# A. Supplementary

This supplementary material contains the source code for networks, more ablation studies, and more visualizations for the affordance heatmaps. All ablations are based on the split of pushing actions and trained categories as in [7, 10].

## A.1. Details for Model Design

We present all the details for the network designs of every module in MAAL, as shown in Fig. 1 and Fig 2. The dimension of all features $z$, $a$, and $q$ is 128. According to the intermediate fusion strategies as in [11, 6, 2], our network design involves both multi-modal fusion and multi-level fusion. These fully consider the multi-modal inputs and provide better learning ability for solving the 3D object affordance problem.

## A.2. Ablation Study for Model Design

We provide ablations for different designations of modules in MAAL, as shown in Tab 1. First, the BN layer is valuable for fusing features. Only with the object learner, MAAL with BN layers achieves $3.85\%$ improvements than MAAL without BN layers in F-score as in Tab 1. This reveals the effectiveness of the BN layers in our network design, which normalizes different distributions [8] and empowers better learning ability for the networks [1, 4]. Then, in the interact learner, MAAL applies the bilinear operation [5, 12] to fuse $z_o$ and $z_a$. We change the bilinear layer to a concatenate operation with a fully-connected layer (Concat + FC), a concatenate operation with batch normalization (Concat + BN), and a cross-attention layer (Cross attention) as in [9], respectively. In experiments, our method with a bilinear layer achieves higher performance. The cross-attention layer obtains a comparable performance, but the results are slightly lower than ours. Thus, we apply the bilinear layer in MAAL.

Moreover, we further evaluate the interact learner without multi-level fusion. This indicates that the features aggregated and learned from $f_{o'}$, $f_p$, $f_a$, and $f_h$ are not considered in the interact learner. In this condition, the result of F-score decrease to $6.88\%$, which indicates the fusion of multi-level features is effective as in [6]. Besides, the effectiveness of residual block in the interact learner can be also reflected in Tab. 1. The interact learner with residual block obtains the better performance. The residual block supports the interact learner to achieve better learning ability for 3D object affordance.

Furthermore, we also test MAAL without using adapters, introducing three independent action encoding modules in MAAL, and MAAL decoder without given object information. All results in Tab. 1 show the effectiveness of our network designs.

## A.3. Ablation Study for Memory Module

The memory module aims to record patterns of action features. The memory number $N$ influences the ability of the memory as in [3]. In this part, we conduct experiments for different memory numbers shown in Tab. 2. Generally, a larger memory size leads to better performance. However, larger memory also introduces more learnable parameters and more computational costs. In our work, we set $N = 200$ since further enlarging the memory size brings only a few improvements.

Moreover, we also evaluate MAAL without the memory module, in which the decoder reconstructs actions directly from the queries. Compared with our MAAL, which achieves a $76.63$ F-score in pushing action, the performance of MAAL without the memory module decreases by $8.82\%$. This reveals that the memory module is valuable in learning 3D object affordance.

## A.4. Experiments on Other Datasets

More than PartNet-Mobility, we experiment with additional datasets, including VAT-Mart and GAPartNet) as in Tab. 3. We present Sample-Succ results for the door category. For pushing actions, MAAL shows improvements of $2.58\%$ and $4.02\%$ over the baselines, further indicating its generalization ability.

## A.5. Experiments on Other Baselines

We conduct experiments with VAT-Mart and UMPNet, as shown in Tab. 4. We present the results for Pushing All and Pulling All (train cat.). Our method outperforms all other baseline in experiments. For Pushing All, compared with VAT-Mart and UMPNet, our MAAL achieves better performances with $20.8\%$ and $16.61\%$ gains of F-score, respectively.

## A.6. Visualization for the Predicted Actionability Heatmap

We provide more visualizations for the affordance heatmaps of MAAL, as shown in Fig. 3, for pulling action and pushing action, respectively. All results further prove the effectiveness of MAAL in learning 3D object affordance.

## A.7. Visualization for Interactions

We display visualizations with the gripper in Fig. 4., highlighting the gripper directions. All results show that MAAL correctly predict the interactable points and corresponding gripper directions for the robots.

