Liquid Pouring Monitoring via Rich Sensory Inputs

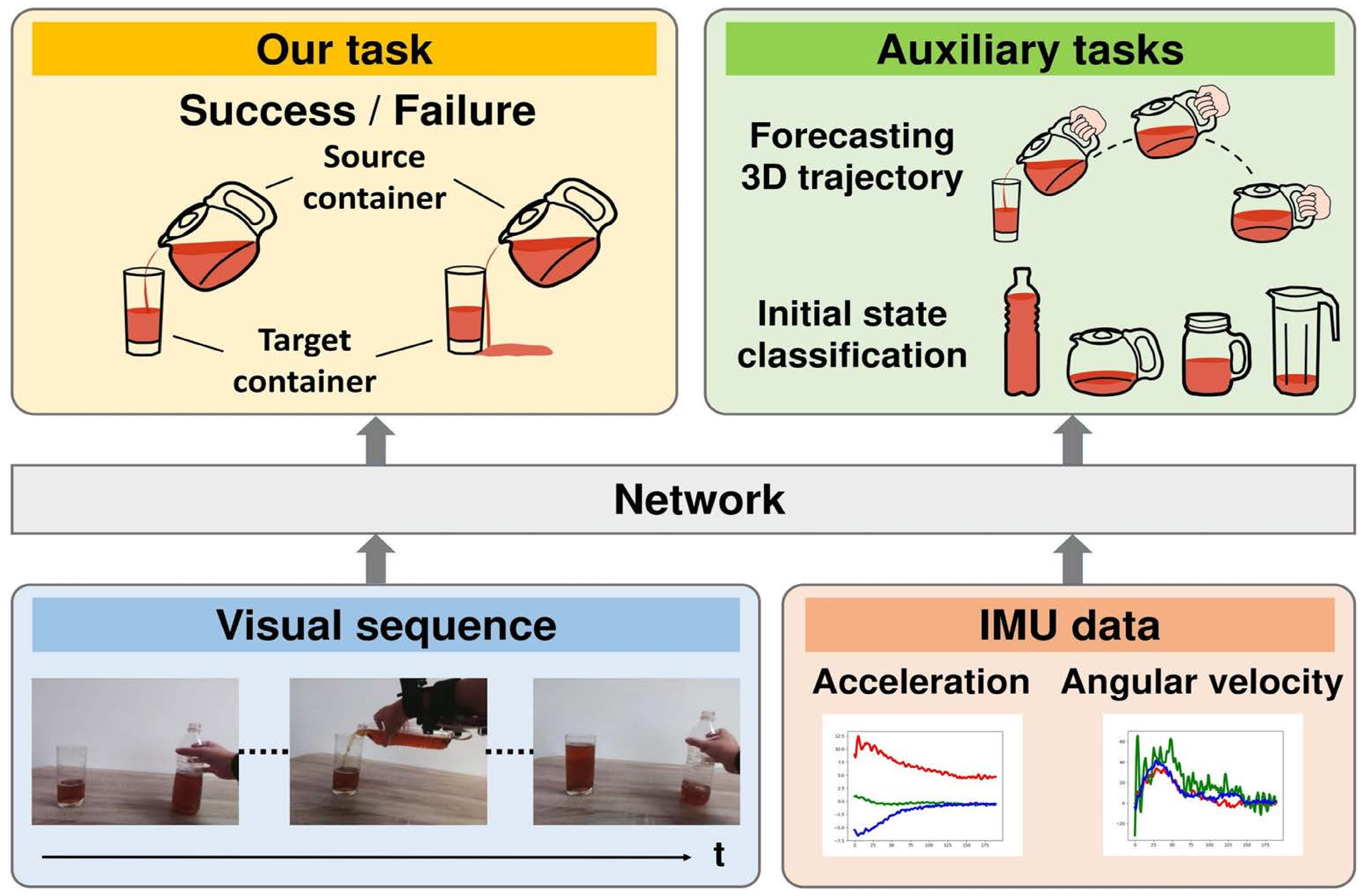
Tz-Ying Wu[1,*], Juan-Ting Lin[1,*], Tsun-Hsuang Wang[1], Chan-Wei Hu[1],
Juan Carlos Niebles[2], Min Sun[1] (*indicate equal contribution)
[1] National Tsing Hua University [2] Stanford University

