

SORNet: Spatial Object-Centric Representations for Sequential Manipulation Dieter Fox^{1,2} Chris Paxton² Karthik Desingh¹ Wentao Yuan¹ ¹University of Washington ²NVIDIA Results Approach **Object Embedding RGB** Image Logical States in_front_of(*blue_block*, *red_block*) stacked(*blue_block*, *green_block*) mbedding Network **Readout Networks** left_of (red_block, yell Object embeddings **Continuous States** in_front_of(*blue_block*, *red_block*) stacked(blue_block, green_block) Spatial relations Canonical Object Views approach(robot, red_block) = True **Canonical Object Views** has_obj(robot, red_block) = False a 📰 📕 🔽 top_clear(*red_block*) = True Objects in the input image **Skill Preconditions Embedding Network** 👔 🧈 Training (Co Permutation-invariant object embedding Cubes ar Cylinders **Relative Direction** • Spheres **Transformer Encoder** Positional Encoding \longrightarrow Testing objects Linear Projection of Flattened Patches **Predicate Classification on Leonardo Canonical Object Views Context Patches** Training split Overall 405 colored blocks. **Learned Attention** One task - stacking 4 blocks. ➤ 133796 sequences. **Testing split** 7 colored blocks (unseen in train) 7 tasks different from training Testing objective (continuous) ➢ 9526 sequences. Objective **Readout Networks** Number of outputs changes adaptively with number of inputs F-1 Score Example: stacked(blue, green) stacked(blue, red)

Overview

We propose **SORNet**: Spatial Object-centric Representation Network to learn object-centric embeddings that encode spatial relationships



SORNet generalizes **zero-shot** to scenes with unseen objects and different number of objects.

Training objects

> SORNet is trained only on classification of logical predicates, but captures **continuous** spatial relationships.

Training objective (logical)



ned_with(cobalt_block, ruby_block) has_obj(robot, cobalt_block) n_approach_region(robot, ruby_block) on_surface(lavender_block, right) on_surface(peach_block, right) on_surface(ruby_block, left) top_is_clear(cobalt_block) top_is_clear(lavender_block) top_is_clear(peach_block) top_is_clear(ruby_block)



Object-object direction

EE-object direction

Binary MLP M_B x 2-layer MLPs binary obje embedding







Spatial Relation Prediction on CLEVR-CoGenT										
Inp	out		Output							
GB frame	Canonical Object Views	S	Spatial Relations							
		SORNet	left_of(e, for a second secon							
ondition A)		Testing (Condition	ר B)							
re gray, blue, brown , or yellow • Cubes are red , green , purple , or cyan										
rs are red , green ,	, purple , or <mark>cyan</mark>	• Cylinders are g	ray, blue, brown, or yellow							
can have any co	olor	 Spheres can had 	ive any color							
	MDETR [34]	MDETR-oracle [34]	SORNet(ours)							
ValA Accuracy	84.950	97.944	99.006							
ValB Accuracy	59.627	98.052	98.222							
	Zero-sh	ot Accuracy								



Method	# pred	all	on_surface	has_obj	top_clear	stacked	aligned	approach
ResNet18 M-Head 100-shot	52	0.0	21.9	0.0	32.6	0.0	0.0	0.0
ViT-B/32 M-View 100-shot	52	0.0	37.7	6.3	46.5	0.0	0.0	7.3
ViT-B/32 M-Head M-View 100-shot	52	0.0	70.5	31.0	73.2	27.2	0.0	23.2
SORNet 0-shot	52	83.2	92.2	79.7	93.0	91.2	63.8	74.9
SORNet M-View 0-shot	52	88.9	97.5	82.0	98.4	97.3	70.5	81.7
SORNet M-View (G) 0-shot	52	89.5	97.1	94.7	96.8	96.4	69.9	76.7
SORNet M-View (G) 5 obj 0-shot	70	85.3	96.0	96.7	91.3	83.6	69.8	78.1
SORNet M-View (G) 6 obj 0-shot	102	79.9	95.5	97.0	87.5	69.2	70.0	77.9



- Randomly chosen 4 blocks in each sequence

- Randomly chosen 4-6 blocks in each sequence

- Classifying logical predicates from RGB input