Appendix

A Individual task success rate of different methods

In this section, we support all experimental results in Sec. 4 of our main paper with individual task success rates for all four levels of evaluation protocol. Specifically, the results for Table 1 and the ablation on Pretraining strategies can be found in Table 3, 4, 5 and 6. Figure 7 that ablates multimodal prompt encoding is based on the results from Table 7, 8, 9 and 10. The results in Figure 8 that ablate model and data sizes are based on the results from Table 11, 12, 13, 14, 15, 16, 17, and 18.

Additionally, we further conduct an ablation study on the transformer architecture of our policy by replacing the decoder-only architecture with encoder-decoder architecture (Our Method w/ Encoder-Decoder). Experimental results in Table 3, 4, 5 and 6 show that this variant does not perform as well as our method on the L1, L2, and L3 tasks, mainly due to its inability to tackle Task 09 (*Twist*) that requires deducting rotation angles from the prompt image sequence (Figure 2). However, it achieves a superior performance on the L4 Task 10 (*Follow Motion*). We hypothesize that it is due to the limit of model capacity. This policy learns a control policy that predicts its action dependent on the object bounding box, while lacking the capability to capture fine-grained visual information that contains the information of object rotation.

Table 3: L1 level generalization results. All methods share the same amount of parameters 92M. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T12	T15	T16	T17	Overall
VIMA	100	100	100	100	65.0	99.5	100	13.5	96.0	94.5	100	88.0	77.0	87.2
Gato OBJ	100	100	100	100	75.5	100	100	16.5	88.5	93.0	100	92.5	80.0	88.2
Our Method														
w/o Pretrain	100	100	100	98.5	88.0	100	100	20.5	100	94.0	99.0	93.0	98.0	91.6
w/ Pretrain	98.5	100	100	99.5	94.0	100	100	100	100	94.0	95.5	94.0	96.5	97.8
w/ Masked Pretrain	100	99.5	100	99.5	97.5	99.5	100	17.5	74.5	94.5	97.0	42.5	96.5	86.0
w/ Encoder-Decoder	100	100	99.5	99.5	96.5	100	100	19.5	99.5	93.5	99.0	93.0	82.5	91.0

Table 4: L2 level generalization results. All methods share the same amount of parameters 92M. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023).

Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T12	T15	T16	T17	Overall
VIMA	100	100	100	99.5	61.0	100	100	12.5	97.5	95.0	100	87.5	77.5	87.0
Gato OBJ	100	100	100	100	73.0	100	100	17.5	88.5	95.0	99.0	94.0	80.5	88.3
Our Method														
w/o Pretrain	100	100	100	99.0	87.0	100	100	23.5	100	94.0	99.5	92.0	98.0	91.8
w/ Pretrain	98.5	100	100	99.0	96.5	99.5	100	100	100	95.5	95.0	93.0	96.0	97.9
w/ Masked Pretrain	99.5	100	100	99.5	96.5	100	99.5	19.5	75.5	95.5	97.0	43.5	96.5	86.3
w/ Encoder-Decoder	99.5	100	99.0	99.5	96.5	100	100	15.5	99.5	94.0	98.5	92.0	82.5	90.5

Table 5: L3 level generalization results. All methods share the same amount of parameters 92M. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023).

Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T15	T16	T17	Overall
VIMA	99.5	100	100	99.5	63.0	99.5	100	12.0	98.5	99.5	58.5	78.0	84.0
Gato OBJ	99.5	100	100	100	72.5	97.5	100	7.5	95.0	99.5	44.5	72.0	82.3
Our Method													
w/o Pretrain	99.5	100	100	100	90.0	100	100	20.5	100	99.5	50.5	99.5	88.3
w/ Pretrain	98.0	99.0	100	99.5	94.0	97.5	99.0	97.0	96.5	95.0	47.0	98.0	93.4
w/ Masked Pretrain	99.0	100	100	100	96.5	99.5	99.0	20.5	76.5	99.0	42.0	100	86.0
w/ Encoder-Decoder	99.0	99.5	100	98.5	95.5	99.5	98.0	20.0	100	95.0	56.0	86.0	87.2

Table 6: L4 level generalization results. All methods share the same amount of parameters 92M. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023).

