Grounding Bodily Awareness in Visual Representations for Efficient Policy Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Learning effective visual representations for robotic manipulation remains a fundamental challenge due to the complex body dynamics involved in action execution. In this paper, we study how visual representations that carry body-relevant cues can enable efficient policy learning for downstream robotic manipulation tasks. We present **I**nter-token **Con**trast (**ICon**), a contrastive learning method applied to the token-level representations of Vision Transformers (ViTs). ICon enforces a separation in the feature space between agent-specific and environment-specific tokens, resulting in agent-centric visual representations that embed body-specific inductive biases. This framework can be seamlessly integrated into end-to-end policy learning by incorporating the contrastive loss as an auxiliary objective. Our experiments show that ICon not only improves policy performance across various manipulation tasks but also facilitates policy transfer across different robots. The project website: `https://anonymous.4open.science/w/ICon/`

1 Introduction

Vision serves not only the awareness of the external environment but also the awareness of one's own self [11]. Through vision, we perceive our bodies, monitor our movements, and maintain a perceptual boundary between self and non-self. This form of bodily awareness, commonly referred to as *visual proprioception* [2], enables agents to respond to their own bodily dynamics in a flexible and adaptive manner. Such responsiveness is essential for planning and executing actions in tasks that require high-level action sensitivity, such as locomotion and manipulation [10]. Going further, incorporating such inductive biases, particularly those arising from the agent's body within the visual field, can be highly beneficial to learning policies for robotic tasks [12, 17, 39]. With awareness of the position and movement of its own body, a robotic agent can efficiently learn structured agent-environment representations from raw pixel observations [12].

However, despite existing efforts in visuomotor policy learning, extracting body-aware information from high-dimensional images remains challenging, especially in end-to-end learning frameworks where visual encoders are jointly optimized with policy networks [24]. Since both components share the same optimization objective, models can easily converge to bottlenecks that inadvertently filter out task-irrelevant cues, including visual signals related to the agent's body. This issue becomes even more pronounced when training data is deficient. To address this, one approach is to augment the policy loss with an agent-centric auxiliary objective [12, 29]. These methods typically involve reconstructing RGB observations or agent masks from latent representations to implicitly disentangle a robotic agent from its environment. While this strategy has proven effective across various tasks, we argue that the reconstruction loss can undermine the training stability of policy learning. This raises a key question: is there a more natural way to derive disentangled agent-environment representations from pixels without sacrificing model performance and training stability?

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

To this end, we propose **I**nter-token **Con**trast (**ICon**), a contrastive learning approach designed to extract agent-centric representations from the Vision Transformer (ViT) [8], a high-capacity visual encoder widely utilized in robotic manipulation [9, 19, 32, 42]. ICon applies contrastive learning to the ViT's token-level features, where features corresponding to the agent are pulled together, and are contrasted against those corresponding to the environment, and vice versa. By explicitly decoupling agent-specific and agent-agnostic features, we implicitly encourage the model to focus on agent-relevant information, rather than information of the entire scene. We further introduce the following technical contributions to enhance the performance of ICon:

- We bring Farthest Point Sampling (FPS) [30] into 2D domains to sample keys from tokens for contrastive learning. By encouraging a wide spatial distribution of keys, FPS ensures that the selected features capture diverse and informative aspects of either the agent or the environment, maintaining a good representation of the overall structure.
- We propose a multi-level design that fuses inter-token contrastive losses from multiple layers of the ViT encoder, enabling a more complete disentanglement between the agent and its environment within the learned visual representations.

