

# Data deep-dive research

Seungwoo Lim

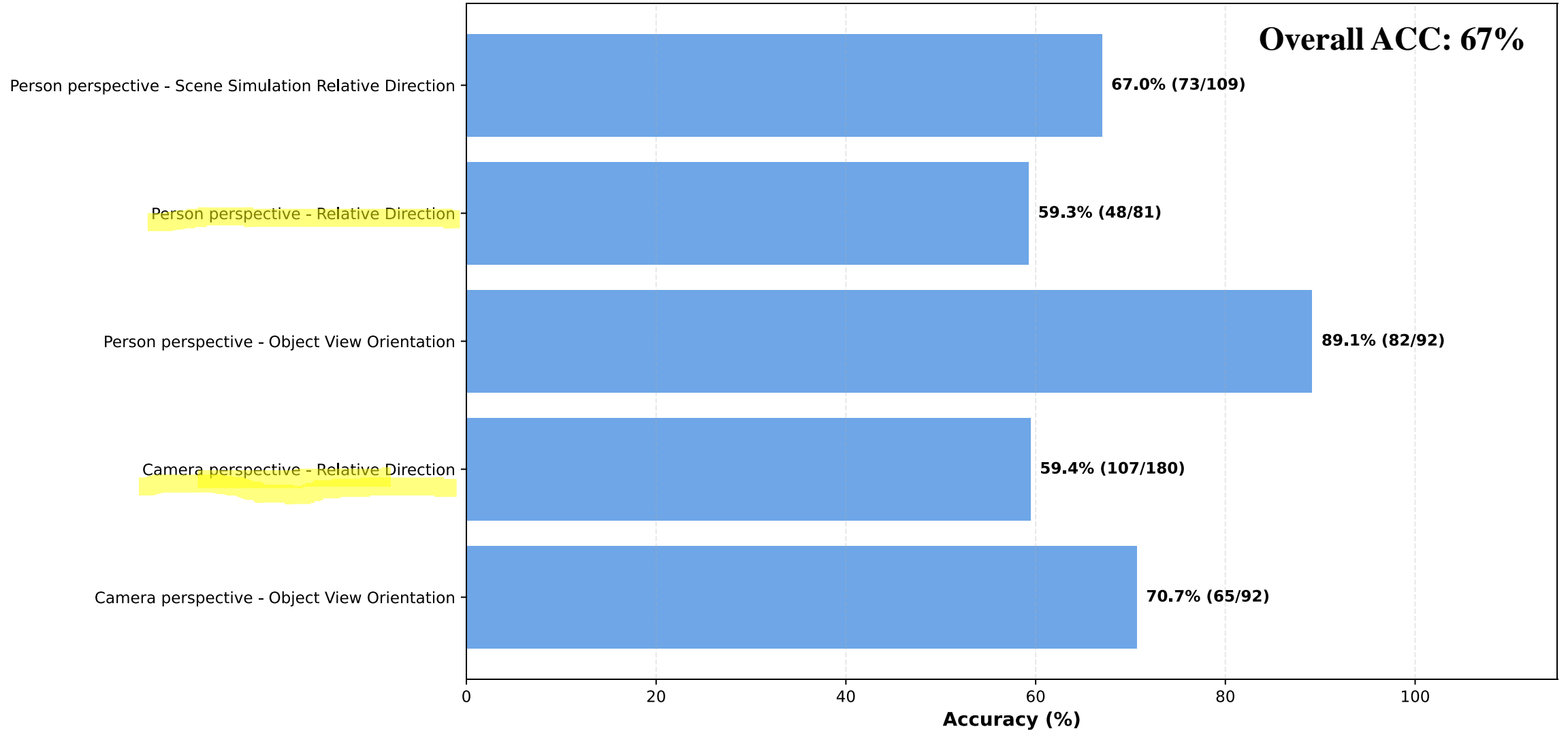
# Short recap

- Problems with the last research
  - Haven't looked into the data in detail.
  - A one-to-one comparison of a different model(MVSM) from the paper

# Introduction

- Is there a task that you get more wrong than others? Why?
- If you get VQA wrong, is there a tendency for the model to give incorrect answers?
  - Left -> Right? Or is it completely wrong?
- Baseline : Vanilla MVSM (fine tuned Qwen2.5)

