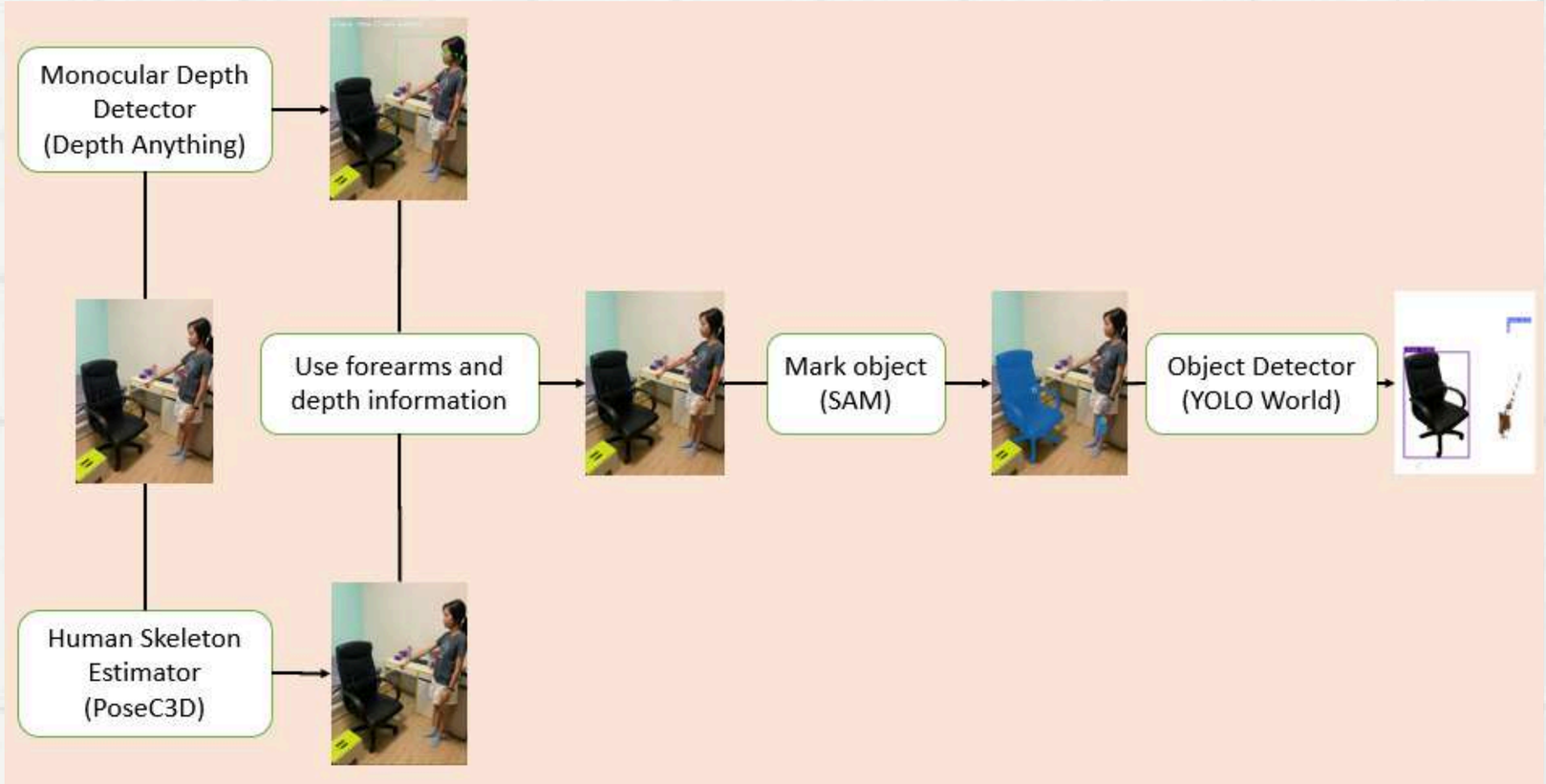
1. 深度圖本身就不準
2. boudnding Box中利用眾數估計代表點深度的方法受到圖片背景的影響而不準

針對情況一，我們認為可以在獲得更好的深度模型後改善

針對情況2，我們提出新方法

Adaptive Approach Ver 1

加入Segment Anything，三維射線從前臂出發碰到的第一個物體mask，切下後讓YOLO辨識(精確知道每個pixel屬於哪個物體，不用估計代表點座標)



Results

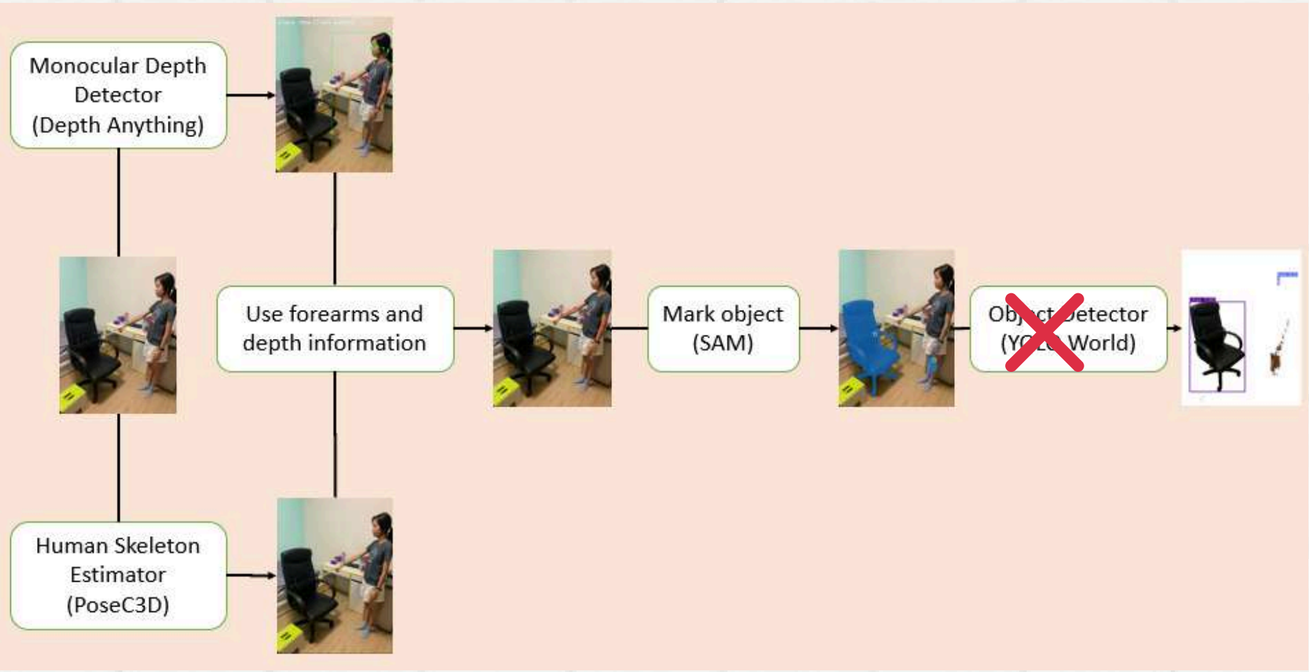
	Ours	Ours Adaptive Approach	LLaVA-v1.5-13b-4-bit Release:2023/10/5	LLaVA-NeXT-72b (LLaVA v1.6) Release:2024/5/10	GPT-4o Release:2024/5/13
Accuracy	18/80=22.5%	17/80=21.25%	53/80=66.25%	53/80=66.25%	64/80=80%

