MICHIGAN STATE UNIVERSITY

ACTION LAB

Procedural Mistake Detection via Action Effect Modeling

Wenliang Guo

01 Background

ACTION LAB

Mistakes are universal and inevitable in procedural tasks.

The coffee cup sleeves fall off onto the table.
(HoloAssist dataset [1])

The corn in the bowl spills onto the table.
(CaptainCook4D dataset [2])

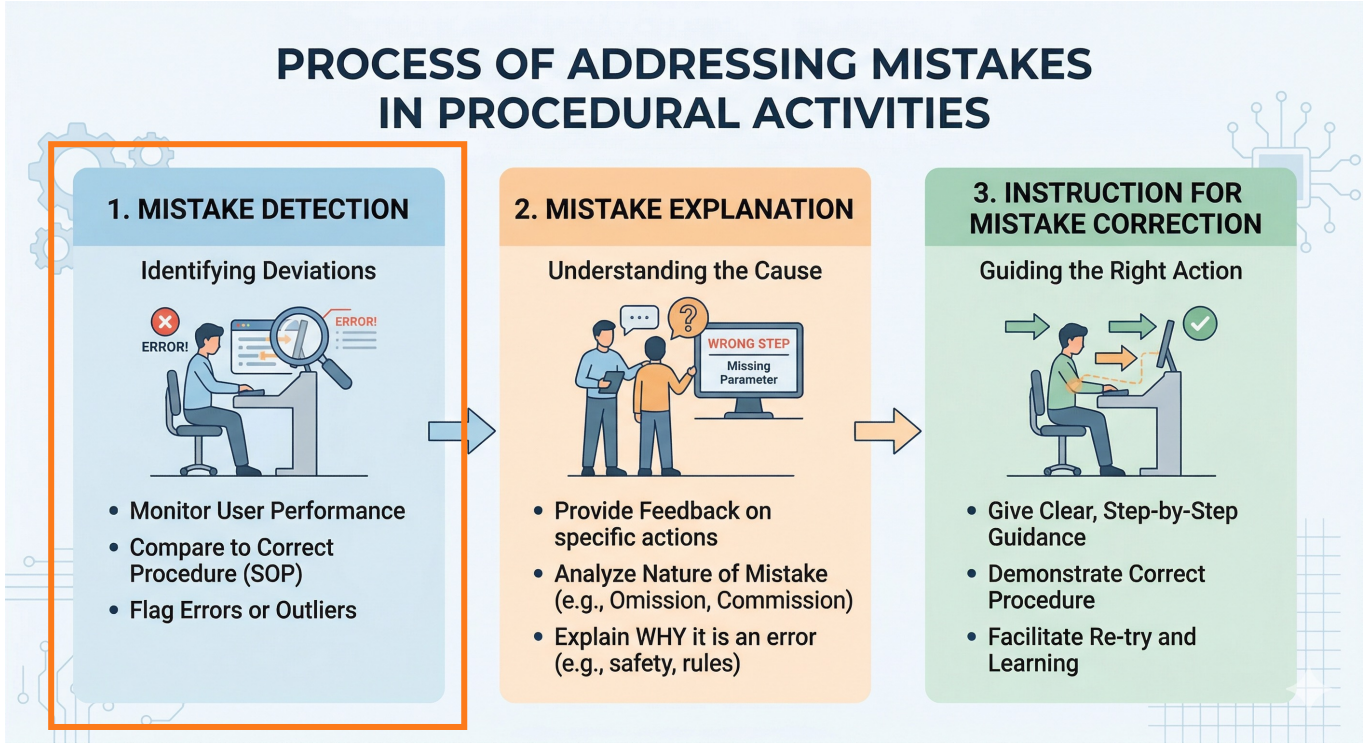
The dripper is knocked over and coffee ground spills out.
(EgoPER dataset [3])

[1] “Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world”, ICCV’23
[2] “Captaincook4d: A dataset for understanding errors in procedural activities.” NeurIPS’24
[3] “Error Detection in Egocentric Procedural Task Videos”, CVPR’24

Background

Mistake detection is essential in improving safety, reliability, and task completion.