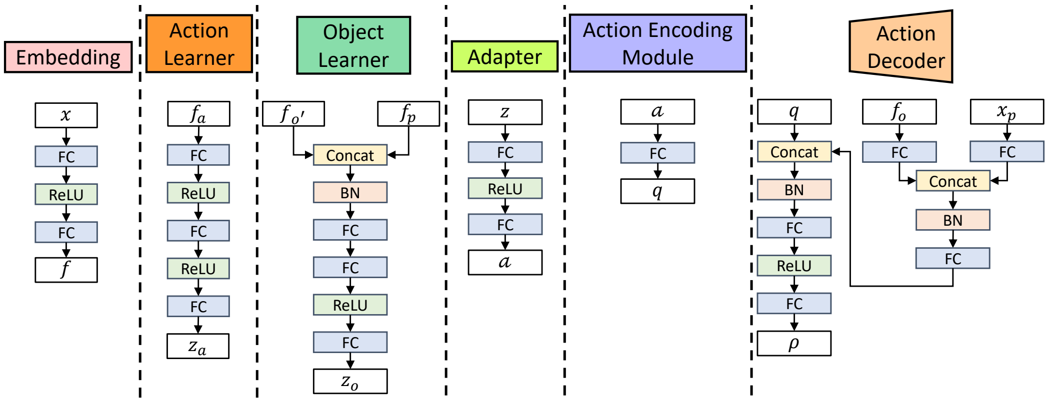


Figure 1. Structures of networks in MAAL. FC, ReLU, Concat, and BN denote the fully-connected layer, ReLU operation, concatenate operation, and batch normalization, respectively. $f_{o'}$ indicates the embedded feature by embedding layer for $f_o$. Feature $z$ is the output from the corresponding learner. ⊣ indicates the feature outputs from adapters.

| Method | | F-score (%) | Sample-Succ (%) |
|---|---|---|---|
| Variations of MME | Object Learner only | 32.47 | 13.54 |
| | Object Learner only w/o BN | 28.62 | 10.95 |
| | Interact Learner only | 58.74 | 24.01 |
| | Interact Learner only (Concat + FC) | 54.07 | 20.47 |
| | Interact Learner only (Concat + BN) | 49.64 | 13.48 |
| | Interact Learner only (Cross attention) | 58.65 | 23.34 |
| | Interact Learner only w/o multi-level fusion | 51.86 | 17.07 |
| | Interact Learner only w/o residual block | 53.90 | 19.25 |
| w/o Adapters | MAAL w/o Adapters | 74.56 | 34.08 |
| Variations of Action Encoding Module | MAAL w/ independent Action Encoding Module | 76.82 | 33.20 |
| | Action Encoding Module w/o ($f_o$ and $x_p$) | 54.91 | 19.20 |

Table 1. Ablation of different network designs in MAAL.

| Dataset | N=100 | N=200 | N=500 | N=1000 | N=2000 |
|---|---|---|---|---|---|
| pushing (train cat.) | 74.63 | 76.63 | 76.64 | 76.87 | 77.07 |
| pushing (test cat.) | 62.19 | 69.88 | 69.82 | 69.83 | 69.82 |
| pulling (train cat.) | 56.64 | 59.26 | 59.22 | 59.32 | 59.34 |
| pulling (test cat.) | 41.59 | 43.57 | 43.77 | 43.71 | 43.75 |

Table 2. Ablation study for the memory size $N$. Larger memory usually leads to better performances but also introduces computational costs. We set $N = 200$ in our works. Memory numbers larger than 200 do not lead to significant improvements.

| Dataset | Where2Act | Ours |
|---|---|---|
| VAT-Mart Dataset (pushing door) | 32.67 | 36.66 |
| VAT-Mart Dataset (pulling door) | 6.02 | 8.83 |
| GAPartNet (pushing door) | 24.08 | 28.10 |

Table 3. Comparison of different datasets.

| Method | Pushing All | | Pulling All | |
|---|---|---|---|---|
| | F-score (%) | Sample-Succ (%) | F-score (%) | Sample-Succ (%) |
| VAT-Mart | 55.83 | 20.15 | 50.25 | 19.46 |
| UMPNet | 60.02 | 26.30 | 54.70 | 21.14 |
| Ours | 76.63 | 34.25 | 69.88 | 28.34 |

Table 4. Comparison of different baselines.

# References


[1] Nils Bjorck, Carla P Gomes, Bart Selman, and Kilian Q Weinberger. Understanding batch normalization. *Advances in neural information processing systems*, 31, 2018.

[2] Changxing Ding and Dacheng Tao. Robust face recognition via multimodal deep face representation. *IEEE transactions on Multimedia*, 17(11):2049–2058, 2015.

[3] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1705–1714, 2019.

[4] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.

[5] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. *Advances in neural information processing systems*, 31, 2018.