### Task-wise Accuracy (Baseline)



# Hypothesis 1

Data Augmentation

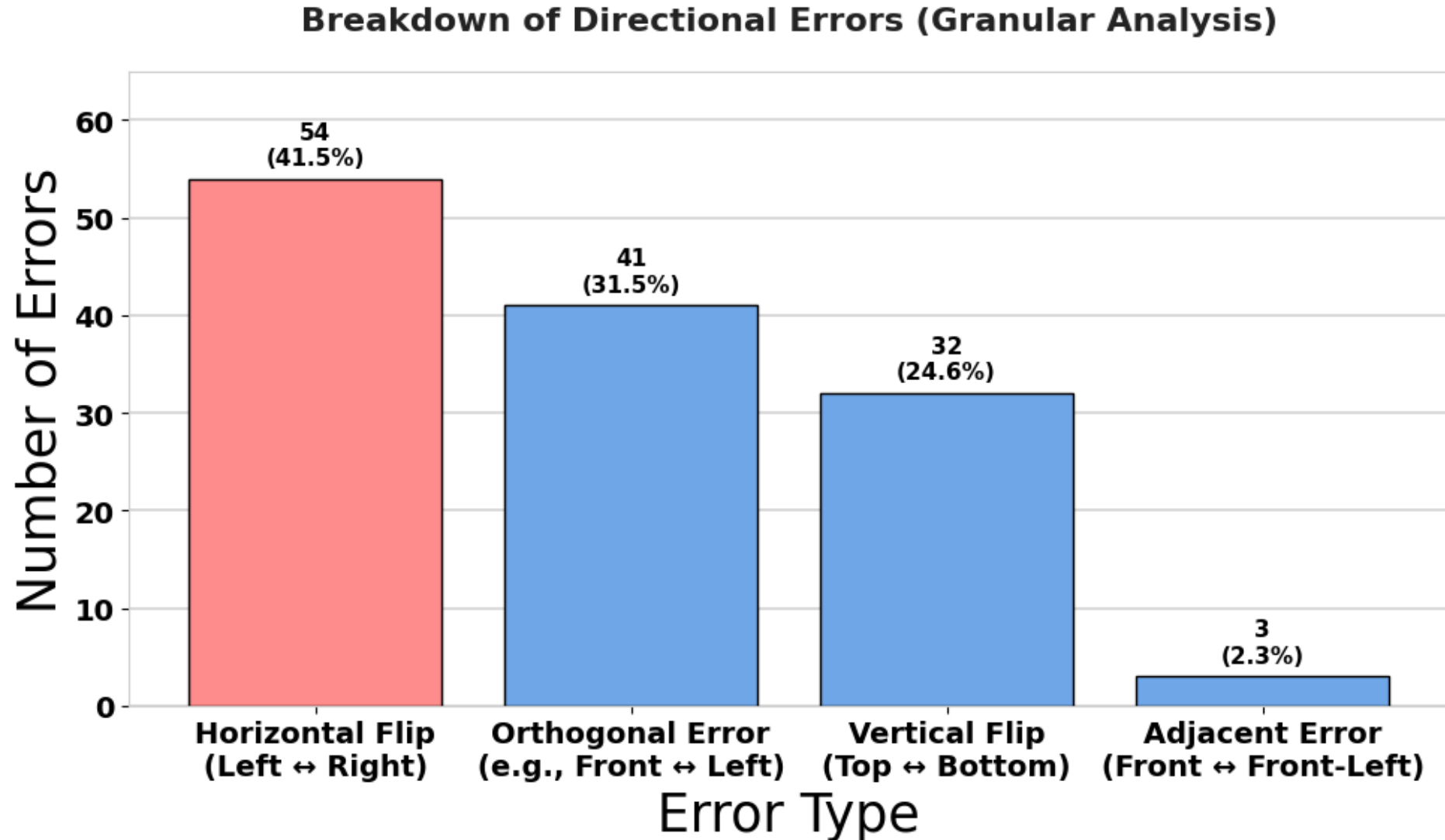
# Hypothesis 1 : Data augmentation

- As demonstrated in Vision-Language research, focusing on **learning hard negatives** that confuse the model is key to improving performance, rather than simply augmenting data.
- Hard negative :
  - The most common incorrect answer the model gets wrong
  - image-text pair that is semantically similar but differs in fine-grained details.

