ViewSpatial-Bench(2026)

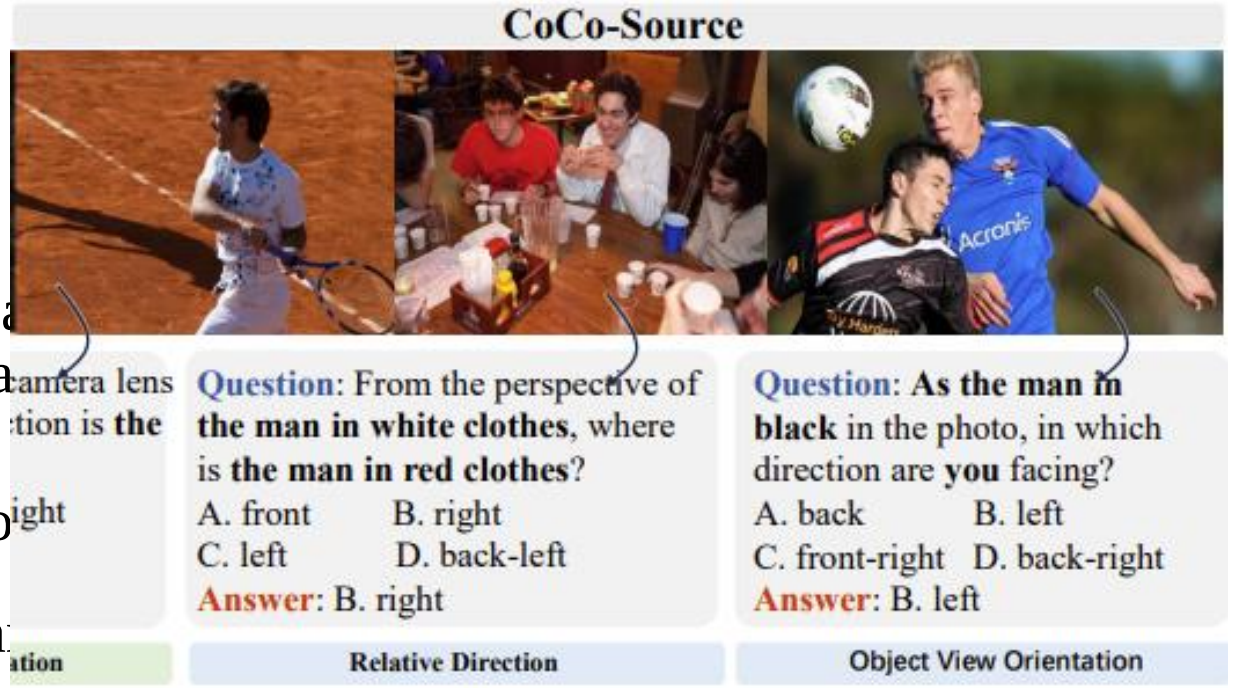
Seungwoo Lim

Index

- Part 1 : Paper review
  - Motivation
  - Contribution
  - **Methodology**
  - Experiments
  - Limitations
- Part 2 : Experiment

Motivation

- [SpatialVLM 2024]
  - VLM lacks spatial reasoning due to the sca
  - Present a new model capable of spatial rea
- [COMFORT 2025]
  - Spatial expressions are ambiguous and mo conventions
  - Present a consistent multilingual benchma



- Still, VLM understands only camera-perspective and lacks allocentric spatial reasoning
  - It was obvious -> there was no training data about it.
  - “popular vision-language corpora contain little reliable data for learning spatial relationships...”*

Contribution

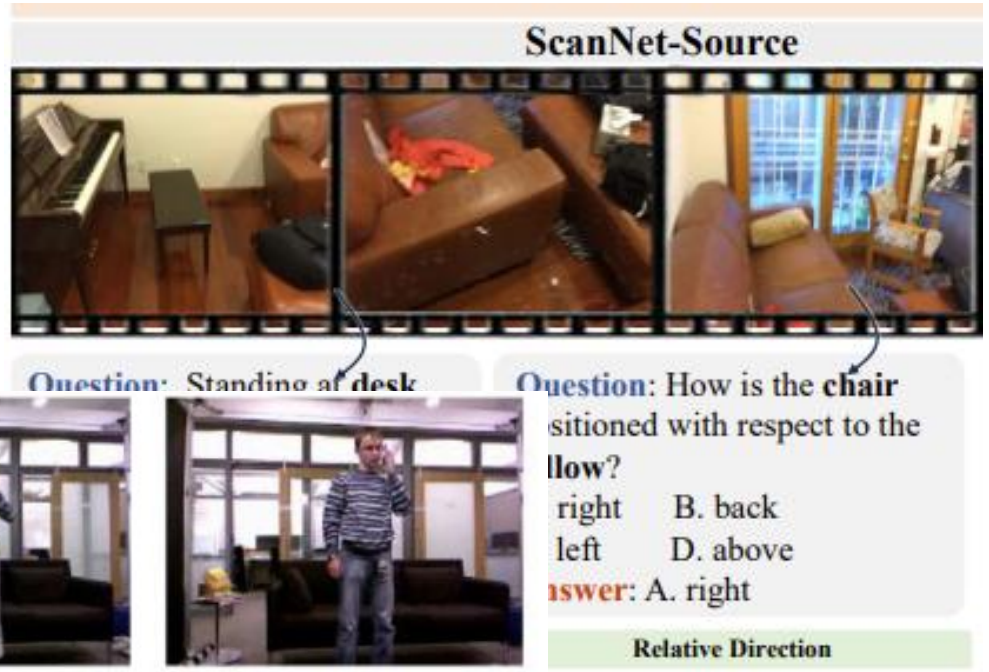
- Present a new model trained by adding data that change perspectives
- Present a new evaluation metric

