CoRL 2021

CLIPort: What and Where Pathways for Robotic Manipulation

Mohit Shridhar[1], Lucas Manuelli[2], Dieter Fox[1, 2]

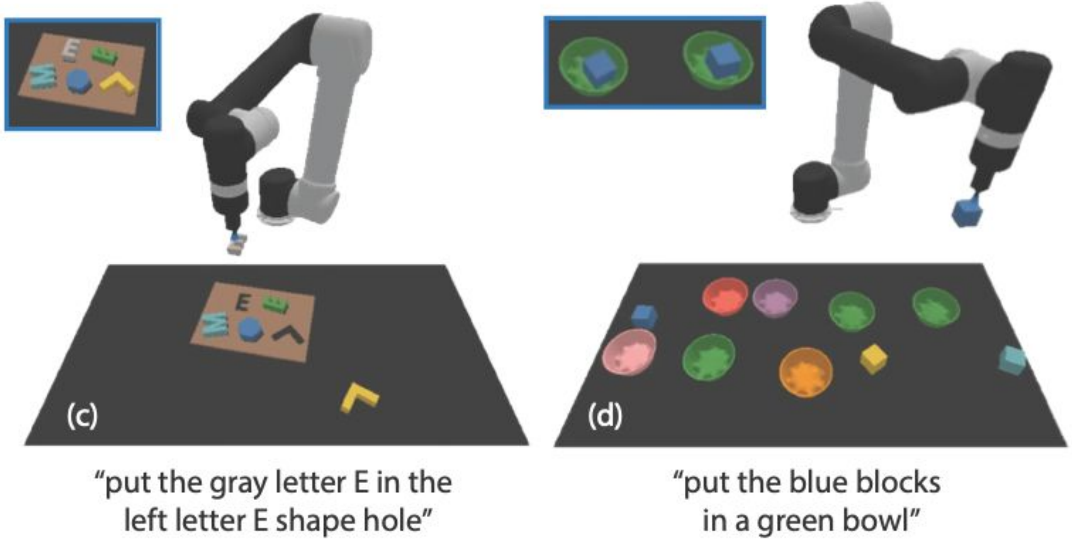
[1]University of Washington, [2]NVIDIA

Presented by Imran Ibrahimli

Contents

Motivation

Language-conditioned learning for precise robotic manipulation from demonstrations



[Source: Shridhar et al. (2021), CLIPort]

Motivation

Real-world tasks such as packing, palletizing, stacking

[Source: Shridhar et al. (2021), CLIPort]

Contributions

Grounding semantic concepts using CLIP

End-to-end with no object models, poses, segmentation

Single multi-task model

Data efficiency (few demonstrations required)

Scope

Work is NOT attempting to solve:

- Handling novel object types
- Arbitrary (out of distribution) language instructions

for which no demonstrations were given

Restricted to 2D pick/place pose prediction

Overview

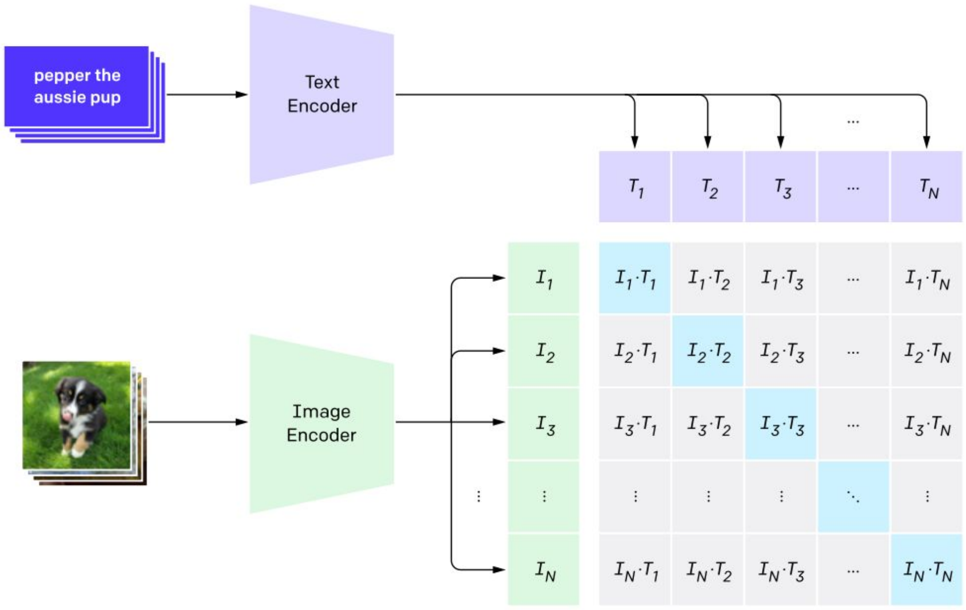
Two-stream architecture:

What (semantic) pathway: CLIP

Where (spatial) pathway: Transporter Net

Background: CLIP

Learns visual concepts from natural language supervision

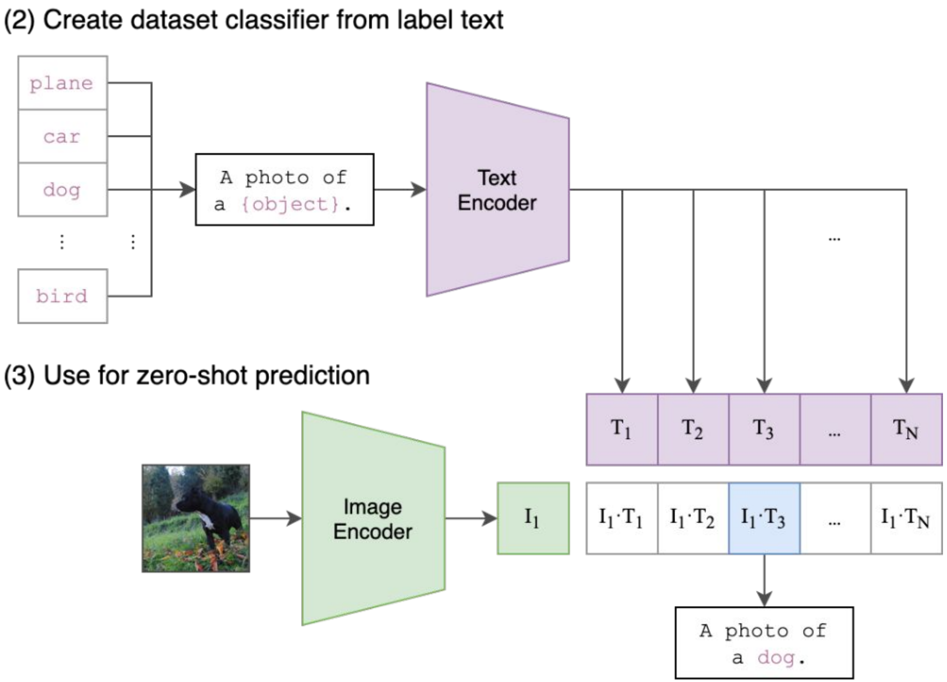


[Source: Radford et al. (2021), CLIP]