Link of the Results:

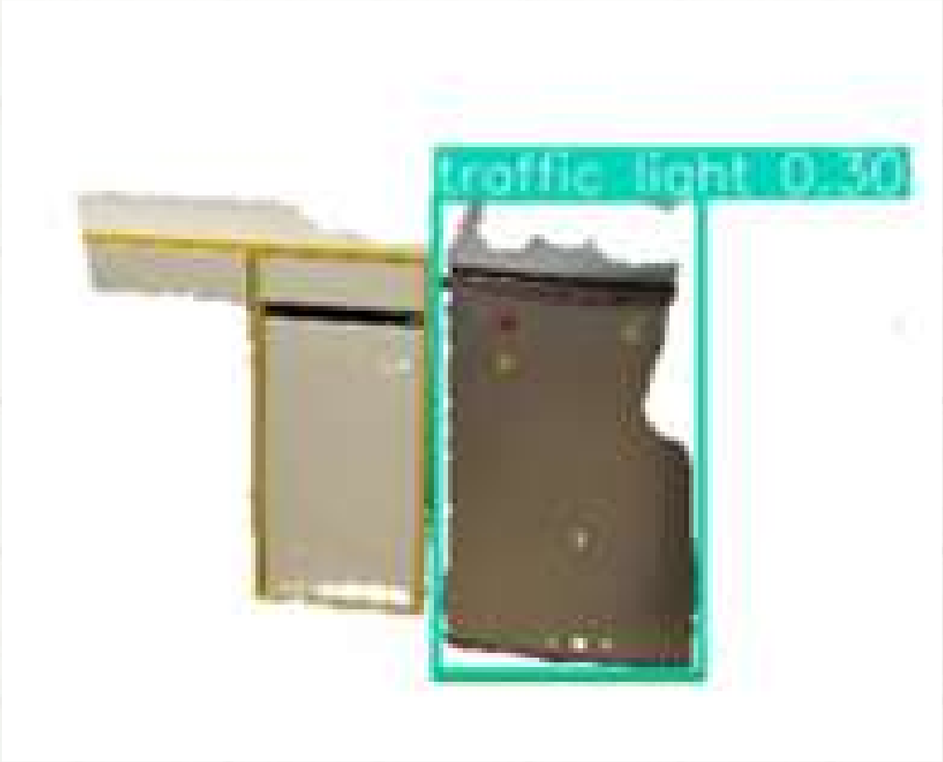
Problems We Encountered

Case 1.