# Hypothesis 1 : Data augmentation

- **Horizontal Errors:**
  - (Front/Back)Left ↔ (Front/Back)Right
- **Orthogonal Errors:**
  - Front ↔ Left/Right
- **Vertical Flip:**
  - Top ↔ Bottom
- **Adjacent Error:**
  - Angles that are not 90 or 180 degrees
  - Front ↔ Front-left

# Problem 1 : Direction analysis

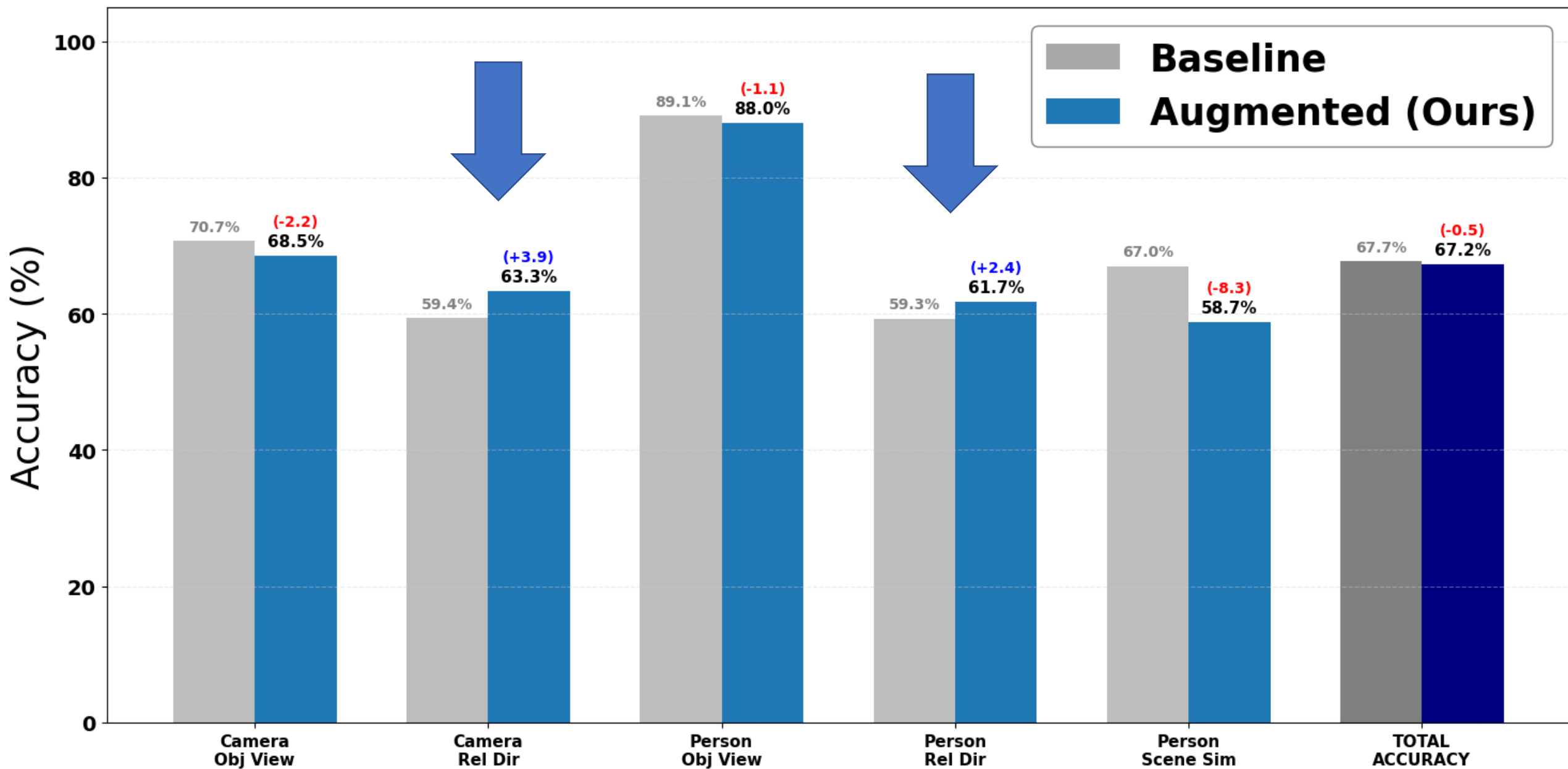




# Hypothesis 1

- Augment Train dataset
  - 4548(baseline train data) -> 9043 (augmented train data)
  - Flip the image, flip the label( left , right)

# Benchmark Performance Comparison (Baseline vs Augmented)



# Result 1 : Data Augmentation

- Although we achieved accuracy improvement in the targeted relative direction,
- We confirmed that the performance of other tasks actually decreased.
  - Which is called **Catastrophic Forgetting**\*
  - -> To solve this, we can try “cognitive replay”
  - It is a method of repeatedly learning previous data.

# Hypothesis 2

Bounding box + Visual prompting

# Problem 2 : Object detection fail



☐ **WRONG**

**[Question]**

From the perspective of the boy, where is the TV located?