Background: CLIP

Trained on image-caption pairs scraped from the internet

Can be used for zero-shot classification & other tasks



[Source: Radford et al. (2021), CLIP]

Background: CLIP

Vision encoder: ResNet or Vision Transformer

Text encoder: CBOW or Text Transformer

Contrastive training: given an image, predict which one of these ~32K text snippets was paired with it

Background: CLIP

Pros:

Leverages massive amounts of weakly-labeled data

Zero-shot generalization to different tasks

Cons:

Bad with abstract / systematic / fine-grained tasks (e.g. counting, classifying car model)

Need to provide choices / classes (unlike image captioning)

Background: Transporter

Rearrange deep features to infer spatial displacements from visual input

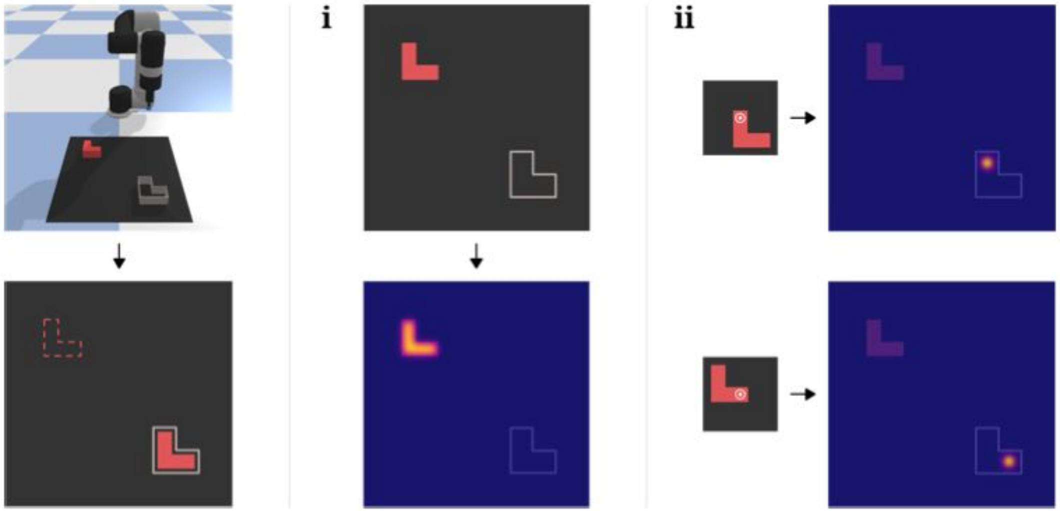


Figure 2. Simple planar pick-and-place task where (i) there is a distribution of successful pick poses, and (ii) for each successful pick pose, there is a corresponding distribution of successful place poses.

[Source: Zeng et al. (2020), Transporter Networks]

Background: Transporter

Problem decomposed into

- Picking $\mathcal{T}_{\text{pick}} = \underset{(u,v)}{\text{argmax}}\ \mathcal{Q}_{\text{pick}}((u,v)|\mathbf{o}_t)$
- Pick-conditioned placing $\mathcal{T}_{\text{place}} = \underset{\{\tau_i\}}{\text{argmax}}\ \mathcal{Q}_{\text{place}}(\tau_i|\mathbf{o}_t, \mathcal{T}_{\text{pick}})$

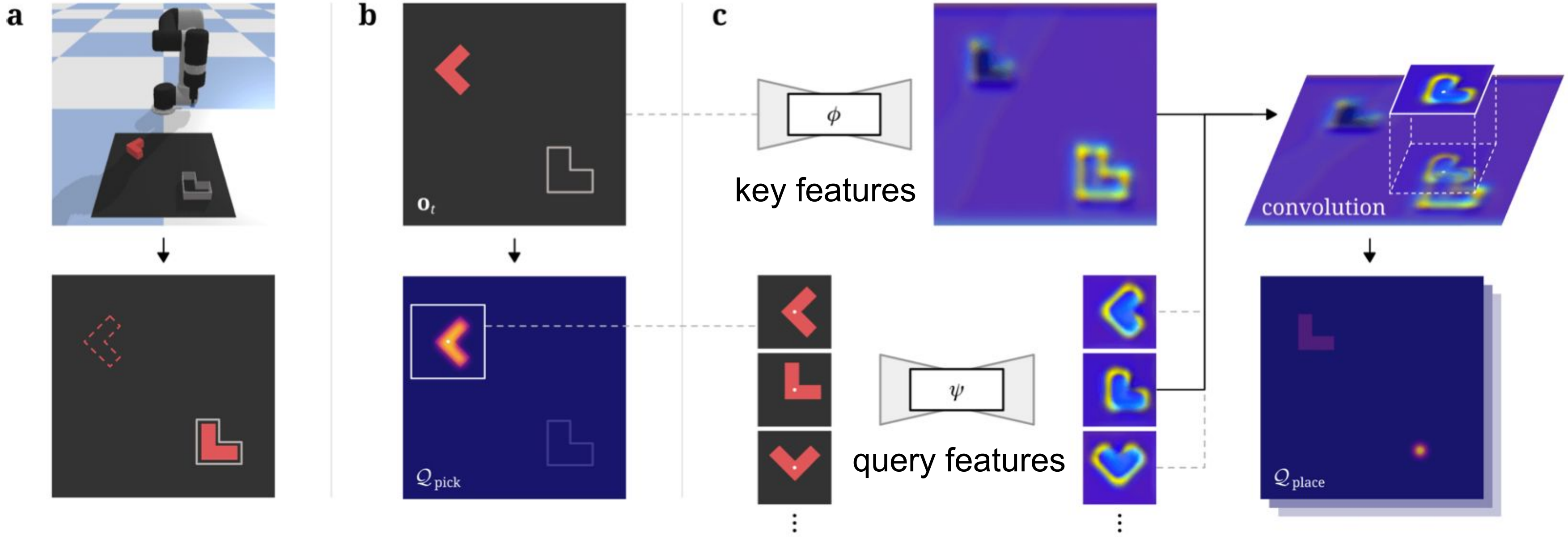
Where o_t is the observation (RGB-D image) and $\mathcal{T}_{\text{pick}}, \mathcal{T}_{\text{place}} \in \mathbf{SE}(2)$