Object Detector Fail



Example



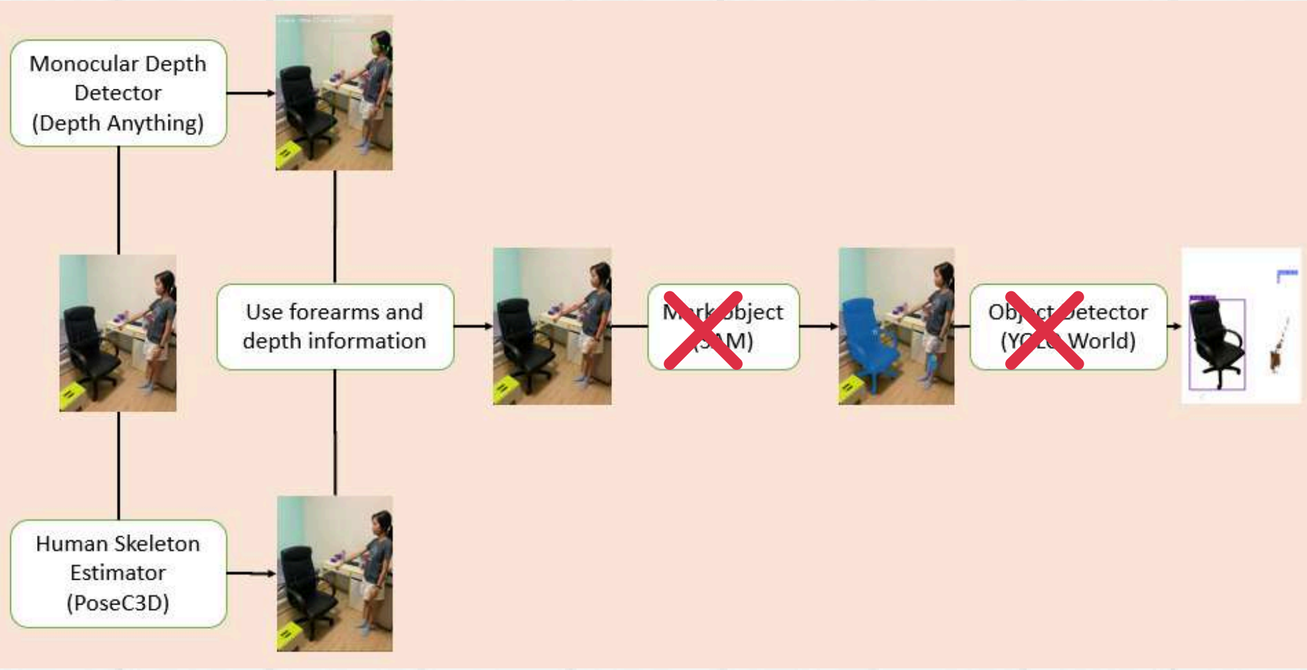
Ans: Refrigerator

Output: Traffic light

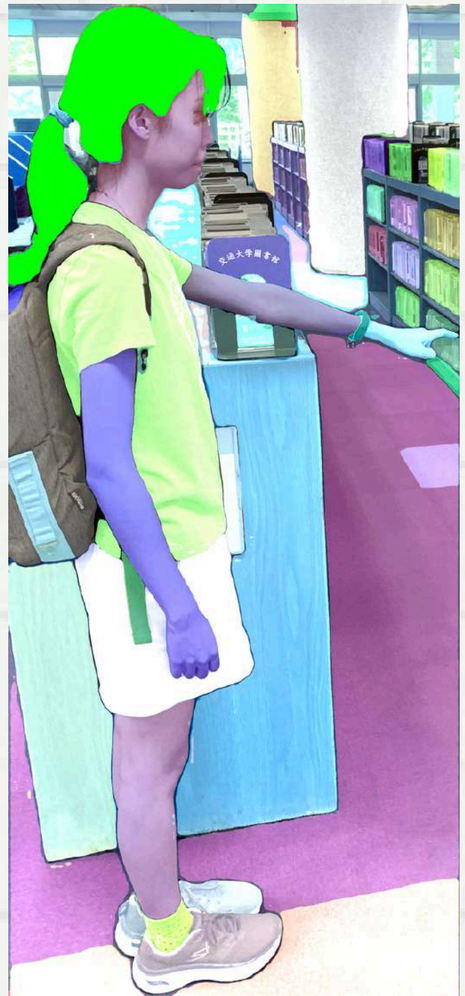
Problems We Encountered

Case 2.

Segment Anything cutting too detailed，Object Detector Fail



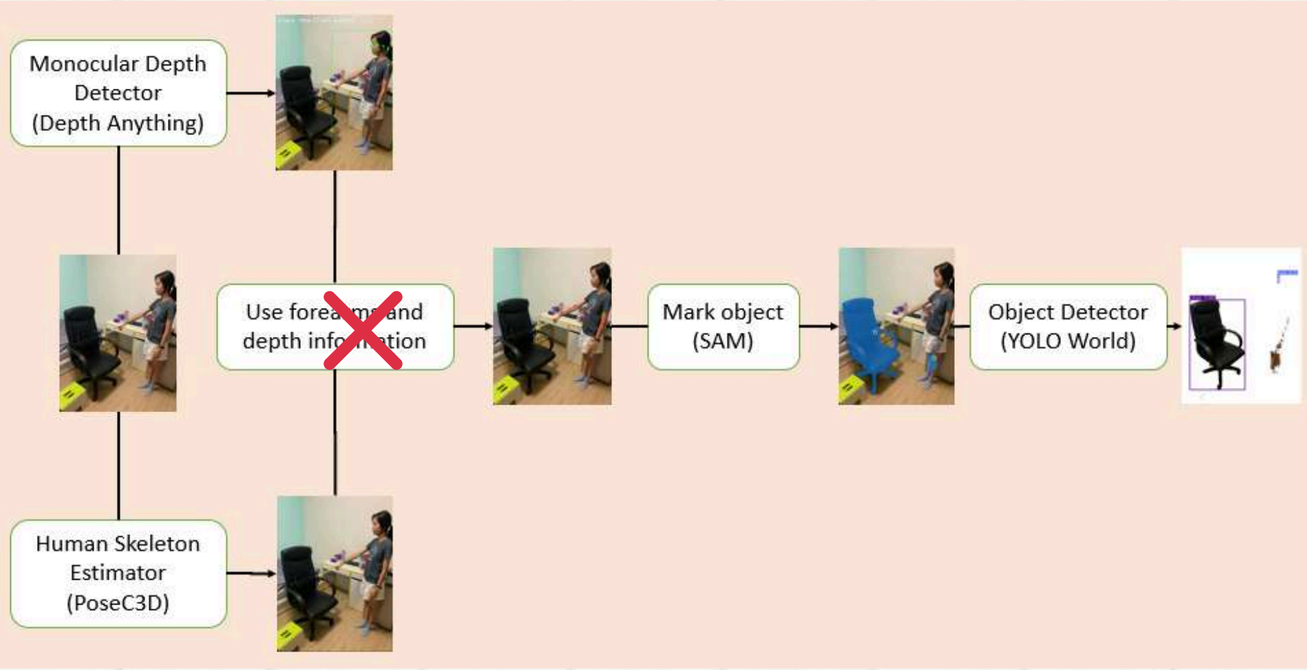
Example



Problems We Encountered

Case 3.

