2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

3D Affordance Keypoint Detection for Robotic Manipulation

Zhiyang Liu[1], Ruiteng Zhao[1], Lei Zhou[1], Chengran Yuan[1], Yuwei Wu[2], Sheng Guo[1], Zhengshen Zhang[1], Chenchen Liu[1], Marcelo H. Ang Jr.[1] and Francis EH Tay[1]

***Abstract*— This paper presents a novel approach for affordance-informed robotic manipulation by introducing 3D keypoints to enhance the understanding of object parts' functionality. The proposed approach provides direct information about *what* the potential use of objects is, as well as guidance on *where* and *how* a manipulator should engage, whereas conventional methods treat affordance detection as a semantic segmentation task, focusing solely on answering the *what* question. To address this gap, we propose a Fusion-based Affordance Keypoint Network (FAKP-Net) by introducing 3D keypoint quadruplet that harnesses the synergistic potential of RGB and Depth image to provide information on execution position, direction, and extent. Benchmark testing demonstrates that FAKP-Net outperforms existing models by significant margins in affordance segmentation task and keypoint detection task. Real-world experiments also showcase the reliability of our method in accomplishing manipulation tasks with previously unseen objects. Our source code and video demo will be public.**

I. INTRODUCTION

Autonomous robotic manipulation requires robots to understand the various potential functions of objects and this understanding is referred to as "affordance" [1]. Unlike other properties such as object pose that solely describes the object itself, affordances consider the functional interactions between an object's parts and humans or robots [2]. According to recent studies, the affordance detection for object parts has been approached as a semantic segmentation problem [3], [4], [5], [6], [7], [8], [9], [10] where affordances are predicted by grouping pixels with similar functionality into a single category. However, to perform manipulation tasks, it is crucial to identify not only *what* the object's affordances are but also *where* a manipulator should manipulate the object with those affordances, as well as *how* the actions should be performed by the manipulator associated with those affordances. For instance, when using a manipulator to cut a sausage, we need to consider not only the affordances of "cut" and "grasp" but also specific details such as the position and orientation for grasping, as well as the cutting contact point position and direction. In short, affordance semantic segmentation provides what affordances exist, but not where or how they should be executed.

To bridge this gap, [11] firstly proposed an action-level approach for object parts, representing objects by pixel-wise affordance labels and 2D image keypoints per affordance, which could yield enhanced performance by combining affordance with keypoints. However, the image keypoints fail to capture the object's geometric features and depend heavily on post-processing techniques for application in real-world scenarios. This deficiency in spatial awareness complicates the differentiation of affordances that appear similar, such as the 'contain' and 'scoop' affordances, which, despite their resemblance, differ significantly in size. Moreover, the reliance on 2D keypoints necessitates further post-processing to translate these points into practical applications, especially for tasks involving object manipulation. An example of this limitation is evident in the wrap-grasp action; if the predicted keypoints deviate slightly from the object's surface—even by a few pixels—the action is likely to fail.

Hence, incorporating both geometric and appearance information into keypoints representation is crucial. It not only helps distinguish between similar affordances but also improves the accuracy of detection for subsequent manipulation tasks. In our work, leveraging this insight, we assign four 3D keypoints to each affordance region, forming the corresponding execution position, direction, and extent, which are explicit representations of where and how to manipulate. For example, as depicted in Figure 1, when cutting a sausage with a knife, we assign two sets of four 3D keypoints for grasp affordance and cut affordance respectively: for grasping, keypoints 3 and 4 determine the position and orientation of the grasp, and keypoint 2 provides the connection with cut affordance; for cutting, keypoint 3 provides a connection with grasp affordance, and keypoint 2 indicates the contact point of the sharp edge to cut, while the direction from keypoint 1 to keypoint 2 indicates the cutting direction.

In summary, the contribution of this work is twofold. Conceptually, to the best of our knowledge, it is the first to introduce four 3D keypoints to affordance detection, providing a novel representation to guide *where* and *how* a manipulator should engage. Regarding implementation, a novel Fusion-based Affordance Keypoint Detection Network (FAKP-Net) is proposed, which outperforms existing models by significant margins in affordance segmentation task and 3D keypoint detection task on the UMDKP dataset, augmented from UMDGT dataset [11]. Real-world experiments of affordance-informed manipulation tasks demonstrate the generalizability of the proposed approach to previously unseen objects.