ECCV 2018 European Conference on Computer Vision

Introduction

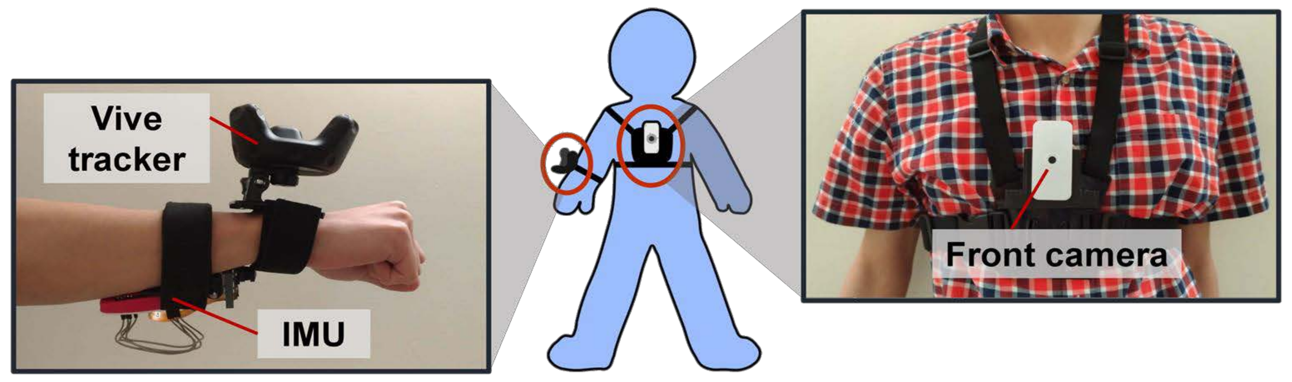
Humans have the amazing ability to perform very subtle manipulation tasks using a closed-loop control system with imprecise mechanics (i.e., our body parts) but rich sensory information (e.g., vision, tactile, etc.).

In this work, we take liquid pouring as a concrete example and aim at learning to continuously monitor whether liquid pouring is successful (e.g., no spilling) or not via rich sensory inputs.

