# Enhancing Language-Conditioned Robotic Manipulation through Prompt Tuning in a Missing Color Environment

Baoming Zhang, Ouyang Xu

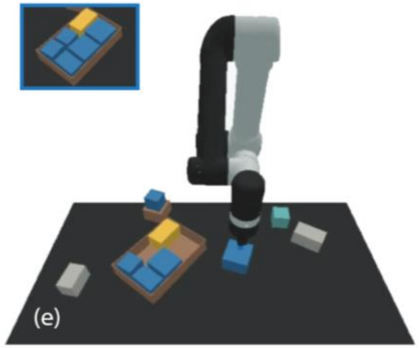
Presented on 12/04/2024

# Motivation

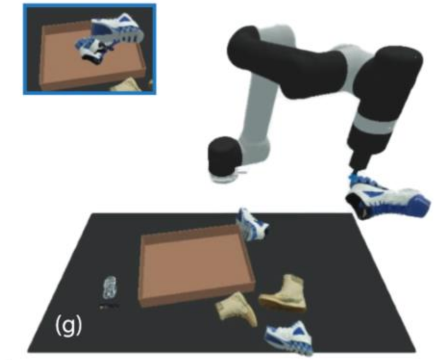
- Leverage large visual-language model, such as CLIP[1], on robotic grasping

- Robustness on text prompt corruptions: prompt engineering

<u>Language-conditioned tasks</u>:




Pack all the yellow and blue blocks in the brown box




Pack all the blue and black sneaker objects in the brown box

<u>Examples from CLIP</u>:

Caltech101

| Prompt | Accuracy |
|---|---|
| a [CLASS]. | 82.68 |
| a photo of [CLASS]. | 80.81 |
| a photo of a [CLASS]. | 86.29 |

Flowers102

| Prompt | Accuracy |
|---|---|
| a photo of a [CLASS]. | 60.86 |
| a flower photo of a [CLASS]. | 65.81 |
| a photo of a [CLASS], a type of flower. | 66.14 |



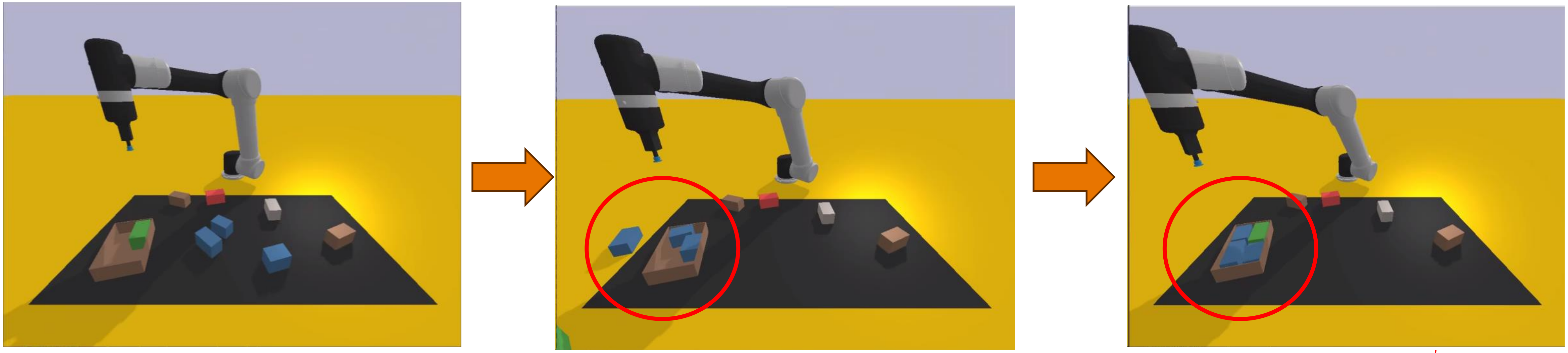
[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal,G. Sastry, A. Askell, P. Mishkin, J. Clark et al., “Learning transferable visual models from natural language supervision,” in International Conference on Machine Learning. PMLR, 2021, pp. 8748–8763

# Motivation

packing box pairs:
all the blue and green blocks

Template 1

Template 3



Fail ×

Success √

| Text Prompt | Task success scores (%) |
|---|---|
| 1. 'pack all the [colors] blocks'; | 90.5 |
| 2. 'pack all the [colors] blocks into the box'; | 92.1 |
| 3. 'pack all the [colors] blocks into the brown box.'. | 97.1 |