II. RELATED WORK

Affordance Detection: Affordance detection has been the subject of numerous studies in the fields of computer vision

[1]Advanced Robotics Centre, National University of Singapore, 117576, Singapore, {zhiyang, ruiteng, leizhou, chengran.yuan, e0576004, zhengsheng_zhang, chenchen.liu}@u.nus.edu, mpeangh@nus.edu.sg, mpetayeh@nus.edu.sg
[2]Department of Mechanical Engineering, National University of Singapore, 117575, Singapore

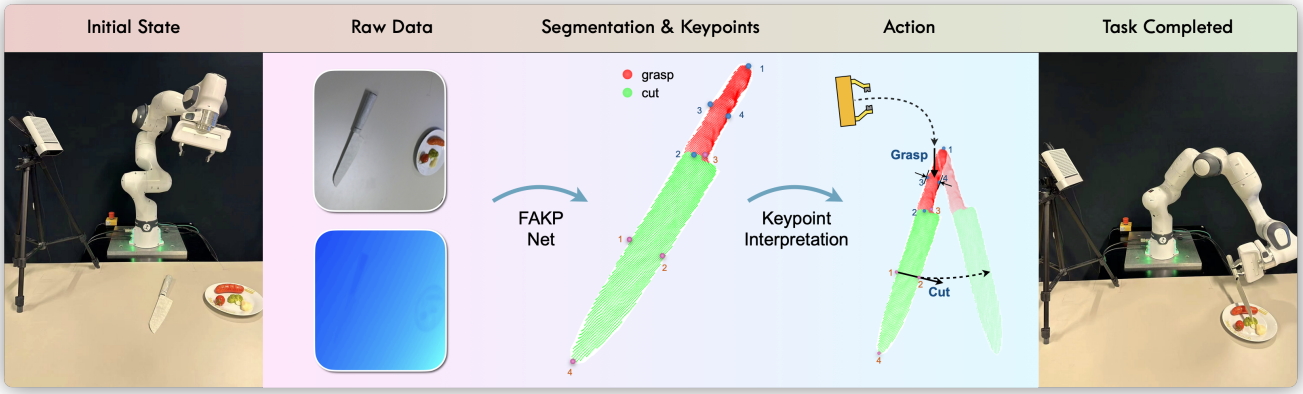

Fig. 1. Affordance-informed Manipulation Pipeline via 3D Keypoints. Each affordance region is represented by four 3D keypoints predicted from RGB-D image by FAKP-Net. Keypoints are interpreted as action position, direction and extent. The manipulator completes the task with provided execution information.

and robotics [3], [5], [9], [6], [12]. [3] finds corresponding affordances both for intra- and inter-class one-shot part segmentation. [6] introduced AffordanceNet, a learning-based end-to-end approach to accurately identify affordances for objects. [5] focused on learning affordance segmentation for real-world robotic manipulation using synthetic images, contributing insights into enhancing robotic manipulation tasks. [9] presented an adaptive binarization technique for weakly supervised affordance segmentation, contributing to the development of efficient detection methods. However, these works only output affordance labels on images, representing what the potential use of objects is.

Keypoint for Manipulation: The main applications of keypoints are to cast articulated human parts to joints, thus framing human pose estimation as a joint detection task. In the context of manipulation, we hypothesize 3D keypoints is able to provide the position and geometric details for manipulation tasks. Keypoint representations have been utilized in previous works [13], [14], [15], [11] to tackle the problem at various levels, including the task-level, category-level, and action-level. The KETO framework [15] is only able to predict a set of task-specific keypoints in a simulation environment, which is not enough to fully specify the robot's action, especially in the real-world scenarios with diverse task contexts and diverse object geometries. Category-level methods[13], [14] utilize a set of 3D keypoints to represent objects. However, these category-level methods are object-dependent, and they greatly benefit from prior information about the object's shape, especially when it comes to shape completion [14].

III. METHOD

The main challenge we aim to address is the simultaneous detection of point-wise affordances and their corresponding 3D keypoints from RGB-D images. To tackle this task, we propose a network with an encoder-decoder architecture called Fusion-based Affordance Keypoint Network (FAKP-Net), which is described in Figure 2. Specifically, given an RGB-D image as input, our network employs a feature extraction module to fuse appearance features and geometry features. The learned features are then passed through an affordance segmentation module and a 3D keypoint detection module respectively. The 3D keypoint detection module is trained to predict per-point offsets relative to keypoints, while the affordance segmentation module is trained to predict per-point semantic labels for object parts corresponding to affordances. Using the learned per-point offsets and labels, we apply the clustering algorithm [16] to predict a set of four 3D keypoints for each affordance region. These keypoints are utilized to interpret the position and direction information related to the affordance and how it can be effectively executed. For example, as illustrated in Figure 2, there are two sets of keypoints associated with the grasp and cut affordances of a knife. For the grasp affordance, keypoints 3 and 4 represent the expected contact points for grasping, while keypoints 2 serve as a connection point with cut affordance. Similarly, for the cut affordance, the direction from keypoint 1 to 2 forms the operating direction, with keypoint 2 also indicating the expected contact point for cut execution. Keypoints 3 in this case also serve as a connection point to connect with grasp affordance.