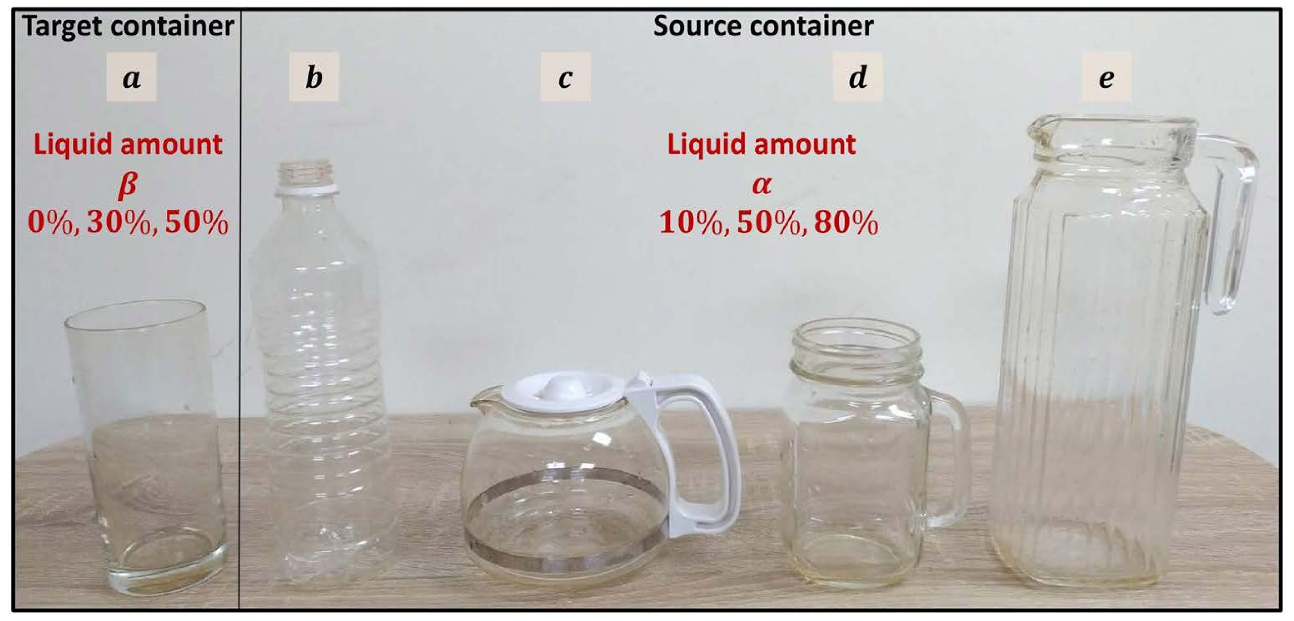


Dataset

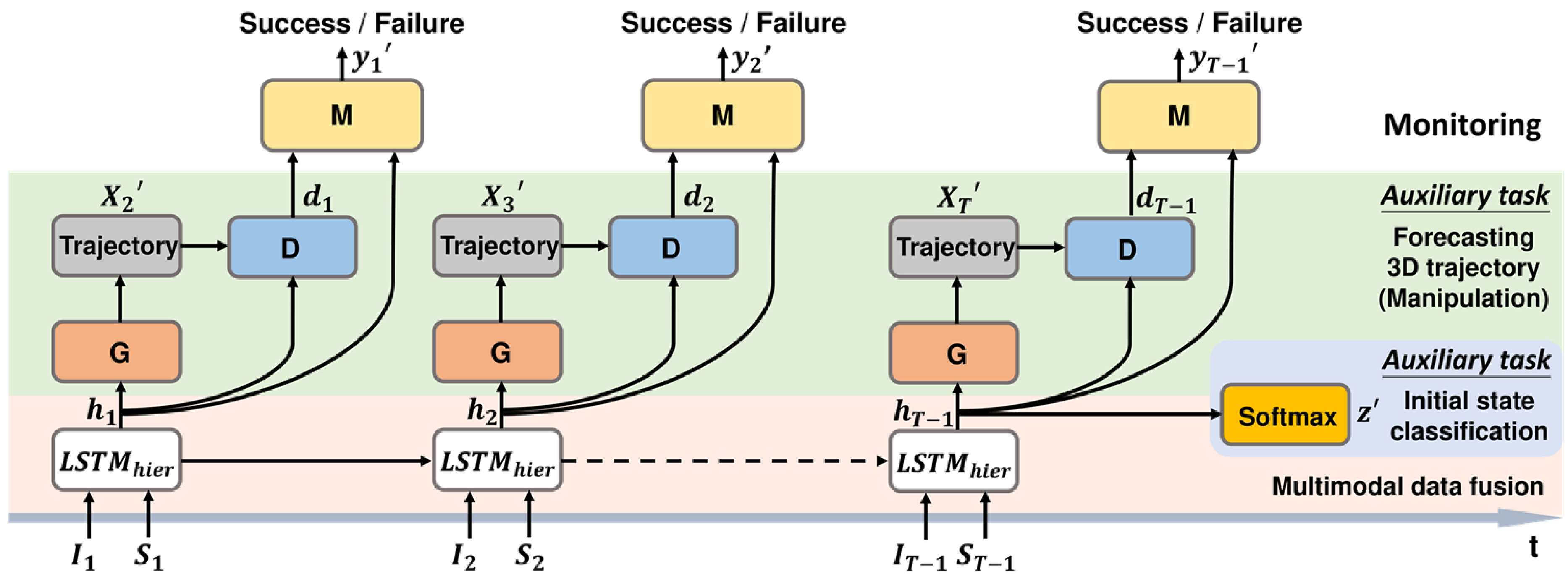
We use rich sensory inputs to collect both **successful** and **failure** pouring sequences.



