Fusion-Perception-to-Action Transformer: Enhancing Robotic Manipulation With 3-D Visual Fusion Attention and Proprioception

Yangjun Liu, Sheng Liu, Binghan Chen, Zhi-Xin Yang, *Member, IEEE*, and Sheng Xu, *Senior Member, IEEE*

***Abstract*—Most prior robot learning methods focus on image-based observations, limiting their capability in 3-D robotic manipulation. Voxel representation naturally delivers rich spatial features but remains underutilized. Specifically, current voxel-based methods struggle with fine-grained tasks, since precise actions are not fully achievable. However, humans can accomplish these tasks well using vision and proprioception. Inspired by this, this article proposed a novel Fusion-Perception-to-Action Transformer (FP2AT) with cross-layer feature aggregation to handle fine-grained manipulation in 3-D space. In particular, a multiscale 3-D visual fusion attention mechanism is devised to draw attention to local regions of interest and maintain awareness of global scenes, thereby boosting the capabilities of visual perception and action planning. Meanwhile, a 3-D visual mutual attention mechanism is designed and it can also enhance spatial perception. Besides, we further explore the potential of FP2AT by developing its coarse-to-fine version, which progressively refines the action space for more precise predictions. In addition, a proprioceptive encoder is developed to mimic the perception of body movements and contact, elevating the effectiveness of the FP2AT. Furthermore, a new metric, the average number of key actions (ANKA), is introduced to evaluate efficiency and planning capability. In various simulated and real-robot examples, our methods significantly outperform state-of-the-art 3-D-vision-based methods in success rate and ANKA metrics.**

***Index Terms*—3-D manipulation, imitation learning, task and motion planning, transformer.**

I. Introduction

VISION-BASED robotic manipulation has presented pleasant generalization ability to scene changes and novel instances, boosting their application in general unstructured task scenarios such as household service, healthcare, safety monitoring, operation, and maintenance. Recently, end-to-end solutions become popular with the development of reinforcement learning [1], [2], [3] and imitation learning [4], [5], [6]. They straightforwardly learn actions from visual observations, escaping from the demanding model linkage as conventional multistage methods, which rely on explicit object-centric information like bounding boxes and poses for motion planning. Nonetheless, most end-to-end methods employ 2-D RGB and depth images as visual observations to predict actions, thereby restricting their applications to 2-D or 2.5-D space. Unlike 2-D images, which often compress visual information into a flat representation, 3-D voxel grids preserve essential details regarding object 3-D structures, positions, and relative poses (see Fig. 1). This richness maintains spatial relations between the robot, objects, and the environment, having the potential to enhance spatial intelligence for robots and plan sophisticated actions for 3-D robotic manipulation tasks.

Nonetheless, end-to-end robotic manipulation based on voxel grids is still a relatively underexplored topic facing hurdles, despite many years of development in the fields of navigation and computer vision [7], [8], [9], [10], [11]. One significant challenge lies in designing neural networks capable of efficiently processing high-dimensional 3-D data. The complexity of this data necessitates innovative architectures that can effectively capture spatial hierarchies and learn from them. Furthermore, scene reasoning and comprehension within an end-to-end learning pipeline becomes increasingly intricate, as robots must discern nuanced interactions between multiple objects in complex 3-D space. This demands not only robust feature

Received 11 October 2024; accepted 20 January 2025. Date of publication 5 February 2025; date of current version 21 February 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62273327, Grant 62461160260, and Grant U22A2064; in part by the Science and Technology Development Fund of Macau SAR under Grant 0092/2024/AFJ, Grant 0075/2023/AMJ, Grant 0003/2023/RIB1, and Grant 001/2024/SKL; in part by the Guangdong Science and Technology Department under Grant 2023A0505030003 and Grant 2020B1515130001; in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2022B1515120010; in part by Shenzhen Science and Technology Program under Grant KCXFZ20211020165003005; and in part by the University of Macau under Grant MYRG-GRG2023-00237-FST-UMDF and Grant MYRG-GRG2024-00299-FST. This article was recommended for publication by Associate Editor M. Walter and Editor J. Bohg upon evaluation of the reviewers' comments. *(Corresponding authors: Zhi-Xin Yang; Sheng Xu.)*

Yangjun Liu is with the State Key Laboratory of Internet of Things for Smart City, Centre for Artificial Intelligence and Robotics, Department of Electromechanical Engineering, University of Macau, Macau 999078, China, and also with the Guangdong Provincial Key Laboratory of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: marco.yj.liu@connect.um.edu.mo).

Sheng Liu is with the Harbin Institute of Technology, Shenzhen 518055, China (e-mail: sheng.liu@siat.ac.cn).

Binghan Chen and Sheng Xu are with the Guangdong Provincial Key Laboratory of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: bh.chen2@siat.ac.cn; sheng.xu@siat.ac.cn).

Zhi-Xin Yang is with the State Key Laboratory of Internet of Things for Smart City, Centre for Artificial Intelligence and Robotics, Department of Electromechanical Engineering, University of Macau, Macau 999078, China (e-mail: zxyang@um.edu.mo).

This article has supplementary downloadable material available at https://doi.org/10.1109/TRO.2025.3539193, provided by the authors.

Digital Object Identifier 10.1109/TRO.2025.3539193

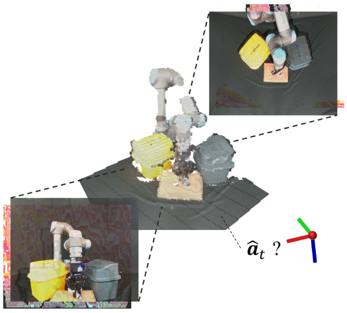


Fig. 1. Voxel grid representation for 3-D scene.

extraction, but also the ability to synthesize and reason about the extracted information. While image-based robot learning can serve pretrained 2-D visual models as effective feature extractors, the lack of pretrained feature extractors for voxel grids raises the bar for 3-D robot learning models themselves. Thus, the existing voxel-grid-to-action methods have difficulty in fine-grained tasks [12], [13]. These methods frequently fail in high-precision manipulation primitives. One possible solution is to take advantage of both global and local voxel grids. However, it has been proven that the progressive zoom-in version of 3-D convolutional networks also cannot achieve satisfactory results in 3-D robotic manipulation [12], [13].

In this work, we explore two 3-D visual attention mechanisms to enhance robot perception and manipulation through fusing global and local 3-D observations. By integrating a proprioceptive encoder and cross-layer feature aggregation, a fusion-perception-to-action transformer (FP2AT) is introduced to handle fine-grained manipulation tasks. Moreover, a coarse-to-fine version of the transformer is proposed to investigate the zoom-in law, i.e., whether the performance can be improved by increasing the prediction resolution of voxel grids. We conduct simulations and real-world experiments varying in objects, manipulating primitives, initial and target poses, obstacles, colors, distractors, and camera setups. These evaluations demonstrate the strengths of the key developments and verify the generalizability and general applicability of our proposed methods.

The major contributions and novelty of this work are summarized as follows.

1) We pioneer two novel 3-D visual attention methods to fuse the global and local voxel grids, and both of them can enhance spatial visual perception in 3-D robot learning. One of these methods, inspired by flexible viewpoints, global awareness, and local attention of humans' visual perception, has shown strong effectiveness.
2) A new attention-based network architecture (FP2AT) with cross-layer feature aggregation is proposed to learn 7-D manipulating actions of fine-grained tasks from fused visual perception and comprehensive proprioception in an end-to-end manner.
3) The proposed FP2AT is further improved by developing a coarse-to-fine variant, named C2F-FP2AT, which progressively refines the action space for more precise predictions.
4) Sufficient simulation and real-world tasks (see Fig. 2) with different robot embodiments and camera placements demonstrate that the proposed method significantly outperforms the state-of-the-art voxel-grid-based method by 34.4% and 38.0% in overall success rate (SR) with fewer key actions in most cases. Furthermore, in simulation, the coarse-to-fine variant evidently surpasses the previous leading 3-D robotic manipulation methods based on either voxel grids or 3-D point clouds, with at least 14.6% higher SR.

The rest of this article is organized as follows. Section II introduces the related work. Section III provides problem formulation and preliminaries. Subsequently, the FP2AT for 3-D robotic manipulation are proposed in Section IV. Next, the experiments and results on two metrics are reported in Section V, as well as the ablation study of the proposed network architecture. Finally, Section VI concludes this article.

II. Related Work

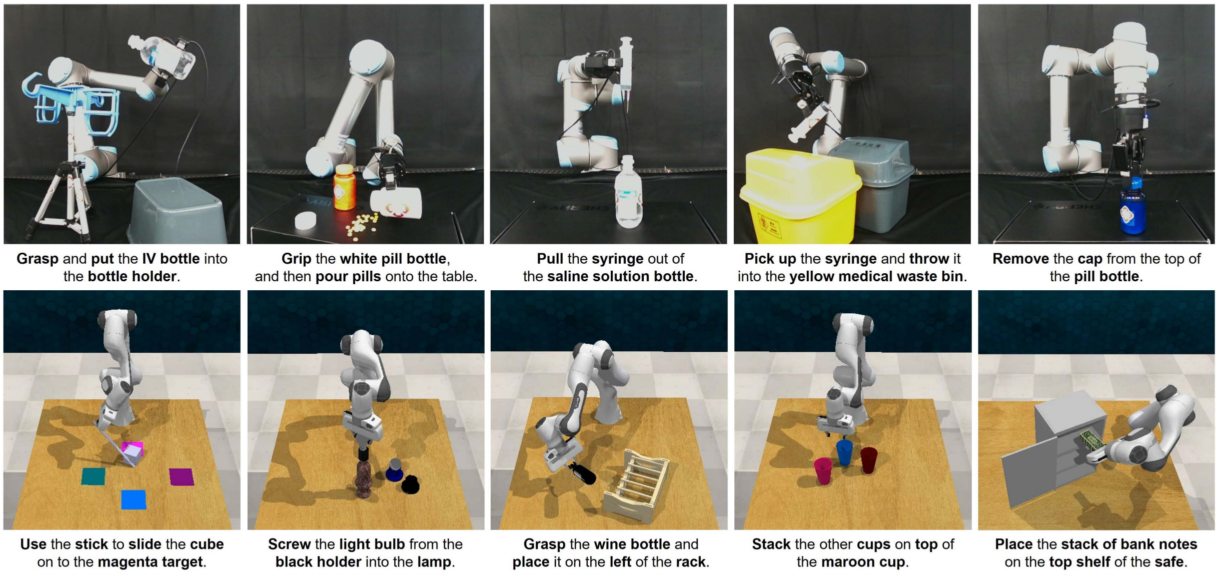
A. Vision-to-Action Robotic Manipulation

Conventional pipelines of vision-based manipulation in 3-D space typically require multiple stages [15], [16], [17], [18], [19], [20], including task planning, object detection or segmentation, 6-D object pose estimation, gripper's grasp pose detection, motion planning. Each stage demands a deep learning model or careful engineering by human experts. Although topics of the aforementioned stages have been well studied in previous work as [21], [22], [23], [24], [25], [26], [27], [28], it is inevitable to finetune or engineer for specific task scenarios in light of the independence of each study. Inferring multiple deep models simultaneously is memory-expensive and time-consuming. Furthermore, the performance of each stage highly relies on former stages, which poses a challenge to maintaining the robustness and efficiency of the whole system.