# Problem Formulation (based on CLIPORT[1])

- **Objective**: Learn a goal-conditioned policy $\pi$ that outputs actions $\mathbf{a}_t$ based on inputs:
  - $\gamma_t = (\mathbf{o}_t, \mathbf{l}_t)$, where:
    - $\mathbf{o}_t$: Visual observation.
    - $\mathbf{l}_t$: English language instruction.
- **Policy Definition**:

$$\pi(\mathbf{o}_t, \mathbf{l}_t) \rightarrow \mathbf{a}_t = (\mathcal{T}_{\text{pick}}, \mathcal{T}_{\text{place}}) \in \mathcal{A}$$

  - $\mathbf{a}_t$: End-effector poses for:
    - $\mathcal{T}_{\text{pick}}$: Picking.
    - $\mathcal{T}_{\text{place}}$: Placing.
- **Task Focus**: Tabletop tasks where:
  - $\mathcal{T}_{\text{pick}}, \mathcal{T}_{\text{place}} \in \mathbf{SE}(2)$.
- **Visual Observations**:
  - Top-down orthographic RGB-D reconstructions.
  - Each pixel corresponds to a point in 3D space.



[1]M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," 2021.

# Problem Formulation

- **Language Instructions**:
  - Single goal descriptions:
    - Example: *"Pack all the blue and yellow boxes in the brown box."*

- **Dataset $\mathcal{D}$**:
  - $n$ expert demonstrations:

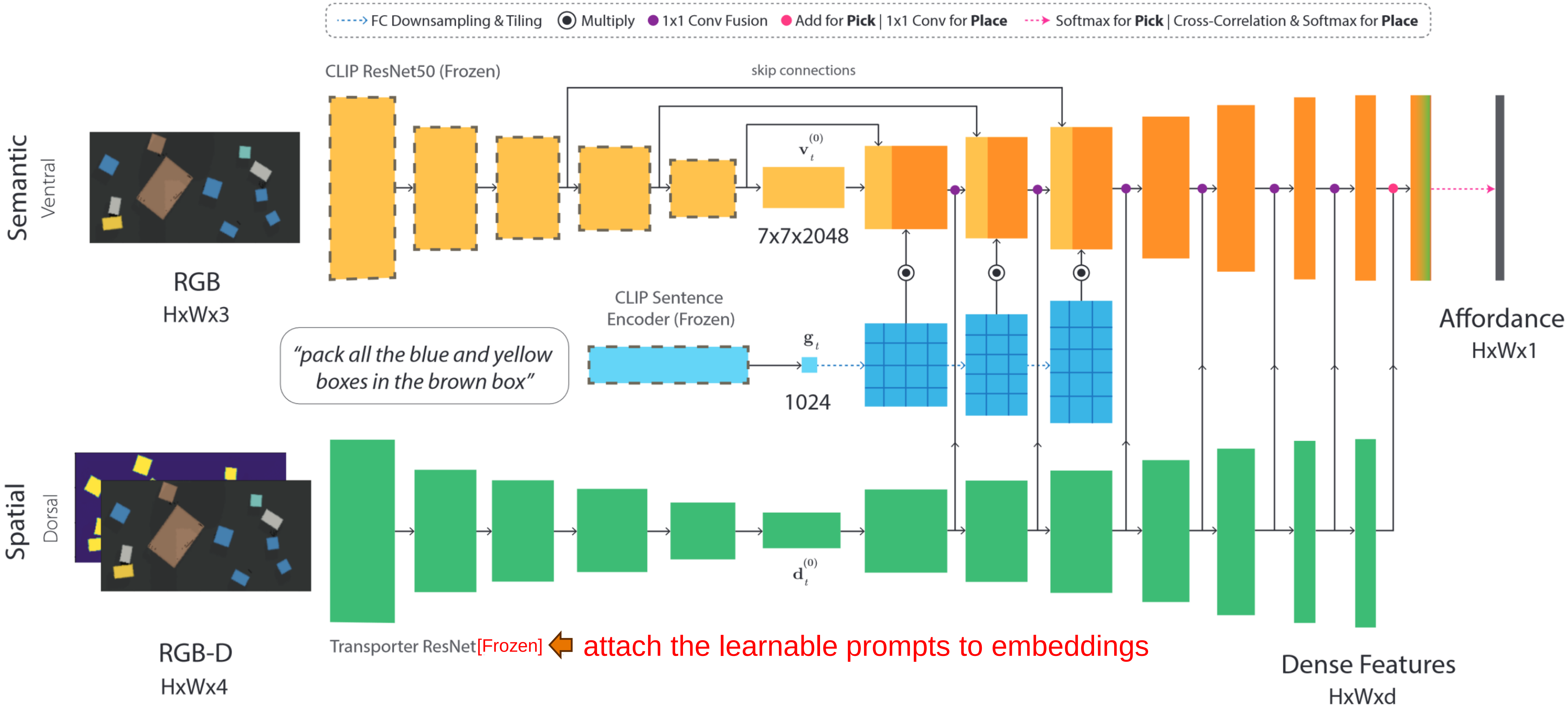
$$\mathcal{D} = \{\zeta_1, \zeta_2, \ldots, \zeta_n\}.$$