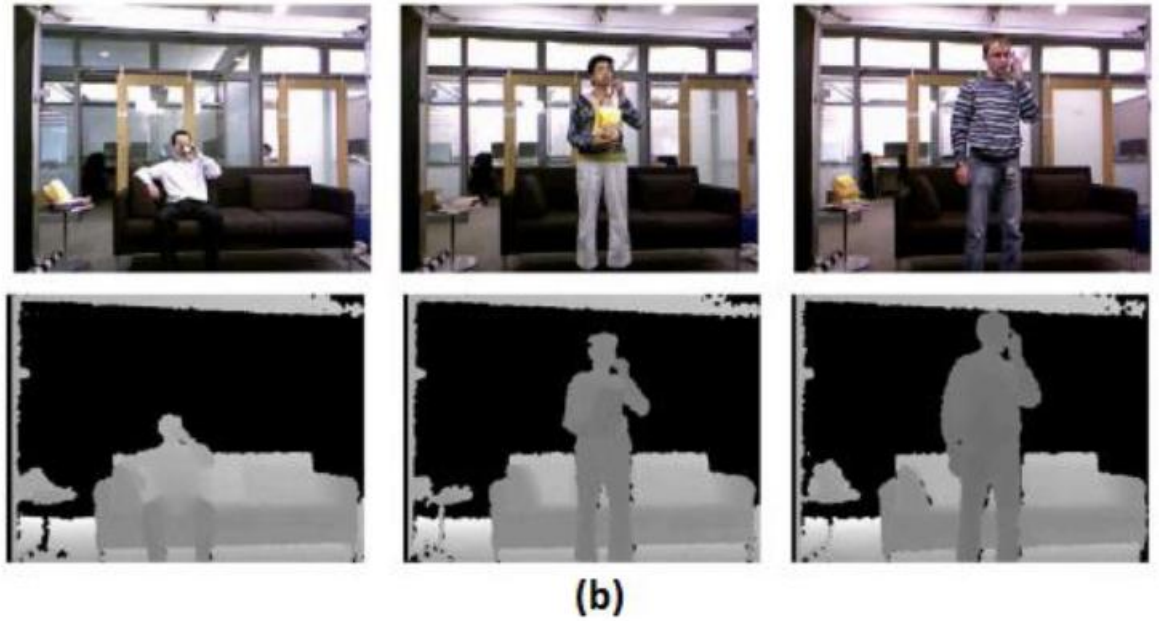
Methodology

- (a) Semantic Filtering (pre – processing)
- (b) Train the model
- (c) New evaluation metric

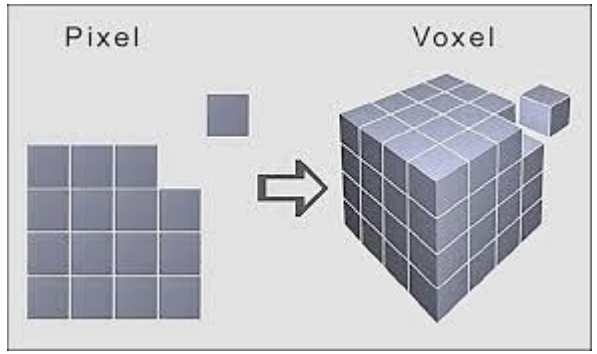
Pre-process the data

- Image source
- 1. ScanNet (for
  - Data taken ind
  - Validation set :
  - Train set : trai
  - Use Maximum
    - To find optin
    - With the most information
    - No overlap





Pre-process the data



Algorithm 1 Maximum Coverage Sampling

Require: Set of frames $F = \{f_1, f_2, \ldots, f_n\}$, voxel sets V_k for each frame f_k, budget K
Ensure: Subset $S \subseteq F$ maximizing voxel coverage

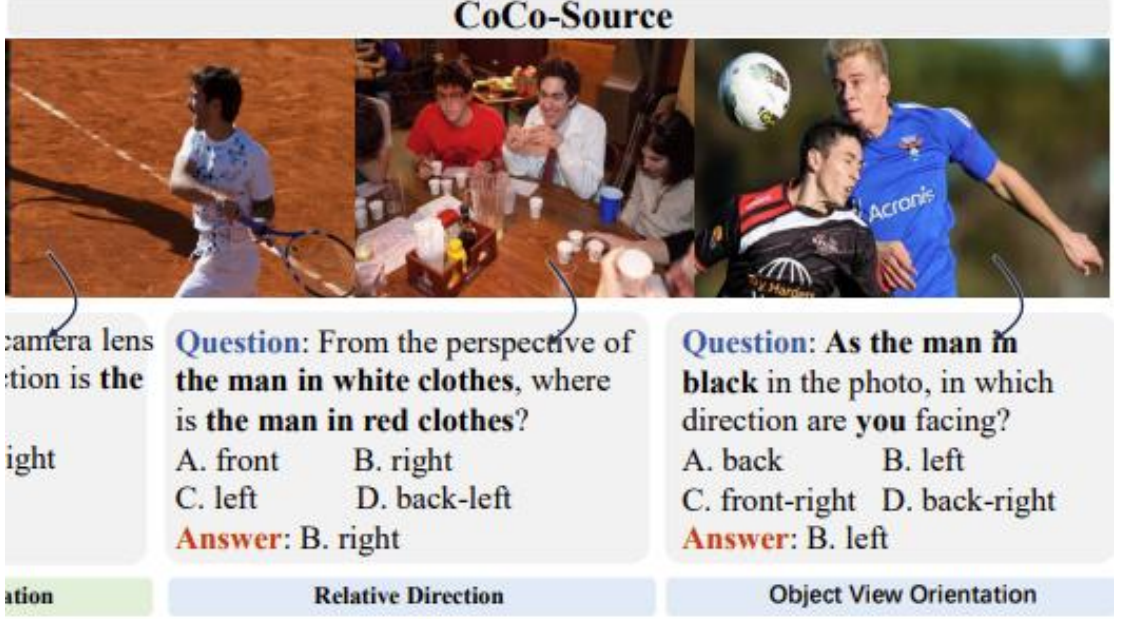
Initialize $S \leftarrow \emptyset$
Initialize $U \leftarrow \emptyset$ {Covered voxels set}
while size of S is less than K **do**
  Select $f^* = \arg\max_{f_k \in F \setminus S} |V_k \setminus U|$
  Add f^* to S
  Update $U \leftarrow U \cup V_{f^*}$
  if Stop condition is met **then**
    break
  end if
end while
return S

- 1. Find f_k that maximizes V_k and put it into U.
- 2. while(choose one of the f_k that is not in U.)