Datasets and Training: The primary source of our data is the UMD dataset [10]. This was expanded as UMDGT dataset [11], which added five keypoints for each affordance region in RGB-D images. Within the UMDGT dataset, the support affordance was excluded due to its inherent vagueness. Essentially, any object with a surface could be deemed supportive, blurring its identification as a primary affordance. Given tables are commonly used in manipulation tasks, there's a risk of misidentifying them with a support affordance, confusing them with background categories. Consequently, the UMDGT dataset offers six affordances: grasp, cut, scoop, contain, pound, and wrap-grasp. These images were obtained using a Kinect camera for the original UMD objects [10]. Within the five 2D keypoints, the center one often isn't on the object surface, impacting the uptake of geometric features. Moreover, this center point can be inferred from the other keypoints. Therefore, we enhanced

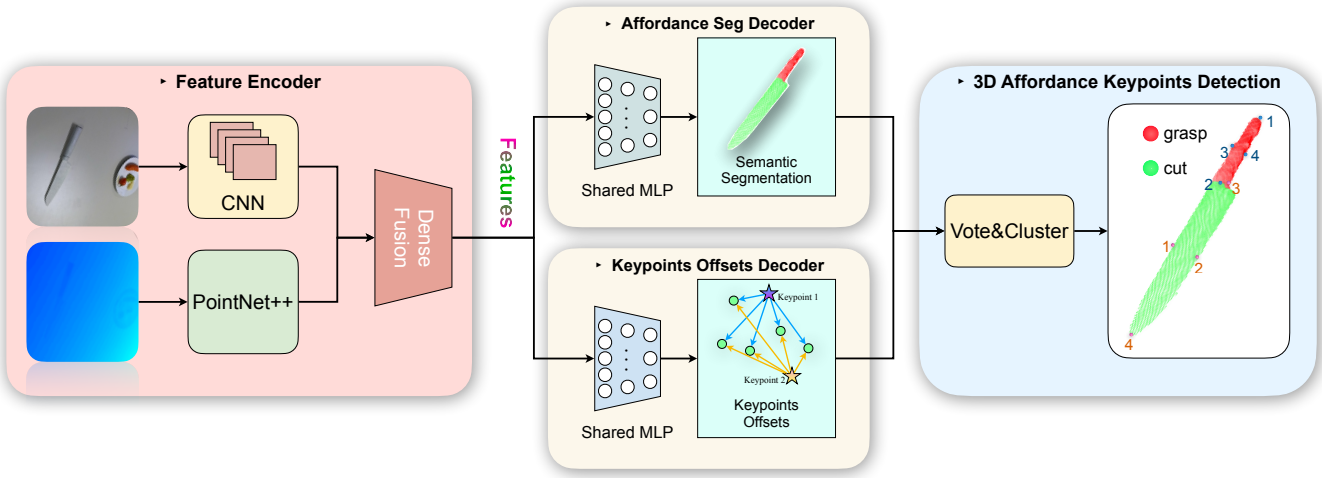

Fig. 2. Overview of FAKP-Net. The feature encoder processes an RGB-D image to extract per-point features. These features are then fed into affordance segmentation decoder and keypoints offsets decoder respectively, predicting per-point semantic labels and per-point translation offsets relative to keypoints. A clustering algorithm is then used to distinguish different affordance regions with the same semantic labels, and points within the same affordance region vote for their keypoints.

the UMDGT dataset by removing the center keypoint and repositioning the others on the object surface. We maintained the initial training-test split from the UMD dataset to ensure consistent comparison in affordance segmentation. Further details are in our source code.

Feature Encoder: As shown in Figure 2, the feature encoder consists of two components: a PSPNet [17] with a pretrained ResNet34 [18] on the ImageNet dataset [19] for extracting appearance information from RGB images, and a PointNet++ [20] for extracting geometric information from point clouds and normal maps. These two components are then fused together using a DenseFusion block [21] to acquire a fused feature representation for each seed point. After the feature encoder block, each point p_i is associated with a corresponding feature f_i.

Affordance Segmentation Decoder: In our formulation of the 3D affordance keypoints problem, we utilize two learning modules for each visible seed point, one for semantic labels and one for translation offsets to keypoints. We utilize shared Multi-Layer Perceptrons (MLPs) for both decoders and train them jointly using a multi-task loss.

Specifically, with the per-point extracted feature from the encoder, the affordance segmentation decoder predicts the per-point affordance labels. Focal Loss[22] is applied for the training:

$$\mathcal{L}_{semantic} = -\alpha\,(1 - c_i \cdot l_i)^{\gamma} \log\,(c_i \cdot l_i) \tag{1}$$

where α represents the balance parameter, γ denotes the focusing parameter, c_i is the predicted confidence that the i_{th} point belongs to each affordance category, and l_i is the one-hot representation of the ground truth affordance label. In our context, we designate seven labels; label zero is assigned for the background, while the other six are assigned for different affordance categories. We employ $\gamma = 2$ and $\alpha = (0.03, 0.12, 0.17, 0.21, 0.17, 0.2, 0.1)$ across all experiments.α is based on the frequency of occurrence of the affordance, the more frequent the occurrence the smaller the value.