**[Options]**

A. back-left **B. front** C. right D. left

## Problem 2 : Object detection fail

- The question is about “person perspective” but the model is likely looking at "the entire image"

# Hypothesis 2 : Bounding box + Visual prompting

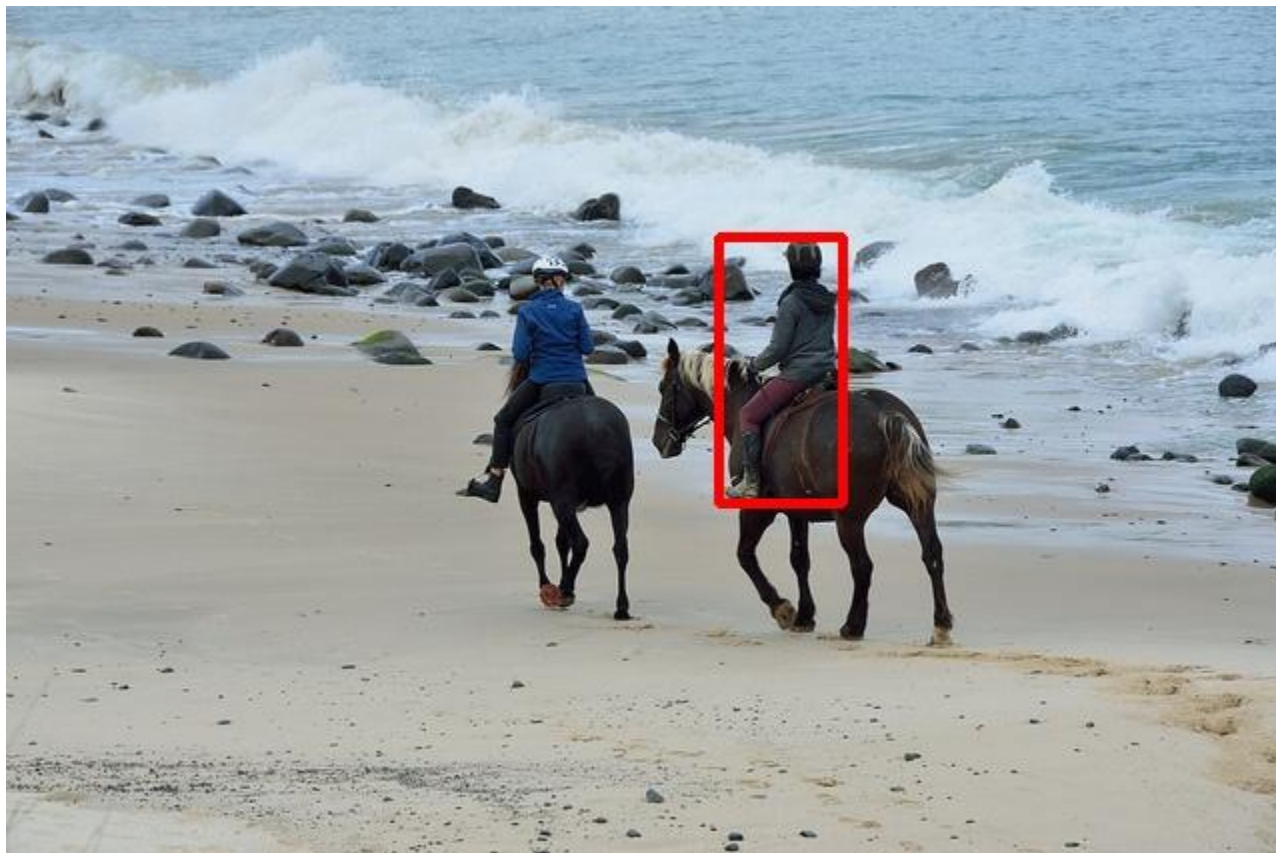
- The model may not be good at recognizing **reference objects**.
  - “From the perspective of **the boy**, where is the TV located?”
- -> Then let's make the reference point recognition more certain!
  - Maybe Bounding box might be helpful!
  - Used “Ultralytics YOLO-World\*”

# How does “Yolo world” works?

- Existing YOLO (v5, v8, etc.) could only find fixed, pre-trained classes (e.g. 80 classes including people, cars, dogs, cats, etc.).
- 1. Text Encoder : Convert user-entered text (e.g., "cat") into vector that the computer can understand. This process utilizes CLIP, a large-scale language-image model, to extract the meaning of the text.
- 2. Image Encoder : Analyzes the input image to extract visual features
- 3. Fusion : If a specific area of the image has a high similarity to the text vector, we determine that it is the object the user is looking for and draw a bounding box there.