- It's basically a greedy algorithm.
- In Maximum coverage problem, it always guarantees optimal solution results at least $63 \approx 1 - \frac{1}{e}$ % of the time*.
- O(N) : checks N, N-1 … K times

An analysis of approximations for maximizing submodular set functions , 1978, George L. Nemhauser et al.

Pre-process the data



- 2. MS-CoCo (for human perspective)
  - Human subjects and annotated keypoints
  - Only use a person that is more than 20%

  - [Orientation task]
    - Use AI called “Orient Anything”

  - [Relative direction task]
    - “From person A’s perspective, where is person B located?”
    - Manually labelled 864 of instances.
    - due to the complexity, insufficient accuracy in automated approaches

What is “Orient Anything”?

- Extract one 3D object from the Objaverse
- Create 5 orthogonal images taken from the front, back, left, right, and top.
- Ask VLM “Which of these is front”?

- Collect 2 million pieces of this data to train it.

Pre-process the data

Algorithm 2 Head-to-body Orientation Offset

Require: Image I, keypoints K, bounding box B, Orient-Anything model D
Ensure: Person gaze direction

$P \leftarrow \text{Crop}(I, B)$
$(L_x, L_y), (R_x, R_y) \leftarrow \text{ExtractShoulders}(K)$
if $\text{Visibility}(L_y) = 0$ OR $\text{Visibility}(R_y) = 0$ **then**
  return False
end if
$H \leftarrow \min(L_y, R_y)$
$P_{head} \leftarrow P[0 : H, :], P_{body} \leftarrow P[H :, :]$
$(az_{head}, conf_{head}) \leftarrow D(P_{head})$
$(az_{body}, conf_{body}) \leftarrow D(P_{body})$
$\Delta \leftarrow (az_{head} - az_{body} + 540) \bmod 360 - 180$
return direction based on Δ thresholds for left, front-left, front, front-right, right

- 1. Extract person by Bounding Box
- 2. Separate head and body based on shoulder coordinates
- 3. Calculate azimuthal head_angle, body_angle using “Orient Anything”
- 4. Calculate delta
- 5. Classify the delta based on the direction’s range
  - “Front-Up” when the horizontal angle is near 0 degrees and the vertical angle is near 45 degrees

Orient Anything: ~ , Zehan Wang et al , arXiv:2412.18605 , 2024

Generate QA data

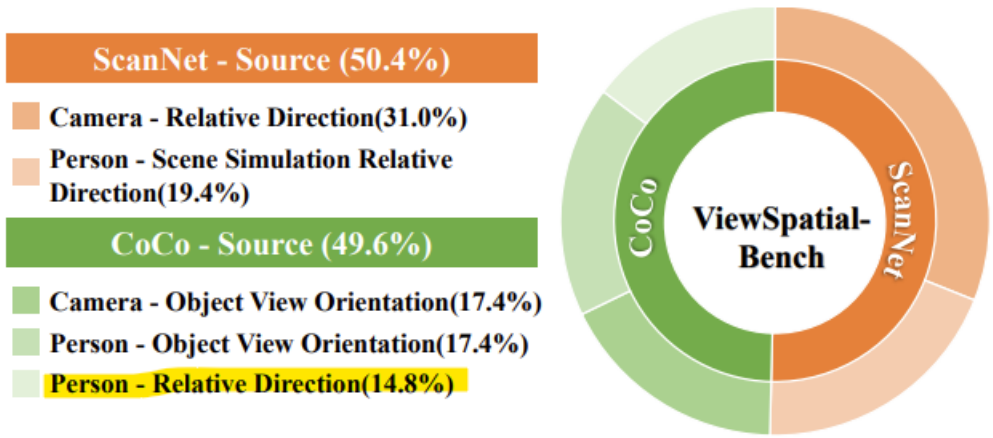
- Now we have a…
  - [ScanNet] Useful Frame and Object direction (x,y,z)
  - [MS-CoCo] Human gaze angle

- Based on these data, we can generate QA data
  - Maps (x,y,z) and angles to one of 18 directions.
  - Create based on the template.
  - Wrong answer have to be completely wrong
    - Front vs Front-up

- Filtering and Human Verification
  - Too overlapped, ambiguous classification(Bottle? Cup?) ,

Task	Question Template
Cam-Rel. Dir.	• Can you describe the position of the {object1} relative to the {object2}? • Could you tell me the location of the {object1} in comparison to the {object2}? • Where is the {object1} in relation to the {object2}? • Where is the {object1} located compared to the {object2} from the camera's perspective? • How is the {object1} positioned with respect to the {object2}? • If you're looking at the {object2}, where would you find the {object1}?
Cam-Obj. Dir.	• With the camera's viewpoint as the front, which direction is {object} facing in the image? • Taking the camera lens as the front, what direction is {object} looking toward? • Taking the camera's viewpoint as the front, which way is {object} facing in the image? • Considering the camera's perspective as the front, what direction is {object} facing within the picture?
Per-Obj. Dir.	• Imagine you're {object} in this image — which direction are you facing? • Suppose you are in {object}'s position, what direction are you facing? • Picture yourself as {object}; which way are you looking in the scene? • As {object} in the photo, in which direction are you facing?
Per-Sce. Sim.	• Imagine standing at {object1} looking towards {object2}, where is {object3}? • When positioned at {object1} facing {object2}, where can you find {object3}? • If you stand at {object1} facing {object2}, where is {object3}? • Standing at {object1}, gazing at {object2}, where should {object3} be?