  - Each demonstration $\zeta_i$ contains input-action pairs:

$$\zeta_i = \{(\mathbf{o}_1, \mathbf{l}_1, \mathbf{a}_1), (\mathbf{o}_2, \mathbf{l}_2, \mathbf{a}_2), \ldots\}.$$

  - Actions $\mathbf{a}_t = (\mathcal{T}_{\text{pick}}, \mathcal{T}_{\text{place}})$: Expert pick-and-place coordinates.
- **Supervision**: Expert demonstrations are used to train the policy $\pi$.

# Foundational work




An overview of the semantic and spatial streams of foundation work CLIPORT[1].
What we did for the missing color environment is in red.



[1]M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," 2021.

# Transporter for Pick-and-Place

- **Overview**: Policy $\pi$ is trained using Transporter [2] for spatial manipulation.
  - Two stages:
    1. Attend to a local region to determine the pick location.
    2. Compute the placement location using cross-correlation of deep visual features.
- **Policy Components**:
  - Two action-value modules (Q-functions):
    - $\mathcal{Q}_{\text{pick}}$: Identifies the pick location.
    - $\mathcal{Q}_{\text{place}}$: Determines the placement location conditioned on the pick action.
- **Place Module**:
  - Query FCN $\Phi_{\text{query}}$ processes:
    - $\gamma_t[\mathcal{T}_{\text{pick}}]$: $c \times c$ crop around $\mathcal{T}_{\text{pick}}$.
    - $\mathbf{l}_t$: Language instruction.
  - Key FCN $\Phi_{\text{key}}$ processes full input $\gamma_t$.
  - Placement action-values $\mathcal{Q}_{\text{place}}$:

$$\boxed{\mathcal{Q}_{\text{place}}(\Delta\tau|\gamma_t, \mathcal{T}_{\text{pick}}) = \left(\Phi_{\text{query}}(\gamma_t[\mathcal{T}_{\text{pick}}]) * \Phi_{\text{key}}(\gamma_t)\right)[\Delta\tau]}$$

THE UNIVERSITY OF TEXAS AT DALLAS

[2] A. Zeng, P. Florence, J. Tompson et al., “Transporter networks: Rearranging the visual world for robotic manipulation,” in Proceedings of the Conference on Robot Learning, 2021, pp. 726–747.

# Prompt Tuning

$$\mathcal{Q}_{\text{place}}(\Delta\tau|\gamma_t, \mathcal{T}_{\text{pick}}) = \left(\Phi_{\text{query}}(\gamma_t[\mathcal{T}_{\text{pick}}]) * \Phi_{\text{key}}(\gamma_t)\right)[\Delta\tau]$$

The prompts are appended as additional dimensions to the query and key embeddings

**Mathematical Formulation:**

$$\Phi'_{\text{query}} = \text{concat}(\Phi_{\text{query}}(\gamma_t[\mathcal{T}_{\text{pick}}]), P_{\text{query}}, \text{dim} = -1)$$
$$\Phi'_{\text{key}} = \text{concat}(\Phi_{\text{key}}(\gamma_t), P_{\text{key}}, \text{dim} = -1)$$

**Dimensionality Impact:**

- $P_{\text{query}} \in \mathbb{R}^{c \times c \times d_p}$: $d_p$ represents the prompt-specific channels.
- $P_{\text{key}} \in \mathbb{R}^{H \times W \times d_p}$: Matches the spatial dimensions of the key embedding.