Through extensive experiments, we demonstrate that integrating ICon with Diffusion Policy [5], a state-of-the-art imitation learning algorithm, leads to consistent performance improvements across 7 out of 8 manipulation tasks spanning 3 different robots from 2 benchmarks. Code, data, and videos can be found: `https://anonymous.4open.science/w/ICon/`

2 Related work and background

2.1 Visuomotor policy learning

Training control policies that map visual sensory inputs directly to motor actions has been widely studied in reinforcement learning (RL) [21, 24, 44] and imitation learning (IL) [5, 23, 26, 37]. Among all, several works have explored learning improved representations for visual control through auxiliary tasks. Dasari and Gupta [7] leverage learned representations to predict the gripper's future location as a 2D keypoint in the image for debugging purposes, although they do not explicitly use this auxiliary objective for representation learning. Extending this line of work, Yarats et al. [45] couple a policy network with an autoencoder to reconstruct raw image pixels from the learned latent space, which has proven effective to improve the sample efficiency of RL algorithms. Building upon this idea, Gmelin et al. [12] incorporate an additional autoencoder to reconstruct binary agent masks, yielding an agent–centric representation that facilitates policy transfer across different robots. More recently, Li et al. [25] introduce the reconstruction approach to the reverse diffusion process [16], where a decoder reconstructs both pixel and state information from the intermediate representations of a U-Net model [34] to enhance the performance of a diffusion-based policy [5]. Our approach is similar to Laskin et al. [22] and Zhu et al. [47], which augment the policy objective with an auxiliary contrastive loss. However, instead of focusing on extracting task-relevant semantics from high-dimensional images, we aim to explicitly encourage the policy to develop a bodily awareness within the learned visual representations.

2.2 Contrastive learning

Contrastive learning is a self-supervised learning paradigm to learn useful representations from high-dimensional data, such as natural language [31], images [3, 4, 15, 31], and videos [27, 36, 43]. It can be interpreted as training an encoder for a dictionary look-up task, whose goal is to pull the query closer to a positive key while pushing it away from all other negative keys. This is typically achieved by minimizing a contrastive loss [6], which serves as an unsupervised objective function for training the encoder networks. Commonly used contrastive losses include Triplet loss [35], N-pair loss [38], Noise Contrastive Estimation (NCE) loss [13], and InfoNCE loss [28]. In this paper, we adopt a variant of the InfoNCE loss proposed by Wang et al. [41]:

$$\mathcal{L}_{\text{InfoNCE}}(q, \mathcal{K}^+, \mathcal{K}^-) = \frac{1}{|\mathcal{K}^+|} \sum_{k^+ \in \mathcal{K}^+} -\log \frac{\exp\left(q \cdot k^+/\tau\right)}{\exp\left(q \cdot k^+/\tau\right) + \sum\limits_{k^- \in \mathcal{K}^-} \exp\left(q \cdot k^-/\tau\right)} \tag{1}$$

where q, $\mathcal{K}^+$, and $\mathcal{K}^-$ denote the query, the set of positive keys, and the set of negative keys, respectively; $(\cdot)$ denotes the dot product; and τ is a temperature hyperparameter.

3 Visually grounded agent-centric representations

In principle, ICon is a general framework compatible with any visuomotor policy that uses vision transformers as visual encoders. In this section, we begin with an overview of the vanilla vision transformer, followed by a detailed explanation of the key design choices of ICon as well as its integration with a policy network. An overview of ICon is shown in Figure 1.