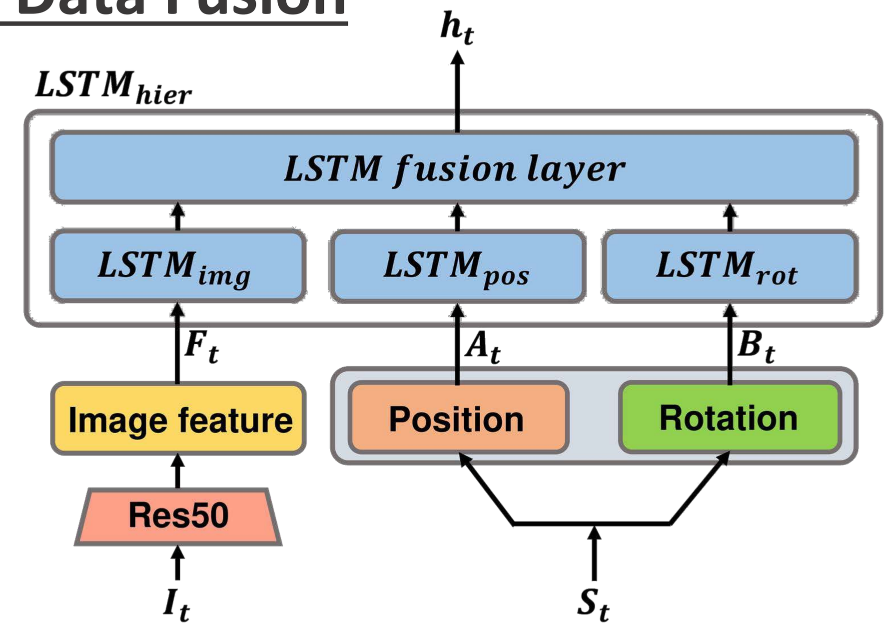
Variations of Pouring Sequences



Our Method



Multimodal Data Fusion



Initial State Classification

We predict the initial state of containers: container type and liquid amount.

Forecasting 3D Trajectory

We forecast the one-step future 3D trajectory of the hand with an adversarial training procedure.

Generator $L_{Gen} = L_{reg} + L_{adv}$

- We generate trajectories that are close to the ground truth demonstration. (with regression loss)

$$L_{reg} = \frac{1}{T-1}\sum_{t=1}^{T-1} dist(X_{t+1}, G_{\theta_G}(h_t))$$

$$dist\left(X_{t+1}, G_{\theta_G}(h_t)\right) = MSE(P, P') + \sum_{k=x,y,z}\left(1 - cos(r_k - r_k')\right)$$

- We aim at fooling the discriminator with the generated trajectory (with adversarial loss)

$$L_{adv} = \frac{1}{T-1}\sum_{t=1}^{T-1} -logD_{\theta_D}\left(h_t, G_{\theta_G}(h_t)\right)$$

Discriminator

- Discriminates real samples and generated samples

$$L_{Dis} = \frac{1}{T-1}\sum_{t=1}^{T-1}\left[-log\left(D_{\theta_D}(h_t, X_{t+1})\right) - log\left(1 - D_{\theta_D}\left(h_t, G_{\theta_G}(h_t)\right)\right)\right]$$

Experiment

Cross Container

	succ./fail. acc.	classification acc.	position error	rotation error
Vanilla RNN	89.16 %	N/A	N/A	N/A
Ours w/o adv.	96.45 %	63.92 %	0.040 m	11.11°
Ours	**97.11 %**	**67.69 %**	**0.038 m**	11.30°