The new dimensions become:

$$\Phi'_{\text{query}} \in \mathbb{R}^{c \times c \times (d+d_p)}$$
$$\Phi'_{\text{key}} \in \mathbb{R}^{H \times W \times (d+d_p)}$$

Updated version: $\mathcal{Q}'_{\text{place}}(\Delta\tau|\gamma_t, \mathcal{T}_{\text{pick}}) = \left(\Phi'_{\text{query}}(\gamma_t[\mathcal{T}_{\text{pick}}]) * \Phi'_{\text{key}}(\gamma_t)\right)[\Delta\tau]$

# Task Details

| Task | multi-step sequencing | unseen colors | unseen objects | language instruction |
|---|---|---|---|---|
| 1. packing-seen-google-objects-seq§ | ✓ | ✗ | ✗ | step |
| 2. packing-unseen-google-objects-seq§ | ✓ | ✓ | ✓ | step |
| 3. packing-seen-google-objects-group*§ | ✗ | ✗ | ✗ | goal |
| 4. packing-unseen-google-objects-group*§ | ✗ | ✓ | ✓ | goal |

§tasks that are commonly found in industry.
*tasks that have more than one correct sequence of actions.

- **Selected Tasks**:
  - 4 out of 10 language-conditioned tasks from the Ravens benchmark set [2].
  - All tasks involve:
    - Precise placing.
    - Multimodal placing.
- **Language Templates for Training**:
  - Language instructions are distributed evenly across three templates:
    - Template 1: `'pack all the [colors] blocks'`.
    - Template 2: `'pack all the [colors] blocks into the box'`.
    - Template 3: `'pack all the [colors] blocks into the brown box'`.

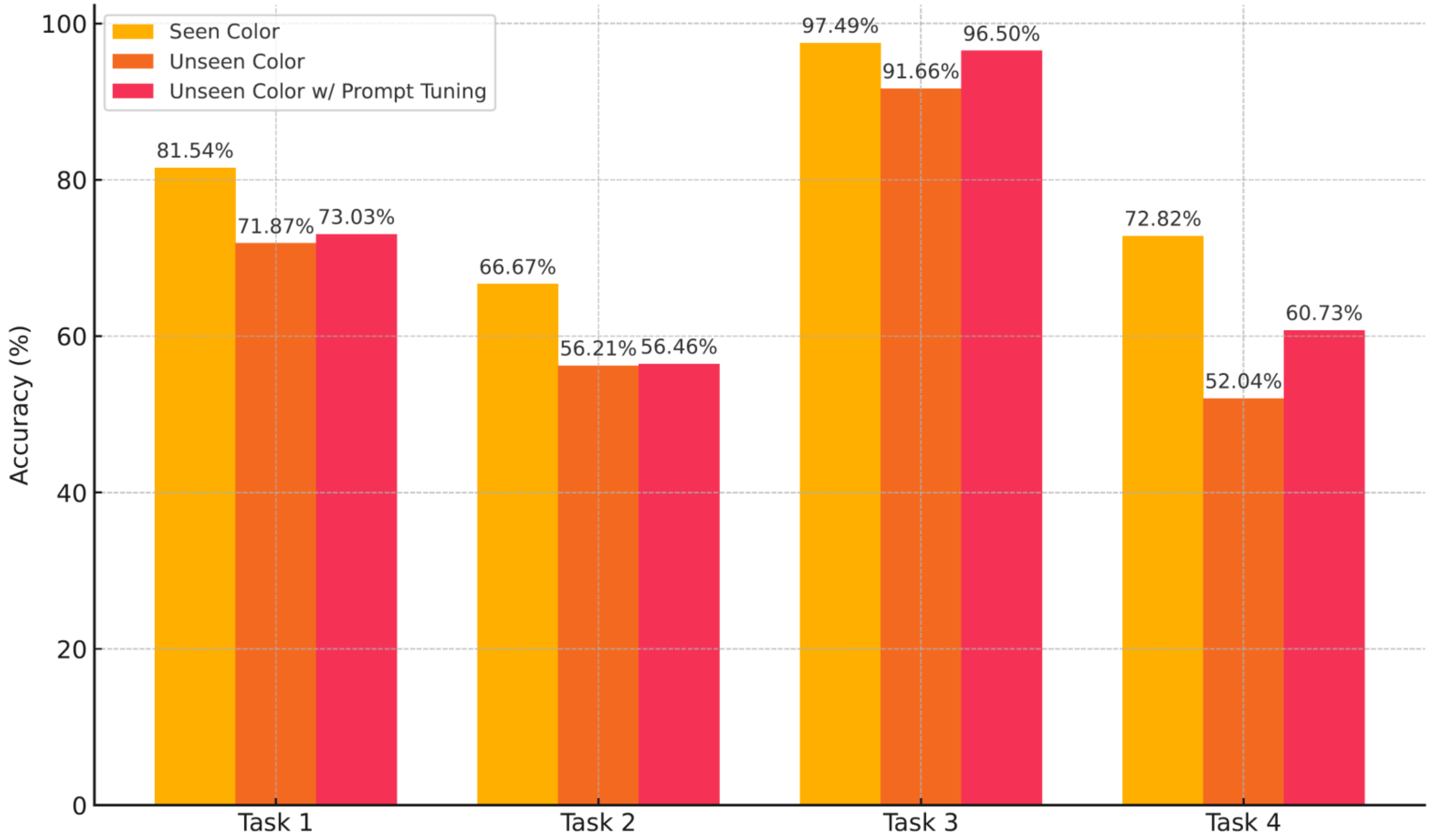
Other Training details:
Simulation environments (Ravens with PyBullet)
The foundation model is frozen, only the prompts are trained