# Example



*MS-CoCo Example*

From the perspective of the person wearing the grey clothes, where is the person wearing the blue clothes?



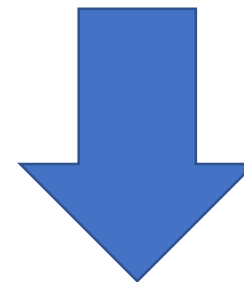
From the perspective of the person **in the red bounding box** wearing the grey clothes, where is the person wearing the blue clothes?

# Example



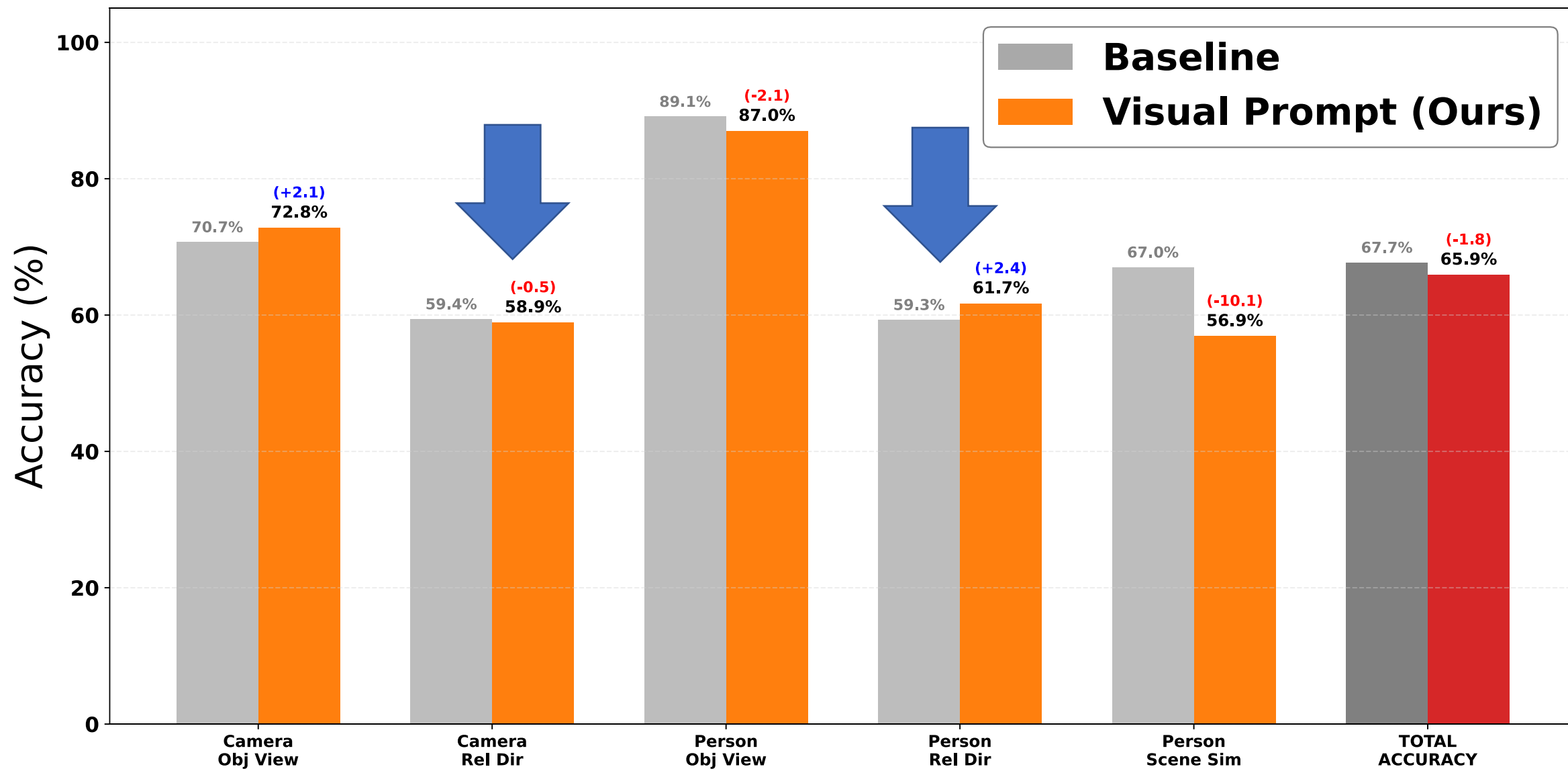
*ScanNet Example*

How is the sofa positioned with respect to the table?



How is the sofa positioned with respect to the table in the **red bounding box**?