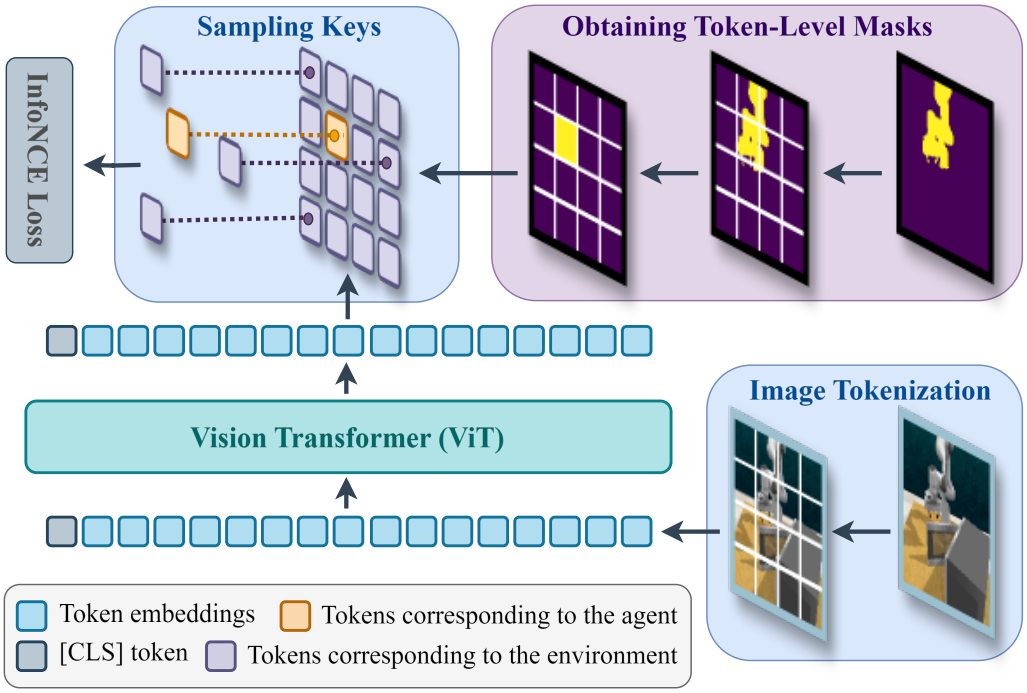


Figure 1: Overview of ICon. A full-scene RGB image containing a robotic agent is tokenized and processed by a vision transformer. The resulting token-level features (excluding the [CLS] token) are reshaped and aligned with a token-level mask derived from the agent's segmentation mask. Tokens corresponding to the agent and the environment are then sampled and used as keys to compute the inter-token contrastive loss.

3.1 Preliminaries: vision transformers

Vision Transformers (ViTs) [8] extract token-level representations from high-dimensional images. As depicted in Figure 1, an image $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$ is first divided into non-overlapping patches, each of size $P \times P$, and then embedded into a sequence of tokens $\mathcal{T} \in \mathbb{R}^{N \times D}$, where $N = HW/P^2$ denotes the number of patches and D is the embedding dimension. The token embeddings, prepended with a learnable classification token [CLS], are subsequently fed into the ViT encoder to produce a sequence of token-level features $[\mathcal{F}_{\text{cls}}, \mathcal{F}]$, where $\mathcal{F}_{\text{cls}} \in \mathbb{R}^{D}$ and $\mathcal{F} \in \mathbb{R}^{N \times D}$ correspond to the [CLS] token and the patch embeddings, respectively.

3.2 Token-level agent masks

While we have obtained token-level features from the vision transformer, how can we determine which features are agent-specific and which are agent-agnostic? Recall that each token corresponds to an image patch consisting of a set of pixels. Each pixel can be classified as belonging to either the agent or the environment based on an agent mask [12, 17, 29]. Therefore, we can propagate these pixel-level assignments to the token level.

Specifically, given the image $\mathcal{I}$ of the full scene, we use a segmentation model to generate a binary mask $\mathcal{M} \in \mathbb{R}^{H \times W}$, where $\mathcal{M}_{i,j} = 1$ for pixels occupied by the agent and 0 otherwise. This mask $\mathcal{M}$

is then patchified into $\mathcal{P}_{\text{mask}} = \{p_{k,l}\}_{k=1,l=1}^{H/P,W/P}$ following the same patchification procedure applied to the image $\mathcal{I}$ in ViT encoding. Since each patch $p_{k,l}$ may contain a mix of agent-related and environment-related pixels, we determine its dominant class based on the majority pixel type: if a patch contains more agent pixels than environment pixels, it is considered agent-dominated and assigned a value of 1; otherwise, it is considered environment-dominated and assigned a value of 0 (Equation 2). This yields a new patch-level (or token-level) mask $\mathcal{M}_{\text{token}} = \{m_{k,l}\}_{k=1,l=1}^{H/P,W/P}$, where $m_{k,l} \in \{0, 1\}$.

$$m_{k,\,l} = \begin{cases} 1 & \text{if sum}(p_{k,\,l}) > P^2/2 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

3.3 Inter-token contrastive loss

Now that we have acquired the token-level features and the agent masks, we introduce an inter-token contrastive loss to help the model distinguish between the agent and its environment. Our intuition is straightforward: features that belong to the same class (agent or environment) should be similar, while features coming from different classes should be dissimilar. To fulfill this, we encourage features of the same class to cluster together while enforcing separation between features of different classes, resulting in a clearer boundary between the agent and its environment in the learned feature space.