Method	T8	T10	T13	T14	Overall
VIMA	98.5	0.0	0.0	100	49.6
Gato OBJ	99.5	0.0	0.0	98.0	49.4
Our Method					
w/o Pretrain	97.0	0.0	0.0	99.5	49.1
w/ Pretrain	97.5	41.0	1.0	97.0	59.1
w/ Masked Pretrain	95.0	0.0	0.0	99.0	48.5
w/ Encoder-Decoder	96.5	85.5	0.0	97.0	69.8

Table 7: Comparison of the performance of our method with different multimodal prompt encoder on L1 level generalization. All methods share the same amount of parameters 92M. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Pretrain iter.	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T12	T15	T16	T17	Overall
0.0K	T5 + RC (Ours)	100	100	100	98.5	88.0	100	100	20.5	100	94.0	99.0	93.0	98.0	91.6
0.0K	T5	100	99.5	98.5	99.0	84.5	100	100	16.0	100	94.5	98.0	49.5	97.0	87.4
0.0K	VL-T5	100	100	98.5	99.5	66.0	100	100	18.0	100	93.0	96.5	94.0	96.0	89.3
3.1K	T5 + RC (Ours)	100	100	100	98.5	85.5	100	99.0	52.0	100	93.5	96.0	93.0	96.0	93.3
3.1K	T5	100	100	100	100	98.5	100	100	18.5	100	93.5	97.0	43.5	96.5	88.3
3.1K	VL-T5	100	99.5	100	100	70.0	99.5	100	23.5	100	94.5	99.5	43.5	96.5	86.7
5.2K	T5 + RC (Ours)	100	100	100	99.0	92.0	99.5	100	99.5	100	95.5	95.0	93.0	97.0	97.7
5.2K	T5	100	100	100	98.5	98.0	100	100	19.5	100	94.0	98.5	42.0	96.5	88.2
5.2K	VL-T5	100	99.5	100	99.0	94.0	99.5	99.5	21.0	100	94.0	95.5	43.0	96.5	87.8
10.3K	T5 + RC (Ours)	98.5	100	100	99.5	94.0	100	100	100	100	94.0	95.5	94.0	96.5	97.8
10.3K	T5	99.5	99.5	100	97.0	98.0	99.0	99.5	22.0	100	94.0	99.5	41.0	97.0	88.2
10.3K	VL-T5	99.5	99.0	100	100	97.5	99.0	100	100	100	93.5	98.0	43.0	97.0	94.3

Table 8: Comparison of the performance of our method with different multimodal prompt encoder on L2 level generalization. All methods share the same amount of parameters 92M. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Pretrain iter.	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T12	T15	T16	T17	Overall
0.0K	T5 + RC (Ours)	100	100	100	99.0	87.0	100	100	23.5	100	94.0	99.5	92.0	98.0	91.8
0.0K	T5	99.5	100	99.0	99.5	87.0	100	99.5	20.0	100	93.5	98.5	48.0	95.0	87.7
0.0K	VL-T5	100	100	98.5	99.0	66.5	99.0	99.5	19.0	100	94.0	96.5	92.0	95.0	89.2
3.1K	T5 + RC (Ours)	99.5	100	100	98.5	89.5	100	99.5	52.0	100	94.0	92.5	92.0	96.0	93.3
3.1K	T5	99.0	100	100	100	97.0	99.0	99.5	22.5	100	94.5	96.0	41.5	96.5	88.1
3.1K	VL-T5	98.0	99.5	99.5	98.5	67.5	99.0	99.5	24.0	100	94.0	96.5	44.0	95.5	85.8
5.2K	T5 + RC (Ours)	100	100	100	97.5	91.0	98.5	99.5	99.5	100	95.5	93.0	92.5	96.5	97.2
5.2K	T5	98.5	100	100	98.0	96.5	99.0	98.5	21.5	100	94.0	93.5	40.0	95.0	87.3
5.2K	VL-T5	99.5	100	100	98.5	94.0	98.5	97.5	20.0	100	94.0	94.0	43.5	96.5	87.4
10.3K	T5 + RC (Ours)	98.5	100	100	99.0	96.5	99.5	100	100	100	95.5	95.0	93.0	96.0	97.9
10.3K	T5	100	100	100	97.0	97.0	100	97.0	19.5	100	94.5	97.0	43.0	95.0	87.7
10.3K	VL-T5	99.0	99.5	100	99.5	97.0	99.0	99.0	99.5	100	94.5	97.5	43.0	95.5	94.1