Dataset



- Overall, evenly distributed
- Train dataset: 42000
  - Due to the scarcity of Person-perspective Relative direction, supplemented with Spatial-MM
- Test dataset : 3700

Experiments

- Use Qwen2.5-VL-3B as a backbone
  - Small size (3B)
  - Reimplementation (Open source model)

- Compare with other VLM

Experiments

Model	Camera-based Tasks			Person-based Tasks				Overall
	Rel. Dir.	Obj. Ori.	Avg.	Obj. Ori.	Rel. Dir.	Sce. Sim.	Avg.	
InternVL2.5 (2B) [30]	38.52	22.59	32.79	47.09	40.02	25.70	37.04	34.98
Qwen2.5-VL (7B) [36]	46.64	29.72	40.56	37.05	35.04	28.78	33.37	36.85
LLaVA-NeXT-Video (7B) [32]	26.34	19.28	23.80	44.68	38.60	29.05	37.07	30.64
LLaVA-OneVision (7B) [33]	29.84	26.10	28.49	22.39	31.00	26.88	26.54	27.49
InternVL2.5 (8B) [30]	49.41	**41.27**	46.48	46.79	42.04	**32.85**	40.20	**43.24**
Llama-3.2-Vision (11B) [34]	25.27	20.98	23.73	51.20	32.19	18.82	33.61	28.82
InternVL3 (14B) [31]	**54.65**	33.63	**47.09**	33.43	37.05	31.86	33.88	40.28
Kimi-VL-Instruct (16B) [35]	26.85	22.09	25.14	**63.05**	**43.94**	20.27	**41.52**	33.58
GPT-4o[37]	41.46	19.58	33.57	42.97	40.86	26.79	36.29	34.98
Gemini 2.0 Flash [38]	45.29	12.95	33.66	41.16	32.78	21.90	31.53	32.56
Qwen2.5-VL (3B) [36] [Backbone]	43.43	33.33	39.80	39.16	28.62	28.51	32.14	35.85
Multi-View Spatial Model	**83.59**	**87.65**	**85.05**	**90.16**	**71.14**	**75.75**	**79.31**	**82.09**
Improvement over backbone	+40.16	+54.32	+45.25	+51.00	+42.52	+47.24	+47.17	+46.24
Random Baseline	25.16	26.10	25.50	24.60	31.12	26.33	27.12	26.33

Table 2: Zero-shot performance on ViewSpatial-Bench. Accuracy comparison across multiple VLMs on camera and human perspective spatial tasks. Our Multi-View Spatial Model (MVSM) significantly outperforms all baseline models across all task categories, demonstrating the effectiveness of our multi-perspective spatial fine-tuning approach.

Limitations

- Annotation Challenges for Human-Perspective Tasks.
  - Labeling by humans directly can be biased and contaminated.
- Domain Constraints in Environmental Coverage.
  - Camera - Relative Direction task is biased in ScanNet(indoor data).
- Static vs. Dynamic Spatial Reasoning.
  - ViewSpatial-Bench evaluates only static spatial orientation comprehension
  - Without addressing dynamic spatial reasoning scenarios
  - “Static” : Least frame sampled by Maximum coverage algorithm
  - “Dynamic” : Much more frame

But ScanNet data, isn’t it dynamic?

- **Discrete Sampling vs. Continuous Stream:** The benchmark extracts discrete, sampled frames to reconstruct static 3D scene layouts, intentionally discarding the temporal continuity of the original video feed.

Part 2

Dynamic Spatial Reasoning

Introduction

- Dataset : EpicKitchens
  - P01_01 (27m 30s) : 59.94 fps, **99029 frame,**
  - Egocentric

- 1. Compare with baseline model (pre-trained Qwen 2.5)
- 2. New methodology
- 3. Experiments

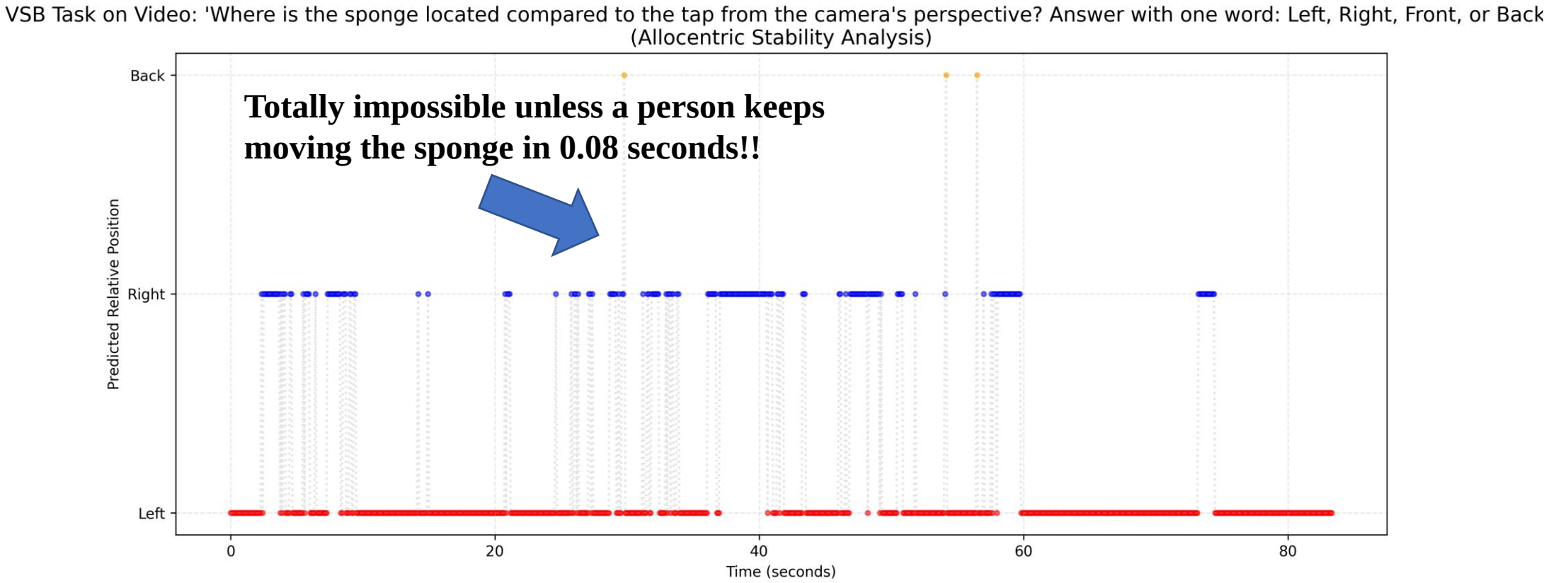

insert coffee holder
stir chicken
close tap
move tap
wipe down counter
stretch dough
open bin

Motivation

Prompt : “Where is the sink? Answer with one word: left, right, center”
Time interval : 0.05s

Motivation(VSB task on video)