By contrast, the end-to-end vision-to-action methods avoid cumbersome engineering and method linkage. However, most of them focus on observations and actions in 2-D or 2.5-D space [1], [2], [3], [4], [5], [6]. For instance, Jangir et al. [29] applied Transformers [30] and Soft Actor–Critic [31] to the reaching, pushing, peg-in-box, and hammering tasks, which are restricted within XYZ action space without rotations. Shridhar et al. [32] considered the in-plane rotation such that it could deal with 4-DOF actions (3-DOF translations and 1-DOF rotation), but their CLIPORT network was purely designed for pick-place affordance predictions. Accordingly, it could only complete two-step actions based on the picking and placing primitives. This article will propose a generic end-to-end pipeline for -3D manipulation tasks requiring 6-DOF pose adjustments and various manipulation primitives as illustrated in Fig. 2.

B. Multiview Robotic Manipulation

Prior works have shown benefits in manipulation tasks when using multiview images instead of single-view ones. According to the simulation results of [33], [34], [35], images from additional global views (e.g., left and right view) were able to provide supplementary visual information, thereby enhancing scene understanding and relative pose estimation. Besides, the


Grasp and **put** the **IV bottle** into the **bottle holder**.

Grip the **white pill bottle**, and then **pour pills** onto the table.

Pull the **syringe** out of the **saline solution bottle**.

Pick up the **syringe** and **throw** it into the **yellow medical waste bin**.

Remove the **cap** from the top of the **pill bottle**.

Use the **stick** to **slide** the **cube** on to the **magenta target**.

Screw the **light bulb** from the **black holder** into the **lamp**.

Grasp the **wine bottle** and **place** it on the **left** of the **rack**.

Stack the other **cups** on **top** of the **maroon cup**.

Place the **stack of bank notes** on the **top shelf** of the **safe**.

Fig. 2. Tasks with language goals varying in manipulating primitives, objects, locations, colors, distractors, etc. Experiments are conducted in both the real world (the first row) and RLBench simulation [14] (the second row), which is a benchmark with multiple challenging tasks and multiview configurations. We propose a generic solution that is applicable to different robots and task scenarios.

combination of global-view and local-view images, such as those captured from wrist-mounted cameras, could improve the localization accuracy of contact points in manipulation tasks [29], [36]. There are only a few studies regarding end-to-end robotic manipulation ba-D vision (e.g., voxel grids and point clouds) [12], [13], [37], [38], [39], even though it can explicitly encode information from diverse viewpoints without changing the mounting position of cameras or installing excessive cameras. Therefore, we endeavor to develop a comprehensive 3-D manipulation pipeline that harnesses the multiview power of 3-D visual inputs to advance perception and planning in robotic manipulation.

C. Coarse-to-Fine Robot Learning

The coarse-to-fine mechanism is essentially hard visual attention, whose main idea involves selecting a discrete subspace of the global feature space for subsequent processing, thoroughly ignoring other parts. The C2FARM-BC [12], [13] implemented a coarse-to-fine hard attention mechanism learnt via Q-Learning, inherited from ARM [2] for 2-D-vision-based robotic manipulation. Similar to this article of ours, it also took 3-D voxel grids as visual inputs. Specifically, C2FARM-BC first took a global voxel grid as visual input and acquired the absolute position of the voxel of most interest v_1. Subsequently, it applied hard attention to a fine voxel grid with a higher resolution centered on v_1. Eventually, the action was predicted based on the attended fine voxel grid. Similar to C2FARM-BC, Act3D [39] worked in a coarse-to-fine manner, but with four main differences. First, Act3D employed 3-D point clouds instead of 3-D voxel grids as visual observations. Second, it used relative positional information [40] rather than absolute one when achieving hard attention. Third, Act3D took one more stage of hard attention than C2FARM-BC. Lastly, Act3D is a Transformer with cross soft attention to 3D visual features, language tokens, and end-effectors' states, while C2FARM-BC comprises only 3-D convolutional and fully connected layers. This article will develop two soft attention mechanisms for 3-D vision in robot learning, and then explore the potential of the proposed FP2AT through hard attention.

III. Problem Formulation and Preliminaries

A. Problem Formulation

In this article, we will develop a generic end-to-end method based on 3-D vision to deal with fine-grained robotic manipulation. It is supposed to directly predict a desired 7D manipulating pose $\hat{\boldsymbol{p}}_t$ of the end effector with an indicator of collision avoidance $\hat{c}_t$ from each observation $\boldsymbol{O}_t$. Consequently, we will obtain high-level key actions $\hat{\boldsymbol{a}}_t$ for each task with a language goal $\boldsymbol{l}$, such that the equipped low-level trajectory generator and actuator of a commercial robot arm will move the end effector to the planned poses step by step. The task will be completed after a sequence of perception–prediction–action. The predicted actions of each step t take the form of

$$\hat{\boldsymbol{a}}_t = [\hat{\boldsymbol{p}}_t, \hat{c}_t] \tag{1a}$$

$$\hat{\boldsymbol{p}}_t = [\hat{\boldsymbol{\tau}}_t, \hat{\boldsymbol{\varphi}}_t, \hat{g}_t] \tag{1b}$$

$$\hat{\boldsymbol{\tau}}_t = [\hat{x}_t, \hat{y}_t, \hat{z}_t] \tag{1c}$$

$$\hat{\boldsymbol{\varphi}}_t = [\hat{\alpha}_t, \hat{\beta}_t, \hat{\gamma}_t] \tag{1d}$$

where $\hat{\boldsymbol{\tau}}_t, \hat{\boldsymbol{\varphi}}_t, \hat{g}_t$ are the predicted 3-D positions, 3-D rotations in Euler angles, and end effector's state, respectively. $\hat{g}_t$ is a binary indicator, and it is set to 1 for opening the gripper while 0 for closing the gripper. The value of $\hat{c}_t$ should be assigned 0 or 1, representing the collision avoidance or tolerance, respectively. In particular, we explicitly use the indicator to enable or avoid collision in simulation, while it is discarded in real-robot experiments since the collaborative robot has a protective mechanism to stop the robot when it collides with the environment. The hat symbol ˆ denotes the predictions that may be different from the ground-truth values.

By learning from expert demonstrations, we aim to improve the overall SR and lower the number of key actions on multiple challenging tasks in RLBench benchmark [14] and the real world. Therefore, the key objective of this article becomes to develop an effective method to select optimal actions $\hat{\boldsymbol{a}}_t$ for completing manipulation tasks effectively and efficiently. Therefore, a novel FP2AT will be developed in the next section.

B. Data Collection

The key action discovery strategy [2] is adapted to select the ground-truth key actions $\boldsymbol{a}_t$ for the real-robot and RLBench tasks. Specifically, the actions at timesteps with joint velocities close to zero or state change of the end effector are denoted as key actions. During the data collection from expert demonstrations, we record 7-D manipulating poses $\boldsymbol{p}_t$ with the indicator of collision avoidance c_t, multiview RGBD images for the reconstruction of voxel grids, and proprioceptive data acquired from internal sensors of robots and external force–torque (FT) sensors. These data are contained in $\boldsymbol{a}_t$ and $\boldsymbol{O}_t$, and then will be applied to the training process.

C. Action Encoding

Given the cameras' intrinsics and extrinsics, the task scenarios can be reconstructed and voxelized into voxel grids from the multiview RGBD images. Following [12] and [13], we one-hot encode ground-truth action $\boldsymbol{a}_t$ in discretized global action space $\boldsymbol{A}$, and subsequently we can acquire the ground-truth confidence score distributions $\boldsymbol{Q}_t$ by

$$\boldsymbol{Q}_t = [\boldsymbol{Q}_{\boldsymbol{\tau}}, \boldsymbol{Q}_{\boldsymbol{\varphi}}, \boldsymbol{Q}_g, \boldsymbol{Q}_c] = \text{onehot}(\boldsymbol{a}_t) \tag{2}$$

which will be employed for training.

D. Language Encoder

Contrastive language image pretraining network (CLIP) jointly trains an image encoder and a text encoder from visuals with language descriptions, demonstrating competitive transferring ability on downstream visual tasks [41]. Hence, we tokenize and encode natural language goals $\boldsymbol{l}$ to language features $\boldsymbol{F}_l$ via a frozen CLIP text encoder, which is pretrained in [41] and will not be further fine-tuned during training

$$\boldsymbol{F}_l = \text{CLIP}(\boldsymbol{l}). \tag{3}$$

In this way, the model can understand natural language goals for different tasks, thereby facilitating its application in multiple scenarios.

IV. Proposed FP2AT

Humans usually handle fine-grained manipulation with adjustable views, namely that they can easily change the focal length of their eyes and their physical distance toward objects, observing them from coarse to fine. Therefore, we likewise take advantage of both global and local visual features in our network. Meanwhile, human beings can comprehensively observe the task scenes from different directions around. Motivated by this, we use 3-D voxel grids rather than 2-D images to obtain rich spatial information implicitly (see Fig. 3), including positions, 3-D structures, and relative poses of objects. Therefore, the robot can perform collision-free actions in manipulation tasks, especially those sensitive to rotations and depth. Proprioceptive sensory provides humans with information on the limb positions, muscle tensions and states of hands, which elevate the perception of contact with the surroundings, body movement and coordination. In this article, we likewise encode proprioceptive data as part of the observations.

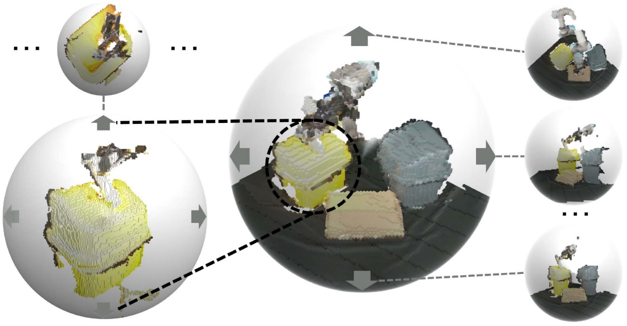


Fig. 3. Human-inspired visual observation. The combination of global and local 3-D vision delivers spatial information from different distances and angles of view, working like humans.

A. Overall Network Architecture

We propose the FP2AT (as shown in Fig. 4), which takes proprioceptive observations $\boldsymbol{O}_p$, global voxel grid $\boldsymbol{O}_{\text{glo}}$, and local voxel grid $\boldsymbol{O}_{\text{loc}}$ as inputs, conditioned on language goals $\boldsymbol{l}$.

Vision and Proprioception Encoder: Initially, the network passes the reconstructed voxel grids $\boldsymbol{O}_{\text{glo}}$ and $\boldsymbol{O}_{\text{loc}}$ through a 3-D visual fusion attention module, acquiring fusion feature map $\boldsymbol{F}_{\text{fus}}$, whose spatial dimension $d_{\text{fus}} \times h_{\text{fus}} \times w_{\text{fus}}$ is then flattened into $1 \times d_{\text{fus}} h_{\text{fus}} w_{\text{fus}}$. Simultaneously, fully connected layers are applied to encode proprioceptive inputs $\boldsymbol{O}_p$, viz., joint positions, joint forces, 3-DOF force and 3-DOF torque of the end effector, gripper's open-close state, the distance between fingers, and timestep

$$\boldsymbol{F}_p = \mathcal{F}_p(\boldsymbol{O}_p) \tag{4}$$

where $\boldsymbol{F}_p$ are encoded proprioceptive features, which are then concatenated with visual features $\boldsymbol{F}_{\text{fus}}$ and language features $\boldsymbol{F}_l$.