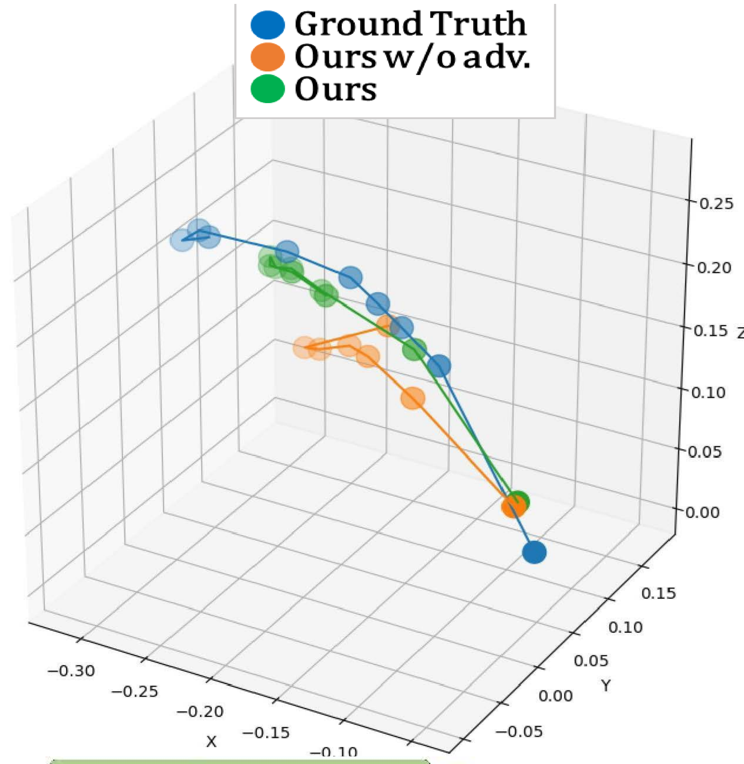
Cross User

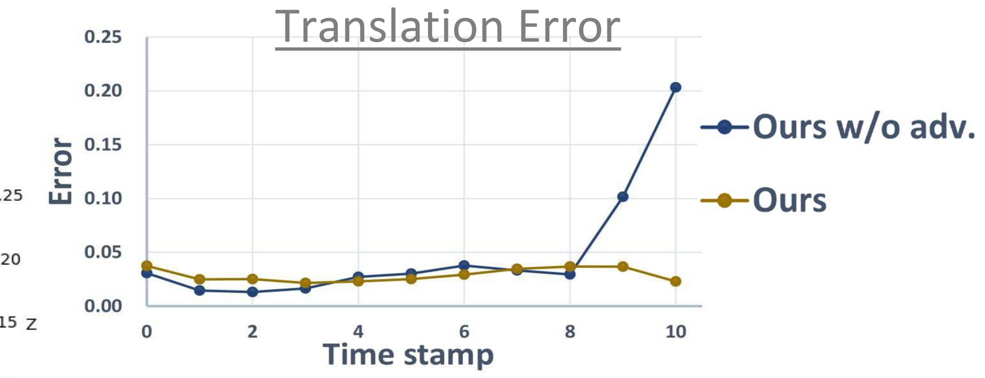
	succ./fail. acc.	classification acc.	position error	rotation error
Vanilla RNN	81.95 %	N/A	N/A	N/A
RNN w/ IOSC	89.25 %	68.51 %	N/A	N/A
RNN w/ TF	90.82 %	N/A	0.033 m	14.15°
Ours w/o adv.	92.97 %	64.15 %	0.033 m	14.20°
Ours	**93.25 %**	**75.69 %**	0.033 m	**14.06°**

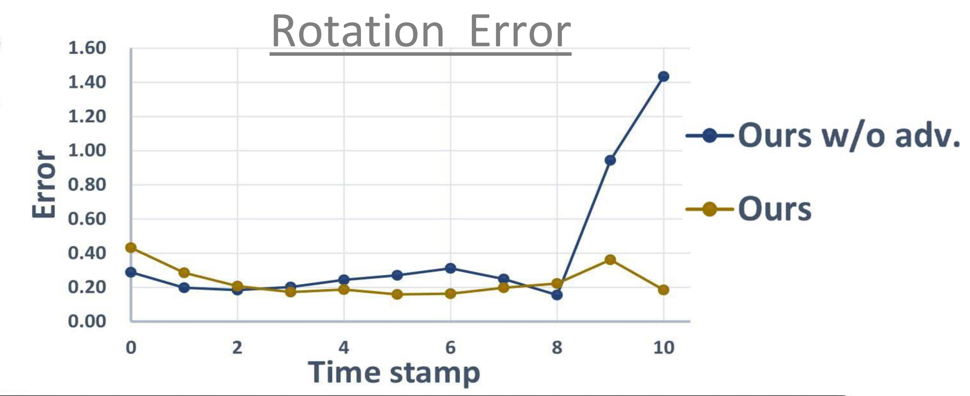
Vanilla RNN: Our fusion RNN without any auxiliary tasks.
RNN w/ IOSC: Our fusion RNN with an aux. task, initial object state classification.
RNN w/ TF: Our fusion RNN with an aux. task, trajectory forecasting
Ours w/o adv.: Our fusion RNN with two aux. tasks. In this setting, we treat one-step trajectory forecasting as a regression task.
Ours: Our fusion RNN with two aux. tasks. In this setting, we introduce the adversarial training loss to generate more diverse trajectories.