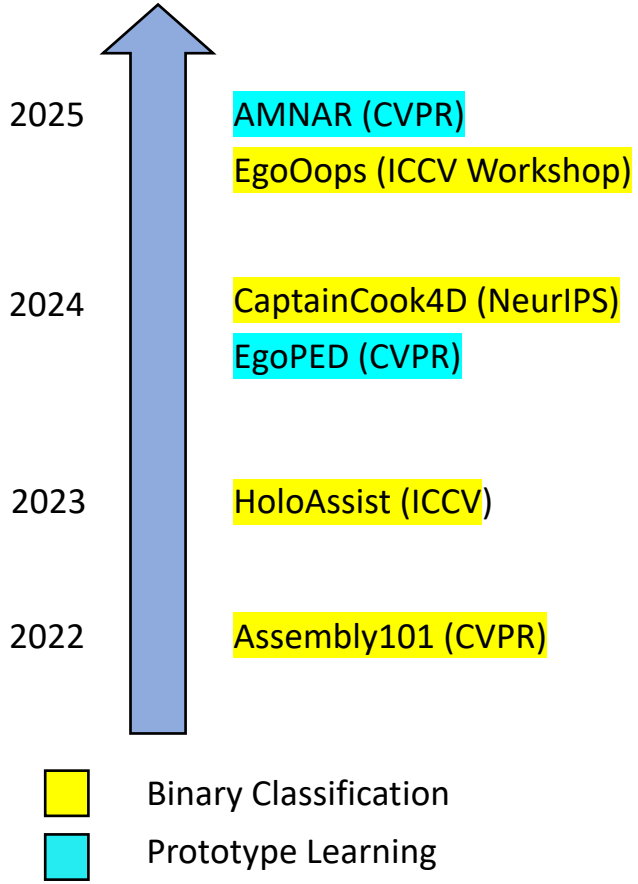
★ Here we are



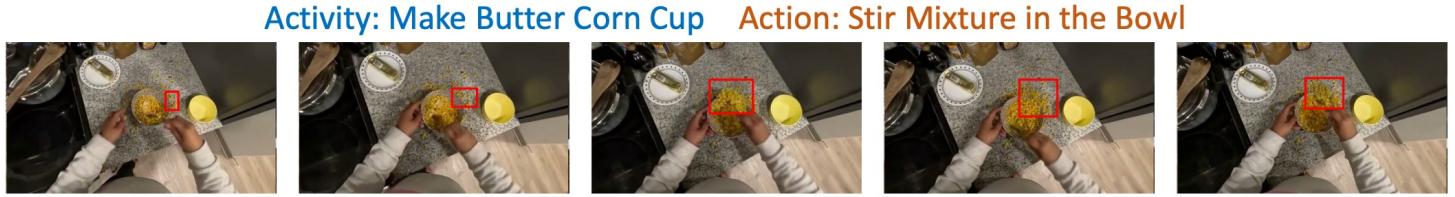
Motivation

ACTION LAB

The execution appears correct, yet minor deviations can lead to flawed outcomes.



Existing works assume mistakes can be identified solely from the execution process, without verifying if the *outcome aligns with the intended result*.

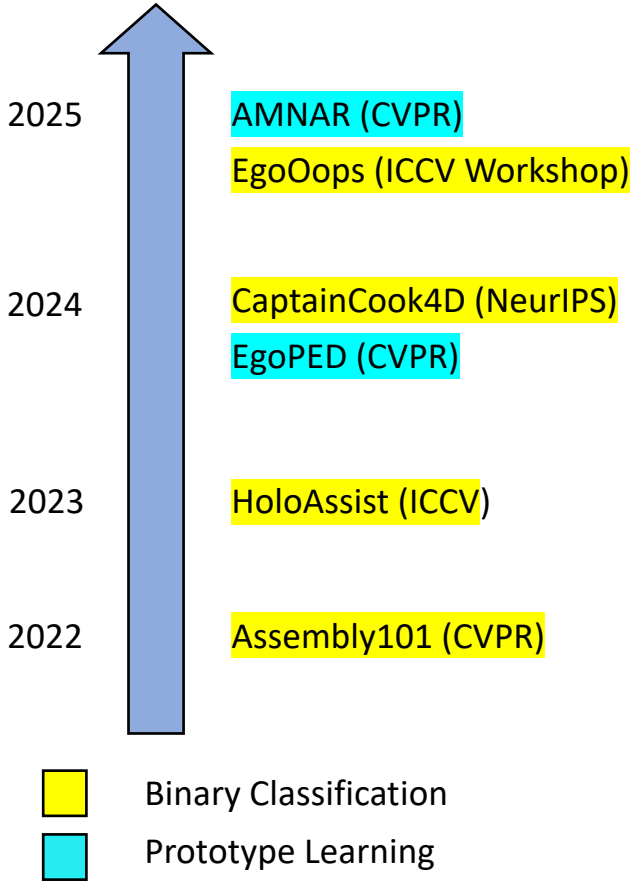


(a) Positional Mistake, i.e., spill out mixture onto table.

(b) State mistake, i.e., cucumber is cut into unexpected shape.

Motivation

The execution appears correct, yet minor deviations can lead to flawed outcomes.



Existing works assume mistakes can be identified solely from the execution process, without verifying if the *outcome aligns with the intended result*.

Our solution: **Action-effect Modeling**

Methodology

Model MD as a marginalization problem over latent variables of Action-effects.