Depth Map Inaccurate



Example

Analysis

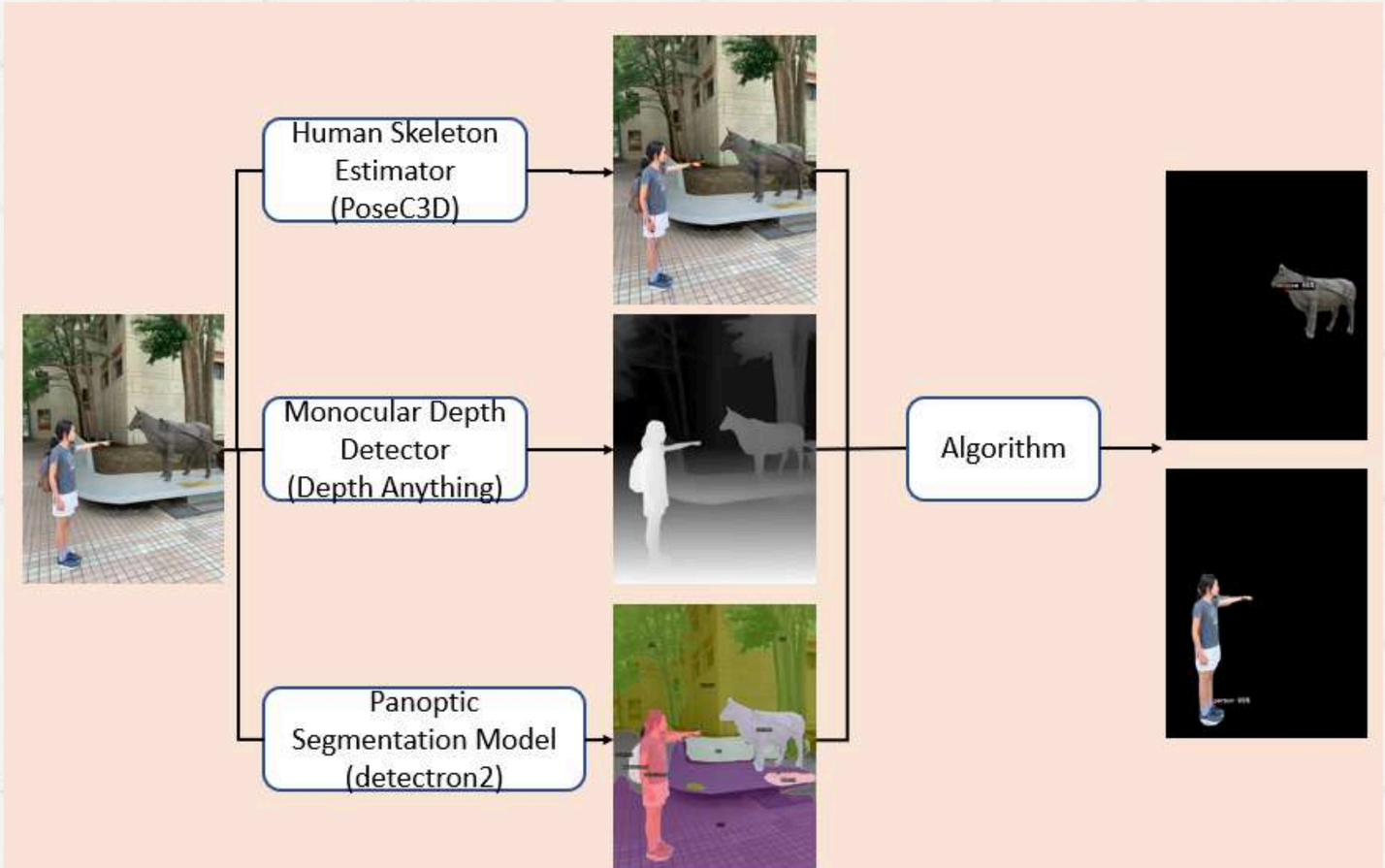
Case 1 : Object Detector Fail我們認為可以在獲得更好的物體辨識模型後解決

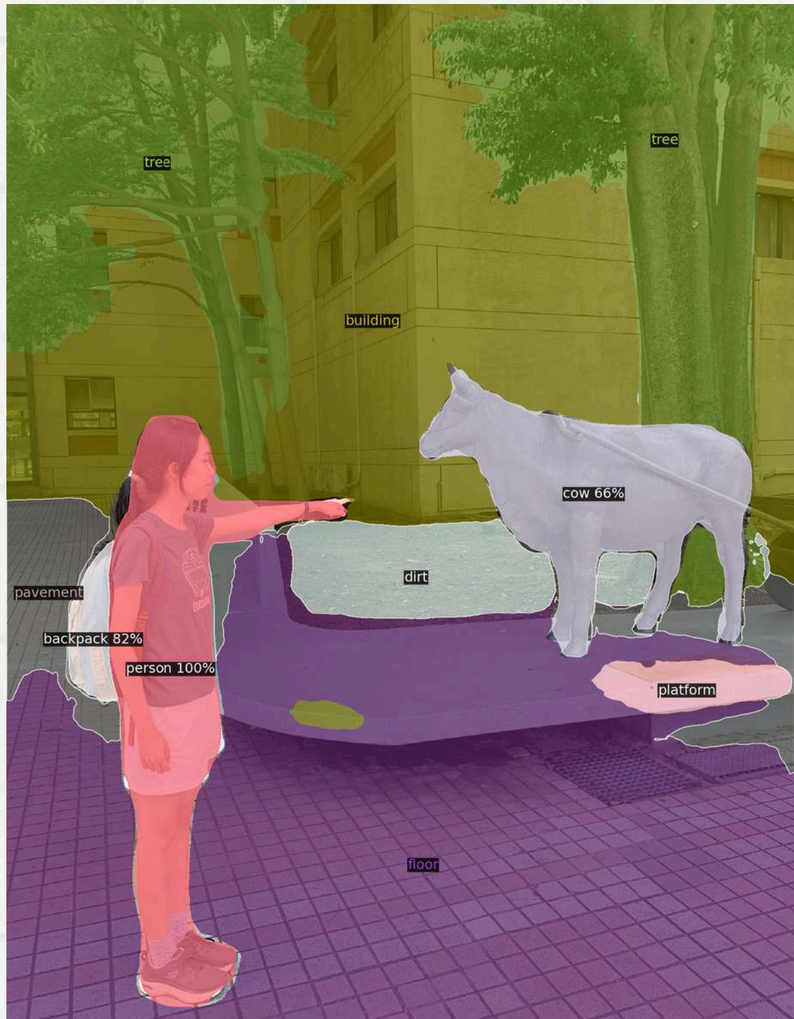
Case 2 : Segment Anything cutting too detailed，Object Detector Fail，我們認為可以透過更換成其他圖片分割模型解決