3D Keypoints Offsets Decoder: As depicted in Figure 2, given the per-point extracted feature from feature encoder, a 3D keypoints offsets decoder predicts per-point Euclidean translation offset from visible points to four target keypoints. The coordinates of visible seed points, combined with the predicted per-point translation offsets, yield possible keypoint quadruplets—this is the voting process. Within a specific affordance region, the voted points are subsequently collected through clustering algorithms, and these clusters' centers are chosen as the voted keypoints.

Specifically, given input of visible points $\{p_i\}_{i=1}^{N}$ and a keypoint quadruplet $\{kp_j\}_{j=1}^{M=4}$ within a specific affordance region $R_{aff, aff \in \{'grasp','cut','scoop','contain','pound','w-grasp'\}}$, we represent $p_i = [x_i; f_i]$ where x_i refers to the xyz coordinates of seed point and f_i refers to the extracted feature, and $kp_j = [y_i]$ where y_i refers to the 3D coordinates of keypoint. Keypoints offsets decoder processes features f_i to output translation offset $\{of_i^j\}_{j=1}^{M=4}$ for each seed point, where $\{of_i^j\}$ represents translation offsets from i_{th} visible seed point to the j_{th} keypoint. The voted points can then be represented as $vkp_i^j = x_i + of_i^j$. To optimize of_i^j, we employ an L1 loss function:

$$\mathcal{L}_{keypoints} = \frac{1}{N}\Sigma_{i=1}^{N}\Sigma_{j=1}^{M}||of_i^j - of_i^{j*}||^{\amalg(p_i \in R_{aff})} \tag{2}$$

with of_i^{j*}, the ground truth of translation offsets; of_i^j, the predicted translation offsets; $M = 4$, the number of target keypoints for one affordance region; N, the total number of seed points; $\amalg$, the indicator function equal to 1 only when point p_i belonging to affordance region R_{aff}, and 0 otherwise.

Multi-task Loss: To jointly supervise the learning of per-point affordance semantic labels and per-point keypoints offsets, a multi-tasks loss is applied by a loss weighting λ ($\lambda = 100$):

$$\mathcal{L}_{multi-task} = \mathcal{L}_{keypoints} + \lambda\mathcal{L}_{semantic} \tag{3}$$

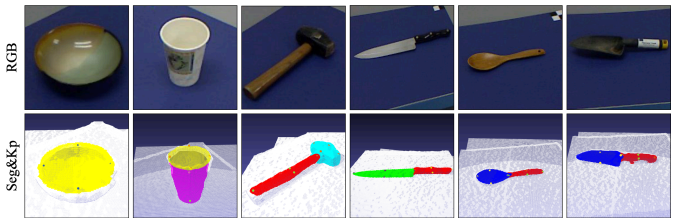

Fig. 3. Visualization of Affordance Segmentation and 3D numbered keypoints on UMDKP test dataset. The color coding for the affordance categories is as follows: white for the background; yellow for contain; purple for w-grasp; red for grasp; cyan for pound; green for cut; blue for scoop. Four numbered 3D keypoints attach with each affordance region.

IV. EVALUATION

Affordance Segmentation: The evaluation of affordance segmentation requires a comparison between valued probability masks and binary labels of ground truth for each class. We employ the F_β^ω metric [23] to evaluate these masks:

$$F_\beta^\omega = \left(1+\beta^2\right)\frac{P^\omega \cdot R^\omega}{\beta^2 \cdot P^\omega + R^\omega}. \tag{4}$$

where P^ω denotes weighted precision and R^ω denotes weighted recall, with $\sigma^2 = 5$, $\alpha = \frac{ln0.5}{5}$, and $\beta = 1$. Higher weights are assigned to points closer to the ground-truth foreground.

3D Affordance Keypoints: Normalized Mean Squared Error (NMSE) and Percentage of Correct 3D Keypoints(PCK3D) are commonly used evaluation metrics for keypoints in human pose estimation. Similarly, for each $aff, aff \in \{'grasp','cut','scoop','contain','pound','w-grasp'\}$, we employ these two metrics to evaluate 3D affordance keypoints.

$$NMSE_{aff} = \frac{1}{N_{aff}M}\Sigma_{i=1}^{N_{aff}}\Sigma_{j=1}^{M}\frac{||kp_i^j - kp_i^{j*}||_2}{d_{aff}} \tag{5}$$

where kp_i^{j*} denotes the ground-truth keypoints; kp_i^j denotes the predicted keypoints; $M = 4$ denotes the number of target keypoints for each affordance region; N_{aff} denotes the total number of affordance regions for each aff in the test dataset.