## Performance Comparison: Baseline vs Visual Prompting



# Result 2 : Bounding Box

- Result :
  - [Per] Rel dir (+2.4%), [Camera] Obj ori(+2.1%): Performance improved in problems where the anchor point must be clear.
- Limitations:
  - [Per] Scene simulation (−10.1%) : The surrounding environment (walls, obstacles, spatial structure) must be grasped as a whole, but this is interpreted as a kind of **tunnel vision** phenomenon where the model's gaze is trapped within a red box.

# Do bounding boxes induce tunnel vision?

$$S_i = \left| \frac{\partial f(x)}{\partial x_i} \right|$$

$f(x)$  : Logit of prediction  
 $x_i$  : pixel value

- Used Saliency map
  - A saliency map is a way to visualize how sensitive each element of the input is to the prediction by calculating the gradient of the input  $x$  with respect to the model output  $y$ .
  - “If this pixel value changes even a little, how much does the correct answer (Logits) change?”
- Why not attention map?
  - It shows where the model "routed" during its computation, doesn't guarantee that it influenced the correct answer.



**[Rel Dir] From the perspective of the man who is looking at the computer, where is the man wearing a hat positioned?**



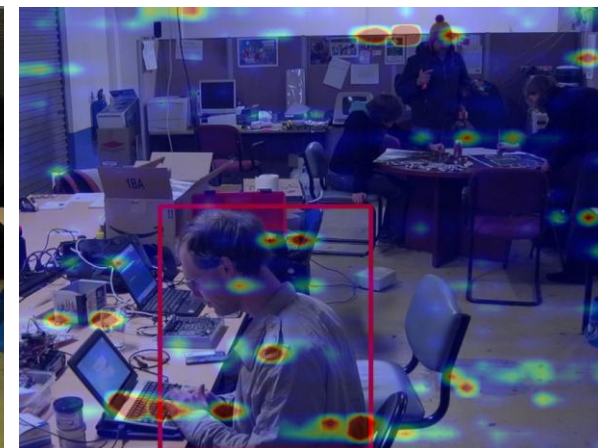
**Original Image**



**Original Saliency**

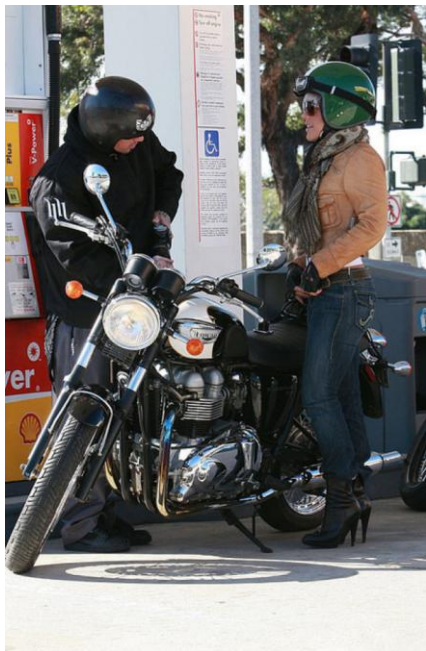


**Visual Prompt Image**



**Boxed Saliency**

**[Rel Dir] From the perspective of the woman wearing the green helmet, where is the person in black clothes?**



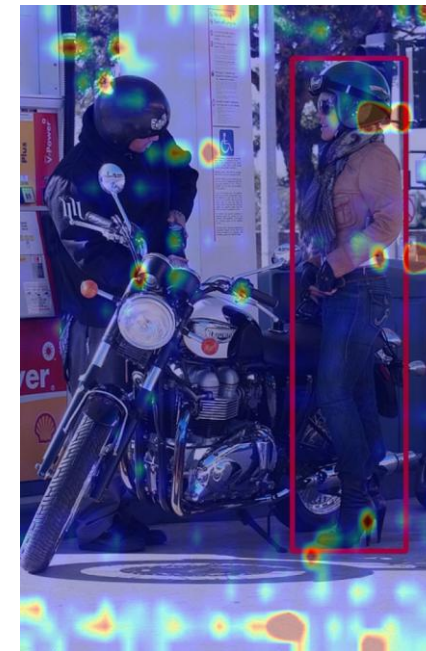
**Original Image**



**Original Saliency**



**Visual Prompt Image**



**Boxed Saliency**

# Hypothesis 3

Chain of thought



# Hypothesis 3 : Chain of Thought



Q : With the camera's viewpoint as the front, which direction is the elephant facing in the image? A. right B. front C. back-left D. left  
Answer with the option letter.