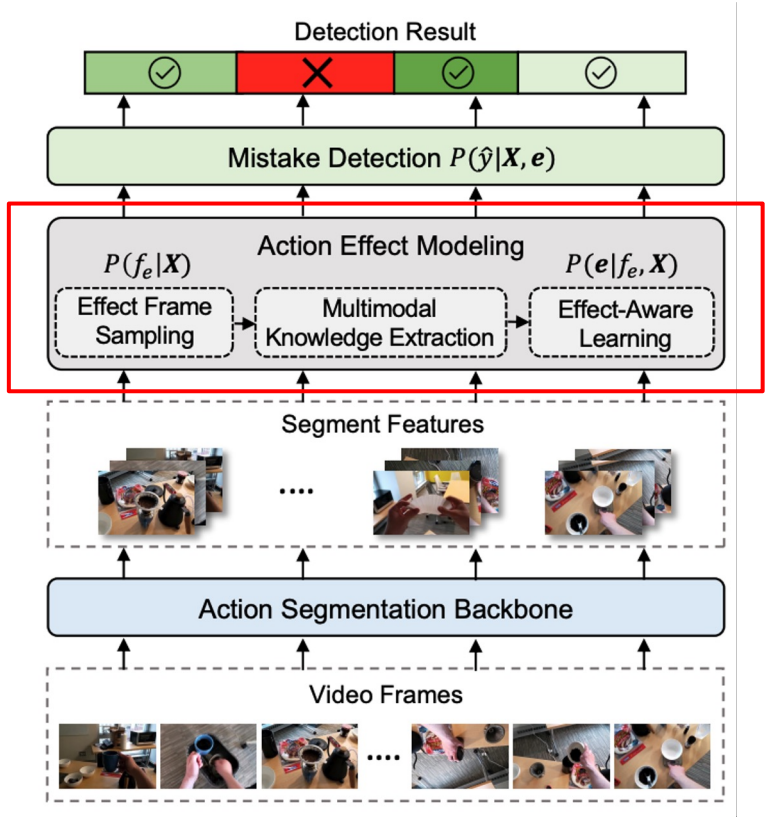
Previous formulation:

$P(\hat{y} \mid \mathbf{X})$ $\boldsymbol{X}$: Input video, $\hat{y}$: Framewise Prediction (normal vs. error)

Our formulation:

$$P(\hat{y} \mid \mathbf{X}) = \sum_{i=1}^{K} \sum_{f_e} \underbrace{P(\hat{y} \mid \mathbf{X}, e_i)}_{\text{mistake detection}} \cdot \underbrace{P(e_i \mid f_e, \mathbf{X})}_{\text{effect-aware learning}} \cdot \underbrace{P(f_e \mid \mathbf{X})}_{\text{frame sampling}},$$

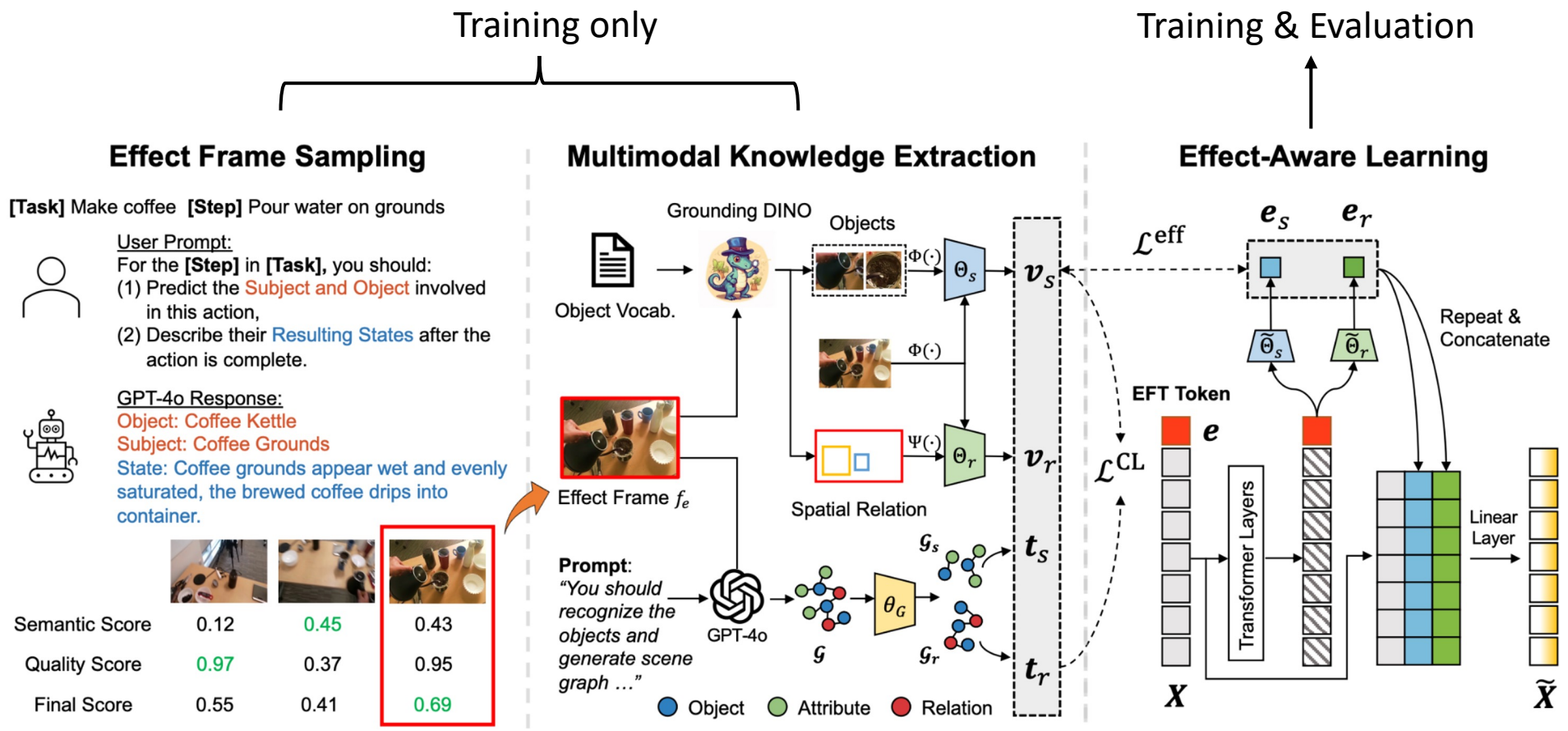
f_e: Effect frames e: Action-effect descriptor
K: Number of effect frames