The normalized factor d_{aff}:

$$\begin{aligned} d_{aff} &= \frac{1}{N_{aff}M}\sum_{i=1}^{N_{aff}}\sum_{j=1}^{M}\left\|kp_i^{j*} - \bar{kp}_i\right\|_2, \\ \bar{kp}_i &= \frac{1}{M}\sum_{j=1}^{M} kp_i^{j*} \end{aligned} \tag{6}$$

The normalized factor d_{aff} is also utilized as the threshold of $PCK3D$

$$PCK_{aff}@0.3 = \frac{C_{0.3d_{aff}}}{C_{aff}} \tag{7}$$

where $0.3d_{aff}$ determines the maximum allowable distance between the predicted keypoint and the ground truth keypoint to be considered correct; $C_{0.3d_{aff}}$ counts the number of correct keypoint quadruplets; C_{aff} counts the total number of ground-truth keypoint quadruplets. The keypoints that are not predicted and misclassified are regarded as incorrect keypoints.

TABLE I

AFFORDANCE SEGMENTATION PERFORMANCE

Affordance	Weighted F-measures						
	Region-proposal		Encoder-Decoder				
	AffNet	SRF	AffCorrs	ED-RGB	DeepLab	AffKP*	Ours
grasp	0.73	0.31	0.65	0.72	0.62	0.72	**0.74**
cut	0.76	0.41	0.81	0.74	0.60	**0.89**	**0.89**
scoop	0.79	0.48	0.81	0.74	0.80	0.78	**0.83**
contain	0.83	0.64	0.87	0.82	0.90	0.86	**0.92**
pound	0.84	0.67	0.87	0.81	**0.88**	0.79	**0.88**
w-grasp	0.81	0.26	0.89	0.77	0.73	0.82	**0.92**
average	0.80	0.46	0.82	0.77	0.76	0.81	**0.86**

TABLE II

AFFORDANCE-ASSOCIATED 3D KEYPOINT PERFORMANCE

Affordance	grasp	cut	scoop	contain	pound	w-grasp	mean
AffKP NMSE	0.37	0.26	0.33	0.42	0.82	0.31	0.42
AffKP PCK@0.3	56.37	65.96	59.78	58.49	24.52	71.06	56.03
FAKP NMSE	0.39	0.27	0.28	0.18	0.35	0.16	0.27
FAKP PCK@0.3	42.91	64.07	66.06	82.96	52.57	97.10	67.61

V. RESULTS

In this section, we discuss the performance of FAKP-Net based on the metrics of affordance segmentation and associated 3D keypoint detection. Visualizations of FAKP-Net outputs on the UMDKP test dataset in Figure 3 show that FAKP-Net can simultaneously predict affordance semantic labels and per-affordance 3D keypoints. The performance of affordance segmentation is summarized in Table I and the performance of 3D keypoints is shown in Table II and Figure 4.

Affordance Segmentation Performance: In Table I, this evaluation includes a comparison with several baseline methods: AffordanceNet(AffNet)[6], SRF[10], AffCorrs[3], ED-RGB[8], DeepLab[24] and AffKP[11]. Except for AffKP, others follow a same train-test split rule as UMD dataset. For a fair comparison, we retrained AffKP and tested it with same split rule. The first two methods utilize region-proposal architectures, whereas the last four are based on simpler encoder-decoder structures. Those region-proposal methods simplify the problem by pre-processing the image to obtain regions of interest (RoIs) to get better performance. Nonetheless, as indicated by the results in Table I, our proposed FAKP-Net significantly outperforms not only existing encoder-decoder methods but also region-proposal methods, achieving state-of-the-art results across all affordance categories. Of particular note is the superior performance of our method in the wrap-grasp affordance category, where we surpass other methods by a margin of at least 10%. This can be attributed to the heightened sensitivity of the w-grasp affordance to geometric features, as opposed to appearance features. Our FAKP-Net is capable of effectively capturing these geometric features, which sets it apart from other methods. For instance, a mug can have complex patterns printed on its surface, leading to complicated appearance features and potentially hindering

the detection of wrap-grasp affordance regions. However, when geometric information such as the size and shape of the mug is fused with the appearance features, the detection of wrap-grasp affordance becomes more straightforward.

Affordance-associated 3D Keypoints Performance: In addition to affordance segmentation, the FAKP-Net concurrently generates 3D keypoints for each identified affordance region. For the evaluation of our predicted 3D keypoints, we employ metrics NMSE (Normalized Mean Squared Error) and PCK3D (Percentage of Correct 3D Keypoints). Notably, AffKp outputs 2D keypoints for each affordance region, and to fairly compare with our 3D affordance keypoints, we use the same way as how we get the point clouds to cast the 2D keypoints generated from AffKp to 3D keypoints. The results is shown in Table II. The NMSE score quantifies the average error, normalized with respect to the constant d_{aff}, which is specific to each affordance class. This constant reflects the average distance of the four keypoints from their mean center point. The PCK3D score quantifies the percentage of correct keypoints.