[6] Dana Lahat, Tülay Adali, and Christian Jutten. Multimodal data fusion: an overview of methods, challenges,

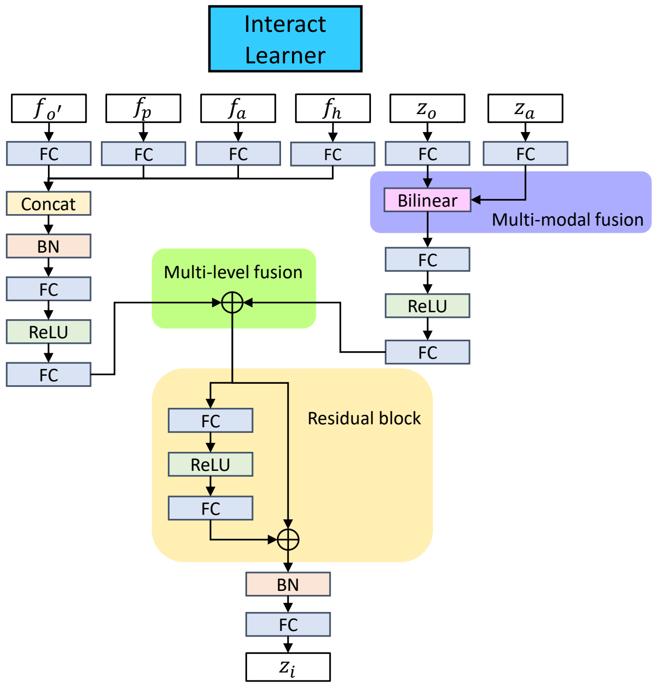


Figure 2. Structures of the interact learner in MMA. Bilinear denote the bilinear layer [5, 12]. Corresponding to the intermediate fusion [11, 6], our network design considers both multi-modal fusion and multi-level fusion of different features.

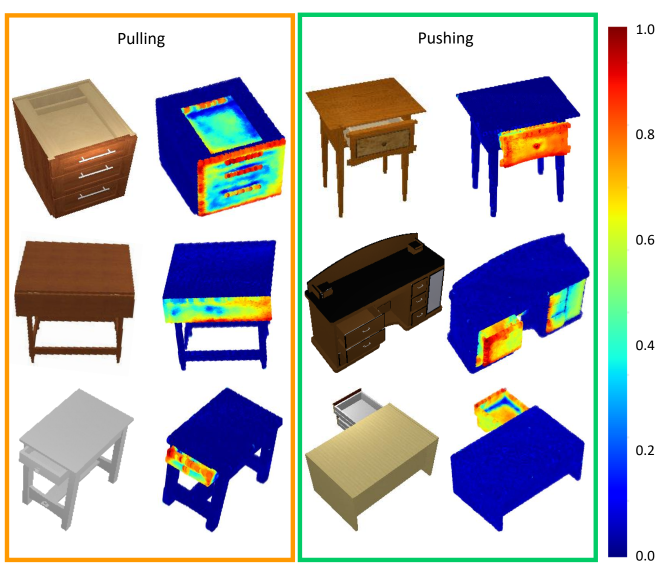


Figure 3. More visualizations for the affordance heatmap.

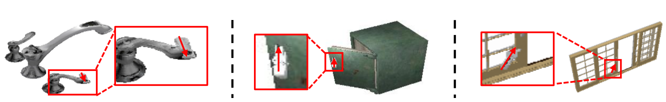

Figure 4. Visualization of predicted interaction.

and prospects. *Proceedings of the IEEE*, 103(9):1449–1477, 2015.

[7] Kaichun Mo, Leonidas J Guibas, Mustafa Mukadam, Abhinav Gupta, and Shubham Tulsiani. Where2act: From pixels to actions for articulated 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6813–6823, 2021.

[8] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? *Advances in neural information processing systems*, 31, 2018.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[10] Yian Wang, Ruihai Wu, Kaichun Mo, Jiaqi Ke, Qingnan Fan, Leonidas J Guibas, and Hao Dong. Adaafford: Learning to adapt manipulation affordance for 3d articulated objects via few-shot interactions. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIX*, pages 90–107. Springer Nature Switzerland Cham, 2022.

[11] Dong Yi, Zhen Lei, and Stan Z Li. Shared representation learning for heterogenous face recognition. In *2015 11th IEEE international conference and workshops on automatic face and gesture recognition (FG)*, volume 1, pages 1–7. IEEE, 2015.

[12] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multimodal factorized bilinear pooling with co-attention learning for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 1821–1830, 2017.