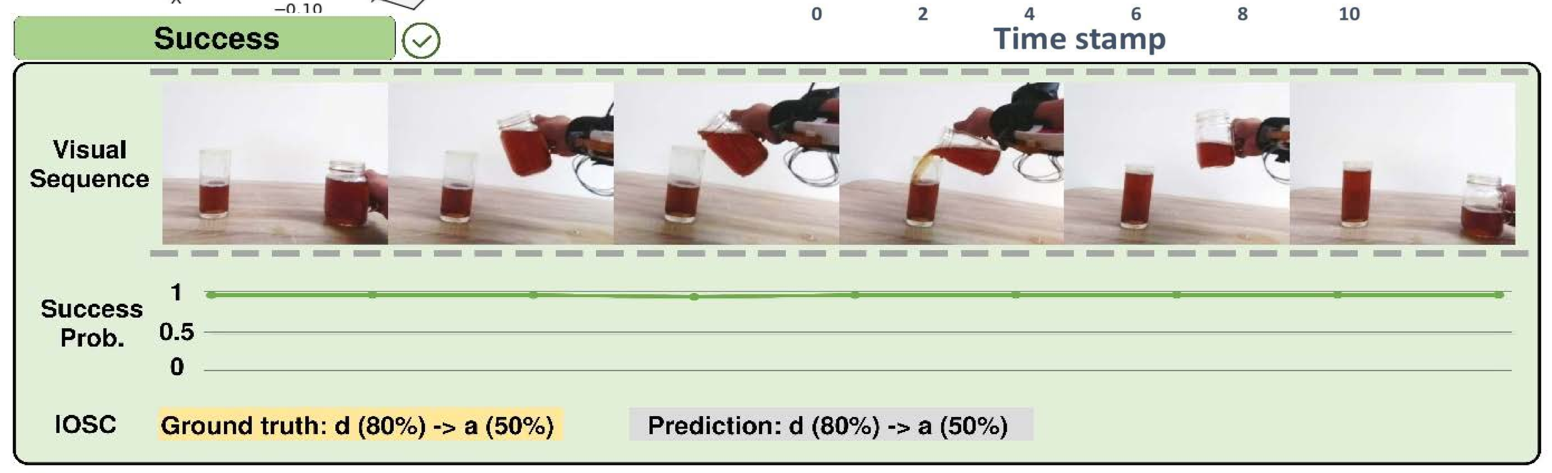
Ablation Study

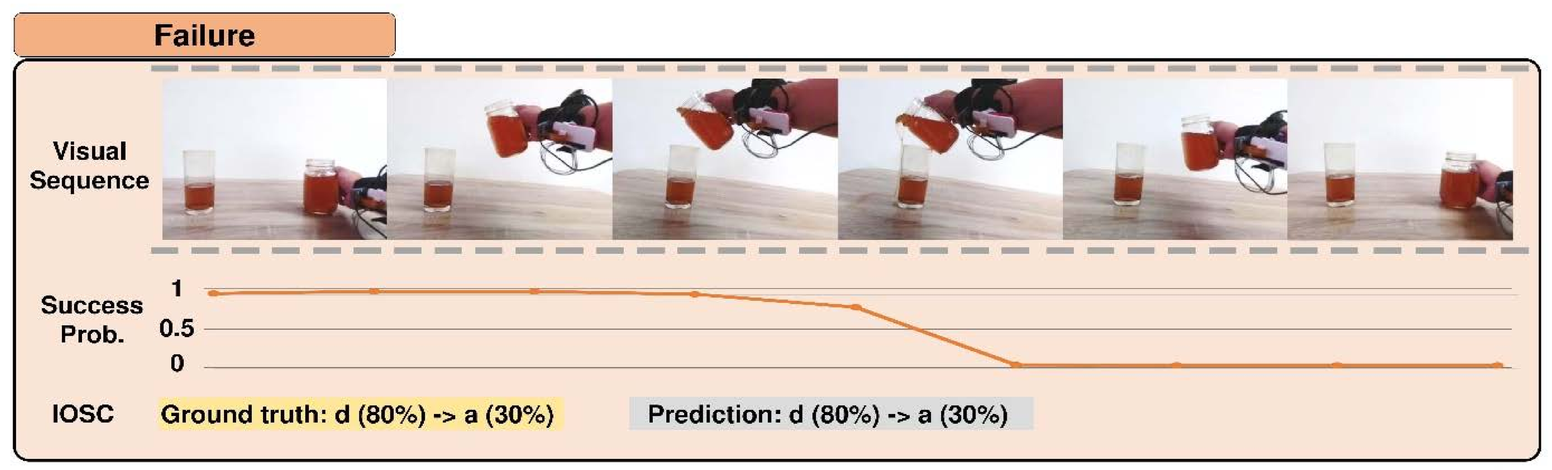
LSTM architecture	succ./fail. acc.	classification acc.	position error	rotation error
2-layer	87.06 %	58.92 %	0.033 m	14.72°
hierachical	**93.25 %**	**75.69 %**	0.033 m	**14.06°**











Acknowledgement

Ministry of Science and Technology
Panasonic
MEDIATEK

Project Page: http://aliensunmin.github.io/project/monitoring/