Table 9: Comparison of the performance of our method with different multimodal prompt encoder on L3 level generalization. All methods share the same amount of parameters 92M. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Pretrain iter.	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T15	T16	T17	Overall
0.0K	T5 + RC (Ours)	99.5	100	100	100	90.0	100	100	20.5	100	99.5	50.5	99.5	88.3
0.0K	T5	98.5	98.5	100	100	85.5	99.5	98.5	19.5	100	98.5	42.0	57.5	83.2
0.0K	VL-T5	98.5	98.5	100	100	68.5	99.5	100	21.5	100	99.0	54.5	99.5	86.6
3.1K	T5 + RC (Ours)	97.0	98.0	99.0	99.0	90.5	96.0	99.5	45.5	98.0	97.0	47.5	96.5	88.6
3.1K	T5	96.0	99.0	99.5	100	98.0	97.0	96.0	21.5	100	95.5	42.0	99.5	87.0
3.1K	VL-T5	96.5	97.0	99.5	99.5	69.0	94.5	95.0	21.0	99.5	96.5	42.0	100	84.2
5.2K	T5 + RC (Ours)	99.5	99.0	100	99.0	93.0	98.0	99.0	98.0	98.0	95.5	46.0	97.0	93.5
5.2K	T5	96.5	96.0	99.5	100	97.5	98.5	97.0	17.5	100	97.0	38.5	100	86.5
5.2K	VL-T5	96.0	98.5	99.5	100	96.5	95.5	95.5	21.0	100	98.0	41.5	100	86.8
10.3K	T5 + RC (Ours)	98.0	99.0	100	99.5	94.0	97.5	99.0	97.0	96.5	95.0	47.0	98.0	93.4
10.3K	T5	99.0	97.5	100	99.5	96.5	96.0	95.0	15.5	100	95.5	43.5	99.5	86.5
10.3K	VL-T5	98.0	97.0	100	99.5	96.0	97.0	96.5	84.5	100	99.5	41.0	99.5	92.4

Table 10: Comparison of the performance of our method with different multimodal prompt encoder on L4 level generalization. All methods share the same amount of parameters 92M. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Pretrain iter.	Method	T8	T10	T13	T14	Overall
0.0K	T5 + RC (Ours)	97.0	0.0	0.0	99.5	49.1
0.0K	T5	95.0	0.0	0.0	98.5	48.4
0.0K	VL-T5	99.0	0.0	0.0	97.5	49.1
3.1K	T5 + RC (Ours)	97.5	12.5	0.0	98.5	52.1
3.1K	T5	98.0	45.0	0.0	95.5	59.6
3.1K	VL-T5	98.5	64.0	0.0	96.0	64.6
5.2K	T5 + RC (Ours)	98.0	40.5	0.0	96.5	58.8
5.2K	T5	98.5	55.5	0.0	96.0	62.5
5.2K	VL-T5	98.0	37.5	0.0	96.5	58.0
10.3K	T5 + RC (Ours)	97.5	41.0	1.0	97.0	59.1
10.3K	T5	98.5	39.5	0.0	98.5	59.1
10.3K	VL-T5	97.5	53.5	0.0	98.0	62.3