“A”



Let's think step by step to determine the spatial relationship.

1. First, I identify the reference point: the camera viewpoint.
2. Next, I locate the target object: man.
3. By analyzing their relative positions in the 3D space, man is positioned to the right of the camera viewpoint.

Therefore, the correct option is **A**.

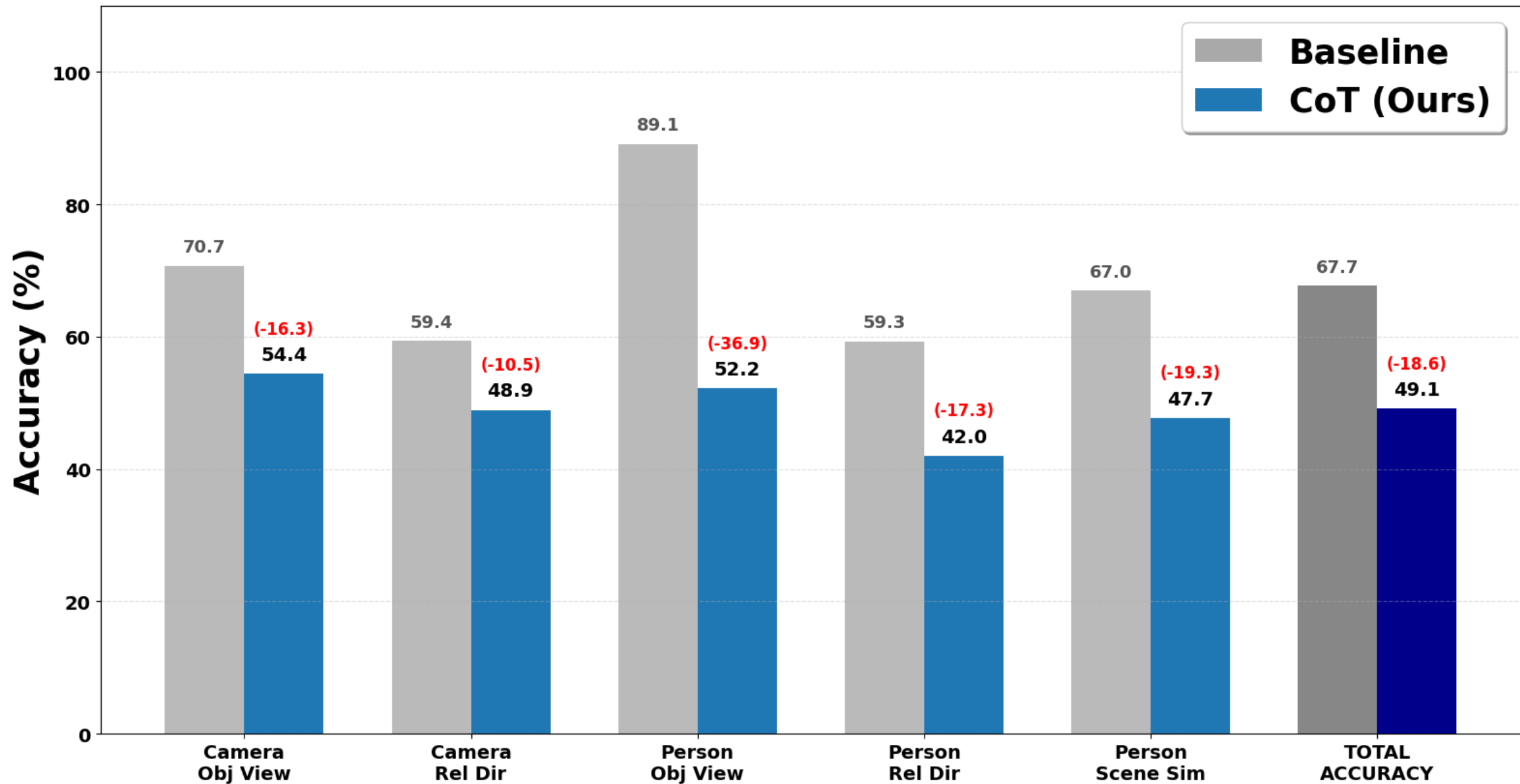
# How can we teach chain of thought?