Methodology

ACTION LAB

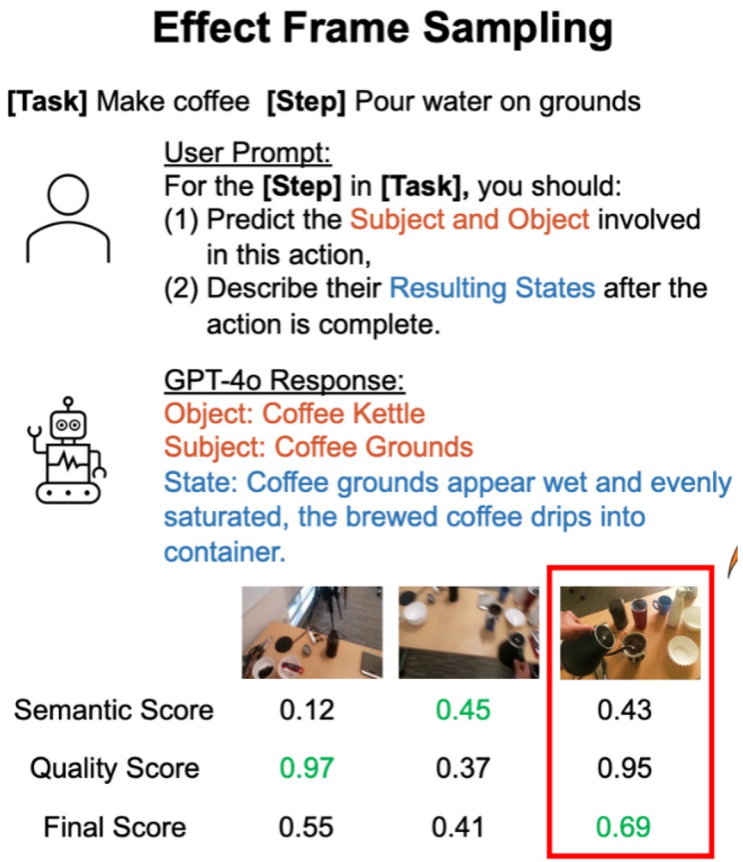
Action-effect Modeling Framework.



ACTION LAB

Effect-frame Sampling.

Step 1: Effect Frame Sampling
Perform rule-based sampling of Effect Frames *capturing the object* after each action in a procedure.