Table 11: Comparison of the performance of our method with different model sizes ranging from 2M to 92M on L1 level generalization results. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Model size.	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T12	T15	T16	T17	Overall
2M	Ours w/o Pretrain	100	98.5	99.0	89.5	48.5	100	100	19.5	97.0	91.0	98.0	36.0	24.0	77.0
2M	Ours	99.5	99.0	97.5	99.0	67.5	100	99.5	18.5	91.5	93.0	99.0	38.0	64.5	82.0
4M	Ours w/o Pretrain	100	100	99.5	97.0	55.0	100	100	18.0	96.0	95.0	99.5	44.0	40.0	80.3
4M	Ours	100	100	86.5	99.0	63.5	99.5	100	20.5	92.0	95.5	98.0	83.5	57.0	84.2
9M	Ours w/o Pretrain	100	100	96.0	99.0	57.0	100	100	23.0	98.0	94.0	98.5	47.0	94.0	85.1
9M	Ours	100	100	99.0	99.0	87.0	100	100	19.0	100	95.5	98.5	92.5	97.0	91.3
20M	Ours w/o Pretrain	100	100	100	98.5	67.5	100	100	30.5	98.5	95.0	99.0	49.5	85.0	86.4
20M	Ours	100	100	100	97.0	90.0	100	99.5	19.0	100	94.0	99.5	93.5	97.5	91.5
43M	Ours w/o Pretrain	100	100	100	98.5	67.0	100	100	17.0	100	94.0	99.0	92.5	96.5	89.6
43M	Ours	99.5	100	99.5	95.5	89.0	97.5	100	100	100	94.5	96.0	94.5	96.5	97.1
92M	Ours w/o Pretrain	100	100	100	98.5	88.0	100	100	20.5	100	94.0	99.0	93.0	98.0	91.6
92M	Ours	98.5	100	100	99.5	94.0	100	100	100	100	94.0	95.5	94.0	96.5	97.8

Table 12: Comparison of the performance of our method with different model sizes ranging from 2M to 92M on L2 level generalization results. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Model size.	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T12	T15	T16	T17	Overall
2M	Ours w/o Pretrain	95.5	84.5	99.0	87.0	50.0	96.5	91.0	21.0	97.0	91.0	88.0	33.5	11.5	72.7
2M	Ours	99.5	98.5	98.5	98.5	59.0	100	98.5	20.5	92.0	92.5	99.0	39.5	61.5	81.3
4M	Ours w/o Pretrain	99.0	98.5	100	97.0	55.0	99.5	98.5	21.0	96.5	95.5	97.0	44.0	35.0	79.7
4M	Ours	100	100	87.0	99.0	67.5	99.5	99.5	19.0	92.5	95.5	97.0	84.0	60.0	84.7
9M	Ours w/o Pretrain	100	100	96.5	98.5	58.0	99.5	99.0	25.5	97.5	94.5	94.5	47.0	88.0	84.5
9M	Ours	100	100	99.5	98.5	86.5	99.5	100	19.0	100	94.5	97.0	91.5	95.0	90.8
20M	Ours w/o Pretrain	100	100	100	98.5	72.0	100	100	29.5	98.0	95.5	99.0	46.0	83.5	86.3
20M	Ours	100	100	100	97.0	86.5	99.0	99.0	19.5	100	95.0	97.0	91.5	96.5	90.8
43M	Ours w/o Pretrain	100	100	100	98.5	72.5	100	100	18.5	100	93.5	99.5	92.0	96.0	90.0
43M	Ours	99.0	100	100	97.0	90.5	98.0	100	99.5	100	95.5	94.0	93.0	96.5	97.2
92M	Ours w/o Pretrain	100	100	100	99.0	87.0	100	100	23.5	100	94.0	99.5	92.0	98.0	91.8
92M	Ours	98.5	100	100	99.0	96.5	99.5	100	100	100	95.5	95.0	93.0	96.0	97.9