Pick model is an encoder-decoder 43-layer ResNet

Place model has the same architecture as pick model, but outputs 2 feature maps (key & query)

Background: Transporter

Cross-entropy on pick and place one-hot encodings



[Source: Zeng et al. (2020), Transporter Networks]

Background: Transporter

Pros:

No object-centric representations

Generalization to unseen objects

Cons:

Sensitive to noise & camera-robot calibration

Restricted actions defined by 2D keypoints

Background: Transporter

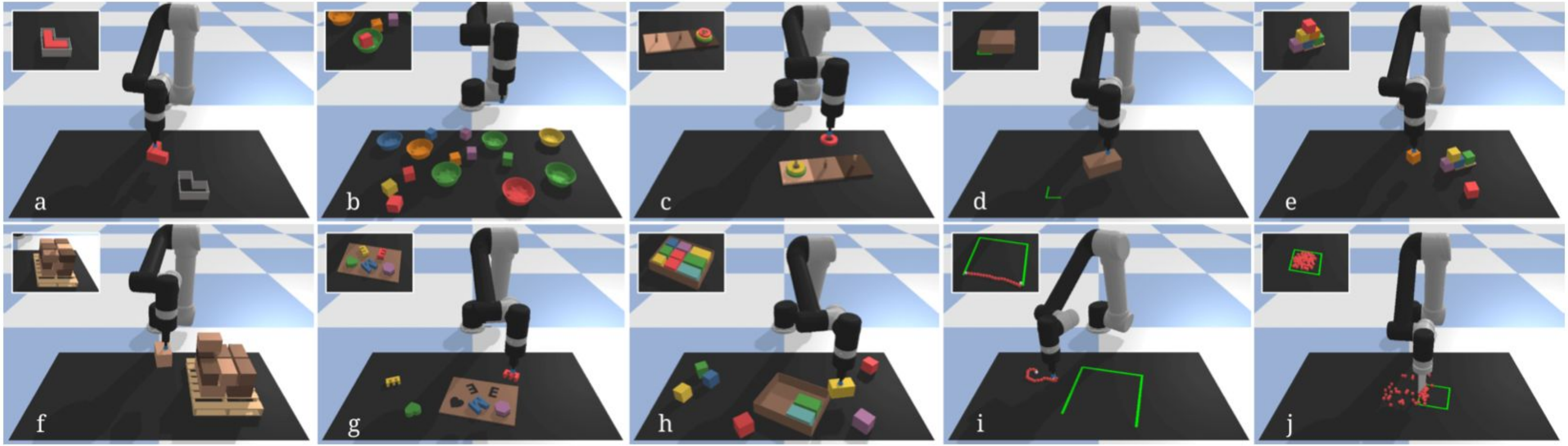
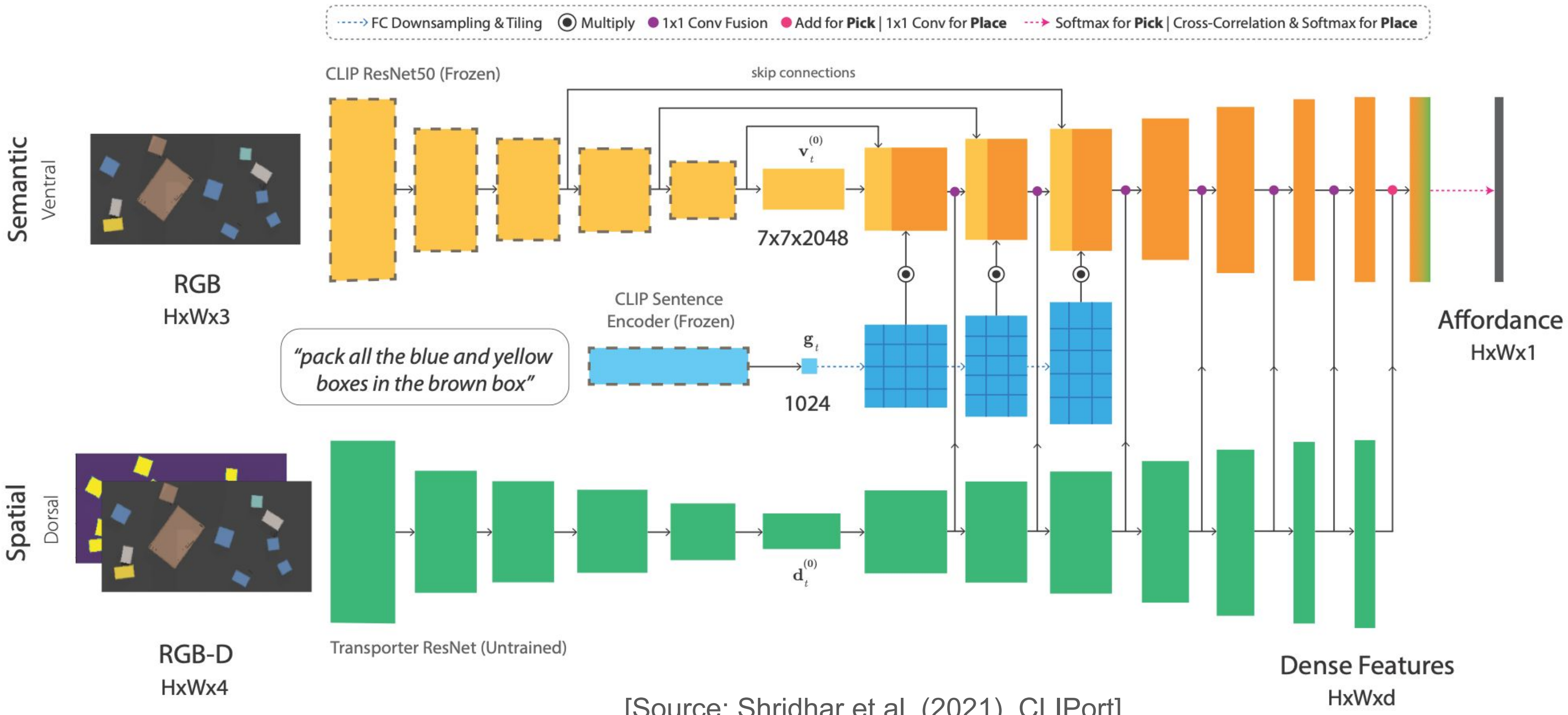