Methodology

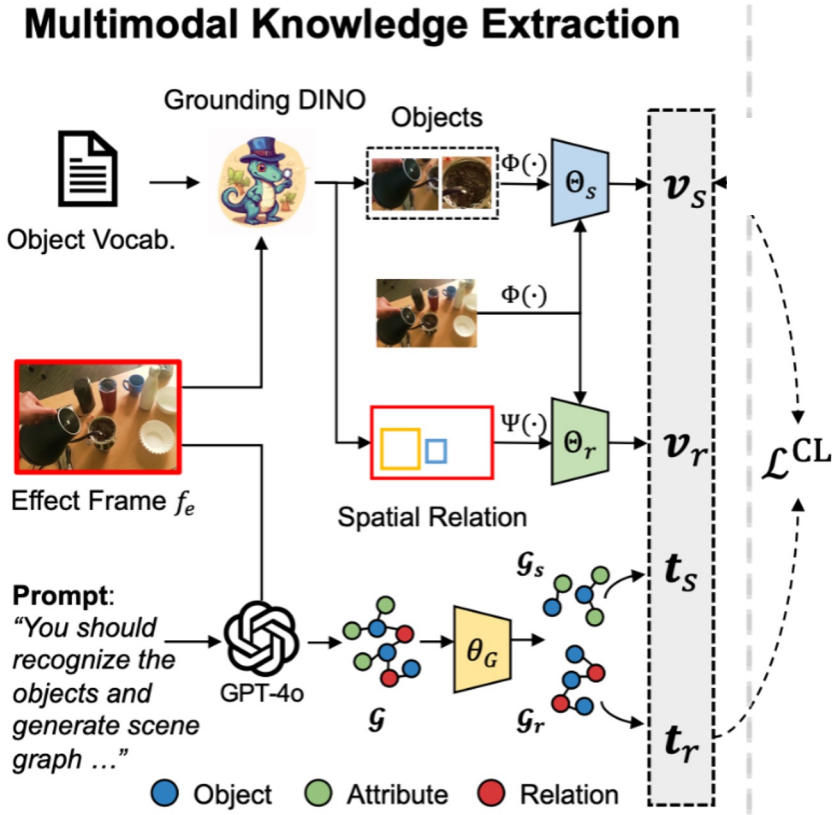
ACTION LAB

Action-effect Feature Extraction.

Step 2: Multimodal Feature Extraction

- Model Action-effect from *object-state and spatial-relationship* perspectives.
- Extract *image and text features* for each perspective with object-detector (Grounding DINO) and scene-graph constructor (GPT-4o).
- Cross-modal feature alignment.

$$\mathcal{L}_s^{\mathrm{CL}} = \sum_i \left[-\log \frac{\exp\left(\cos(\mathbf{v}_s^i, \mathbf{t}_s^i)/\rho\right)}{\sum_j \exp\left(\cos(\mathbf{v}_s^i, \mathbf{t}_s^j)/\rho\right)} \right]$$



Methodology

ACTION LAB

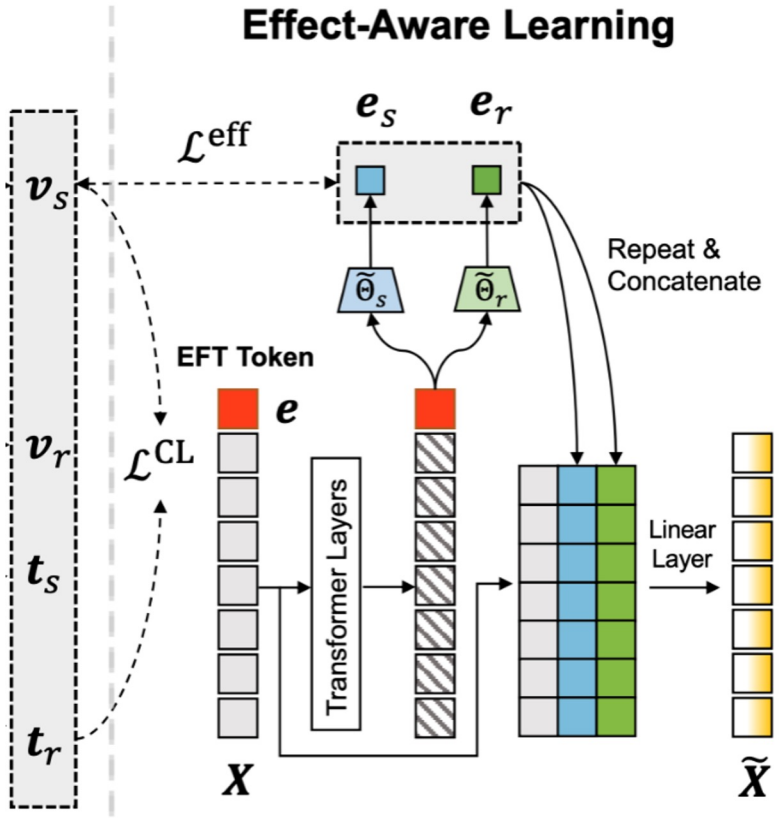
Effect-aware Representation Learning by Feature Alignment.

Step 3: Effect-aware Representation Learning
Append an *EFT token* at the beginning of the video embedding sequence.

Learn *State- and Relation-projectors* during the training to align with the VLM features.

$$\mathcal{L}_s^{\text{eff}} = ||\widetilde{\Theta}_s(\mathbf{e}) - \mathbf{v}_s||^2 + ||\widetilde{\Theta}_s(\mathbf{e}) - \mathbf{t}_s||^2,$$
$$\mathcal{L}_r^{\text{eff}} = ||\widetilde{\Theta}_r(\mathbf{e}) - \mathbf{v}_r||^2 + ||\widetilde{\Theta}_r(\mathbf{e}) - \mathbf{t}_r||^2,$$

Fuse the projected token to the original video embedding sequence via repetition and concatenation.



Experimental Results

ACTION LAB

Evaluate on EgoPER (CVPR'24) and CaptainCook4D (NeurIPS'24) datasets.