Case 3 : Depth Map Inaccurate，我們認為更好的深度預測模型可以改善。

Adaptive Approach Ver 2

利用Detectron 2的Panoptic FPN R101做全景分割，取代SAM+YOLO





Results

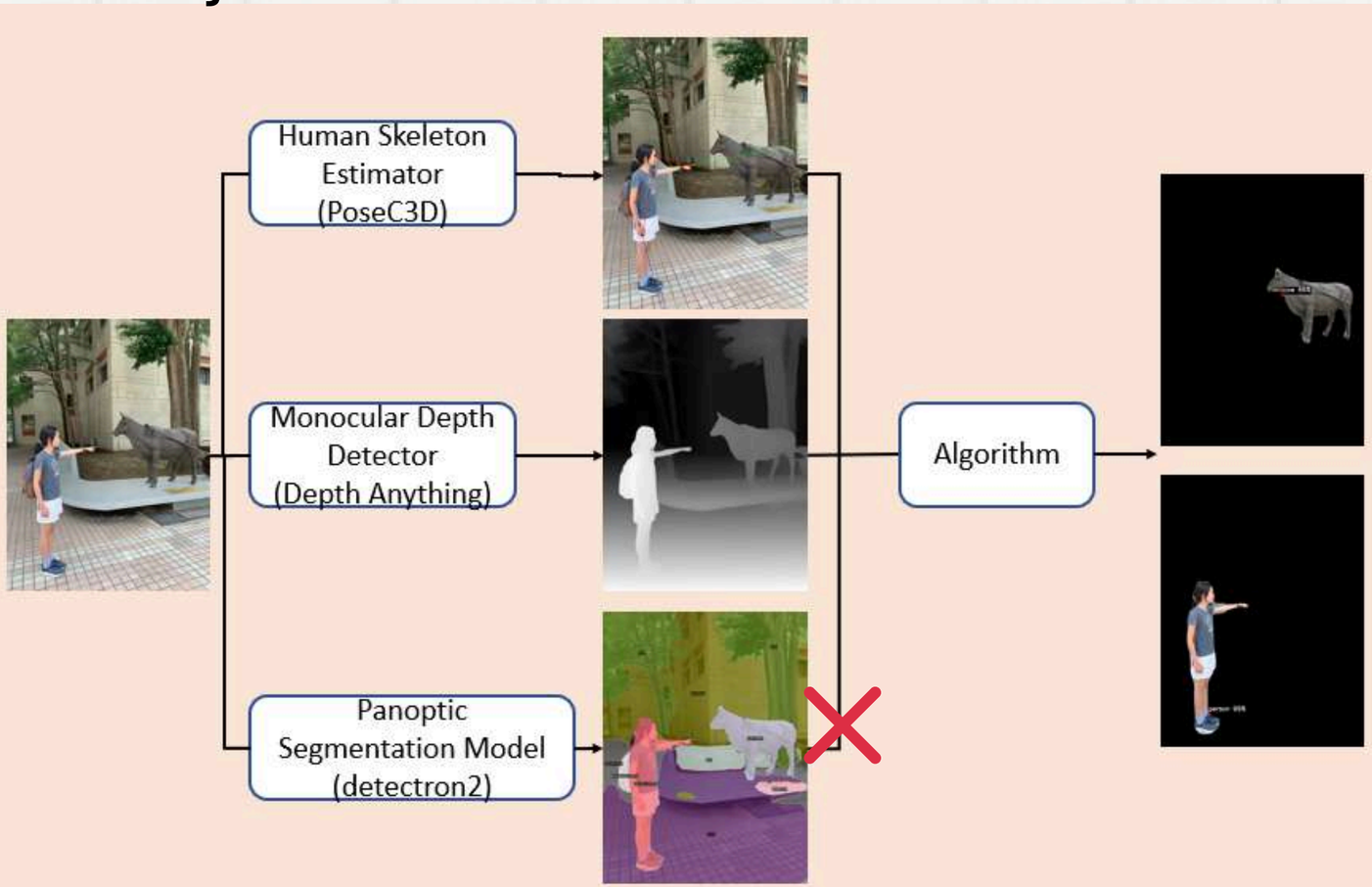
	Ours	Our Adaptive Approach v1	Our Adaptive Approach v2	LLaVA-v1.5-13b-4-bit Released: 2023/10/5	LLaVA-NeXT-72b (LLaVA v1.6) Released: 2024/05/10	GPT-4o Released: 2024/05/13
Accuracy	18/80=22.5%	17/80=21.25%	20/80=25%	53/80=66.25%	53/80=66.25%	64/80=80%

Link of the Results:

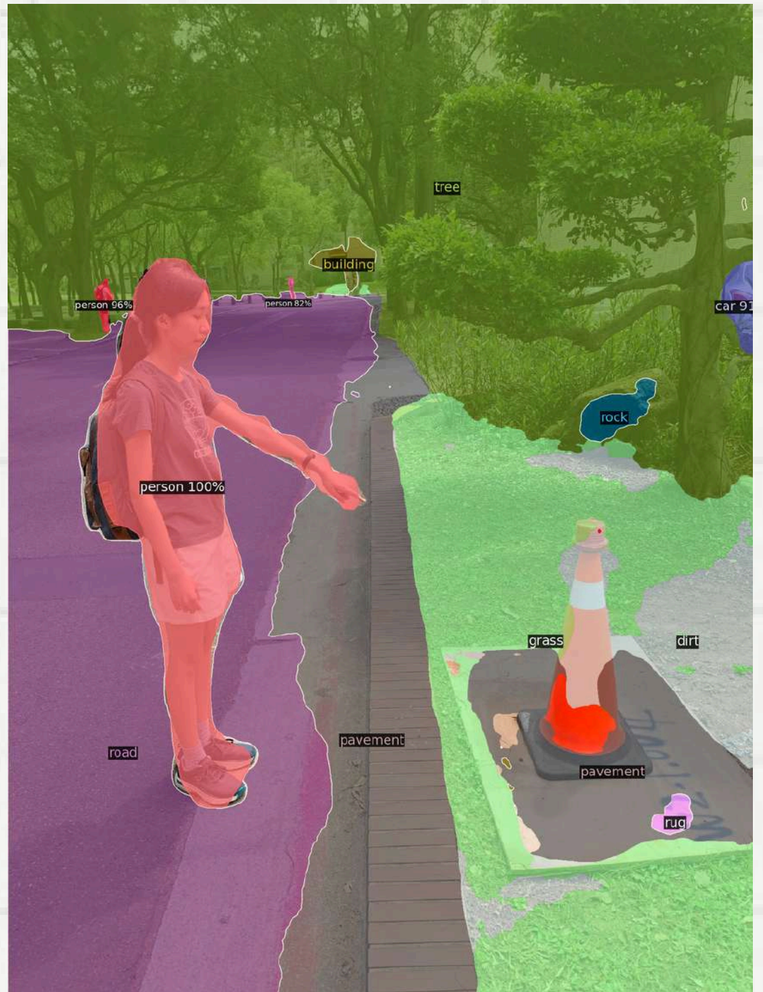
Problems We Encountered

Case 1.

Object detector fail



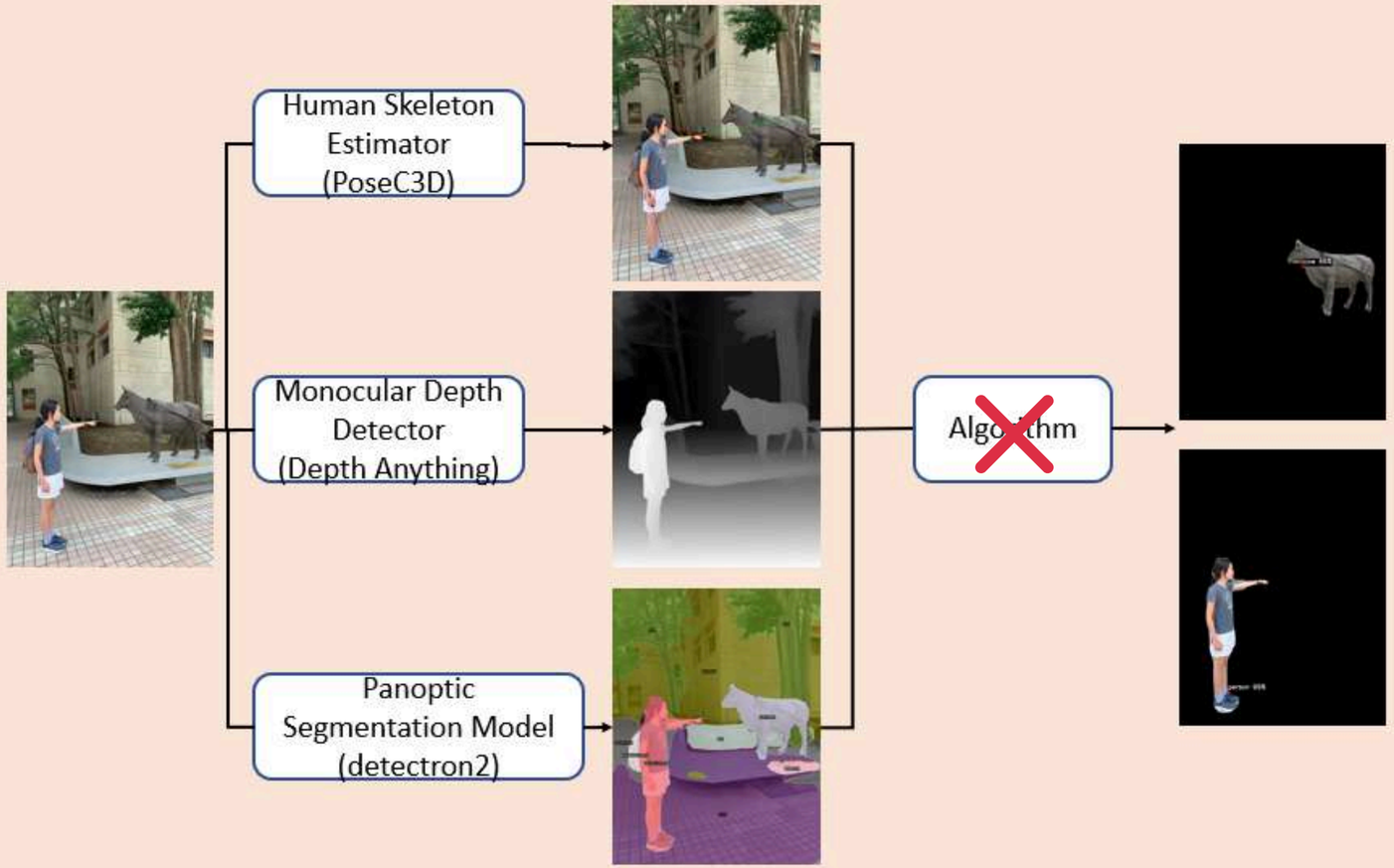
Example



Problems We Encountered

Case 2.