Attention-Based Feature Extractor: Afterward, we will learn a latent feature map with typically smaller and more balanced size ($h_{\text{lat}} \times w_{\text{lat}}$) than the concatenated multimodal one via a feature extractor based on Perceiver-IO [42], considering its inclusiveness for inputs and outputs with different spatial structures. Compared with mainstream vision transformers like ViT [43], the feature extractor consists of not only the transformer encoder but also a decoder. As illustrated in Fig. 4, the encoder maps the

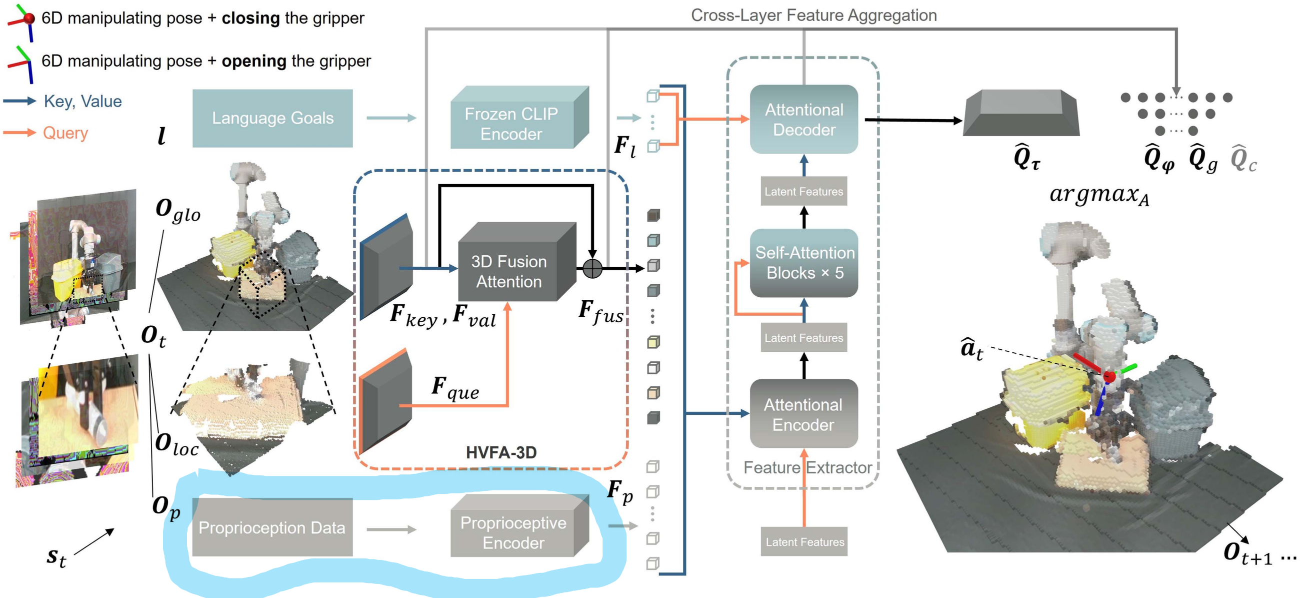


Fig. 4. Architecture of the proposed FP2AT. The main idea is to design a Transformer-like architecture to predict confidence scores in the action space, and the optimal 7-D poses will be achieved step by step with the help of the confidence scores. It first transforms observations into multimodal features through the human-inspired 3-D visual fusion attention module, proprioceptive encoder, and a frozen language encoder. Subsequently, as the key and value, the multimodal features are further learned in a latent feature space. In addition, cross-layer feature aggregation is applied before output heads to aggregate the features from different layers.

concatenated flattened multimodal features into the latent space, and then the decoder maps the latent features into the flattened feature space. At the end of the decoder, a 3-D transposed convolutional block following a reshaping layer is employed to expand the features to the global voxel grid space with the same shape but more channels than inputs. To leverage the expressive power of latent attention and extract better features as [42], five self-attention blocks are inserted between the attentional encoder and decoder.

Cross-Layer Feature Aggregation: With the purpose of shortening the information transition path between low and high layers, we concatenate obtained features across the input encoder, the 3-D visual fusion attention, and the attention-based feature extractor. As a consequence, the features at different levels are aggregated and then exploited as inputs of the heads of rotations, end effector's state, and collision avoidance indicator.

Output Heads and Confidence Score Prediction: At the end of the model, multiple heads are designed for desired outputs. In contrast to utilizing four regression heads to predict a definite action $\hat{\boldsymbol{a}}_t$ for each observation $\boldsymbol{O}_t = [\boldsymbol{O}_{\text{glo}}, \boldsymbol{O}_{\text{loc}}, \boldsymbol{O}_p]$, we develop the heads to predict a group of confidence score distributions $\hat{\boldsymbol{Q}}_t$ in discretized space $\boldsymbol{A}$ of possible actions $\boldsymbol{a}$, conditioned on observations $\boldsymbol{O}_t$ and language goals $\boldsymbol{l}$, i.e.,

$$\hat{\boldsymbol{Q}}_t = [\hat{\boldsymbol{Q}}_{\boldsymbol{\tau}}, \hat{\boldsymbol{Q}}_{\boldsymbol{\varphi}}, \hat{\boldsymbol{Q}}_g, \hat{\boldsymbol{Q}}_c] = \mathcal{MODEL}(\boldsymbol{a}|\boldsymbol{O}_t, \boldsymbol{l}) \qquad (5)$$

where $\mathcal{MODEL}$ denotes the FP2AT without optimal action selection. In the output head of 3-D positions, a convolutional block is applied to predict the confidence score distribution of discrete positions ($\hat{\boldsymbol{Q}}_{\boldsymbol{\tau}}$). By contrast, we predict confidence scores for discrete rotations as $\hat{\boldsymbol{Q}}_{\boldsymbol{\varphi}}$ via fully connected layers due to the inconsistency between rotations' spatial structure and that of the global voxel grid. For predictions of the end effector's state and collision indicator, $\hat{\boldsymbol{Q}}_g$ and $\hat{\boldsymbol{Q}}_c$, two simple fullyconnected blocks are applied for two-neuron outputs.

Optimal Action Selection: The confidence score prediction of possible actions $\boldsymbol{a}$ and selection of optimal actions $\hat{\boldsymbol{a}}_t$ are similar to the decision-making process of humans. Specifically, there are usually different policies to achieve the same goal in one task scenario. Humans tend to accomplish the task by the most convenient policy based on their experience and intuition, but other policies may still be feasible. Technically, the proposed FP2AT considers actions with the highest confidence scores as the optimal ones, i.e.,

$$\hat{\boldsymbol{a}}_t = \underset{\boldsymbol{a} \in \boldsymbol{A}}{\arg\max}\, \mathcal{MODEL}(\boldsymbol{a}|\boldsymbol{O}_t, \boldsymbol{l}) \qquad (6)$$

and $\hat{\boldsymbol{a}}_t$ may change with observations. The optimization in (6) will be resolved by supervised learning with proper loss function and optimizer. Compared with methods that regress a definite action, it is more robust to unseen scenes and observational noises.

As the global and local voxel grids are the most informative observations, exploring an effective method to fuse and take advantage of them is crucial. Next, we will introduce two 3-D visual fusion attention methods in detail, which are key developments of the proposed network.

B. 3-D Visual Fusion Attention Mechanisms

Human-Inspired 3-D Visual Fusion Attention (HVFA-3D): A person can perceive their own pose in the whole working scenario when taking accurate actions based on nuanced observations. In this way, subsequent actions are smoothly predicted and executed. To imitate this efficient workflow, we propose a Human-Inspired 3D Visual Fusion Attention module (HVFA-3D block in Fig. 4), which first learns global and local feature

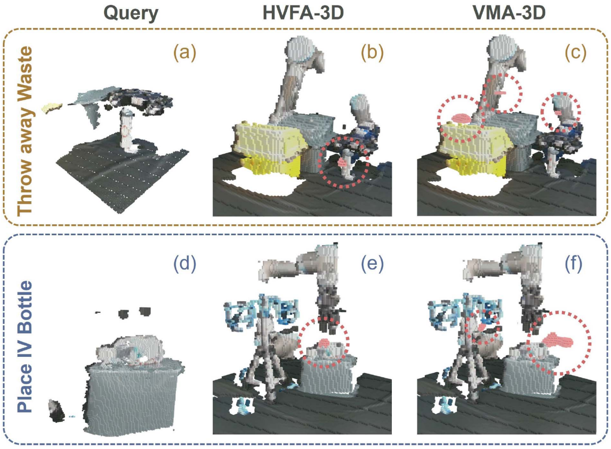


Fig. 5. Attention map of the proposed HVFA-3D and VMA-3D. The HVFA-3D tends to pay attention to one region of interest [(b) and (e)], while the VMA-3D often attends to multiple regions and fails to focus on the desired region [(c) and (f)] with identical queries [(a) and (d)].

maps from voxel grids $\boldsymbol{O}_{\text{glo}}$ and $\boldsymbol{O}_{\text{loc}}$. Subsequently, it takes a global feature map $\boldsymbol{F}_{\text{key}}$ as the key, which is queried by the local feature map, thereby providing a global view of the workspace and paying attention to specific parts according to the query $\boldsymbol{F}_{\text{que}}$. The attention map $\boldsymbol{F}_{\text{att}}$ is the matrix product of the query and transposed key, scaled by the square root of the number of channels of the key and then normalized by softmax function, i.e.,

$$\boldsymbol{F}_{\text{att}} = \text{softmax}\left(\frac{\boldsymbol{F}_{\text{que}}\boldsymbol{F}_{\text{key}}^T}{\sqrt{n_{\text{cha}}}}\right). \tag{7}$$

$\boldsymbol{F}_{\text{att}}$ provides attention weights on the value $\boldsymbol{F}_{\text{val}}$, which is assigned by another global feature map learned from $\boldsymbol{O}_{\text{glo}}$. Consequently, the fused feature map is calculated by

$$\boldsymbol{F}_{\text{fus}} = \boldsymbol{F}_{\text{att}}\boldsymbol{F}_{\text{val}}. \tag{8}$$

The aforementioned maps $\boldsymbol{F}_{\text{que}}$, $\boldsymbol{F}_{\text{key}}$, and $\boldsymbol{F}_{\text{val}}$ are obtained after feeding the global and local voxel grids into 3-D convolutional blocks as follows:

$$\boldsymbol{F}_{\text{que}} = \mathcal{F}_{\text{que}}(\boldsymbol{O}_{\text{loc}}) \tag{9a}$$

$$\boldsymbol{F}_{\text{key}} = \mathcal{F}_{\text{key}}(\boldsymbol{O}_{\text{glo}}) \tag{9b}$$

$$\boldsymbol{F}_{\text{val}} = \mathcal{F}_{\text{val}}(\boldsymbol{O}_{\text{glo}}). \tag{9c}$$

In addition, a shortcut connection is added between the value and fused feature maps to aggregate the low-level and high-level features through identity mapping [44]. Rather than using the local vision merely as a separate complement to the global vision, our HVFA-3D draws attention to the local region of interest when perceiving the global scene, as shown in Fig. 5(a)–(b) and (d)–(e). In our implementation, the $\boldsymbol{O}_{\text{glo}}$ are in the 1m^3 space with a resolution of 100^3 voxels. Meanwhile, the region of 0.25^3 m^3 below the gripper's manipulation space is voxelized into different 100^3 local voxel grids ($\boldsymbol{O}_{\text{loc}}$), whose directions of coordinate axes align with those of the global grids.