Stride : 5 (0.08s)
Time Duration : first 5000 frame
Prompt : Allocentric.
Model : Qwen2.5



Algorithm 1

- “How to find a most dynamic frame?”
- **Mean Absolute Difference (MAD) : O(N), N is pixel #**

$$S(t) = \frac{1}{H \times W} \sum_{y=1}^{H} \sum_{x=1}^{W} \left| I(x, y, t) - I\left(x, y, t - \Delta t\right) \right|$$

- $S(t)$: Motion score at frame t
- $I(x,y,t)$: Pixel intensity value at (x,y,t), uses L1 distance for diff
- $H,\ W$: Pixel’s height and width for normalization
  - Without this, high resolution image will get high value

Automatic partitioning of full-motion video , Zhang et al, 1993

Algorithm 1

- Why L1 distance? or Why MAD?
  - Computational Efficiency : have to investigate over 0.1M frame for one video

- How about another algorithm?
  - Optical flow
  - RAFT : In two consecutive images, it shows where each pixel moved.
    - Uses RNN-GRU
  - The goal is not to perfectly know the changes in every pixel, but to find the most dynamic frame.
  - And these are too time-consuming job comare to the MAD method.

Algorithm 2

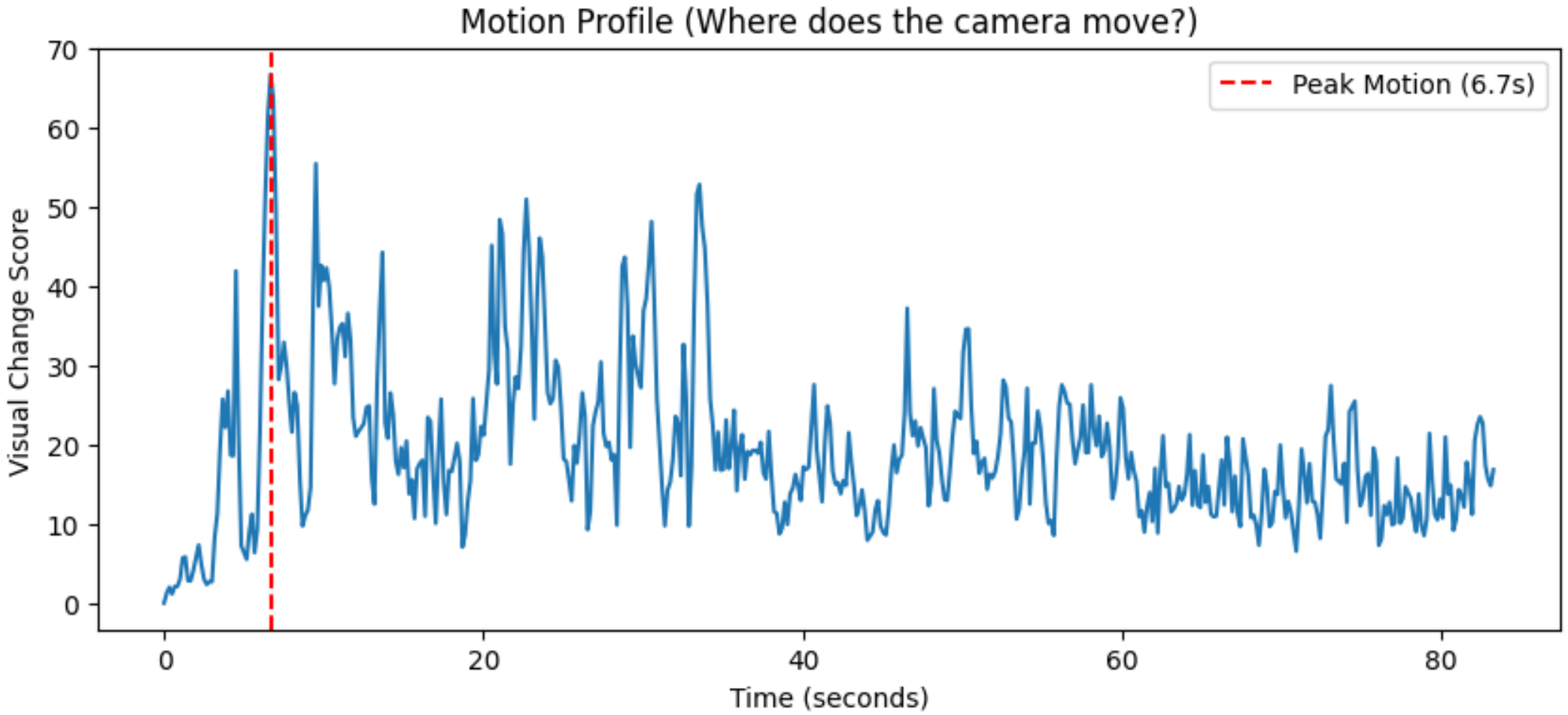
$$f_{sampling} > 2 \times f_{max}$$

- Sampling period
  - Based on Nyquist-Shannon Sampling Theorem:
    - Human Saccade's period is 0.2s = 5hz*
    - So, Sampling frequency have to be bigger than 10hz = 0.1s
  - However we don't need to perfectly restore the signal,
  - [GoPro is attached at head not eye] Due to the inertia of head, the rate of change is much slower than eye (Newton's 1st law)
  - So, I set the stride as 10 = search frame by 0.16s of period

Leigh, R. John, and David S. Zee. *The Neurology of Eye Movements*. 5th ed. Contemporary Neurology Series. New York: Oxford University Press, 2015.

Algorithm

- My algorithm contains pre-processing before TMAD
  - 1. Convert into Gray-scale
    - We don’t care about color change just the dynamic change
  - 2. Down sampling
    - 1920x1080 -> 160x120
  - 3. TMAD
    - Find the most dynamic scene


Motion Profile (Where does the camera move?)
Peak Motion (6.7s)
Visual Change Score
Time (seconds)
0
10
20
30
40
50
60
70
0
20
40
60
80

Experiments

- Frame by Frame
  - Stateless : doesn’t understand the context in t-1 at t frame**(1st assumption)**

- Prompt : “Where is the sink? Answer with one word: left, right, center”
  - Ambiguity : Left, Mostly left, left side…
  - The sink is usually in the center, so I think it can be easily moved from left to center with just a little movement.