[2] A. Zeng, P. Florence, J. Tompson et al., “Transporter networks: Rearranging the visual world for robotic manipulation,” in Proceedings of the Conference on Robot Learning, 2021, pp. 726–747.

# Results

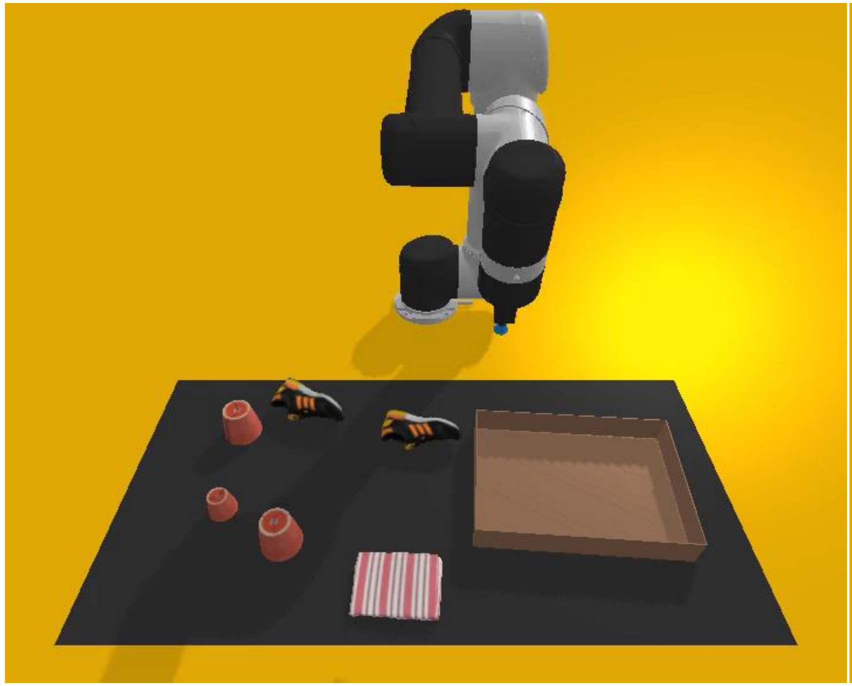
| Task | Seen Color | Unseen Color | Unseen Color w/ Prompt Tuning |
|---|---|---|---|
| **Task1** | 81.54% | 71.87% | **73.03%** |
| **Task2** | 66.67% | 56.21% | **56.46%** |
| **Task3** | 97.49% | 91.66% | **96.50%** |
| **Task4** | 72.82% | 52.04% | **60.73%** |