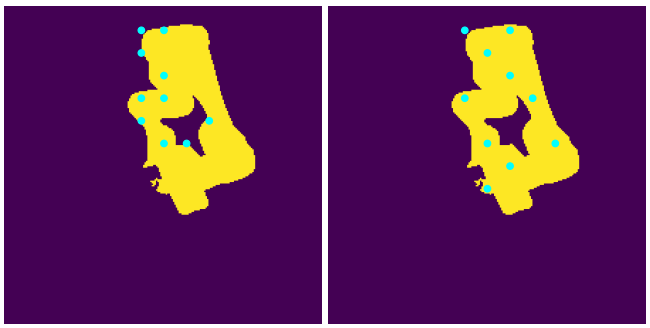
Algorithm 1 2D Farthest Point Sampling

Input: 2D indices $\mathcal{V} = \{(k, l)\}_{k=1,l=1}^{H,W}$, a binary mask $\mathcal{M} = \{m_{k,l} \in \{0, 1\}\}_{k=1,l=1}^{H,W}$ indicating sampling regions, number of samples N ($N \leq \sum m_{k,l}$)
Output: Indices of samples $\mathcal{V}'$
$\mathcal{D} \leftarrow \{d_{k,l} = \infty\}_{k=1,l=1}^{H,W}$ ▷ Distance map
Randomly select $(\tilde{k}, \tilde{l})$ where $m_{\tilde{k},\tilde{l}} = 1$
$\mathcal{V}' \leftarrow \{(\tilde{k}, \tilde{l})\}$
for $s = 1$ **to** $N - 1$ **do**
  $(\hat{k}, \hat{l}) \leftarrow \mathcal{V}'[-1]$
  for $k = 1$ **to** H, $l = 1$ **to** W **do**
    $\hat{d}_{k,l} \leftarrow |\hat{k} - k| + |\hat{l} - l|$
    if $\hat{d}_{k,l} < d_{k,l}$ **then**
      Update $d_{k,l} \leftarrow \hat{d}_{k,l}$
    end if
  end for
  $(k^*, l^*) \leftarrow \arg\max_{k,l}(m_{k,l} \cdot d_{k,l})$
  $\mathcal{V}' \leftarrow \mathcal{V}' \cup \{(k^*, l^*)\}$
end for
return $\mathcal{V}'$

Specifically, given the token-level features $\mathcal{F}$ and the corresponding agent masks $\mathcal{M}_{\text{token}}$, we first rearrange the sequence-like features $\mathcal{F}$ into a 2D feature map $\mathcal{F}_{\text{map}} = \{f_{k,l}\}_{k=1,l=1}^{H/P,W/P}$ for subsequent processing. We then compute the agent-specific query q_a and environment-specific query q_e by averaging the corresponding features, as defined in Equation 3, where $\mathbb{I}(\cdot)$ stands for the indicator function. As for key selection, we adapt the Farthest Point Sampling (FPS) method [30] from point cloud sampling to the 2D domain (see Algorithm 1). Compared with random sampling, FPS promotes diversity through selecting points that are spatially well-distributed (see Figure 2), ensuring that the sampled keys capture diverse and representative features of the agent and the environment. By applying FPS within the feature map $\mathcal{F}_{\text{map}}$ while restricting sampling regions using $\mathcal{M}_{\text{token}}$ and $(1 - \mathcal{M}_{\text{token}})$, we obtain a set of agent-specific keys $\mathcal{K}_a$ and a set of environment-specific keys $\mathcal{K}_e$, respectively. Note that for the agent-specific query q_a, the agent-specific keys $\mathcal{K}_a$ serve as positive keys, while the environment-specific keys $\mathcal{K}_e$ serve as negative keys, and vice versa for the environment-specific query q_e. Finally, we compute two symmetric InfoNCE losses (Equation 1) for the queries using their respective positive and negative keys, and combine them together to form the ICon objective (see Equation 4).