Table 1: Mistake detection results on the EgoPER dataset (Lee et al., 2024). AUC and EDA are in percentage (%). Best results are **bold**, and second-best are underlined.

Method	Quesadilla		Oatmeal		Pinwheel		Coffee		Tea		All	
	AUC	EDA	AUC	EDA	AUC	EDA	AUC	EDA	AUC	EDA	AUC	EDA
Random	50.0	19.9	50.0	11.8	50.0	15.7	50.0	8.2	50.0	17.0	50.0	14.5
HF^2-VAD	62.6	34.5	62.3	25.4	52.7	29.1	59.6	10.0	62.1	36.6	59.9	27.1
HF^2-VAD + SSPCAB	60.9	30.4	61.9	25.3	51.7	33.9	60.1	10.0	63.2	35.4	59.6	27.0
S3R	51.8	52.6	61.6	47.8	52.4	50.5	51.0	16.3	57.9	47.8	54.9	43.0
EgoPED	65.6	62.7	65.1	51.4	55.0	59.6	58.3	55.3	66.0	56.0	62.0	57.0
AMNAR	71.9	61.4	75.4	65.0	65.4	**65.0**	67.8	**73.5**	61.9	57.0	68.5	64.4
Ours	**80.8**	**68.1**	**77.0**	**68.6**	**69.9**	61.2	**70.3**	66.4	**71.1**	**69.4**	**73.8**	**66.7**

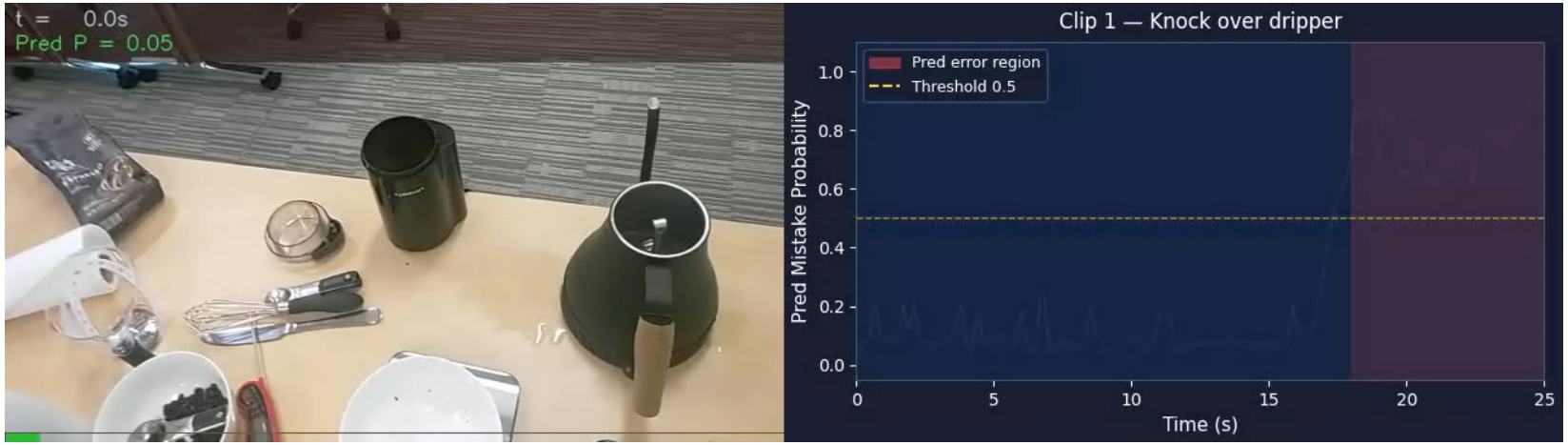
Table 2: Mistake detection results on the CaptainCook4D dataset (Peddi et al., 2023).

Method	Precision	AUC	EDA
Random	44.9	51.2	49.7
EgoPED	56.5	54.9	69.8
AMNAR	66.8	60.2	**72.3**
Ours	**68.1**	**62.5**	71.9