- Use Distilling the knowledge

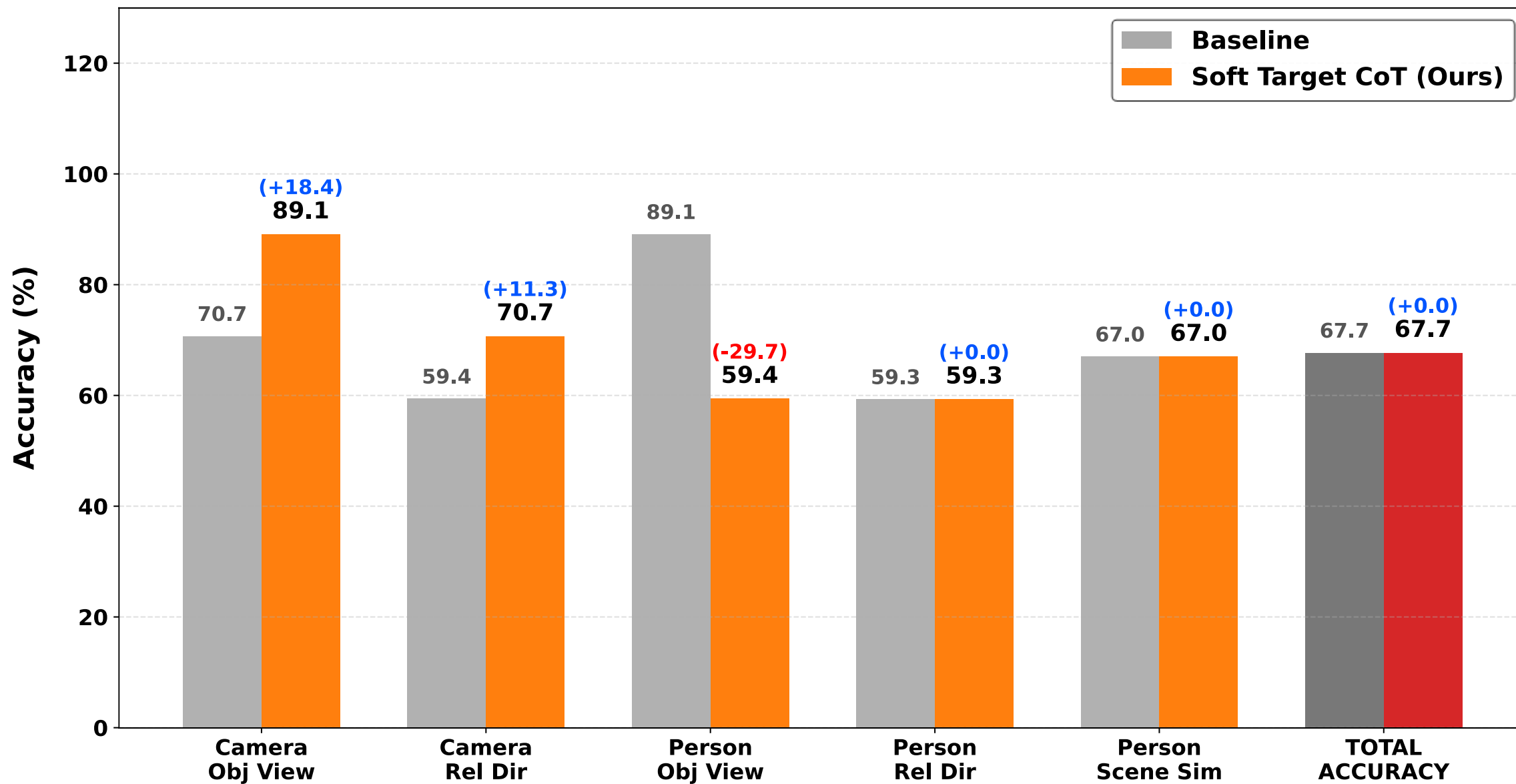
# Distilling the knowledge

- Used Qwen 2.5 **7B** model for **teacher** , and our Qwen 2.5 3B model will be a student
  - Qwen2.5 73B model was unusable due to OOM issue.
- Hard target : train with one-hot vector
  - Ex) [0,0,1]
- Soft target : train with prob. Distribution(after softmax)
  - Ex) [0.1, 0.2, 0.7]

## Benchmark Performance: Baseline vs CoT(w Hard target)



# Performance Comparison: Baseline vs Soft Target CoT



# Result 3 : CoT

- Why did the results turn out this way?
  - Inductive Bias Transfer in the Teacher Model (7B)
- Limitations : Used the Qwen 2.5 **7B** model instead of the 72B model or other higher end models.
  - How can we improve performance even further?
    - Better open source model (InternVL2-26B, Llama-3.2-11B-Vision, Qwen3)

# Conclusion

- We confirmed that **data augmentation(for hard negatives)** and **bounding boxes**, and **CoT** helped improve performance

# Thank you!

[daniel73919@gmail.com](mailto:daniel73919@gmail.com)