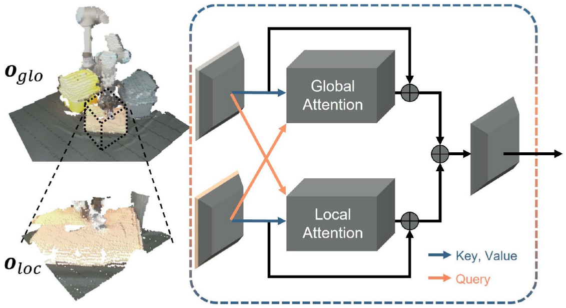


Fig. 6. Architecture of the proposed VMA-3D.

3D Visual Mutual Attention (VMA-3D): As far as we know, there is no satisfactory work leveraging both global and local voxel grids. Previous studies in 2-D-vision-based robotic manipulation and natural language processing have proven the superiority of crosswise attention mechanisms, improving performance over nonattentional methods that learn features from separate global and local branches [29], [45]. We agree with this prior knowledge in 3-D vision and wonder if crosswise attention is more effective than HVFA-3D in 3-D-vision-based robot learning. Thus, 3-D VMA is devised to learn the mutual features from global and local voxel grids. To the best of our knowledge, this is the first time that cross-scale attention has been introduced in 3-D-vision-based robot learning, especially for voxel-grid approaches. As shown in Fig. 6, $\boldsymbol{O}_{\text{glo}}$ and $\boldsymbol{O}_{\text{loc}}$ are fed into two 3-D convolutional blocks at first. Subsequently, local-guided global attention and global-guided local attention are applied, followed by feature aggregation and another 3-D convolutional block to merge features. It mutually uses global and local voxel grids as queries for each other to learn mutual features, enabling spatial information to flow bidirectionally between global and local vision. Compared with the HVFA-3D, the VMA-3D has limited performance in the 3-D case (see Fig. 5), struggling to pay attention to one region of interest. Moreover, Section V will further present comparative experimental results of the two methods, which are consistent with the attention maps. As for the rest of the network that implements the VMA-3D, it shares the architecture with FP2AT as Fig. 4, denoted as FP2AT-VMA.

Different from hard attention utilized in previous 3-D robot manipulation work [12], [39], the proposed VMA-3D and HVFA-3D are both soft attention solutions. Our methods continuously assign weights to all parts of the global feature space (a.k.a. attention map), allowing the model to focus more on important local regions while still considering the context provided by the rest of the 3-D voxel space. By contrast, the hard attention methods discretely crop or zoom into interested local spaces with higher resolution while disregarding the rest of the global space. During training, soft visual attention is differentiable and can be learned using standard backpropagation techniques, while hard visual attention is nondifferentiable. Moreover, 3-D hard attention predicts actions at the final stage of

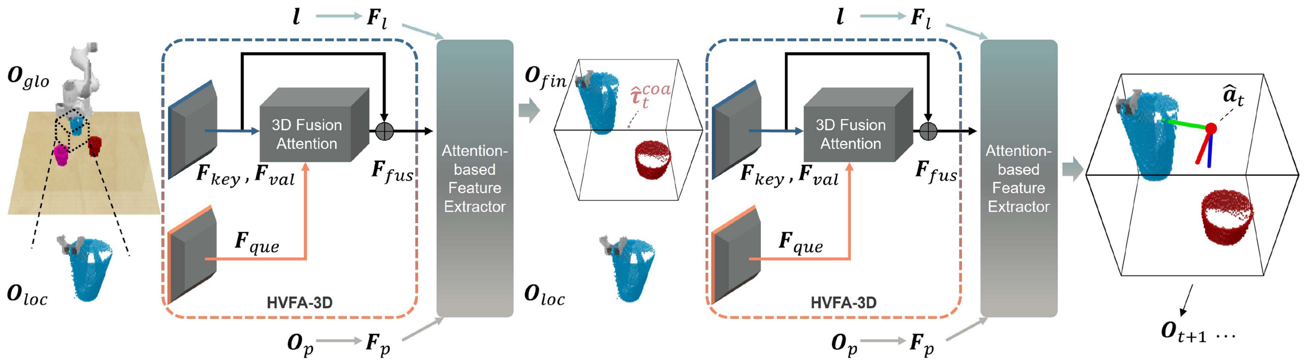


Fig. 7. Architecture of the proposed Coarse-to-Fine FP2AT.

the attention process, with smaller receptive fields than our soft attention.

To conclude, we pioneered two soft attention mechanisms for global and local voxel grids. Among them, HVFA-3D, a method inspired by human perception, can visibly enhance the visual perception of objects and the environment for manipulation.

C. Coarse-to-Fine FP2AT

Unlike 3-D point clouds, which can theoretically sample arbitrary points in 3-D space, voxel grid representation voxelizes the workspace into fixed spatial resolution. The higher resolution provides finer visual observation and more precise prediction resolution than the lower one, whereas a simple increase in resolution will incur huge memory consumption. Therefore, to further explore the potential of our network, we propose Coarse-to-Fine FP2AT (C2F-FP2AT) as presented in Fig. 7, consuming less extra memory than crude high-resolution voxelization. It works in a coarse-to-fine manner, predicting the next key action in two stages. The first stage is almost the same as standard FP2AT, utilizing $\boldsymbol{O}_{\text{glo}}$ and $\boldsymbol{O}_{\text{loc}}$ as visual observations. Differently, C2F-FP2AT selects the voxel $\hat{\boldsymbol{\tau}}_t^{\text{coa}}$ with the highest confidence and does not take any actions in the first stage

$$\hat{\boldsymbol{\tau}}_t^{\text{coa}} = \underset{\boldsymbol{\tau}\in\boldsymbol{A}_{\boldsymbol{\tau}}(\boldsymbol{O}_{\text{glo}})}{\arg\max}\ \hat{\boldsymbol{Q}}_{\boldsymbol{\tau}}^{\text{glo}}(\boldsymbol{\tau}|\boldsymbol{O}_{\text{glo}}, \boldsymbol{O}_{\text{loc}}, \boldsymbol{O}_p, \boldsymbol{l}) \tag{10}$$

where $\hat{\boldsymbol{Q}}_{\boldsymbol{\tau}}^{\text{glo}}$ is predicted confidence score distribution of the 1m^3 global voxel grid $\boldsymbol{O}_{\text{glo}}$. Centered on $\hat{\boldsymbol{\tau}}_t^{\text{coa}}$, a fine cubic space is further voxelized finely into $\boldsymbol{O}_{\text{fin}}$, which is regarded as part of visual observations in the second stage

$$\hat{\boldsymbol{\tau}}_t = \underset{\boldsymbol{\tau}\in\boldsymbol{A}_{\boldsymbol{\tau}}(\boldsymbol{O}_{fin})}{\arg\max}\ \hat{\boldsymbol{Q}}_{\boldsymbol{\tau}}^{\text{fin}}(\boldsymbol{\tau}|\boldsymbol{O}_{\text{fin}}, \boldsymbol{O}_{\text{loc}}, \boldsymbol{O}_p, \boldsymbol{l})\,. \tag{11}$$

$\hat{\boldsymbol{\tau}}_t$ is the predicted next position to be achieved. Benefit from standard FP2AT, the first stage preliminarily provides a decent prior. The second stage pays soft attention to hard-attended local spaces with higher resolution, disregarding the rest of the global space. The confidence distribution of rotations $\hat{\boldsymbol{Q}}_{\boldsymbol{\varphi}}^{\text{fin}}$, effector's state $\hat{\boldsymbol{Q}}_g^{\text{fin}}$ and collision indicator $\hat{\boldsymbol{Q}}_c^{\text{fin}}$ are also predicted in the second fine stage, i.e.,

$$\left[\hat{\boldsymbol{Q}}_{\boldsymbol{\tau}}^{\text{fin}}, \hat{\boldsymbol{Q}}_{\boldsymbol{\varphi}}^{\text{fin}}, \hat{\boldsymbol{Q}}_g^{\text{fin}}, \hat{\boldsymbol{Q}}_c^{\text{fin}}\right] = \mathcal{MODEL}^{\text{fin}}(\boldsymbol{a}|\boldsymbol{O}_t, \boldsymbol{l}). \tag{12}$$

The observation resolution of the fine voxel grid $\boldsymbol{O}_{\text{fin}}$ is configured to match that of the local voxel grid $\boldsymbol{O}_{\text{loc}}$. Hence, the 3-D visual fusion attention works across different scales in the first stage, while it works across different spatial locations in the second stage.

D. Loss Function Design

As mentioned in Section IV-A, FP2ATs are designed for predicting confidence scores for possible discrete actions instead of regressing a definite action. The action with the highest confidence score will be executed. Therefore, we consider the prediction of optimal actions as a classification problem. Below, we describe loss functions using the notation of standard FP2AT, which can be easily adapted to C2F-FP2AT and FP2AT-VMA.

For 3-D positions, the task space has been naturally discretized by the global voxel grid in the visual perception phase, so the confidence score at each voxel will be learned and predicted. The cross-entropy loss is utilized to characterize the divergence between labels and predictions. In other words, the loss function can interpret raw classifying confidence scores as probabilities by the softmax function, which takes the form

$$\mathcal{L}_{\boldsymbol{\tau}} = -\sum_{k_{\boldsymbol{\tau}}=\boldsymbol{\tau}_0}^{\boldsymbol{\tau}_m} q_{k_{\boldsymbol{\tau}}} \log \frac{e^{\hat{q}_{k_{\boldsymbol{\tau}}}}}{\sum_{i_{\boldsymbol{\tau}}=\boldsymbol{\tau}_0}^{\boldsymbol{\tau}_m} e^{\hat{q}_{i_{\boldsymbol{\tau}}}}} \tag{13}$$

where $\boldsymbol{\tau}_m$ denotes the class with the maximum discrete value range in the discretized positional space $\boldsymbol{A}_{\boldsymbol{\tau}}$, and the indices $k_{\boldsymbol{\tau}}$ and $i_{\boldsymbol{\tau}}$ start from the class with the minimum discrete value range ($\boldsymbol{\tau}_0$). Besides, q and $\hat{q}$ represent the ground-truth and predicted confidence scores, respectively. On the basis of (2) and (5), $q_{j_{\boldsymbol{\tau}}}$ and $\hat{q}_{j_{\boldsymbol{\tau}}}$, the ground-truth and predicted confidence scores of specific discrete position $\boldsymbol{\tau}_j$ are obtained by

$$q_{j_{\boldsymbol{\tau}}} = \boldsymbol{Q}_{\boldsymbol{\tau}}(\boldsymbol{\tau}_j) \tag{14a}$$

$$\hat{q}_{j_{\boldsymbol{\tau}}} = \hat{\boldsymbol{Q}}_{\boldsymbol{\tau}}(\boldsymbol{\tau}_j) \tag{14b}$$

$$\boldsymbol{\tau}_j \in \boldsymbol{A}_{\boldsymbol{\tau}} \tag{14c}$$

where $\hat{\boldsymbol{Q}}_{\boldsymbol{\tau}}$ are confidence score distributions in the action space $\boldsymbol{A}$ learned and predicted following (5) and Section IV-A.