Table 13: Comparison of the performance of our method with different model sizes ranging from 2M to 92M on L3 level generalization results. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Model size.	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T15	T16	T17	Overall
2M	Ours w/o Pretrain	97.0	91.0	100	92.5	46.0	96.5	95.5	15.5	95.5	95.0	36.0	8.0	72.4
2M	Ours	99.0	98.5	99.5	99.5	67.0	99.5	99.5	16.5	83.5	98.5	33.5	54.0	79.0
4M	Ours w/o Pretrain	98.5	98.0	100	94.5	53.5	95.0	99.0	17.0	99.0	87.5	47.0	5.5	74.5
4M	Ours	95.5	99.0	75.5	95.5	61.0	95.5	96.5	19.0	92.5	79.5	42.0	32.0	73.6
9M	Ours w/o Pretrain	93.5	97.5	96.0	100	64.0	95.0	96.0	17.5	97.5	85.0	43.0	44.0	77.4
9M	Ours	97.0	97.0	100	96.5	86.0	99.0	99.0	23.5	98.5	96.0	52.5	100	87.1
20M	Ours w/o Pretrain	99.5	100	100	100	71.5	100	100	26.0	98.5	98.5	43.5	87.5	85.4
20M	Ours	97.0	97.0	99.5	99.5	89.0	98.0	98.0	24.0	100	99.0	53.5	98.0	87.7
43M	Ours w/o Pretrain	99.5	100	100	100	74.0	100	99.5	25.0	100	99.5	54.0	99.0	87.5
43M	Ours	95.0	98.0	99.5	96.5	86.0	95.5	96.5	97.0	99.5	96.0	40.0	99.0	91.5
92M	Ours w/o Pretrain	99.5	100	100	100	90.0	100	100	20.5	100	99.5	50.5	99.5	88.3
92M	Ours	98.0	99.0	100	99.5	94.0	97.5	99.0	97.0	96.5	95.0	47.0	98.0	93.4

Table 14: Comparison of the performance of our method with different model sizes ranging from 2M to 92M on L4 level generalization results. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Model size.	Method	T8	T10	T13	T14	Overall
2M	Ours w/o Pretrain	78.5	0.0	0.0	95.5	43.5
2M	Ours	47.5	35.5	0.5	97.5	45.2
4M	Ours w/o Pretrain	99.5	0.0	0.0	95.5	48.8
4M	Ours	96.0	0.5	0.0	92.5	47.2
9M	Ours w/o Pretrain	96.5	1.0	0.0	95.0	48.1
9M	Ours	99.5	15.5	0.5	98.0	53.4
20M	Ours w/o Pretrain	99.0	0.0	0.0	99.0	49.5
20M	Ours	97.0	22.0	0.0	95.5	53.6
43M	Ours w/o Pretrain	99.0	0.0	0.0	98.5	49.4
43M	Ours	95.5	6.0	0.0	96.0	49.4
92M	Ours w/o Pretrain	97.0	0.0	0.0	99.5	49.1
92M	Ours	97.5	41.0	1.0	97.0	59.1

Table 15: Comparison of the performance of our method with different scales of training data on L1 level generalization results. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Data Size	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T12	T15	T16	T17	Overall
10%	Ours w/o Pretrain	100	98.5	96.5	97.0	74.0	97.5	100	19.0	100	93.0	93.0	88.0	93.0	88.4
10%	Ours	100	99.5	96.5	89.0	65.5	98.0	98.5	73.5	97.5	93.5	92.0	89.0	93.0	91.2
50%	Ours w/o Pretrain	100	99.5	97.5	98.0	74.0	99.5	99.5	20.0	100	92.5	98.0	82.0	91.0	88.6
50%	Ours	100	99.5	98.5	98.0	86.5	99.5	99.5	98.5	100	93.5	96.5	95.5	96.5	97.1
100%	Ours w/o Pretrain	100	100	100	98.5	88.0	100	100	20.5	100	94.0	99.0	93.0	98.0	91.6
100%	Ours	98.5	100	100	99.5	94.0	100	100	100	100	94.0	95.5	94.0	96.5	97.8