$$\begin{aligned} q_a &= \frac{1}{\text{sum}(\mathcal{M}_{\text{token}})} \sum_{k=1}^{H/P} \sum_{l=1}^{W/P} \mathbb{I}(m_{k,l} = 1) f_{k,l} \\ q_e &= \frac{1}{\text{sum}(1 - \mathcal{M}_{\text{token}})} \sum_{k=1}^{H/P} \sum_{l=1}^{W/P} \mathbb{I}(m_{k,l} = 0) f_{k,l} \end{aligned} \tag{3}$$

$$\mathcal{L}_{\text{ICon}} = \mathcal{L}_{\text{InfoNCE}}(q_a, \mathcal{K}_a, \mathcal{K}_e) + \mathcal{L}_{\text{InfoNCE}}(q_e, \mathcal{K}_e, \mathcal{K}_a) \tag{4}$$



(a) Random sampling (b) Farthest Point Sampling

Figure 2: Visualization of point distributions sampled from the agent mask. (a) Random sampling may result in points clustered within a small region. (b) Farthest Point Sampling (FPS) produces points that are well-distributed across the entire agent.

3.4 Multi-level contrast

In the standard ICon formulation, inter-token contrastive learning is applied only at the final layer of the vision transformer. However, we argue that this is insufficient to fully decouple the agent and its environment within the visual representations. To achieve a more complete agent-environment disentanglement, we extend ICon to each transformer encoder layer [40] of the vision transformer. Specifically, let $\mathcal{F}^{(i)}$ represent the token-level output features (excluding the [CLS] token) from the i-th layer. The inter-token contrastive loss for this layer, $\mathcal{L}^{(i)}_{\text{ICon}}$, is computed as described in Section 3.3. The overall contrastive objective is then obtained by taking a weighted sum of the layer-wise contrastive losses:

$$\mathcal{L}_{\text{ICon}} = \sum_i \frac{\exp(\gamma \cdot i)}{\sum_i \exp(\gamma \cdot i)} \mathcal{L}^{(i)}_{\text{ICon}} \tag{5}$$

Here, γ is a hyperparameter that controls the disentangling degree across transformer encoder layers. Prior work has shown that the shallow layers of a vision transformer primarily capture positional information, while deeper layers shift toward encoding more semantic features [1]. This implies that shallower layers tend to produce more entangled agent-environment representations, resulting in larger inter-token contrastive losses. To strike a balance, we set $\gamma > 0$ to assign greater weights to the contrastive losses from deeper layers.

3.5 Training

As described above, ICon enhances a policy's visual representations by introducing an agent-centric contrastive loss as an auxiliary objective during policy optimization. We utilize the widely adopted Diffusion Policy [5] to demonstrate how ICon can be incorporated into its training pipeline. Let $\mathcal{D} = \{(o_t \in \mathcal{O}, a_t \in \mathcal{A})\}_{t=1}^{T}$ denote a dataset consisting of observation-action pairs, where the observation space $\mathcal{O}$ comprises both image observations $\mathcal{I}$ and low-dimensional state information $\mathcal{S}$. Diffusion Policy learns a mapping $\pi : \mathcal{O} \rightarrow \mathcal{A}$ by training a visual encoder $\mathcal{E}$ jointly with a diffusion model [16] using a prediction loss $\mathcal{L}_{\text{pred}}$. In our framework, the visual encoder is instantiated as a vision transformer, whose output features $\mathcal{F}_{\text{cls}}$ and $\mathcal{F}$ are used to condition on the denoising diffusion process and compute the contrastive objective $\mathcal{L}_{\text{ICon}}$, respectively. By combining the prediction loss and the contrastive loss together with a weighting coefficient λ, we derive the following training objective for policy update:

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \lambda \mathcal{L}_{\text{ICon}} \tag{6}$$

In practice, we precompute the agent masks $\mathcal{M}$ and store them alongside the observations o_t and actions a_t in the dataset $\mathcal{D}$. During training, for each mini-batch sampled from $\mathcal{D}$, we apply identical image augmentations to the image observations and their corresponding masks before computing the training objective.