As in Table II, FAKP exhibits varying performance across six distinct affordances. The wrap-grasp and contain affordances both demonstrate low NMSE scores and high PCK@0.3. Notably, they both also exhibit high weighted F-measures at 0.92, indicating successful capture of their respective appearance and geometric features. The scoop and cut affordances demonstrate relatively moderate performance. In contrast, the grasp and pound affordances exhibit poorer performance in terms of weighted F-measures, NMSE, and PCK@0.3. This poorer performance can be attributed to their inherently ambiguous nature. For instance, both the handle and the head of a hammer are black and graspable, and distinguishing features between the handle and the head of the hammer are not always clear.

Compared with AffKP, FAKP showcases enhanced performance, with a notable increase of 11.58% in PCK@0.3 and a reduction of 0.15 in NMSE. This improvement is especially pronounced in the affordances related to contain, pound, and w-grasp, which are more susceptible to geometric discrepancies compared to grasp, cut, and scoop. A case in point is the affordance for contain, which can be easily mistaken for the scoop due to their visually similar circular shapes, albeit with different sizes. This underscores the importance of integrating geometric feature fusion in our approach to detecting affordance keypoints. The comparative visualizations of FAKP and AffKP, as shown in Figure 4, highlight capability of FAKP to navigate complex scenarios effectively. For example, in scenario of first row, the spoon's handle, obscured by a mug, leads to erroneous detection of a grasp keypoint within the contain area by AffKp, mistaking part of the contain affordance for grasp. In another scenario of row two, featuring a tomato in a bowl, AffKP fails to detect the tomato as graspable. This limitation stems from its reliance solely on appearance features, which are insufficient for distinguishing the tomato from a pattern on the bowl. The Table II and Figure 4 demonstrate the superiority of FAKP compared with AffKp from accuracy and visualization.

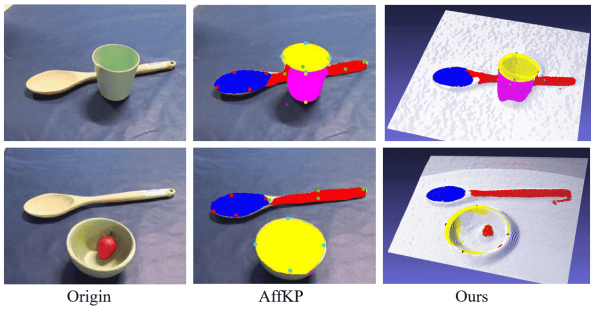

Fig. 4. Outputs of FAKP and AffKp on previously unseen objects in real-world (objects from IKEA).

VI. EXPERIMENTS

In this work, we follow the experiment pipeline in [11] to showcase the reliability of our method in accomplishing manipulation tasks with previously unseen objects. We manually interpret the network outputs into real-world robot's actions. The outputs as a representation of *what*,*where* and *how* details of affordances contribute to manipulation planning and execution.

Set-up in the Real-world: For our real-world experiments, as shown in Figure 1, we use a Franka Emika 7-DoF robotic arm fitted with a 2-fingered gripper and the robot's maximum payload of 3kg. A Kinect Azure camera captures the RGB-D image from the workspace and camera parameters are calibrated before experiments. To test the generalizability of our network, all objects in real-world experiments are unseen before and sourced from IKEA.

Keypoint Interpretation: We mainly employ keypoints quadruplet and the center point of it to determine two axes (x, y) and the origin of the frame. We use black, green and blue axes to represent the x, y and z axes, as shown in Figure 5. In order to avoid confusion, the x-axis always represents the principal axis of the object part. However, the axis required for different tasks won't always be the x-axis.

- **Grasp:** The origin of the frame is determined by the center point, which is the average coordinate of the 4 keypoints. The task-dependent axis which is the y-axis is computed by keypoints 3 and 4. Then the x-axis is computed by y-axis and keypoints 1 and 2. Lastly, the z-axis is computed by cross-product.
- **Contain and scoop:** The origin of the frame is computed by the average coordinate of the 4 keypoints. The y-axis is computed by keypoints 3 and 4. Then the z-axis is computed by y-axis and keypoints 1 and 2. Lastly, the x-axis is computed by cross-product.
- **Wrap-grasp:** The origin of the frame is determined by the average coordinate of keypoint 3 and 4. The task-dependent axis which is the y-axis is computed by keypoints 1 and 2. Then the x-axis is computed by y-axis and keypoints 3 and 4. Lastly, the z-axis is computed by cross-product.
- **Cut:** For cut affordance, the y-axis, which indicates the operational direction, is computed by the keypoints 1 and 2. Similarly, the x-axis is computed by both y-axis and

TABLE III

ROBOT EXPERIMENT STATISTICS

Task	#Trails	#Failure	#Planning Failure	#Grasp Failure	#Execution Failure
1	30	1	1	0	0
2	30	4	2	2	0
3	30	10	2	3	5
4	30	6	4	0	2

keypoints 3 and 4. Lastly, the z-axis is computed by cross-product.