Table 16: Comparison of the performance of our method with different scales of training data on L2 level generalization results. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Data Size	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T12	T15	T16	T17	Overall
10%	Ours w/o Pretrain	99.5	99.0	97.0	95.5	72.5	97.5	99.5	20.5	98.5	93.0	91.5	88.5	91.0	88.0
10%	Ours	99.0	99.5	94.5	90.0	62.5	98.5	99.0	77.5	98.5	94.0	90.0	87.0	89.0	90.7
50%	Ours w/o Pretrain	98.5	100	97.0	98.0	72.0	99.5	99.5	16.5	100	91.5	97.5	84.5	88.5	87.9
50%	Ours	100	100	99.0	97.5	88.5	99.0	99.0	98.5	100	95.0	96.0	95.5	96.5	97.3
100%	Ours w/o Pretrain	100	100	100	99.0	87.0	100	100	23.5	100	94.0	99.5	92.0	98.0	91.8
100%	Ours	98.5	100	100	99.0	96.5	99.5	100	100	100	95.5	95.0	93.0	96.0	97.9

Table 17: Comparison of the performance of our method with different scales of training data on L3 level generalization results. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Data Size	Method	T1	T2	T3	T4	T5	T6	T7	T9	T11	T15	T16	T17	Overall
10%	Ours w/o Pretrain	98.0	97.0	97.5	98.5	74.5	97.5	99.0	18.0	100	96.5	53.5	99.0	85.8
10%	Ours	90.0	93.5	98.5	93.0	71.0	90.0	90.5	56.0	90.0	83.0	52.0	20.5	77.3
50%	Ours w/o Pretrain	97.5	99.5	99.0	99.5	70.5	98.5	99.0	19.0	100	97.0	57.5	93.5	85.9
50%	Ours	97.0	97.5	99.0	99.5	86.0	97.5	96.5	95.5	98.0	97.0	47.5	100	92.6
100%	Ours w/o Pretrain	99.5	100	100	100	90.0	100	100	20.5	100	99.5	50.5	99.5	88.3
100%	Ours	98.0	99.0	100	99.5	94.0	97.5	99.0	97.0	96.5	95.0	47.0	98.0	93.4

Table 18: Comparison of the performance of our method with different scales of training data on L4 level generalization results. Integers in the first row refer to indices of tasks defined in the VIMA paper (Jiang et al., 2023)

Data Size	Method	T8	T10	T13	T14	Overall
10%	Ours w/o Pretrain	92.0	0.0	0.0	94.5	46.6
10%	Ours	91.0	39.0	0.0	88.0	54.5
50%	Ours w/o Pretrain	91.5	0.0	0.0	97.0	47.1
50%	Ours	95.0	12.5	0.0	96.0	50.9
100%	Ours w/o Pretrain	97.0	0.0	0.0	99.5	49.1
100%	Ours	97.5	41.0	1.0	97.0	59.1

B PSEUDO-CODES & TRAINING DETAILS

Algorithm 1 Robot Control with multimodal prompts through pretraining and multitask FT