Figure 6. **Tasks:** (row-major order) block-insertion, place-red-in-green, towers-of-hanoi, align-box-corner, stack-block-pyramid, palletizing-boxes, assembling-kits, packing-boxes, manipulating-rope, sweeping-piles. Goal states (not provided to learners) are shown in top left corner of each image.

[Source: Zeng et al. (2020), Transporter Networks]

CLIPort: Architecture



[Source: Shridhar et al. (2021), CLIPort]

CLIPort: Training

ζ = zeta

Given: a set of expert demonstrations $D = \{\zeta_1, \zeta_2, \ldots, \zeta_n\}$

consisting of discrete time input-action pairs

$$\zeta_i = \{(o_1, l_1, a_1), (o_2, l_2, a_2), \ldots\}$$

o_t : observation, RGB-D image

l_t : language instruction

a_t : action = $(\mathcal{T}_{\text{pick}}, \mathcal{T}_{\text{place}})$ such that $\mathcal{T}_{\text{pick}}, \mathcal{T}_{\text{place}} \in \mathbf{SE}(2)$

CLIPort: Training

k is for rotations (k=36)

One-hot pixel encode Y_{pick} and Y_{place} (shape = H x W x k)

Cross entropy loss $\mathcal{L} = -\mathbb{E}_{Y_{\text{pick}}}[\log \mathcal{V}_{\text{pick}}] - \mathbb{E}_{Y_{\text{place}}}[\log \mathcal{V}_{\text{place}}]$

$$\mathcal{V}_{\text{pick}} = \text{softmax}(\mathcal{Q}_{\text{pick}}((u, v)|\gamma_t))$$

(u, v) = pixel-space coordinate

$$\mathcal{V}_{\text{place}} = \text{softmax}(\mathcal{Q}_{\text{place}}((u', v', \omega')|\gamma_t, \mathcal{T}_{pick}))$$

$$\mathcal{T}_{\text{pick}} = \underset{(u,v)}{\text{argmax}}\, \mathcal{Q}_{\text{pick}}((u, v)|\gamma_t)$$

action $\in$ SE(2)

$$\gamma_t = (\mathbf{o}_t, \mathbf{l}_t)$$

Observation & language

CLIPort: Prediction examples

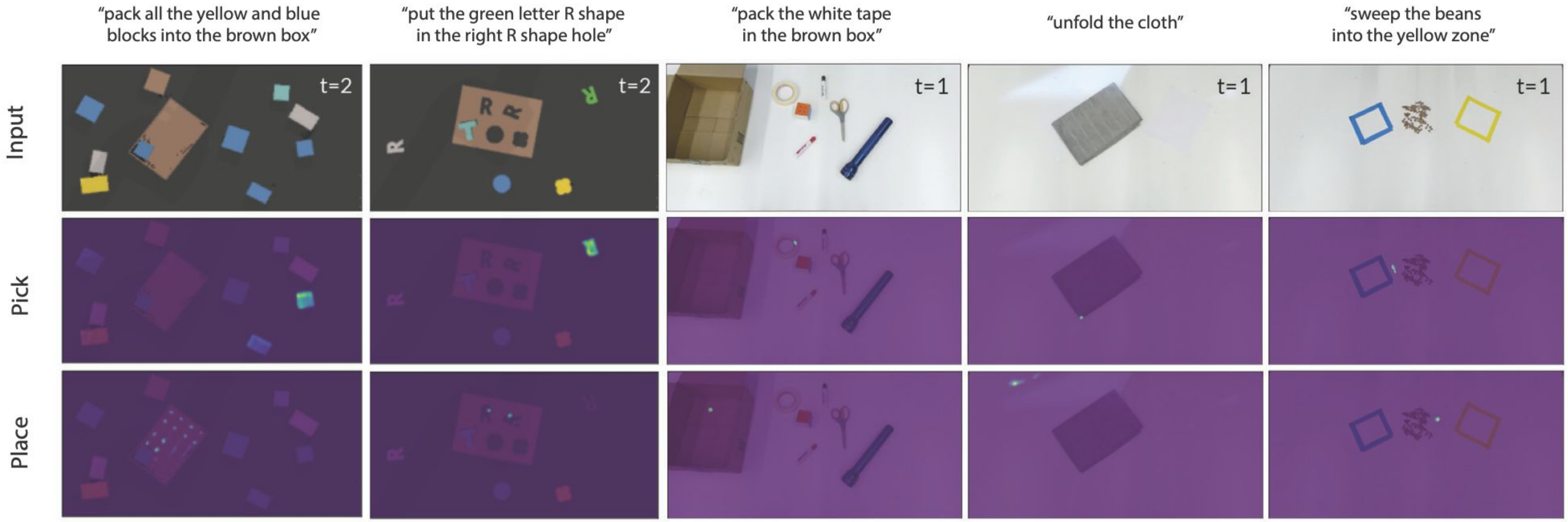


Figure 4. Affordance predictions from CLIPORT (multi) models in sim (left two) and real settings (right three). More examples in Appendix H.