Similarly, the classifications of rotations, end effector's state, and collision indicator in corresponding discretized spaces $\boldsymbol{A}_{\boldsymbol{\varphi}}$, $\boldsymbol{A}_g$, $\boldsymbol{A}_c$ are trained with loss functions $\mathcal{L}_{\boldsymbol{\varphi}}$, $\mathcal{L}_g$, and $\mathcal{L}_c$, respectively, i.e.,

$$\mathcal{L}_{\boldsymbol{\varphi}} = -\sum_{k_{\boldsymbol{\varphi}}=\boldsymbol{\varphi}_0}^{\boldsymbol{\varphi}_m} q_{k_{\boldsymbol{\varphi}}} \log \frac{e^{\hat{q}_{k_{\boldsymbol{\varphi}}}}}{\sum_{i_{\boldsymbol{\varphi}}=\boldsymbol{\varphi}_0}^{\boldsymbol{\varphi}_m} e^{\hat{q}_{i_{\boldsymbol{\varphi}}}}} \tag{15}$$

$$\mathcal{L}_g = -\sum_{k_g=0}^{1} q_{k_g} \log \frac{e^{\hat{q}_{k_g}}}{\sum_{i_g=0}^{1} e^{\hat{q}_{i_g}}} \tag{16}$$

$$\mathcal{L}_c = -\sum_{k_c=0}^{1} q_{k_c} \log \frac{e^{\hat{q}_{k_c}}}{\sum_{i_c=0}^{1} e^{\hat{q}_{i_c}}}. \tag{17}$$

We employ an overall loss function $\mathcal{L}_{\text{all}}$, which is a linear combination of $\mathcal{L}_{\boldsymbol{\tau}}$, $\mathcal{L}_{\boldsymbol{\varphi}}$, $\mathcal{L}_g$, and $\mathcal{L}_c$:

$$\mathcal{L}_{\text{all}} = \lambda_{\boldsymbol{\tau}}\mathcal{L}_{\boldsymbol{\tau}} + \lambda_{\boldsymbol{\varphi}}\mathcal{L}_{\boldsymbol{\varphi}} + \lambda_g\mathcal{L}_g + \lambda_c\mathcal{L}_c. \tag{18}$$

The $\lambda_{\boldsymbol{\tau}}$, $\lambda_{\boldsymbol{\varphi}}$, λ_g, and λ_c are weights of the corresponding loss functions, sharing the equivalent value of 1 in our case.

E. Implementation Details of Proposed FP2AT

The training process is a particular case of optimization, i.e., minimizing the value of $\mathcal{L}_{\text{all}}$ by finding the proper parameters of the FP2ATs. To resolve this optimization, the LAMB optimizer (Layer-wise Adaptive Moments optimizer for Batch training) is applied owing to its efficiency for attention-based models [46], cooperating with a cosine learning scheduler with a maximum learning rate of 0.0005. After the FP2ATs are trained well from expert demonstrations with low and stable overall loss, it is capable of selecting optimal actions from the discretized action space for unseen scenes according to predicted confidence score distributions and (6). Even for those with the randomness of the surroundings and observational noises caused by the sensing hardware system, the method will still select satisfactory actions.

We select proper regularization methods involving SE(3) data augmentation, dropout, and layer normalization to avoid overfitting when training. There is an inevitable gap between reality and simulation, and thus the real-world tasks are learned from scratch. The FP2AT was trained distributedly with a total batch size of 8 on 4 NVIDIA Tesla V100 GPUs for 250 K iterations (approximately 5 days) for RLBench and real-world manipulation tasks, respectively. The trained network can work using a single GPU with more than 5 GB of memory.

V. Simulation and Experimental Verifications

A. Simulated and Real-Robot Experimental Setup

The proposed FP2AT can work together with the position control mode of mainstream commercial robot arms, regardless of their configurations. To validate the generalization ability of the proposed method, we conducted simulated and real-robot experiments with different setups (see Fig. 8).

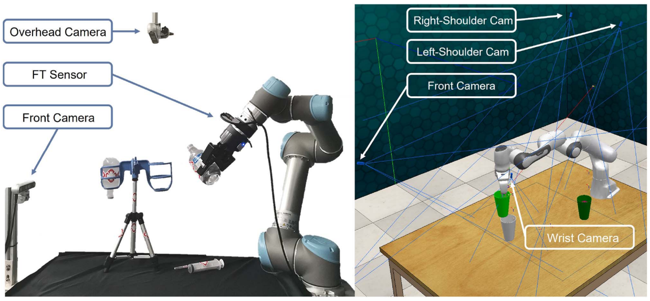


Fig. 8. Experimental setup of real robot (left) and simulation (right). Cameras were installed for visual observations. Proprioceptive data was obtained from the internal sensors of the robots and grippers, as well as the external FT sensor.

In the simulation, we implemented the FP2AT with the same experimental configurations as that of baselines [12], [13] for benchmarking in RLBench tasks [14]. More specifically, the 7-DOF Franka Emika Panda robot, camera setup (front, right-shoulder, left-shoulder, and wrist cameras), scene bounds, and language goals were uniform in different methods.

For real-world experiments, our algorithm was verified by a 6-DOF UR5 robot with a Robotiq 2F-85 gripper and an NBIT GLL93003BB0 FT sensor. In view of the limited minimum working distance of the depth channel, we installed two RealSense D455 cameras in front and overhead settings instead of the four-camera setting. Hence, in the voxel grid reconstruction stage, images from the front, wrist, left, and right cameras are applied for RLBench tasks, while only the front-view and overhead-view images are used for real-world manipulation tasks. The desired observations with actions were collected via our self-developed joystick demonstration package to avoid visual occlusion in drag-and-replay demonstration production. Note that the collected demonstration data was selected and thus the cases with unsatisfactory performance were eliminated.

For a fair comparison, we compared the FP2AT with baselines under the same experimental setup in simulation and real-robot experiments, respectively.

B. Baselines and Metrics

Voxel-Grid-Based Methods: The proposed FP2ATs were first compared with two baselines, which took voxel grids as visual inputs but employed them in distinct ways. The C2FARM-BC [12], [13] first voxelized the whole 1 m^3 scene into a 32^3 voxel grid and predicted the most confident location, then took that location as the center and zoomed into a local 32^3 voxel grid within 0.15^3 m^3. The PerAct [13] was purely fed with a 100^3 global voxel grid in the 1 m^3 space, lacking local vision for fine-grained learning.

Point-Cloud-Based Methods: PolarNet [37] downsampled dense point clouds to sparse point clouds with the resolution of one point per cm^3, then applied architecture and pretrained weights of PointNext [47] to encode preprocessed point clouds and decode actions. Act3D analogously [39] utilized a coarse-to-fine mechanism, which progressively paid hard attention to local 3-D point clouds in three stages (i.e., 1-meter-side cube → 16-centimeter-diameter sphere → 4-centimeter-diameter sphere), and then predicted the next action. Hence, local 3-D vision is an intermediate output of hard attention methods (C2FARM-BC and Act3D).

Compared with our FP2ATs, the baselines have not developed any soft attention mechanisms or multiscale inputs for 3-D visual perception. Furthermore, we provided some detailed comparisons in Section V-F to investigate the effectiveness of the major innovations of our FP2AT, i.e., Human-Inspired 3-D Visual Fusion Attention, cross-layer feature aggregation, and sufficient proprioception data (including joint positions, joint forces, forces, and torques of the end effector).

Two metrics were utilized to evaluate the performance of different methods. In the simulation, we first trained networks on multiple RLBench tasks with 100 demonstrations for each one and then tested each task on 25 new scenarios with all variation settings to calculate the SR following [13], [37]. Moreover, the average number of key actions (ANKA) was introduced to represent the efficiency and planning capabilities of methods, and those with fewer key actions for successful execution constituted better ones. For the same key action sequence predicted by one method, different low-level motion planners and kinematic settings (e.g., velocities and accelerations) will take different execution times. In other words, many external factors may impact the execution time, and thus, it is inappropriate to choose execution time as a metric in this study. Hence, the ANKA is more generic than execution time, providing a unified evaluation metric for different methods deployed in different robot embodiments, implementation platforms (simulation or real robots), motion planners, kinematic settings, etc. As for real-robot experiments, we trained each task from 10 demonstrations and evaluated the performance with the two aforementioned metrics on another 10 unseen scenarios. Different methods were compared in 10 fine-grained manipulation tasks (five in the real world and five in simulation, as shown in Fig. 2) involving diverse high-precision manipulation primitives (we defined nine different primitives, as the x-axis in Fig. 9).

C. Increase in SR

In order to verify the merits of our methods, five challenging RLBench tasks were tested. The common reasons for task failure included low-precision manipulation, over contact, and collision between the robot and objects. Table I compares our FP2AT with the voxel-grid-based baselines on the SR of each task. The FP2AT outperformed the previous leading methods by at least 28% in nine of the 10 tasks, including grasping and placing the wine on the specific location of the rack, screwing the bulb into the lampstand, and stacking cups on the table. They are such challenging tasks with complex contact dynamics and uncertainties of the relative poses between objects and targets, acquiring extraordinarily low SRs (less than 25%) by baselines.

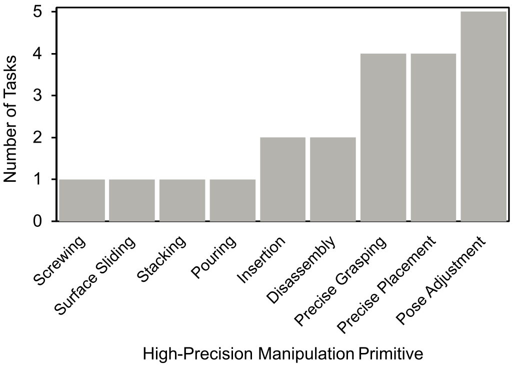


Fig. 9. High-precision manipulation primitives required in the ten different tasks. Primitives with taller bars denote the more popular basic manipulations among the different tasks in Fig. 2. For example, there are two tasks requiring insertion, i.e., tasks of screwing the bulb and stacking cups in Fig. 2, and four tasks requiring precise placement, i.e., tasks of putting the IV bottle into the holder, throwing away waste, grasping and placing the wine bottle, and putting bank notes on the shelf in Fig. 2.

TABLE I
SR Comparison for Voxel-Grid-Based Methods (%)

RLBench Tasks	**Slide Block by Stick**	**Put on Shelf**	**Grasp and Place Wine**	**Screw Bulb in**	**Stack Cups**	**Avg**
C2FARM-BC	24	12	8	8	0	10.4
PerAct	68	44	12	24	0	29.6
FP2AT-VMA	44	56	32	36	24	38.4
FP2AT	**84**	**72**	**60**	**56**	**48**	**64.0**
Real-robot Tasks	**Throw away Waste**	**Pour Pills**	**Pull out Syringe**	**Open Pill Bottle**	**Put Bottle into Holder**	**Avg**
PerAct	40	50	20	40	10	32.0
FP2AT	**80**	**80**	**70**	**70**	**50**	**70.0**

Overall, the proposed FP2AT achieved state-of-the-art performance in these RLBench tasks with average enhancements of approximately 34.4% and 53.6% compared with PerAct [13] and C2FARM-BC [12], [13], respectively. Meanwhile, FP2AT-VMA also outperformed PerAct and C2FARM-BC with improvements of 8.8% and 28.0% in overall SR, respectively. This suggests that VMA-3D can also enhance voxel grid observation for manipulation, but not as effectively as HVFA-3D.