Manipulation Task Specifications: Four affordance-informed manipulation tasks are introduced to test our pipeline. The video demo is in supplementary materials.

1) **Putting a tomato into a bowl:** The first manipulation task aims to evaluate the contain affordance. The objective is to place a tomato into a bowl. After grasping the tomato, the desired position is determined by the origin of the contain affordance. The gripper then releases the tomato above this target position. The task is considered successful if the tomato is contained within the bowl.
2) **Cutting sausage with a Knife:** The second manipulation task focuses on assessing the grasping and cutting affordance. The goal is to first grasp the knife and then use its blade to make contact with the sausage. The grasp position and orientation are determined by the origin and y-axis of grasp affordance. Keypoint 2 of the cut affordance serves as the contact point between the knife and the sausage, while the direction of the y-axis indicates the cutting direction. If the sharp side of the blade makes contact with the sausage, the task is considered successful.
3) **Scooping from a bowl:** The third manipulation task tests the grasp, scoop, and container affordance. Initially, the gripper grasps the tool. Subsequently, the y-axis of the scoop affordance is aligned with the y-axis of the contain affordance. Success is achieved if the tool can enter and exit the bowl without colliding with its edges.
4) **Wrap-grasping a cup:** The final manipulation task examines the wrap-grasp affordance. The objective is to wrap-grasp, lift, and hold a cup. The desired position for this task is determined by the origin of the wrap-grasp affordance, while the grasp width is determined by the distance between keypoint 1 and 2 and grasp orientation is determined by the y-axis of wrap-grasp affordance. The task is considered successful if the manipulator can hold the cup in the air for a duration of 3 seconds.

Manipulation Tasks Results: The visualization results from captured RGB-D image of the workspace are shown in Figure 5, while the statistics for the four distinct manipulation tasks are summarized in Table III. Most fail cases are from failed motion planning of motion planners which can be trapped in local minima due to its reliance on non-convex optimization, which leads the path planer consistently failed to produce a valid path. This issue could potentially be mitigated by employing sampling-based motion planners. Grasp failure

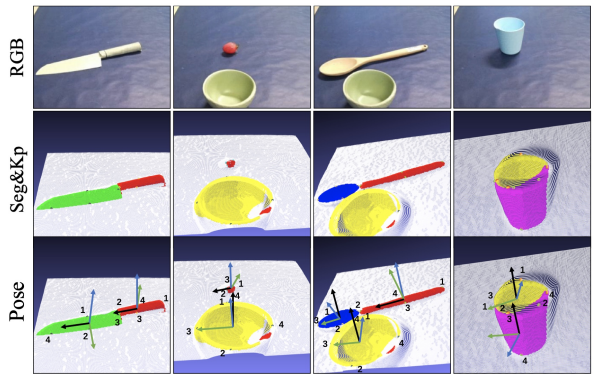

Fig. 5. Visualization of Affordance Segmentation and 3D keypoints quadruplet on previously unseen objects in real-world (objects from IKEA).

in Table III means the robot was unable to successfully grasp the object, or excessive relative motion between the gripper and the grasped object occurred. This issue could potentially be addressed by re-perceiving on-hand objects. The execution failure pertains to scenarios where the robot collides with an object despite successful planning, or the success requirements for the respective tasks are not met.

VII. LIMITATIONS

We believe that the limitations of our approach are mainly from the dataset and the hardware:

Dataset: Since our UMDKP dataset is enhanced from UMD dataset. UMDKP dataset has limited diversity in terms of object categories, scenes, and environments. This can restrict the generalizability of models trained on the dataset to real-world scenarios. For example, UMD dataset only includes limited affordance categories; the background is monotonous; there are no cluttered scenes.

Hardware: This limitation is mainly from Kinect camera and Franka Emika manipulator. Kinect Azure camera used in our real-world experiments may have difficulties capturing reflective objects and black objects, because it uses infrared depth sensors to capture depth information, and infrared sensors may struggle with reflective surfaces and absorbent black objects. The gripper and manipulator specifications also restrict the generalizability in real-world experiments. For example, the manipulator has difficulty manipulating the objects which exceed the 3Kg maximum payload of the manipulator; the two-finger gripper also restricts the manipulation tasks, for example, sliding when grasping the hammer owing to its uneven weight distribution nature as our video demo shows.

VIII. CONCLUSION

To address downstream planning and action challenges associated with affordances, we introduced the Fusion-based Affordance Keypoint Network(FAKP-Net), which can output affordance-associated 3D keypoints to support affordance-informed manipulation tasks. Real-world manipulation experiments showcase the reliability of our method in accomplishing manipulation tasks with never-before-seen objects. Future work will explore FAKP-Net in a more complex environment. All codes and data will be public.

References

[1] J. J. Gibson, "The theory of affordances," in *Perceiving, Acting, and Knowing: Toward an Ecological Psychology*, R. Shaw and J. Bransford, Eds. Erlbaum, 1977, pp. 67–82.