[Source: Shridhar et al. (2021), CLIPort]

CLIPort: Experiments in sim

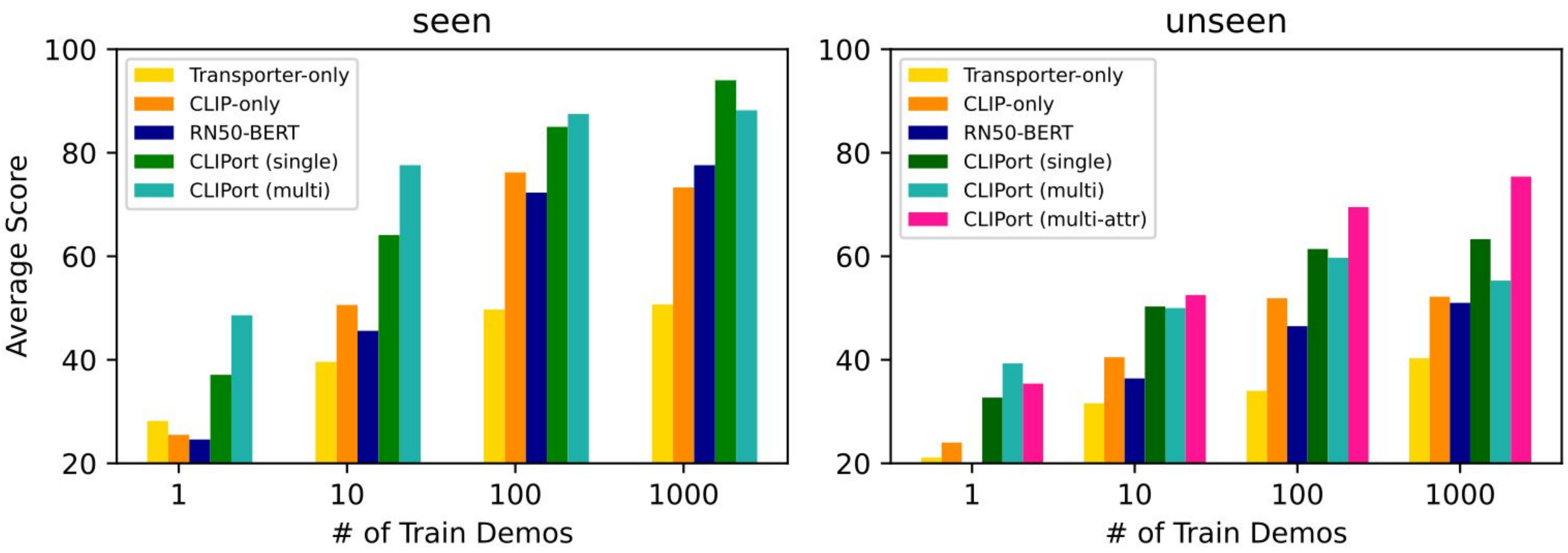
UR5e with a suction gripper

RGB-D reconstructed from 3 cameras (640x480)

Ravens benchmark from PyBullet (extended by 10 language -conditioned tasks) with an **oracle**

Evaluation based on 0 to 100 score (partial credit)

CLIPort: Results (simulation)



[Source: Shridhar et al. (2021), CLIPort]

CLIPort: Experiments on real robot

Franka Panda with parallel gripper

Kinect2 RGB-D Camera

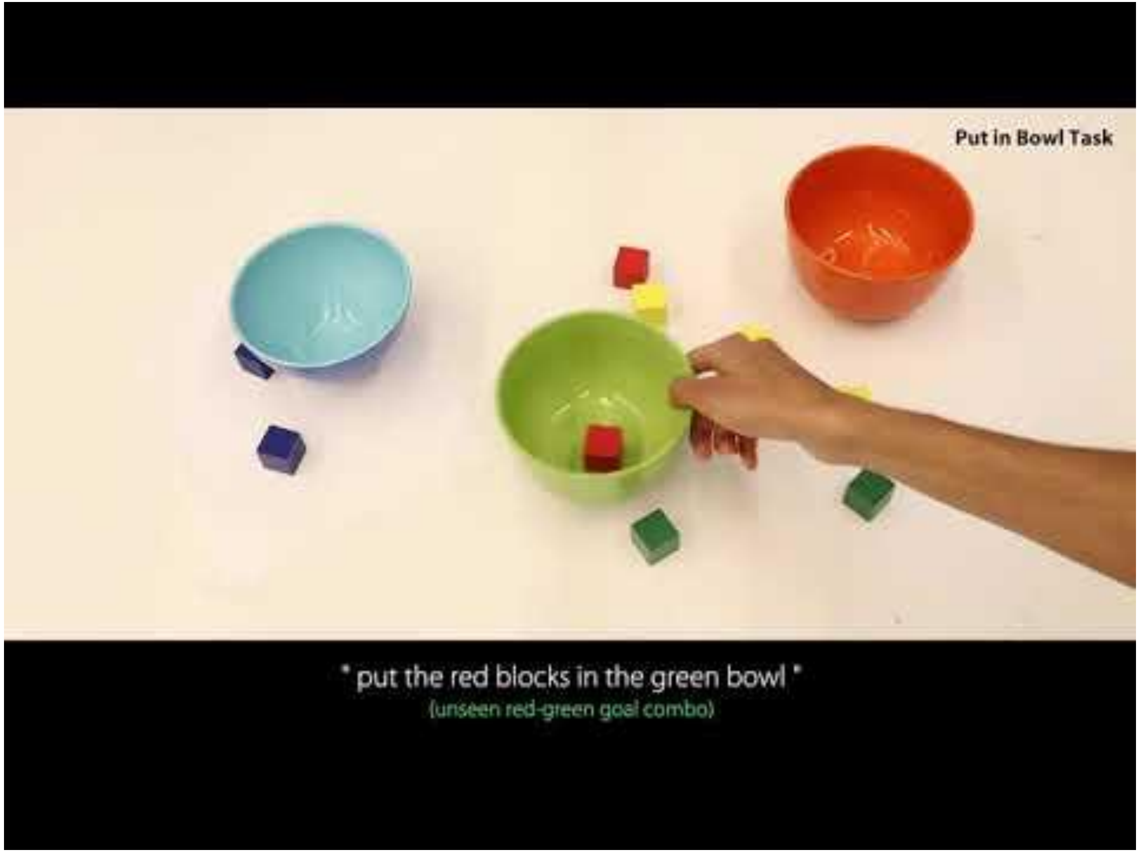
5-10 demos for training, **5-10** test runs per task

Predict one out of **36** rotations for pick too (unlike simulation)

Figure 8. Real-Robot Experimental Setup.