- Model : Qwen 2.5 (Backbone model in ViewSpatialBench)

Algorithm 3

$$P_{smooth}^{(t)} = \alpha \cdot P_{smooth}^{(t-1)} + (1-\alpha) \cdot P_{raw}^{(t)}$$

- EMA(Exponential Moving Average):
  - Idea : “How much should we trust our memories of the past?”
  - Alpha is the value we need to find the optimal value for.
  - Inspired by **Markov model**

$$p(x_t|x_{t-1}, x_{t-2}, \dots, x_1) = \boldsymbol{p(x_t|x_{t-1})}$$

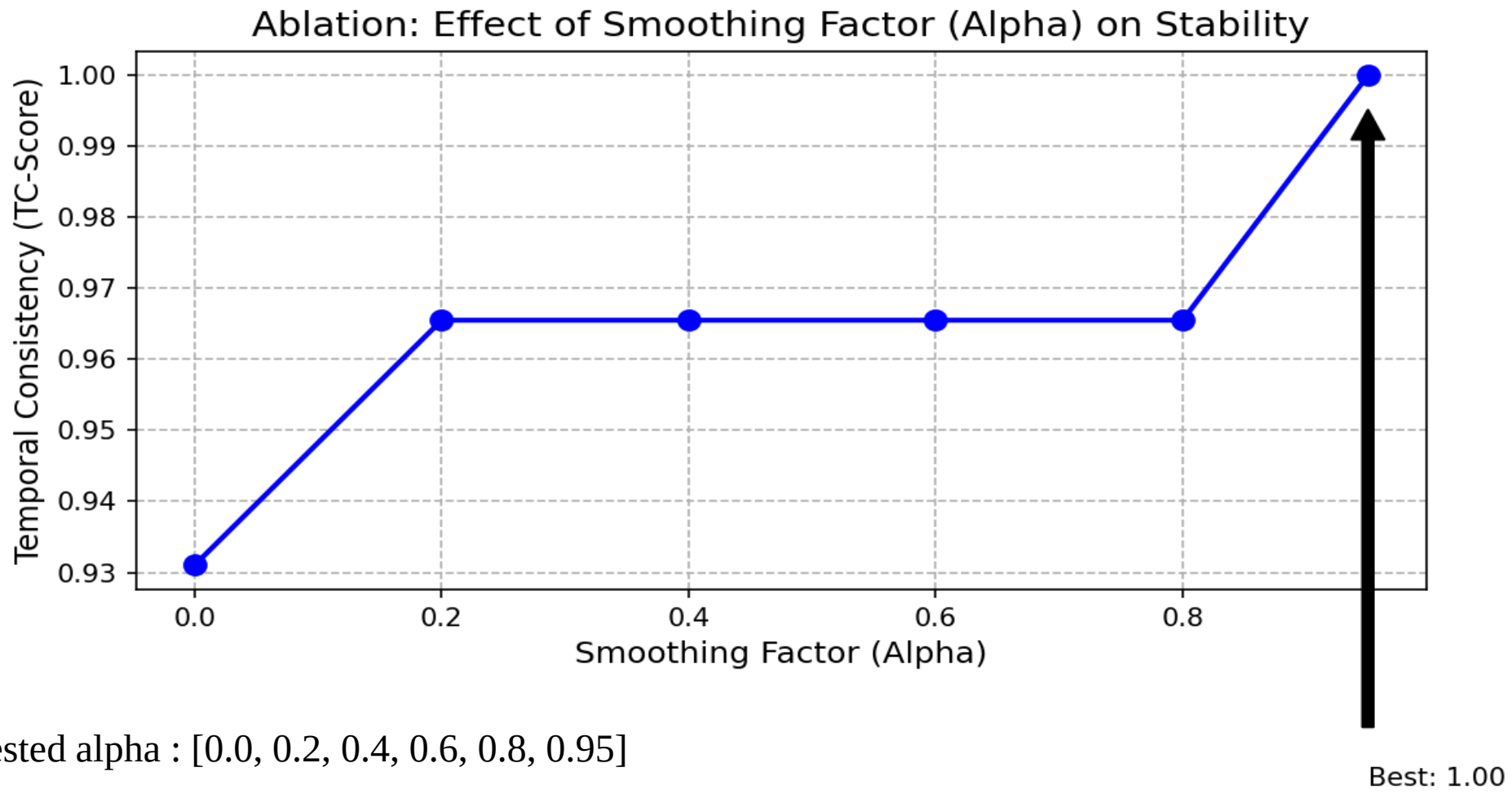
Metric

$$TC = \frac{1}{T-1}\sum_{t=1}^{T-1} I(y_t = y_{t+1})$$

- TC(Tyler-Cuzick) score*:
  - Indicator function is in sigma, it returns 1 if the prediction matches, otherwise 0
  - $\frac{1}{T-1}$ for normalize
  - So basically, this experiment is about consistency