Estimate Point Inaccurate



Example

Analysis

Case 1 : Object Detector Fail
我們認為可以在獲得更好的物體辨識模型後解決

Case 2 : Estimate Point Inaccurate
我們觀察到問題分成以下幾種情況

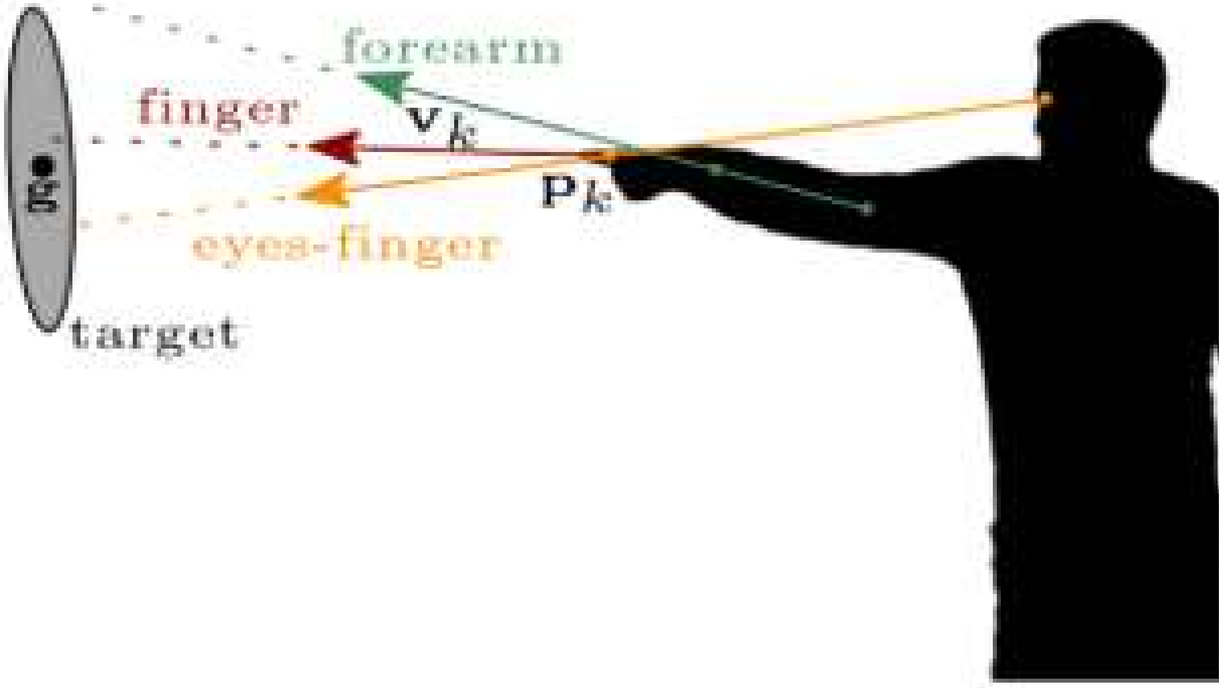
- 人指的不準
- 深度圖不準
- 射線計算的不準

針對情況三，我們提出改進作法三

Adaptive Approach Ver 3

參考Related work中第三篇提到的問題

實作射線修正



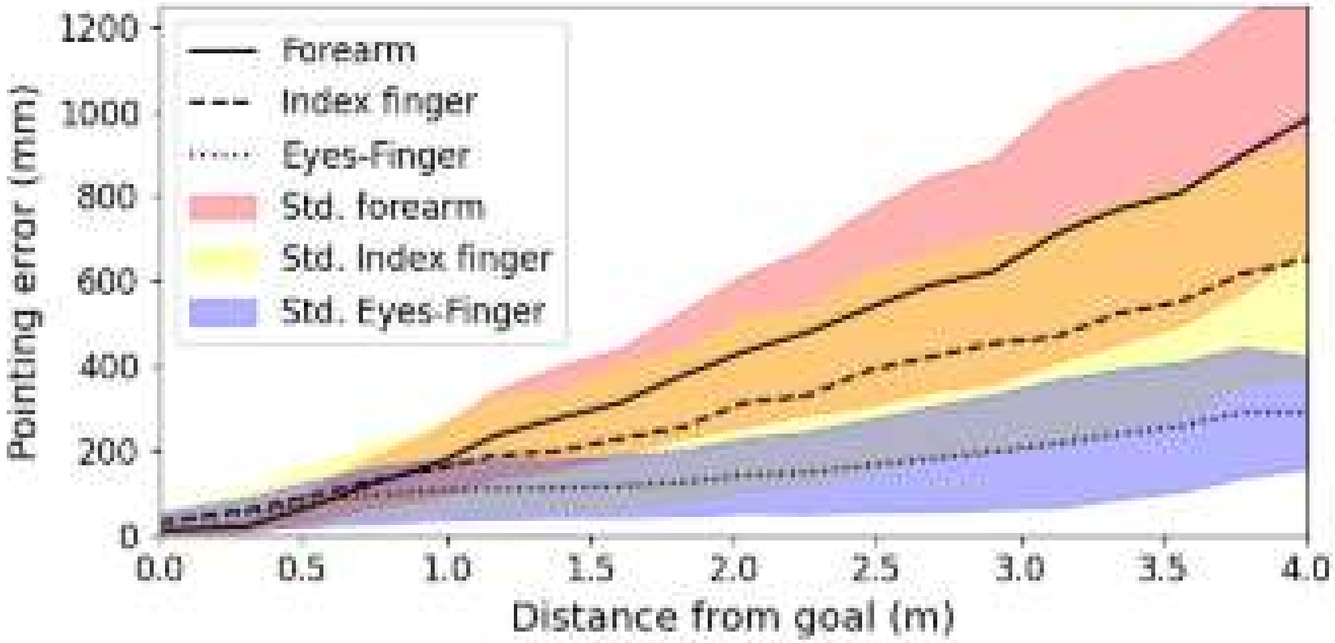


Fig. 4. Pointing accuracy with regards to the distance of the user from the pointed target for three measurement approaches.