Afterward, five real-world manipulation tasks were trained and tested as exemplified in Fig. 2. 1) grasping and putting the intravenous (IV) bottle into the bottle holder, 2) griping the specific pill bottle and pouring pills out of the bottle, 3) pulling the syringe out of the saline solution bottle, 4) picking up the waste and throwing it into the specific waste bin, and 5) removing the cap from the bottle. The training and testing scenes differed in terms of the object instances, 6-D poses of objects, the initial 7-D pose of the gripper, colors, and distractors. The methods were supposed to plan actions without obvious collisions, sometimes dealing with pose adjustment and rich contact. Consistent with the results of the RLBench simulation, the proposed FP2AT showcased a notable improvement of 38.0% on the averaged SR over the global-voxel-grid-based PerAct.

Qualitative results in Fig. 10 further presented the advantages of our method compared to the baselines. The previous

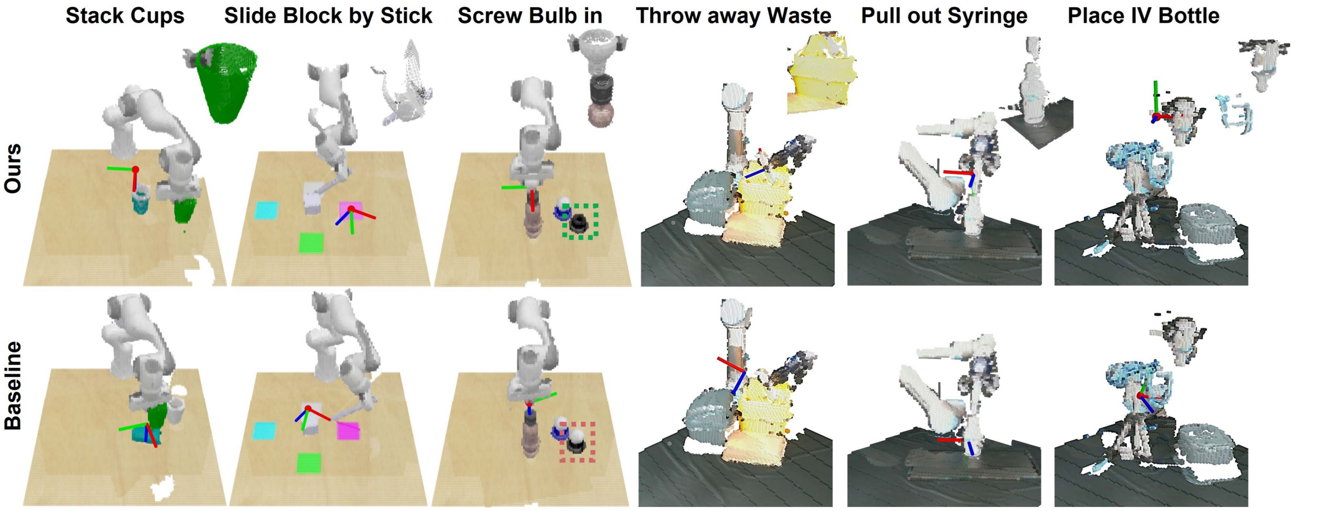


Fig. 10. Qualitative results of our FP2AT and PerAct. The methods were tested with the same initial scenarios for each task, and the next 7D manipulating poses were predicted based on observations. The FP2AT leveraged both global and local voxel grids as visual observations, while the baselines only took global voxel grids as visual observations. It performed better than baselines in planning, precise manipulation and collision avoidance.

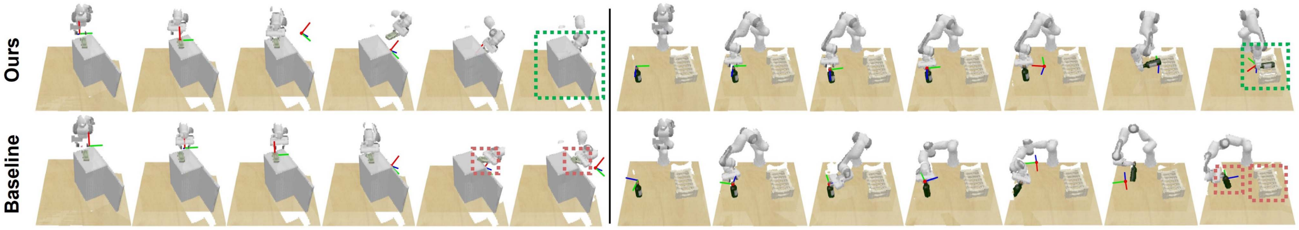


Fig. 11. Key actions performed by FP2AT and PerAct in different tasks: put the bank notes on the top shelf of the safe, grasp and place wine bottle on the middle of the rack. The baseline often requires more actions than the proposed FP2AT to achieve the goal, which is marked with a green dotted box.

methods faced challenges in planning, so they sometimes could not achieve the goals (see the examples of sliding the block and pulling out the syringe). The FP2AT dealt with precise manipulation better than the baselines, such as grasping in the task of screwing the bulb, stacking in the task of stacking cups, and placing in the task of throwing away the waste. Furthermore, the proposed method was more likely to avoid collisions than its counterparts, as shown in stacking cups and placing the IV bottle. To conclude, the HVFA-3D module in our proposed FP2AT provided excellent scene understanding and precise observation of relative poses between objects, coupled with proprioception of body movement and contact, thereby improving the overall performance.

D. Reduction in ANKA

The FP2AT can finish each kind of task with fewer actions than the baselines, benefiting from its robust planning capability. Fig. 11 showcases two example tasks with different predicted actions using diverse methods, which both succeeded in handling the two scenarios. However, the method requiring more actions (PerAct) tends to struggle with obstacles and rotations when FP2AT accomplishes the same tasks. For instance, in the put-on-shelf task, the baseline took additional steps to navigate around the obstacle of the safe's structure, leading to extra movements. Regarding rotations, as seen in the grasp-and-place-wine task, PerAct struggled with adjusting the end-effector's orientation, resulting in unnecessary rotational corrections compared to the concise actions of FP2AT. Generally, the fewer actions the robot takes, the less execution time is required for the identical implementation details (e.g., low-level motion planners and kinematic settings). The Appendix provides visual confirmation of this positive correlation. The average number of ground-truth key actions taken by experts (either joystick demonstrations or hard-coded programs) and different methods were recorded in Table II. The ground-truth numbers for real-robot tasks were not collected since our system directly inferred and executed actions during the testing phase without a test dataset as RLBench. We only counted the key actions for successful execution, and thus the examples with zero SR were not recorded. As the PerAct failed to accomplish any trials on the cup-stacking task, the average number (Avg-4) only factored in the remaining four tasks from the RLBench.

It has been demonstrated that the proposed FP2AT has a superior efficiency in both RLBench and real-robot tasks, surpassing the baselines with an average decrease in the number of key actions of 3.2 and 1.7, respectively. As for FP2AT-VMA, it could accomplish RLBench tasks with about two fewer

TABLE II
ANKA Comparison for Voxel-Grid-Based Methods

RLBench ***Tasks***	**Slide Block by Stick**	**Put on Shelf**	**Grasp and Place Wine**	**Screw Bulb in**	**Stack Cups**	**Avg-4**
Ground-truth	10.7	9.8	10.4	13.7	16.6	11.2
PerAct	**9.2**	19.1	9.9	19.8	–	14.5
FP2AT-VMA	10.6	18.3	7.1	13.5	21.4	12.4
FP2AT	10.1	**16.2**	**6.4**	**12.4**	**17.2**	**11.3**
Real-robot ***Tasks***	**Throw away Waste**	**Pour Pills**	**Pull out Syringe**	**Open Pill Bottle**	**Put Bottle into Holder**	**Avg**
PerAct	12.8	11.2	8.0	**4.5**	16.0	10.9
FP2AT	**10.8**	**10.3**	**6.6**	5.1	**13.4**	**9.2**

actions on average than PerAct. In most tasks, the proposed methods had fewer key action numbers than the other state-of-the-art methods. In addition, the FP2AT even took fewer actions than the expert demonstrations in some cases, viz., grasping and placing the wine bottle and screwing the bulb. These indicate that it has the potential to comprehend manipulation tasks and exceed the experts to some extent. Moreover, there is a correlation between the ANKA and the SR, namely that the method with concise actions usually has high SRs. This might be caused by precise planning and awareness of obstacles.

E. Performance Comparison With Point-Cloud-Based Methods

Sufficient simulation and real-world experiments demonstrated FP2AT's superiority over previous voxel-based approaches. Subsequently, we further compared FP2ATs with existing leading methods based on 3-D point clouds. The letters "v" and "p" were utilized to distinguish the two modalities, voxel grid and point cloud, respectively. A performance table of 3-D visual robot learning was provided, taking into account the prediction resolution, which uniformly represented the number of minimum representation units (points or voxels) per unit volume for prediction. For instance, Act3D [39] randomly samples 333 and 3333 candidate points for prediction within the 4-cm-diameter sphere at each action step t during training and testing, equivalent to approximately 10 and 100 candidate points in 1cm^3, respectively. In contrast, the proposed FP2AT merely predicts actions from just one glance at a resolution of 1 candidate voxel per cm^3, which is at least $\frac{1}{10}$ of Act3D. However, according to the results in Table III, FP2AT still showcased noticeable advantages in bulb-screwing and cup-stacking tasks, which are contact-rich and require stable control of rotations and translations. As for other fine-grained tasks, our FP2AT likewise showed decent performance leveraging the visual fusion attention mechanism (HVFA-3D), though with extremely lower resolution. Overall, Act3D performed somewhat better than FP2AT in some tasks due to its coarse-to-fine hard attention and 10× (100× for testing) resolution of candidate prediction.

As for the coarse-to-fine adaptation of the FP2AT, C2F-FP2AT, we voxelized its fine cubic space $\boldsymbol{O}_{\text{fin}}$ with a volume of 0.25^3 m^3 into a 100^3 voxel grid and acquired 64 candidate voxels per cm^3 for prediction. Subsequently, five RLBench tasks were trained, tested, and then compared in Table III, using the same evaluation episodes as Act3D. In line with FP2AT, C2F-FP2AT maintains the same prediction resolution during training and testing. By combining coarse-to-fine hard attention and improving the resolution of candidate prediction, C2F-FP2AT overall outperformed the previous SOTA point-cloud-based methods PolarNet and Act3D by 25.6% and 14.6% on average SR, respectively. In addition, it demonstrated an improvement of 15.2% over the standard FP2AT. These confirm the validity of the zoom-in law for our FP2AT. Nevertheless, it is noteworthy that the increase in prediction resolution will not always raise the performance. Namely, the zoom-in law has an upper limit for a specific method. For instance, Act3D has reported a performance drop when reducing the prediction volume to $\frac{1}{8}$ of original sizes and retaining the number of candidate points in the medium and finest stages (with a prediction resolution of 79.5 candidates/cm^3). Similar phenomena were observed in our C2F-FP2AT. Fig. 12 visualizes the training process of C2F-FP2AT and FP2AT on selected tasks, both undergoing 250 K iterations. The two models exhibited a similar overall trend, while the C2F-FP2AT required approximately 1.7x the training time of FP2AT to fully converge. C2F-FP2AT ultimately achieved a lower loss and fewer fluctuations than FP2AT, benefiting from its coarse-to-fine feature refinement and enhanced handling of complex tasks. Overall, these suggest that the zoom-in law of our method comes at the expense of time.