# Demo

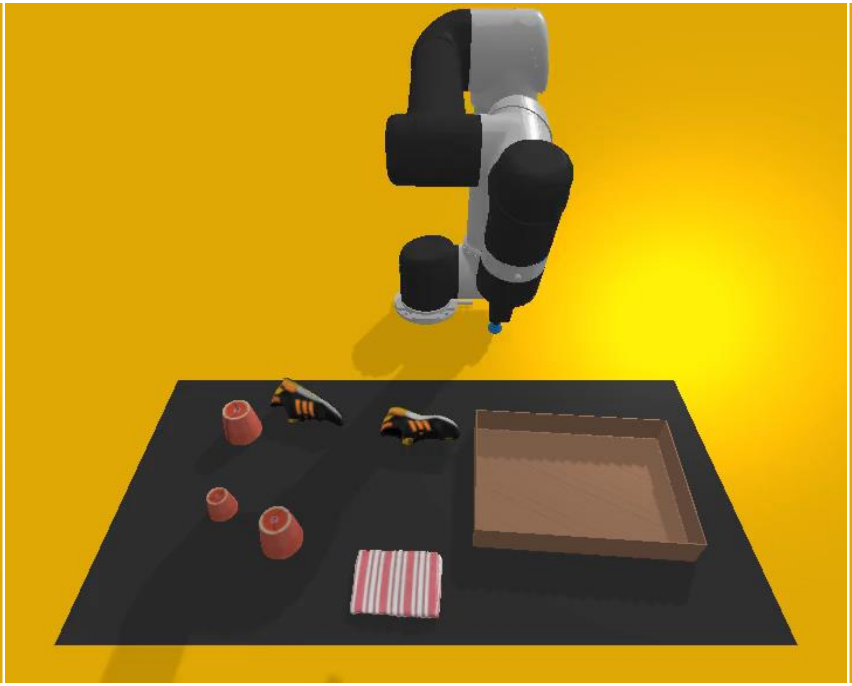
Task: Pack all the 'objects' in the brown box

Pack all the red cups in the brown box



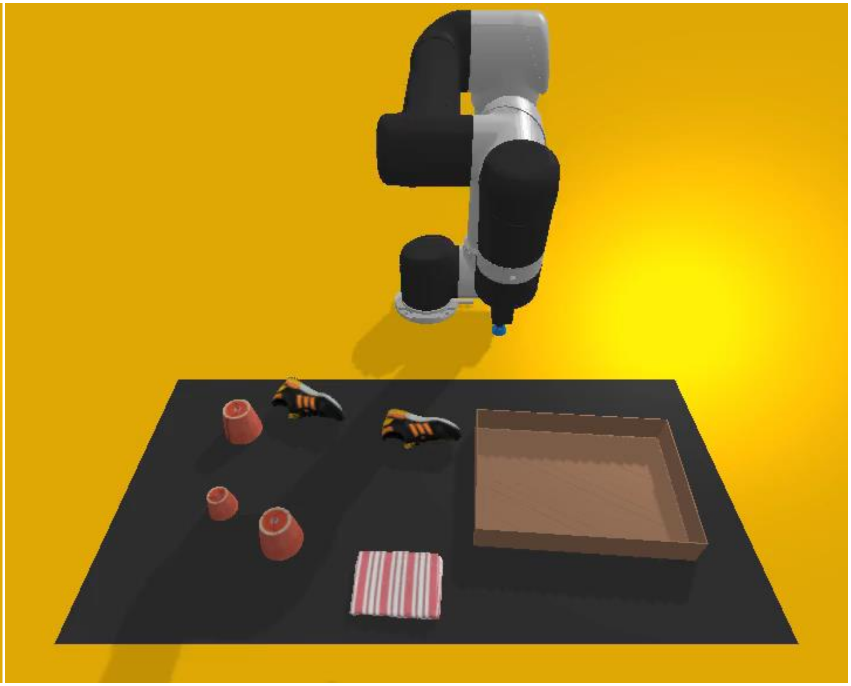
**Seen Color**

Pack all the red cups in the box



**Unseen Color**

Pack all the red cups in the box w/ Prompt Tuning



**Unseen Color w/ Prompt Tuning**

# Thanks Everyone!