[Source: Shridhar et al. (2021), CLIPort]

CLIPort: Video



CLIPort: Results (real robot)

Task	# Train (Samples)	# Test	Succ. %
Stack Blocks	5 (13)	10	70.0
Put Blocks in Bowl	5 (10)	10	65.0
Pack Objects	10 (31)	10	60.0
Move Rook	4 (29)	10	70.0
Fold Cloth	9 (9)	10	57.0
Read Text	2 (26)	10	55.0
Loop Rope	4 (12)	10	60.0
Sweep Beans	5 (23)	5	60.6
Pick Cherries	4 (26)	5	75.0

Table 2. Success rates (%) of a multi-task model trained an evaluated 9 real-world tasks (see Figure 1). Samples indicate total image-action pairs, e.g 1 in Figure 9.

[Source: Shridhar et al. (2021), CLIPort]

CLIPort: Limitations

Need for balanced datasets (exploiting biases)

Sensitive to hand-eye calib (due to action space being 2D+rotation)

Limited to SE(2) poses for pick/place

Limited to simpler object relations (‘on’, ‘in’)

Relies on expert to detect task completion (& stop)

Conclusion

Semantic priors (e.g. CLIP) help data-efficient generalization

No symbolic states

No “objectness” assumptions (pose, segmentation, etc.)

Works on a real robot

Questions

Appendix A

CLIP hyperparameters

F. Model Hyperparameters

Hyperparameter	Value
Batch size	32768
Vocabulary size	49408
Training epochs	32
Maximum temperature	100.0
Weight decay	0.2
Warm-up iterations	2000
Adam β_1	0.9
Adam β_2	0.999 (ResNet), 0.98 (ViT)
Adam ϵ	10^{-8} (ResNet), 10^{-6} (ViT)

Table 18. Common CLIP hyperparameters

Model	Learning rate	Embedding dimension	Input resolution	ResNet blocks	ResNet width	Text Transformer layers	Text Transformer width	Text Transformer heads
RN50	5×10^{-4}	1024	224	(3, 4, 6, 3)	2048	12	512	8
RN101	5×10^{-4}	512	224	(3, 4, 23, 3)	2048	12	512	8
RN50x4	5×10^{-4}	640	288	(4, 6, 10, 6)	2560	12	640	10
RN50x16	4×10^{-4}	768	384	(6, 8, 18, 8)	3072	12	768	12
RN50x64	3.6×10^{-4}	1024	448	(3, 15, 36, 10)	4096	12	1024	16

Table 19. CLIP-ResNet hyperparameters

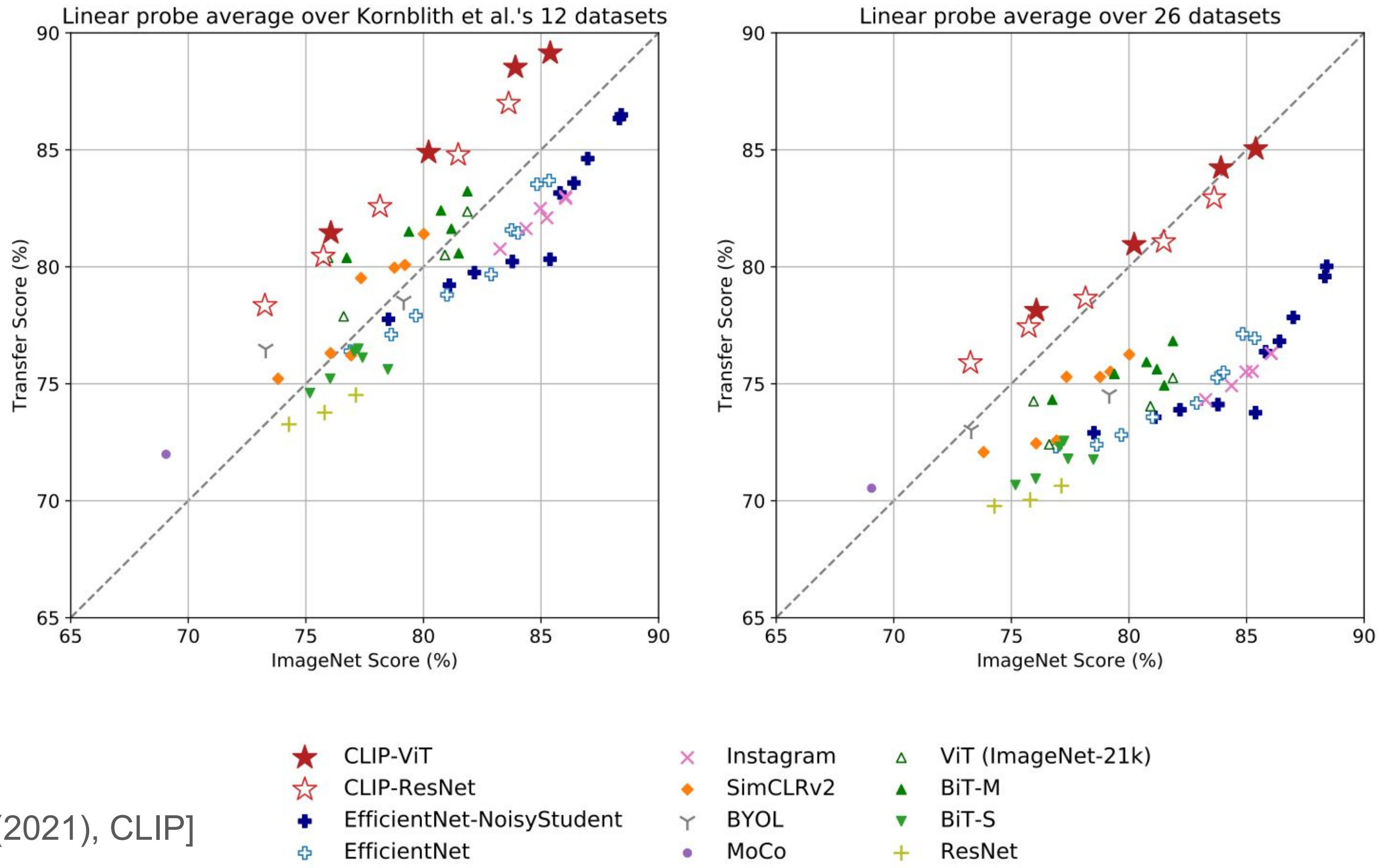
Model	Learning rate	Embedding dimension	Input resolution	Vision Transformer layers	Vision Transformer width	Vision Transformer heads	Text Transformer layers	Text Transformer width	Text Transformer heads
ViT-B/32	5×10^{-4}	512	224	12	768	12	12	512	8
ViT-B/16	5×10^{-4}	512	224	12	768	12	12	512	8
ViT-L/14	4×10^{-4}	768	224	24	1024	16	12	768	12
ViT-L/14-336px	2×10^{-5}	768	336	24	1024	16	12	768	12