Input: Dataset $\mathcal{D} = \{\zeta_1, \zeta_2, \ldots\}$, policy parameter θ, number of pretraining iterations N_{pretrain}, number of multi-task imitation finetuning iterations N_{FT}

for $i = 1, \ldots, N_{\text{pretrain}}$ **do**
  Sample a mini-batch $\mathcal{B}$ from $\mathcal{D}$
  Minimize $L_{\text{pretrain}}(\theta)$ defined in Eq. 3 on $\mathcal{B}$
end for
for $i = 1, \ldots, N_{\text{FT}}$ **do**
  Sample a mini-batch $\mathcal{B}$ from $\mathcal{D}$
  Minimize $L_{\text{Imitatation}}(\theta)$ defined in Eq. 4 on $\mathcal{B}$
end for

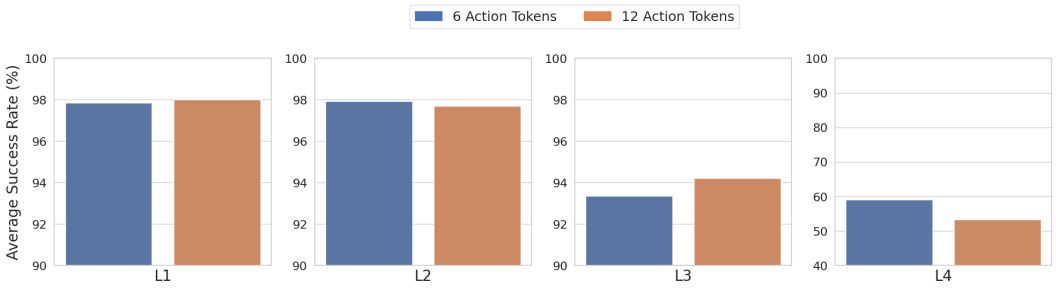

Figure 9: Ablation on the number of action tokens.

Algorithm 1 presents the pseudo-codes for the training pipeline, which includes a pretraining phase and a multi-task FT phase. We set our training HP following the recipe provided by VIMA, which open-sourced its policy architectures without providing the training codes. We conduct our experiments on cluster nodes, each with 8 NVIDIA-A10G. Table 19 presents the HP for our training pipeline. As we build our policy based on the VIMA Policy, we refer interested readers to Tables 2 and 3 in Appendix C of VIMA paper (Jiang et al., 2023) for all model parameters.

Additionally, the action space $\mathcal{A}$ includes initial pose $\mathcal{T}_{\text{initial}} \in \mathcal{R}^6$ and $\mathcal{T}_{\text{target}} \in \mathcal{R}^6$. Each pose is a 6-dimension vector with 2 for xy position and 4 for rotation represented in quaternion. Since the VIMA-BENCH focuses on tabletop manipulation, the rotation quaternion of $\mathcal{T}_{\text{initial}}$ is always a constant vector. So is the first two dimensions of the rotation quaternion of $\mathcal{T}_{\text{initial}}$. Therefore, we only tokenize the other 6 action dimensions to improve computational efficiency. Thus, each action worth 6 tokens. Moreover, we conduct an ablation study to show that this choice will not affect the task success rate. As shown in Figure 9, modeling each of the 12 action dimensions as a single token achieves almost the same performance as modeling the 6 active action dimensions.

Table 19: Hyper-parameters for our training pipeline

Phase	Hyperparameter	Value
Shared	Learning Rate (LR)	1e-4
	Minimum LR	1e-7
	Warmup Steps	7K
	Weight Decay	0
	Dropout	0.1
	Gradient Clip Threshold	1.0
	Optimizer	AdamW (Loshchilov & Hutter, 2017)
	Batch Size	128
	Iterations per epochs	5158
Pretrain	Training epochs	20
	Training iterations N_{pretrain}	20 $\times$ Iterations per epochs = 103160
	LR Cosine Annealing Steps	N_{pretrain} - Warmup Steps = 96160
Finetune	LR Cosine Annealing Steps	17K
	Training epochs	10
	Training iterations N_{FT}	10 $\times$ Iterations per epochs = 51580