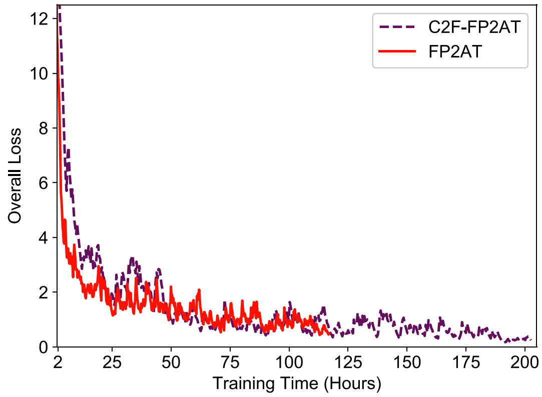


Fig. 12. Training curves of C2F-FP2AT and FP2AT on RLBench tasks.

F. Ablation Studies

We conducted ablation experiments with different modifications of the proposed FP2AT to study the importance of its key components, summarized in Table IV. The ablated models were trained on three tasks, i.e., grasping and placing the wine on the specific location of the rack, screwing the bulb into the lampstand, and grasping the stick to slide the cube onto a target position. These tasks required most of the manipulation primitives we performed in this article, involving grasping, placement, pose adjustment, insertion, screwing, and sliding. Hence, the results are relatively informative and practical for further reference.

First, the Human-Inspired 3D Visual Fusion Attention of our FP2AT was replaced by the VMA-3D, facing a considerable

TABLE III

PERFORMANCE COMPARISON WITH DIFFERENT 3D-VISION-BASED ROBOT LEARNING METHODS (%)

RLBench Tasks	**Modality**	**Testing Prediction Resolution (/cm^3)**	**Training Prediction Resolution (/cm^3)**	**Slide Block by Stick**	**Put on Shelf**	**Grasp and Place Wine**	**Screw Bulb in**	**Stack Cups**	**Avg**
C2FARM-BC	v	10	10	24	12	8	8	0	10.4
PerAct	v	1	1	68	44	12	24	0	29.6
FP2AT-VMA	v	1	1	44	56	32	36	24	38.4
PolarNet	p	1	1	92	84	40	44	8	53.6
FP2AT	v	1	1	84	72	60	56	48	64.0
Act3D	p	100	10	92	**95**	80	47	9	64.6
C2F-FP2AT	v	64	64	**99**	93	**87**	**63**	**54**	**79.2**

TABLE IV

ABLATIONS OF FP2AT

Model	**Avg Success Rate**	**Avg Num of Key Actions**
1 FP2AT-VMA	41.3	10.7
2 FP2AT w/o cross-layer feature aggregation	52.0	11.8
3 FP2AT using part of proprioceptive data	58.7	**9.1**
4 FP2AT	**65.3**	9.4

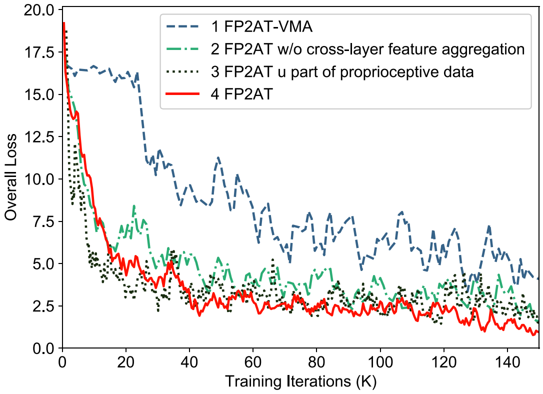


Fig. 13. Training curves of different ablated models.

reduction of over one-third in the average SR and a rise of 13.8% in the averaged key action number. Subsequently, the FP2AT without the cross-layer feature aggregation was tested and obtained a SR of 52.0%, with around 13.3% reduction compared with the proposed FP2AT method. Moreover, the ANKA was 11.8, necessitating around two additional actions for each execution. We then followed the PerAct [13] to train the FP2AT with only the gripper's open-close state, fingers' positions, and timestep rather than all of those proprioceptive data mentioned in Section IV-A. As a result, the performance of this method declined in average SR while slightly improved in the number of key actions. The results revealed that the HVFA-3D contributes the most to overall performance. With the assistance of cross-layer feature aggregation and comprehensive proprioception, the proposed FP2AT is able to achieve optimal performance.

Fig. 13 comprises training curves of different ablations. For simplification, the four methods in Table IV are denoted as Models 1-4. The proposed FP2AT tended to understand tasks fast with few proprioceptive data, according to the leading state of Model 3 in the initial stage. However, the FP2AT with all of the proprioceptive data converged stabler and slightly better than Model 3 from the middle stage of training. In contrast, Model 1 and Model 2 learned noticeably worse than the abovementioned two models, which is in line with the results in Table IV.

Moreover, some interesting phenomena were discovered in these ablation studies. For example, the FP2AT with VMA-3D initially struggled with large overall losses, as shown in Fig. 13. In addition, the FP2AT without cross-layer feature aggregation could achieve comparable performance to the optimal architecture after training for sufficient iterations (more than twice of FP2AT) in the bulb-screwing task, which has hardly ever been observed in other tasks. To conclude, the proposed method has outstanding performance on SR and ANKA, which are demonstrated by simulation and real-robot experimental results.

VI. CONCLUSION

This article has resolved 3-D robotic manipulation by sequentially predicting and achieving 7-D manipulating poses. It conducted a comprehensive study on multiscale 3-D visual attention mechanisms. HVFA-3D, inspired by flexible viewpoints, global awareness, and the local attention of humans, enhances visual perception and action planning by facilitating attention to the region of interest. Meanwhile, VMA-3-D also strengthens the voxel grid observations for manipulation, though less effective than HVFA-3-D. Besides, a proprioceptive encoder is developed to perceive robot movement and contact with its surroundings, leveraging internal, and external sensors. Combined with the cross-layer feature aggregation, the proposed FP2AT can effectively deal with fine-grained manipulation tasks in an end-to-end manner. Furthermore, the network's capabilities are boosted through Coarse-to-Fine FP2AT, which refines the action space incrementally for more precise predictions. Finally, sufficient simulation and real-world tasks have demonstrated the merits of the proposed FP2ATs in diverse manipulating primitive skills on different robot embodiments and camera placements. Specifically, they can achieve higher SRs with fewer key actions than previous SOTA methods based on either voxel grids or point clouds. The ablation studies verify the importance of HVFA-3D, cross-layer feature aggregation, and comprehensive proprioception.

According to the qualitative results, the proposed method can often plan collision-free actions. This merit benefits from the attended visual perception, which takes voxelized objects and environments as inputs. Consequently, the model is trained and inferred with knowledge of its surroundings despite not being

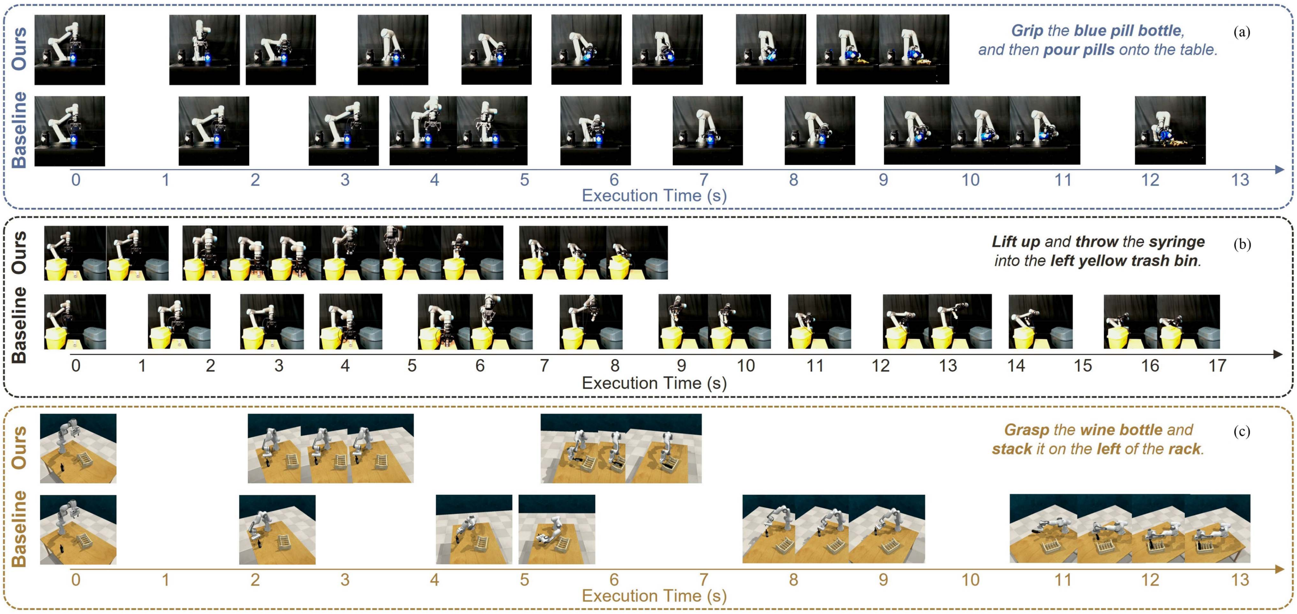


Fig. 14. Key actions with execution timestamps of FP2AT and PerAct across multiple manipulation tasks. The method with fewer key actions usually completes the same tasks in a shorter execution time.

explicitly pregiven in the workspace. Further, the ANKA can be applied as a metric for most methods of achieving task and motion planning with a sequence of actions, not just voxel-based ones. With actions formulated to 6-D poses of the end effector and 1-D functional indicators, our method can be adapted in a wide range of working scenarios for various robots with other tools except grippers, e.g., a vacuum suction cup with sucking and releasing functions, a welding torch with starting and stopping functions, and an ultrasound probe with scanning and lifting functions.

Although the FP2AT has achieved pleasant results in fine-grained RLBench and real-robot tasks, it remains challenging to reduce the computational costs. The voxel-grid-based deep model occupies more GPU memory than image-based models, and the training process is relatively time-consuming. This problem might be optimized by combining sparse representations or randomly sampled candidates of voxels in our future work. In addition, the quasi-static assumption limits the applicability of FP2AT in dynamic environments or those with moving obstacles. Since our method plans key actions in a step-wise manner, it has the potential to replan subsequent actions when the positions of objects change. Dynamic tasks would likewise benefit from real-time decision-making and adaptability. Addressing these issues would enable the system to perform more fluidly in scenarios that demand dynamic responses. Furthermore, since the action space is discrete, nonsmooth and inefficient issues may exist in tasks, which require continuous, curved motions. Besides, the reliance on a sampling-based motion planner introduces randomness in trajectory generation, which may affect consistency and performance in tasks that require precise, repeatable motions. For instance, in door-opening tasks, where the gripper must follow a smooth, continuous trajectory to handle the rotating hinge mechanism, the discrete action space can lead to inefficient and jerky movements. The sampling-based motion planner can result in inconsistent trajectories, making it difficult to reproduce the same precise movements reliably across trials. To address these issues, our method could benefit from incorporating continuous trajectory refinement after key actions. By learning a residual policy on top of discrete predictions, we may achieve smoother, more accurate, and more reproducible motions.