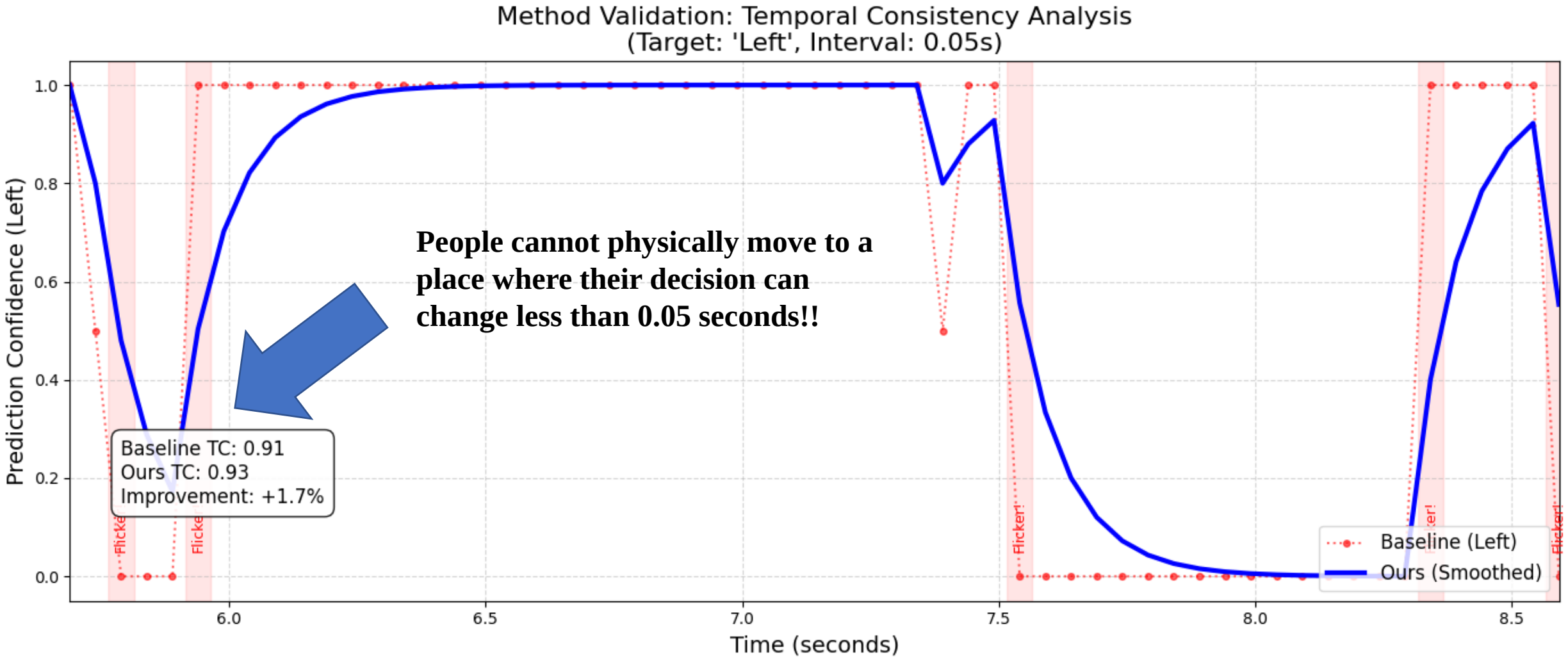
Yifan Liu et al., "Efficient Semantic Video Segmentation with Per-Frame Inference," *ECCV 2020*

Ablation study



Tested alpha : [0.0, 0.2, 0.4, 0.6, 0.8, 0.95]

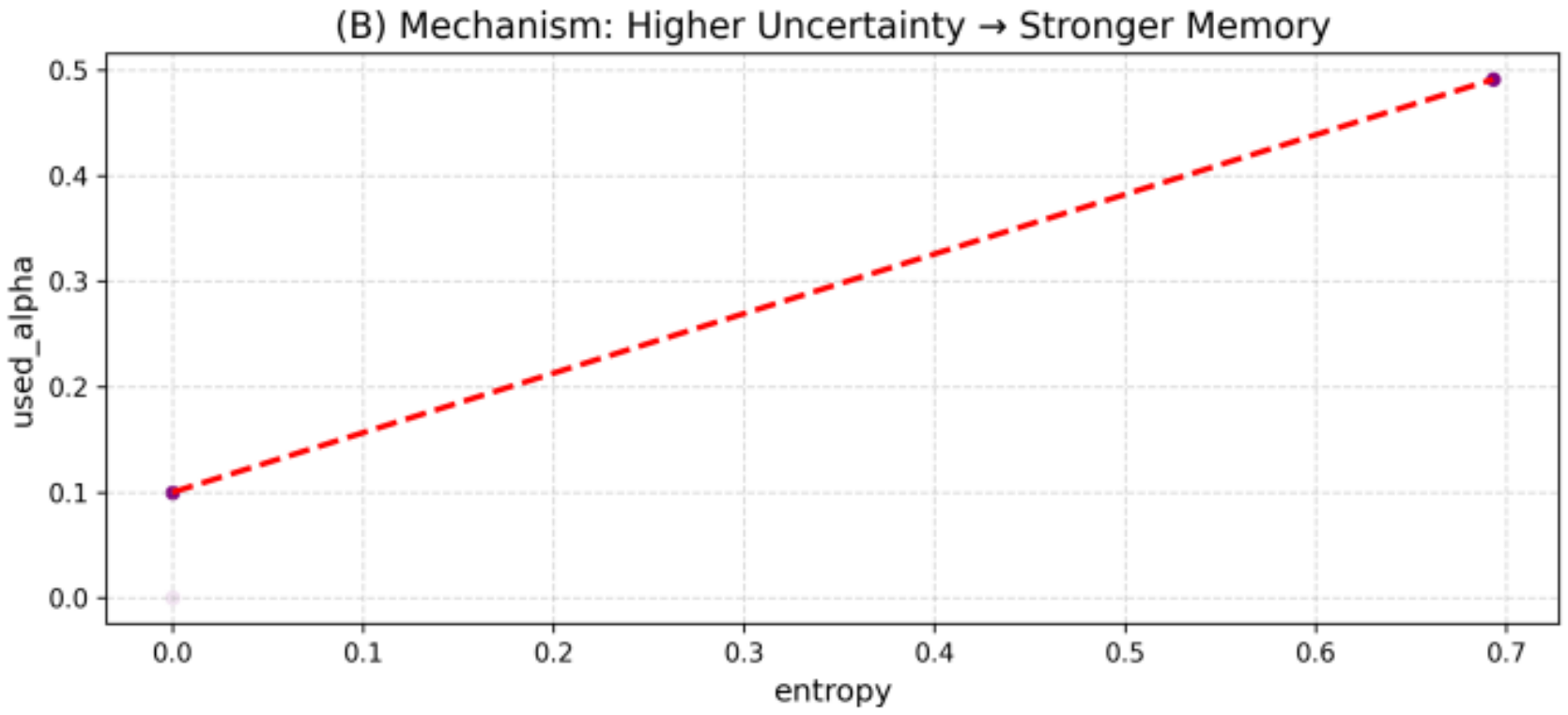
Results



Y axis : Confidence rate is from softmax value at final layer's logit.