Results

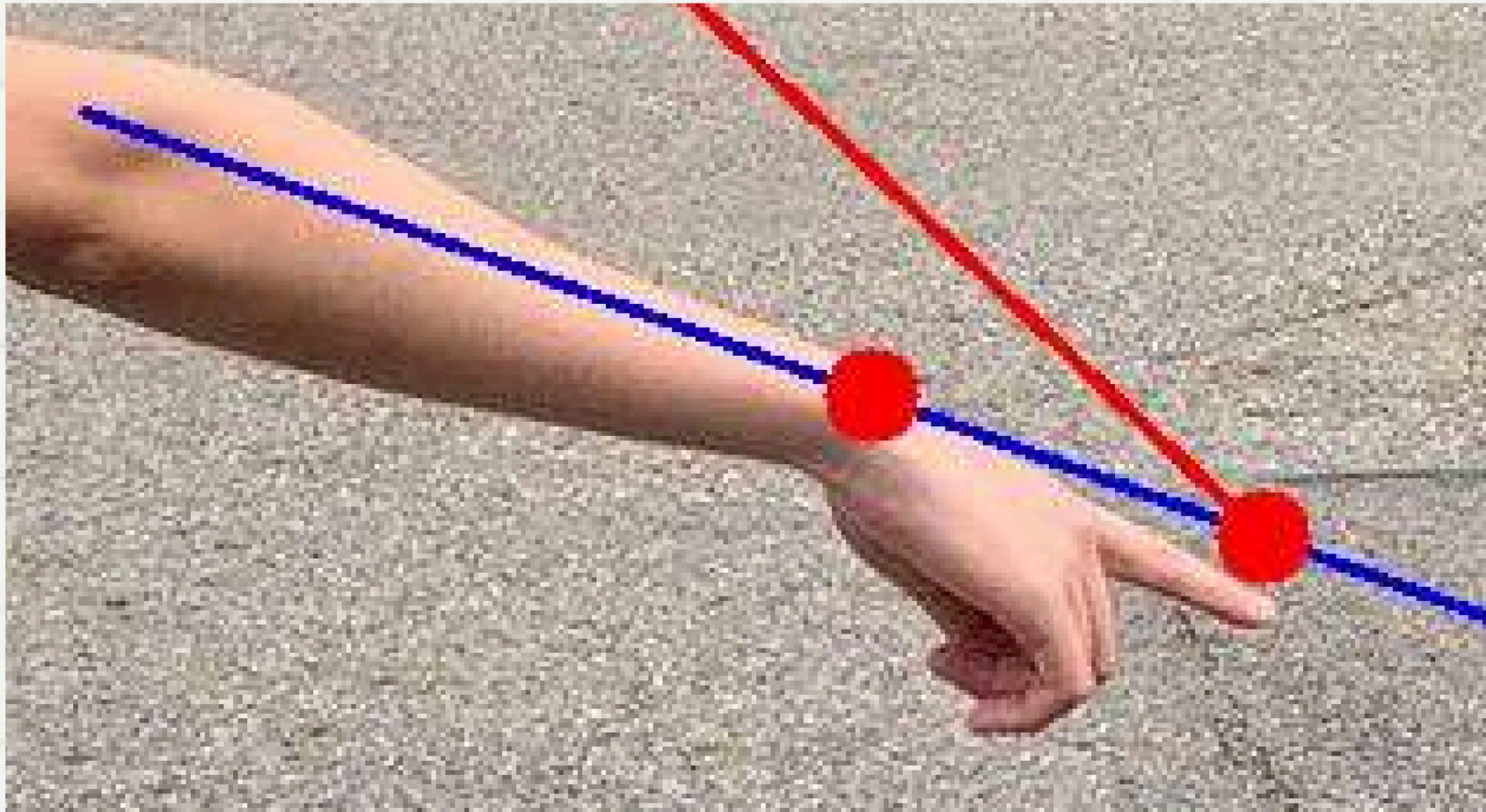
	Ours	Our Adaptive Approach v1	Our Adaptive Approach v2	Our Adaptive Approach v3	LLaVA-v1.5-13b-4-bit Released: 2023/10/5	LLaVA-NeXT-72b (LLaVA v1.6) Released: 2024/05/10	GPT-4o Released: 2024/05/13
Accuracy	18/80=22.5%	17/80=21.25%	20/80=25%	9/80 = 11.25%	53/80=66.25%	53/80=66.25%	64/80=80%

Link of the Results:

Analysis

受限於PoseC3D沒有手指的位置資訊

使用手腕+0.5*前臂向量作為估計，深度失真



Limitation of Our Work

1. 圖片中有多人時，無法確定是誰在指
2. 人物必須幾乎完整入鏡(PoseC3D的限制)
3. 20-30秒才能產生結果，實時性仍需改善
4. Accuracy較低，實際使用的效果仍需改善
5. 模塊化設計導致Error Propagation

Conclusion

What have we done?

我們實做了從RGB照片中以物理方法辨識人物指向的物體的方法，但由於這個task太難，準確率低於30%。考慮到前面提到的第一篇準確率也只有27%，我們的各種改進方法都低於30%

Conclusion

What have we observed?

圖像辨識種類不足是最主要的錯誤原因。我們認為這也是多模態大模型如此好的原因之一。三維交點不準確也很嚴重，需要更好的深度圖。而多模態大模型端到端的特性，也使其免於Error Propagation的問題。

Conclusion

Contribution

我們提出多種方法，做出詳細的觀察與分析。我們的work在人機互動與場景理解上很有用，同時我們利用模塊化設計，以後表現更好的各模型都可以提高正確率。

Our Github

- https://github.com/solocat17/PointAnything

Reference

- YOLO-World: https://github.com/AILab-CVC/YOLO-World
- YOLOv10: https://github.com/THU-MIG/yolov10
- Segment-anything: https://github.com/facebookresearch/segment-anything
- Depth-Anything: https://github.com/LiheYoung/Depth-Anything
- PoseC3D: https://github.com/kennymckormick/pyskl
- detectron2: https://github.com/facebookresearch/detectron2

Contribution of Each Member

- **(25%) 李宗諺(111550089):** 題目發想、程式實作
- **(25%) 黃仁駿(111550125):** 程式實作、資料集收集和標記、跑實驗、製作簡報及影片
- **(25%) 劉冠言(111705069):** 程式實作
- **(25%) 游惠晴(111550101):** 資料集收集和標記、跑實驗、製作簡報及影片

Thanks for listening !

Any Questions ?