Table 20. CLIP-ViT hyperparameters

[Source: Radford et al. (2021), CLIP]

Appendix B

CLIP results



[Source: Radford et al. (2021), CLIP]

Appendix C

Task	precise placing	multimodal placing	multi-step sequencing	unseen poses	unseen colors	unseen objects	language instruction
put-blocks-in-bowls-seen-colors*	✗	✓	✗	✓	✗	✗	goal
put-blocks-in-bowls-unseen-colors*	✗	✓	✗	✓	✓	✗	goal
assembling-kits-seq-seen-colors	✓	✓	✓	✓	✗	✓	step
assembling-kits-seq-unseen-colors	✓	✓	✓	✓	✓	✓	step
packing-unseen-shapes	✗	✓	✗	✓	✓	✓	goal
stack-block-pyramid-seq-seen-colors	✓	✓	✓	✓	✗	✗	step
stack-block-pyramid-seq-unseen-colors	✓	✓	✓	✓	✓	✗	step
towers-of-hanoi-seq-seen-colors	✓	✓	✓	✓	✗	✗	step
towers-of-hanoi-seq-unseen-colors	✓	✓	✓	✓	✓	✗	step
packing-box-pairs-seen-colors*§	✓	✓	✓	✓	✗	✓	goal
packing-box-pairs-unseen-colors*§	✓	✓	✓	✓	✓	✓	goal
packing-seen-google-objects-seq§	✗	✓	✓	✓	✗	✗	step
packing-unseen-google-objects-seq§	✗	✓	✓	✓	✓	✓	step
packing-seen-google-objects-group*§	✗	✓	✗	✓	✗	✗	goal
packing-unseen-google-objects-group*§	✗	✓	✗	✓	✓	✓	goal
align-rope*†	✓	✓	✓	✓	✗	✗	goal
separating-piles-seen-colors*†	✓	✓	✓	✓	✗	✗	goal
separating-piles-unseen-colors*†	✓	✓	✓	✓	✓	✗	goal

§ tasks that are commonly found in industry.
* tasks that have more than one correct sequence of actions.
† tasks that require manipulating deformable objects and granular media.

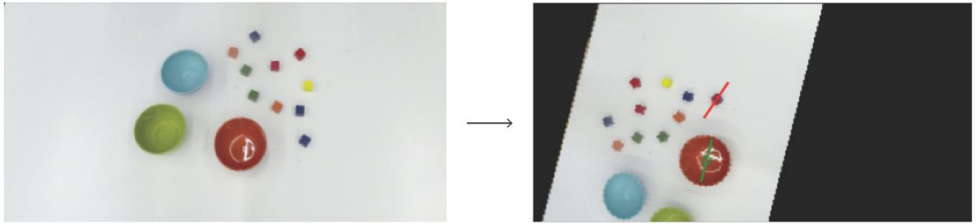


Figure 9. **Data Augmentation**: $\mathbf{SE}(2)$ transform applied to RGB-D input. The left image shows the original input, and the right image shows the transformed input along with expert $\mathcal{T}_{\text{pick}}$ (red) and $\mathcal{T}_{\text{place}}$ (green) actions.

[Source: Shridhar et al. (2021), CLIPort]

Appendix D

CLIPort results

[Source: Shridhar et al. (2021), CLIPort]

	packing-box-pairs seen-colors				packing-box-pairs unseen-colors				packing-seen-google objects-seq				packing-unseen-google objects-seq				packing-seen-google objects-group				packing-unseen-google objects-group			
Method	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000
Transporter-only [2]	44.2	55.2	54.2	52.4	34.6	48.7	47.2	54.1	26.2	39.7	45.4	46.3	19.9	29.8	28.7	37.3	60.0	54.3	61.5	59.9	46.2	54.7	49.8	52.0
CLIP-only	38.6	69.7	88.5	87.1	33.0	65.5	68.8	61.2	29.1	67.9	**89.3**	95.8	37.1	49.4	60.4	57.8	52.5	62.0	**89.6**	**92.7**	43.4	65.9	73.1	70.0
RN50-BERT	36.2	64.0	**94.7**	90.3	31.4	52.7	65.6	**72.1**	32.9	48.4	87.9	94.0	29.3	48.5	48.3	56.1	46.4	52.9	76.5	86.4	43.2	52.0	66.3	73.7
CLIPORT (single)	51.6	82.9	92.7	**98.2**	45.6	65.3	68.6	71.5	14.8	59.5	86.8	**96.2**	27.2	50.0	65.5	**71.9**	52.7	67.0	84.1	94.0	61.5	66.2	78.4	**81.5**
CLIPORT (multi)	**66.8**	**88.6**	94.1	96.6	**59.0**	**69.7**	**76.2**	71.4	**41.6**	**78.4**	85.0	84.4	**40.7**	**51.1**	**65.8**	70.3	**71.3**	**84.6**	**89.6**	88.3	**68.4**	**69.6**	**78.4**	80.3
CLIPORT (multi-attr)	–	–	–	–	*46.2*	*72.0*	*86.2*	*80.3*	–	–	–	–	*35.4*	*45.1*	*78.9*	*87.4*	–	–	–	–	*48.6*	*69.3*	*84.8*	*89.1*