Appendix

For the average number of key actions (ANKA), assuming identical aforementioned implementation details, fewer key actions usually imply shorter execution times in practice. Fig. 14 provides multiple examples of predicted key actions with execution timestamps of different methods in the same task scenarios.

Acknowledgment

The author would like to thanks Yi Dong, Jiahang Liu, and Guoyao Ma for their valuable discussions of the experiments. This work was performed in part at the high-performance computing cluster, which is supported by the Information and Communication Technology Office of the University of Macau.

References

[1] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1334–1373, 2016.

[2] S. James and A. J. Davison, "Q-attention: Enabling efficient learning for vision-based robotic manipulation," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 1612–1619, Apr. 2022.

[3] T. Dai, K. Arulkumaran, T. Gerbert, S. Tukra, F. Behbahani, and A. A. Bharath, "Analysing deep reinforcement learning agents trained with domain randomisation," *Neurocomputing*, vol. 493, pp. 143–165, 2022.

[4] X. Xu, M. You, H. Zhou, Z. Qian, and B. He, “Robot imitation learning from image-only observation without real-world interaction,” *IEEE/ASME Trans. Mechatron.*, vol. 28, no. 3, pp. 1234–1244, Jun. 2023.

[5] C. Zeng, S. Li, Z. Chen, C. Yang, F. Sun, and J. Zhang, “Multifingered robot hand compliant manipulation based on vision-based demonstration and adaptive force control,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 9, pp. 5452–5463, Sep. 2023.

[6] C. Chi et al., “Diffusion policy: Visuomotor policy learning via action diffusion,” in *Proc. Robot.: Sci. Syst.*, 2023.

[7] J. Wang, X. Jia, T. Zhang, N. Ma, and M. Q.-H. Meng, “Deep neural network enhanced sampling-based path planning in 3D space,” *IEEE Trans. Automat. Sci. Eng.*, vol. 19, no. 4, pp. 3434–3443, Oct. 2022.

[8] Z. Wu et al., “3D shapenets: A deep representation for volumetric shapes,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1912–1920.

[9] S. Shi, Z. Wang, J. Shi, X. Wang, and H. Li, “From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 8, pp. 2647–2664, Aug. 2021.

[10] K. Wada, E. Sucar, S. James, D. Lenton, and A. J. Davison, “MoreFusion: Multi-object reasoning for 6D pose estimation from volumetric fusion,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14528–14537.

[11] Y. Yuan and A. Nüchter, “Uni-Fusion: Universal continuous mapping,” *IEEE Trans. Robot.*, vol. 40, pp. 1373–1392, 2024.

[12] S. James, K. Wada, T. Laidlow, and A. J. Davison, “Coarse-to-fine Q-attention: Efficient learning for visual robotic manipulation via discretisation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 13739–13748.

[13] M. Shridhar, L. Manuelli, and D. Fox, “Perceiver-actor: A multi-task transformer for robotic manipulation,” in *Proc. Conf. Robot Learn.*, 2023, pp. 785–799.

[14] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison, “RLBench: The robot learning benchmark & learning environment,” *IEEE Robot. Automat. Lett.*, vol. 5, no. 2, pp. 3019–3026, Apr. 2020.

[15] S. Yang, W. Zhang, R. Song, J. Cheng, H. Wang, and Y. Li, “Watch and act: Learning robotic manipulation from visual demonstration,” *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 53, no. 7, pp. 4404–4416, Jul. 2023.

[16] K. Fang et al., “Learning task-oriented grasping for tool manipulation from simulated self-supervision,” *Int. J. Robot. Res.*, vol. 39, no. 2-3, pp. 202–216, 2020.

[17] C. Yang, H. Wu, Z. Li, W. He, N. Wang, and C.-Y. Su, “Mind control of a robotic arm with visual fusion technology,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 9, pp. 3822–3830, Sep. 2018.

[18] W. Gao and R. Tedrake, “kPAM 2.0: Feedback control for category-level robotic manipulation,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2962–2969, Apr. 2021.

[19] S. Xu, K. Chen, Y. Ou, Z. Wang, and C. Yang, “A learning-based object tracking strategy using visual sensors and intelligent robot arm,” *IEEE Trans. Automat. Sci. Eng.*, vol. 20, no. 4, pp. 2280–2293, Oct. 2023.

[20] J. Zhao, Z. Wang, L. Zhao, and H. Liu, “A learning-based two-stage method for submillimeter insertion tasks with only visual inputs,” *IEEE Trans. Ind. Electron.*, vol. 71, no. 7, pp. 7381–7390, Jul. 2024.

[21] Z. Zong, G. Song, and Y. Liu, “Detrs with collaborative hybrid assignments training,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 6748–6758.

[22] Z. Wang and G. Tian, “Task-oriented robot cognitive manipulation planning using affordance segmentation and logic reasoning,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 9, pp. 12172–12185, Sep. 2024.

[23] B. Wen et al., “BundleSDF: Neural 6-DoF tracking and 3D reconstruction of unknown objects,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 606–617.

[24] H.-S. Fang et al., “AnyGrasp: Robust and efficient grasp perception in spatial and temporal domains,” *IEEE Trans. Robot.*, vol. 39, no. 5, pp. 3929–3945, Oct. 2023.

[25] X. Wang and Q. Xu, “Transferring grasping across grippers: Learning-optimization hybrid framework for generalized planar grasp generation,” *IEEE Trans. Robot.*, vol. 40, pp. 3388–3405, 2024.

[26] S. Karaman and E. Frazzoli, “Sampling-based algorithms for optimal motion planning,” *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, 2011.

[27] A. H. Qureshi, Y. Miao, A. Simeonov, and M. C. Yip, “Motion planning networks: Bridging the gap between learning-based and classical motion planners,” *IEEE Trans. Robot.*, vol. 37, no. 1, pp. 48–66, Feb. 2021.

[28] T. Marcucci, M. Petersen, D. von Wrangel, and R. Tedrake, “Motion planning around obstacles with convex optimization,” *Sci. Robot.*, vol. 8, no. 84, 2023, Art. no. eadf7843.

[29] R. Jangir, N. Hansen, S. Ghosal, M. Jain, and X. Wang, “Look closer: Bridging egocentric and third-person views with transformers for robotic manipulation,” *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 3046–3053, Apr. 2022.

[30] A. Vaswani et al., “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[31] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2018, pp. 1861–1870.

[32] M. Shridhar, L. Manuelli, and D. Fox, “CLIPort: What and where pathways for robotic manipulation,” in *Proc. Conf. Robot Learn.*, PMLR, 2022, pp. 894–906.

[33] T. R. Savarimuthu et al., “Teaching a robot the semantics of assembly tasks,” *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 5, pp. 670–692, May 2018.

[34] I. Akinola, J. Varley, and D. Kalashnikov, “Learning precise 3D manipulation from multiple uncalibrated cameras,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 4616–4622.

[35] A. Goyal, J. Xu, Y. Guo, V. Blukis, Y.-W. Chao, and D. Fox, “RVT: Robotic view transformer for 3D object manipulation,” in *Proc. Conf. Robot Learn.*, PMLR, 2023, pp. 694–710.

[36] J. Luo et al., “Multistage cable routing through hierarchical imitation learning,” *IEEE Trans. Robot.*, vol. 40, pp. 1476–1491, 2024.

[37] S. Chen, R. Garcia, C. Schmid, and I. Laptev, “PolarNet: 3D point clouds for language-guided robotic manipulation,” in *Proc. Conf. Robot Learn.*, 2023, pp. 1761–1781.

[38] Y. Ze et al., “GNFactor: Multi-task real robot learning with generalizable neural feature fields,” in *Proc. Conf. Robot Learn.*, PMLR, 2023, pp. 284–301.

[39] T. Gervet, Z. Xian, N. Gkanatsios, and K. Fragkiadaki, “Act3D: 3D feature field transformers for multi-task robotic manipulation,” in *Proc. 7th Annu. Conf. Robot Learn.*, 2023, pp. 3949–3965.

[40] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu, “Roformer: Enhanced transformer with rotary position embedding,” *Neurocomputing*, vol. 568, 2024, Art. no. 127063.

[41] A. Radford et al., “Learning transferable visual models from natural language supervision,” in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[42] A. Jaegle et al., “Perceiver IO: A general architecture for structured inputs & outputs,” in *Proc. Int. Conf. Learn. Representations*, 2022.

[43] A. Dosovitskiy et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” in *Proc. Int. Conf. Learn. Representations*, 2021.

[44] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[45] Q. Ma, L. Yu, S. Tian, E. Chen, and W. W. Ng, “Global-local mutual attention model for text classification,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 12, pp. 2127–2139, 2019.

[46] Y. You et al., “Large batch optimization for deep learning: Training BERT in 76 minutes,” in *Proc. Int. Conf. Learn. Representations*, 2020.

[47] G. Qian et al., “PointNext: Revisiting pointnet with improved training and scaling strategies,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 23192–23204.

Yangjun Liu received the B.S. degree in engineering mechanics from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2021. He is currently working toward the Ph.D. degree in electromechanical engineering with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Macau,, China.

He is a Joint Student with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. His research interests include robot learning, multimodal robotic manipulation, and embodied AI.

Sheng Liu received the B.S. degree in automation from Harbin Institute of Technology, Harbin, China, in 2021, and the M.S. degree in electronic science and technology from Southern University of Science and Technology, Shenzhen, China, in 2024. He is currently working toward the Ph.D. degree in mechanical engineering with the Harbin Institute of Technology, Shenzhen.

He is a Joint Student with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His research interests include robot control and manipulation.

Binghan Chen received the B.S. degree in communication engineering from Zhengzhou University, Henan, China, in 2020, the M.S. degree in robotics from Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China, in 2024.

His research interests include magnetic robotic systems and guidewire steering.

Zhi-Xin Yang (Member, IEEE) received the B.Eng. degree in mechanical engineering from the Huazhong University of Science and Technology, Wuhan, China, and the Ph.D. degree in industrial engineering and engineering management from the Hong Kong University of Science and Technology, Hong Kong, in 1992 and 2000 respectively.

He is currently an Associate Professor with the State Key Laboratory of Internet of Things for Smart City and the Department of Electromechanical Engineering, University of Macau, Macau, China. His current research interests include machine vision-based intelligent robot, data-driven fault diagnosis, and Internet of Things-based safety monitoring.

Sheng Xu (Senior Member, IEEE) received the B.S. degree from Shandong University, Shandong, China, in 2011, the M.S. degree from Beihang University, Beijing, China, in 2014, both in electrical engineering, and the Ph.D. degree in telecommunications engineering from ITR, University of South Australia, Australia, in 2017.

He was a Postdoctoral Researcher with the Center for Intelligent and Biomimetic Systems, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China, for two years, where he is currently an Associate Professor. His main research interests include imitation learning, learning-based robot control, target tracking, optimization and statistical signal processing.