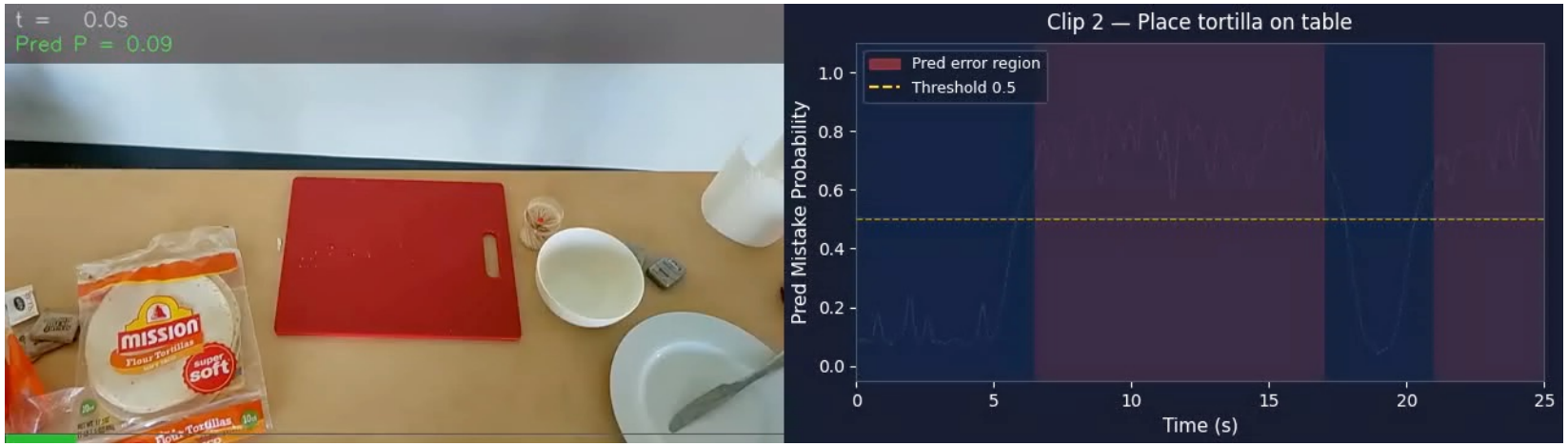
Experimental Results

ACTION LAB

Knock over the dripper.

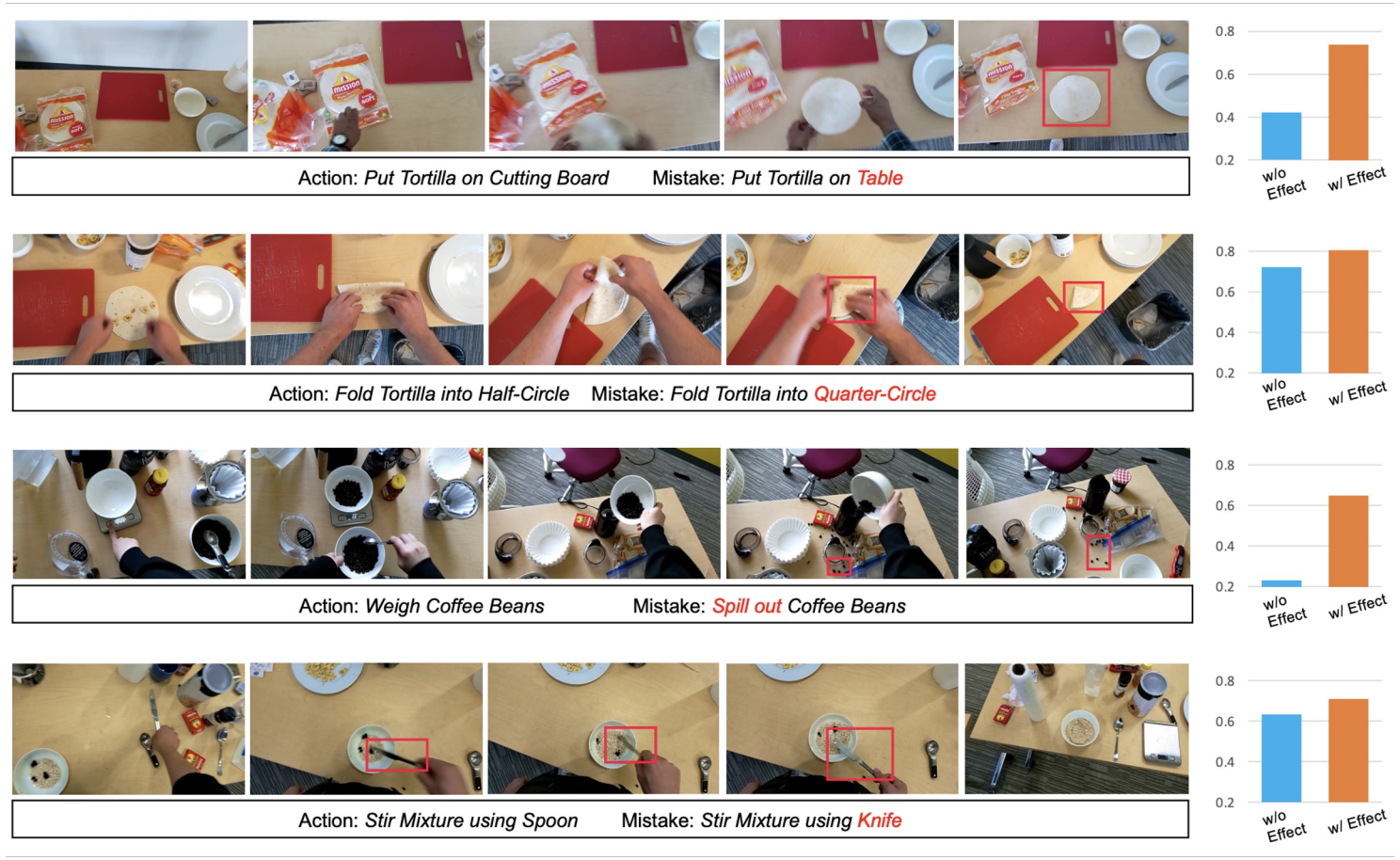


Place tortilla on table instead of cutting board.