	stack-block-pyramid seq-seen-colors				stack-block-pyramid seq-unseen-colors				separating-piles seen-colors				separating-piles unseen-colors				towers-of-hanoi seq-seen-colors				towers-of-hanoi seq-unseen-colors			
	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000
Transporter-only [2]	4.5	2.3	5.2	4.5	3.0	4.0	2.3	5.8	42.7	52.3	42.0	48.4	41.2	49.2	44.7	52.3	25.4	67.9	98.0	99.9	24.3	44.6	71.7	80.7
CLIP-only	6.3	28.7	55.7	54.8	2.0	12.2	18.3	19.5	43.5	55.0	84.9	90.2	**59.9**	49.6	73.0	71.0	9.4	52.6	88.6	45.3	24.7	47.0	67.0	58.0
RN50-BERT	5.3	35.0	89.0	97.5	6.2	12.2	21.5	30.7	31.8	47.8	46.5	46.5	33.4	44.4	41.3	44.9	28.0	66.1	91.3	92.1	17.4	75.1	85.3	89.3
CLIPORT (single)	28.3	64.7	93.3	**98.8**	13.7	24.3	31.2	**41.3**	**54.5**	59.5	**93.1**	**98.0**	47.2	51.0	**76.6**	**75.2**	59.4	92.9	97.4	**100**	56.1	**89.7**	**95.9**	**99.4**
CLIPORT (multi)	**33.5**	**75.3**	**96.8**	96.5	**23.3**	**26.8**	**31.7**	22.2	48.9	**72.4**	90.3	89.0	56.6	**62.6**	64.9	62.8	**61.6**	**96.3**	**98.7**	98.1	**60.1**	65.6	76.7	68.7
CLIPORT (multi-attr)	–	–	–	–	*15.5*	*51.5*	*59.3*	*79.8*	–	–	–	–	*49.9*	*51.8*	*48.2*	*59.8*	–	–	–	–	*56.7*	*78.0*	*88.3*	*96.9*

	align-rope				packing-unseen-shapes				assembling-kits-seq seen-colors				assembling-kits-seq unseen-colors				put-blocks-in-bowls seen-colors				put-blocks-in-bowls unseen-colors			
	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000
Transporter-only [2]	6.9	30.6	33.1	51.5	16.0	20.0	22.0	22.0	5.8	11.6	28.6	29.6	7.8	17.6	25.6	28.4	16.8	33.3	62.7	64.7	11.7	17.2	14.8	18.7
CLIP-only	13.4	48.7	70.4	70.7	13.0	28.0	**44.0**	**50.0**	0.8	9.2	19.8	23.0	2.0	4.6	10.8	19.8	23.5	60.2	93.5	97.7	11.2	34.2	33.2	44.5
RN50-BERT	3.1	25.0	63.8	57.1	19.0	25.0	32.0	44.0	2.2	5.6	11.6	21.8	1.6	6.4	10.4	18.4	13.8	44.5	81.2	91.8	16.2	23.0	30.3	23.8
CLIPORT (single)	**20.1**	**77.4**	**85.6**	**95.4**	21.0	26.0	40.0	37.0	**12.2**	17.8	**47.0**	**66.6**	**16.2**	18.0	**35.4**	**34.8**	23.5	68.3	92.5	**100**	18.0	35.3	37.3	25.0
CLIPORT (multi)	19.6	49.3	82.4	74.9	**25.0**	**35.0**	37.0	31.0	11.4	**34.8**	46.2	52.4	7.8	**21.6**	29.0	25.4	**54.0**	**90.2**	**99.5**	**100**	**32.0**	**48.8**	**55.3**	**45.8**
CLIPORT (multi-attr)	–	–	–	–	–	–	–	–	–	–	–	–	*7.6*	*10.4*	*43.8*	*34.6*	–	–	–	–	*23.0*	*41.8*	*66.5*	*75.7*

Appendix E

CLIPort ablations

[Source: Shridhar et al. (2021), CLIPort]

	stack-block-pyramid seq-seen-colors				stack-block-pyramid seq-unseen-colors				packing-seen-google object-seq				packing-unseen-google object-seq			
Method	1	10	100	1000	1	10	100	1000	1	10	100	1000	1	10	100	1000
One-Stream Transporter-only	4.5	2.3	5.2	4.5	3.0	4.0	2.3	5.8	26.2	39.7	45.4	46.3	19.9	29.8	28.7	37.3
One-Stream CLIP-only	6.3	28.7	55.7	54.8	2.0	12.2	18.3	19.5	52.5	62.0	**89.6**	92.7	43.4	**65.9**	**73.1**	70.0
One-Stream Language Transporter	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.1	0.1	0.0	0.0
One-Stream Image-Goal Transporter	1.8	1.3	7.0	6.8	2.5	4.7	4.2	4.8	**64.5**	**67.0**	81.8	85.4	**47.7**	62.8	71.0	**83.3**
Two-Stream CLIP-Transporter w/o skips	0.0	4.3	3.8	3.3	4.2	5.2	3.2	2.5	22.9	26.1	36.9	38.9	24.4	29.9	33.7	38.3
Two-Stream Untrained-Sem-Transporter	3.0	12.7	61.5	51.2	1.0	6.8	17.2	15.7	28.8	40.5	67.1	79.7	27.2	34.7	33.0	34.8
Two-Stream RN50-BERT-Transporter	5.3	35.0	89.0	97.5	6.2	12.2	21.5	30.7	32.9	48.4	87.9	94.0	29.3	48.5	48.3	56.1
Two-Stream CLIP-Transporter (ours)	**28.3**	**64.7**	**93.3**	**98.8**	**13.7**	**24.3**	**31.2**	**41.3**	14.8	59.5	86.8	**96.2**	27.2	50.0	65.5	71.9

Table 5. **Ablations and Baselines.** Evaluation scores (mean %) for stack-block-pyramid-seq and packing-google-objects-seq tasks from 100 evaluation runs. Stacking block pyramids involves both semantic and precise spatial reasoning, whereas packing objects mostly involves semantic grounding without requiring any precise placements.