[2] S. Hart, P. Dinh, and K. Hambuchen, "The Affordance Template ROS package for robot task programming," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. Seattle, WA, USA: IEEE, May 2015, pp. 6227–6234. [Online]. Available: http://ieeexplore.ieee.org/document/7140073/

[3] D. Hadjivelichkov, S. Zwane, L. Agapito, M. P. Deisenroth, and D. Kanoulas, "One-Shot Transfer of Affordance Regions? AffCorrs!" in *Proceedings of The 6th Conference on Robot Learning*. PMLR, Mar. 2023, pp. 550–560, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v205/hadjivelichkov23a.html

[4] F.-J. Chu, R. Xu, L. Seguin, and P. A. Vela, "Toward Affordance Detection and Ranking on Novel Objects for Real-World Robotic Manipulation," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4070–4077, Oct. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8770077/

[5] F.-J. Chu, R. Xu, and P. A. Vela, "Learning Affordance Segmentation for Real-World Robotic Manipulation via Synthetic Images," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1140–1147, Apr. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8620534/

[6] T.-T. Do, A. Nguyen, and I. Reid, "AffordanceNet: An End-to-End Deep Learning Approach for Object Affordance Detection," Mar. 2018, arXiv:1709.07326 [cs]. [Online]. Available: http://arxiv.org/abs/1709.07326

[7] A. Nguyen, D. Kanoulas, D. G. Caldwell, and N. G. Tsagarakis, "Object-based affordances detection with Convolutional Neural Networks and dense Conditional Random Fields," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Vancouver, BC: IEEE, Sep. 2017, pp. 5908–5915. [Online]. Available: http://ieeexplore.ieee.org/document/8206484/

[8] ——, "Detecting object affordances with Convolutional Neural Networks," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Daejeon, South Korea: IEEE, Oct. 2016, pp. 2765–2770. [Online]. Available: http://ieeexplore.ieee.org/document/7759429/

[9] J. Sawatzky, A. Srikantha, and J. Gall, "Weakly Supervised Affordance Detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, Jul. 2017, pp. 5197–5206. [Online]. Available: http://ieeexplore.ieee.org/document/8100035/

[10] A. Myers, C. Teo, C. Fermüller, and Y. Aloimonos, "Affordance detection of tool parts from geometric features," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015, pp. 1374–1381, Jun. 2015.

[11] R. Xu, F.-J. Chu, C. Tang, W. Liu, and P. A. Vela, "An affordance keypoint detection network for robot manipulation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2870–2877, 2021.

[12] J. Gall and J. Sawatzky, "Adaptive Binarization for Weakly Supervised Affordance Segmentation," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*. Venice: IEEE, Oct. 2017, pp. 1383–1391. [Online]. Available: http://ieeexplore.ieee.org/document/8265374/

[13] L. Manuelli, W. Gao, P. Florence, and R. Tedrake, "kPAM: KeyPoint Affordances for Category-Level Robotic Manipulation," Oct. 2019, arXiv:1903.06684 [cs]. [Online]. Available: http://arxiv.org/abs/1903.06684

[14] W. Gao and R. Tedrake, "kPAM-SC: Generalizable Manipulation Planning using KeyPoint Affordance and Shape Completion," Sep. 2019, arXiv:1909.06980 [cs]. [Online]. Available: http://arxiv.org/abs/1909.06980

[15] Z. Qin, K. Fang, Y. Zhu, L. Fei-Fei, and S. Savarese, "KETO: Learning Keypoint Representations for Tool Manipulation," Oct. 2019, arXiv:1910.11977 [cs]. [Online]. Available: http://arxiv.org/abs/1910.11977

[16] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, May 2002. [Online]. Available: http://ieeexplore.ieee.org/document/1000236/

[17] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid Scene Parsing Network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, Jul. 2017, pp. 6230–6239. [Online]. Available: http://ieeexplore.ieee.org/document/8100143/

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 770–778. [Online]. Available: http://ieeexplore.ieee.org/document/7780459/

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp. 248–255, iSSN: 1063-6919.

[20] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/d8bf84be3800d12f74d8b05e9b89836f-Abstract.html

[21] C. Wang, D. Xu, Y. Zhu, R. Martin-Martin, C. Lu, L. Fei-Fei, and S. Savarese, "DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA: IEEE, Jun. 2019, pp. 3338–3347. [Online]. Available: https://ieeexplore.ieee.org/document/8953386/

[22] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection."

[23] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to Evaluate Foreground Maps," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. Columbus, OH, USA: IEEE, Jun. 2014, pp. 248–255. [Online]. Available: https://ieeexplore.ieee.org/document/6909433

[24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," May 2017, arXiv:1606.00915 [cs]. [Online]. Available: http://arxiv.org/abs/1606.00915