Experimental Results

ACTION LAB



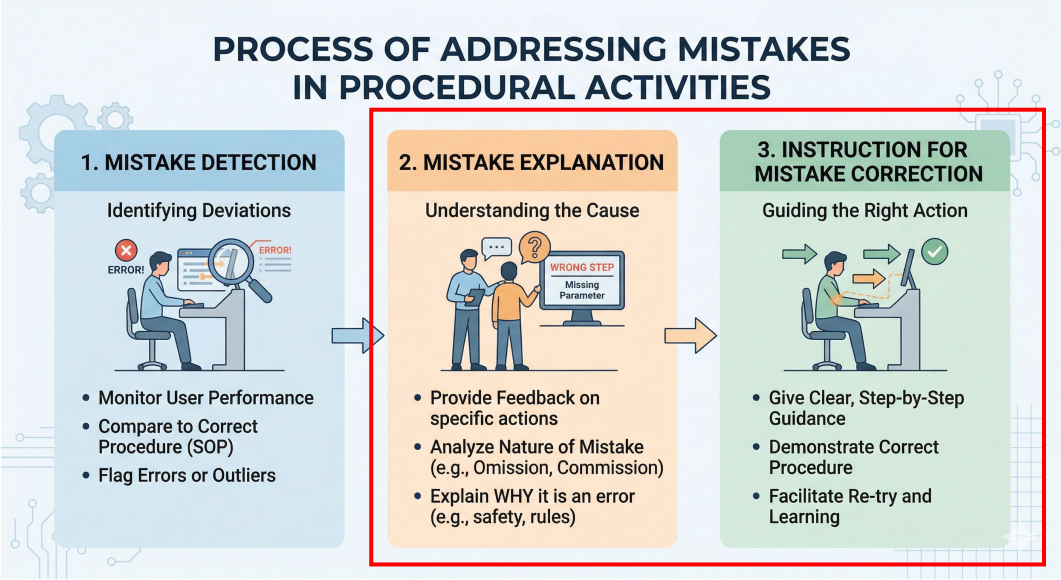
Model effectively detects errors appearing in the final outcomes.

Model generalizes to detect execution errors during the action process.

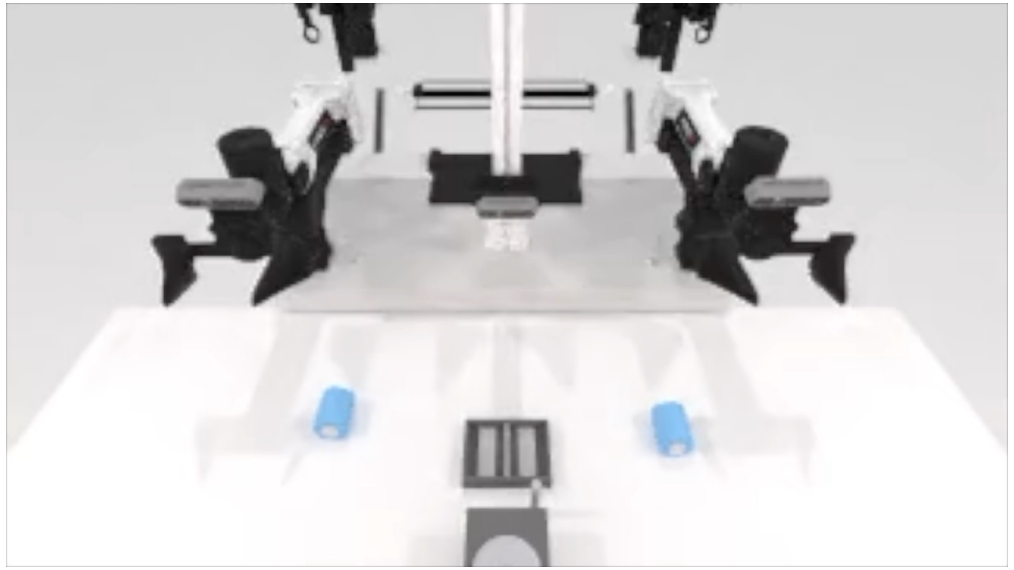
Future Work

1. Spatial-temporal Action-effect Modeling.

2. Mistake Explanation and Correction.



3. Generalizability to other domains.



(Example of robotic mistake on EMBench dataset[1])

[1] “RMBench: Memory-Dependent Robotic Manipulation Benchmark with Insights into Policy Design”, Arxiv 2026