X axis : From peak time 6.7s, designed the duration 5.7s ~ 8.7s

Additional study



- X axis : Entropy scale is calculated by Shannon's Entropy formula

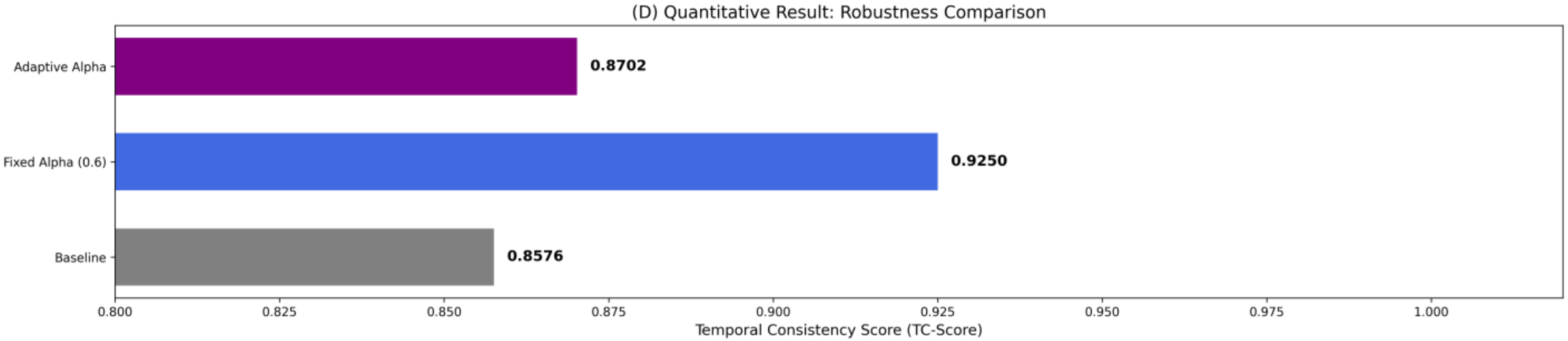
$$H(P) = -\sum_i p_i \log(p_i)$$

Rule based

- If Entropy is greater than 0.7, set alpha as 0.9
- If Entropy is less than 0.7, set alpha as 0.1
- If Entropy is exactly 0.7, then 0.5

Due to the lack of ground truth, I designed a heuristic algorithm based on prediction confidence.

Additional study

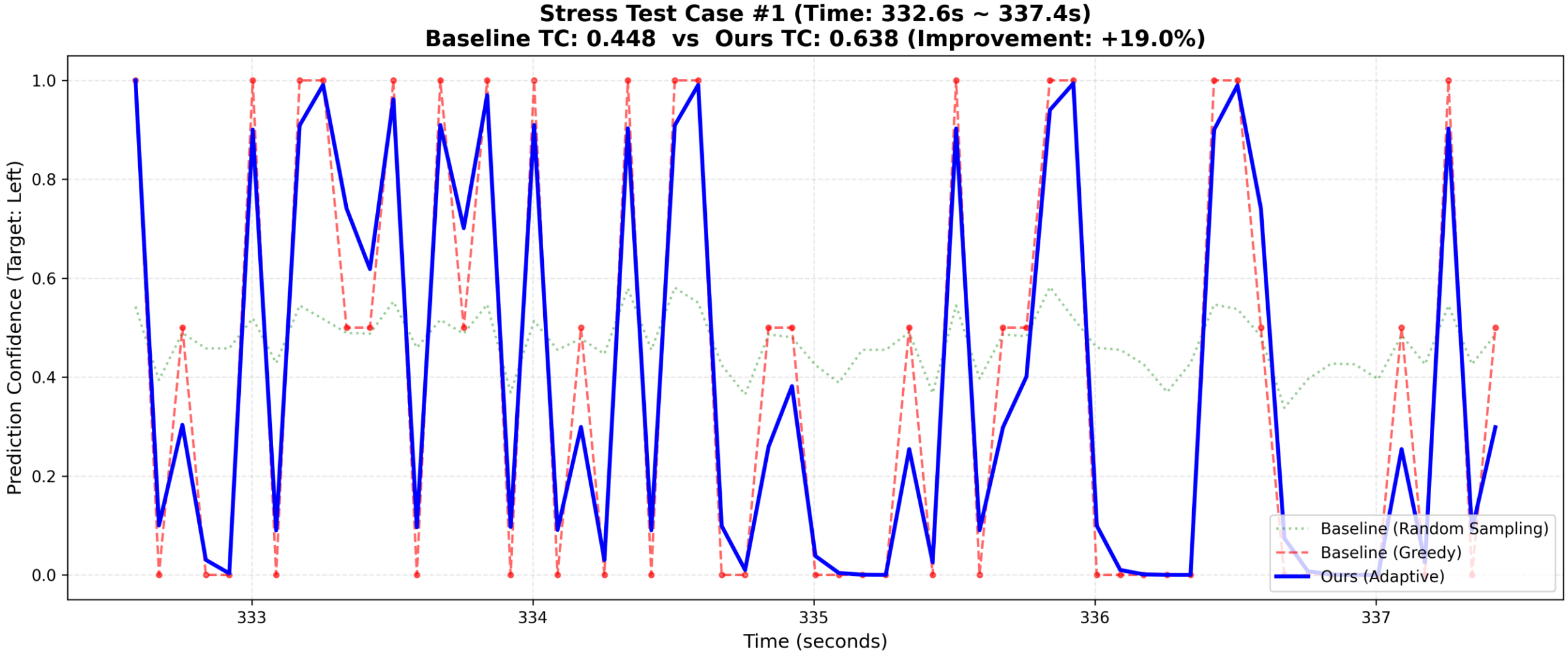


This is the consistency score, Bigger value doesn’t mean better method

Random sampling

- If the model was confident, the probs should be $[0.99, 0.005, 0.005]$
- If the model is confusing, the probs should be $[0.35, 0.33, 0.32]$
- Greedy Algorithm will choose Left no matter what.
- However, we don’t know whether this is confusing or certain situation.

Generalization with Random Sampling



Limitations

- Have to design the whole new model or fine tune some SOTA VLM
- Have to evaluate on Accuracy not just Consistency
  - How to label the answer in every frame?
- Is the TC score really good enough? Is there any better metric?
  - TC score see only the consistency. Though 1.0 might be bad, and 0.0 might not be bad.
- Additional scale-up studies needed