C DETAILS OF EVALUATING THE IN-CONTEXT LEARNING ABILITY

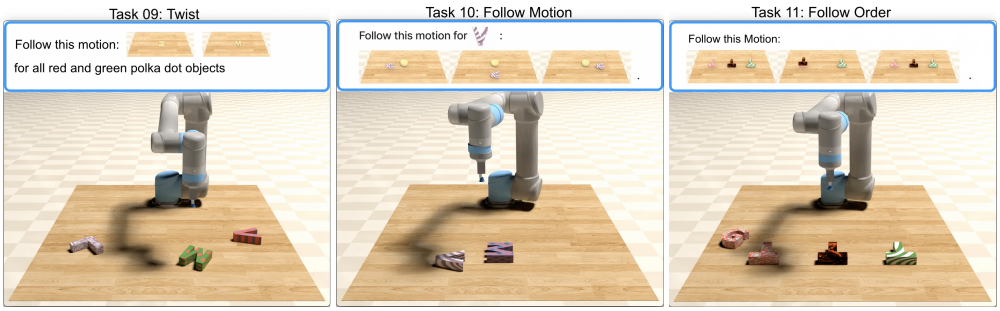

Figure 10: The new set of L4 tasks with in-context examples and modified prompts.

We provide training details for the experiments conducted in Sec. 4.3 by introducing the data augmentation strategies, pretraining, our *Modified FT*, and how we edit the task prompt for Task 09 (*Twist*) and Task 10 (*Follow Order*). Moreover, the L1, L2 and L3 success rate in this settings are given by 97.6%, 97.7%, and 93.0%, respectively.

Data Augmentation To improve the generalizability of the pretrained model, we randomly apply the standard random data augmentation techniques, including Color Jitter and Gray Scale (He et al., 2020) to the prompt images. Since we adopt an object-centric representation, we randomly shift the bounding box location for all objects in the whole trajectory with the same constant value. Note that we only augment the prompt images without modifying the observation images.

Pretraining We empirically find that further dividing the pretraining phase into two steps can improve the performance. We first pretrain a policy for 20 epochs and only extract the object encoder from it. Next, we use the pretrained object encoder to initialize another policy and pretrain it for 5 epochs. And the FT phase remains unchanged.

Modified FT To improve the model's ability to understand both visual and textual object descriptions, we randomly replace the object images in the multimodal prompts with text descriptions during multi-task FT. For example, the task prompt for *Follow Motion* in Figure 10 can be rephrased as

> Follow this motion for the white and purple striped V: $\{\text{frame}_1\}$, $\{\text{frame}_2\}$, $\{\text{frame}_2\}$.

Note that only object images will be converted into text descriptions. Images depicted the scene, e.g., frame_1, frame_2, frame_3, will never be converted to text. We randomly apply this operation to the task prompt of the pretraining tasks during the FT phase.

Edit Prompts As shown in Figure 10, we modify the task prompt for both *Twist* and *Follow order* to make them similar to the pretraining prompts. Specifically, the task prompt for *Twist* is modified as below

> **Original**: "Twist" is defined as rotating object a specific angle. For examples: From $\{\text{before\_twist}_1\}$ to $\{\text{after\_twist}_1\}$. From $\{\text{before\_twist}_2\}$ to $\{\text{after\_twist}_2\}$. From $\{\text{before\_twist}_3\}$ to $\{\text{after\_twist}_3\}$. Now twist all [TEXT OBJ DESCRIPTION] objects.
>
> **Modified**: Follow this motion: $\{\text{before\_twist}_1\}$ to $\{\text{after\_twist}_1\}$ for all [TEXT OBJ DESCRIPTION] objects.

Similarly, the task prompt for *Follow Order* is modified as below:

> **Original**: Stack objects in this order $\{\text{frame}_1\}$, $\{\text{frame}_2\}$, $\{\text{frame}_2\}$.
>
> **Modified**: Follow this motion: $\{\text{frame}_1\}$, $\{\text{frame}_2\}$, $\{\text{frame}_2\}$.