4 Experiments

We conduct a systematic evaluation of ICon across **8** manipulation tasks spanning **3** robots from 2 simulation benchmarks. Through our experiments, we seek to answer the following questions:

1) To what extent does ICon improve the performance of the base policy?

2) What are the advantages of ICon over its counterparts?

3) Does ICon facilitate policy transfer across different robots?

4) What design choices of ICon have the most influence on its performance?

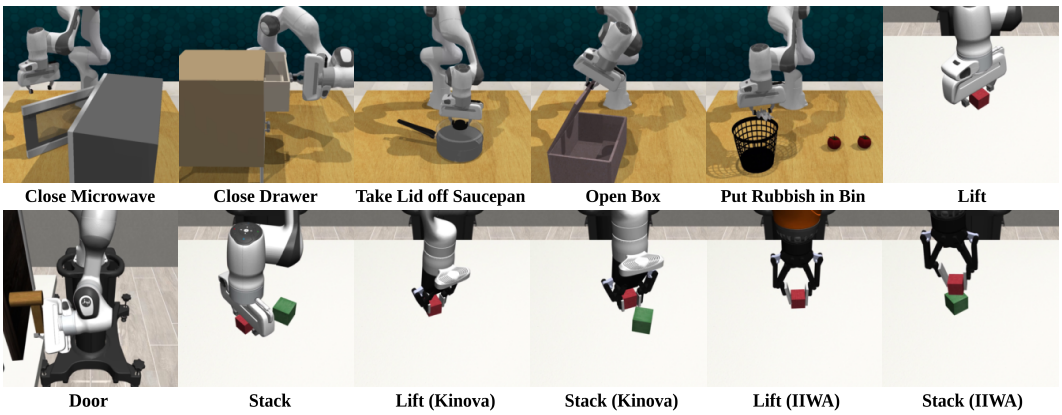


Figure 3: Visualization of simulated environments used for evaluation.

4.1 Simulation benchmarks

RLBench [18]: is a large-scale manipulation benchmark designed for meta learning, reinforcement learning, and imitation learning. It provides more than 100 robotic manipulation tasks ranging from simple target-reaching to complex long-horizon tasks. We select **5** tabletop tasks—*Close Microwave*, *Close Drawer*, *Take Lid off Saucepan*, *Open Box*, and *Put Rubbish in Bin*—which encompass object picking, articulated object manipulation, and long-horizon pick-and-place.

Robosuite [48]: is a widely used manipulation benchmark comprising 19 task environments that span both single-arm and dual-arm manipulation. From this benchmark, We select **3** representative tasks—*Lift*, *Door*, and *Stack*—which involve lifting a cube, opening a door, and stacking one cube on top of another, respectively.

4.2 Datasets

We release a new dataset covering the 8 manipulation tasks across 3 different robots in the RLBench and Robosuite environments. In RLBench, data are collected using the built-in motion planning toolkit, whereas in Robosuite, data are collected via teleoperation. Specifically, we collect **50** human demonstrations per task using a Franka Emika Panda robot, and an additional **5** demonstrations each from a Kinova Gen3 robot and a KUKA LBR IIWA robot for the *Lift* and *Stack* tasks. Each human demonstration comprises a sequence of paired observations and actions, where observations include RGB images from two viewpoints (a third-person and a wrist-mounted camera) and robot proprioception (e.g., joint position, gripper status), and actions correspond to the end-effector poses. For each RGB image, we use the Segment Anything Model (SAM) [20, 33] to extract a segmentation mask of the robot in the scene, and store the robot mask alongside the observation-action pairs in the dataset, forming a sequence of observation-mask-action triplets. In the following experiments, we train different policies using the Franka-specific data for performance comparison and fine-tune the pre-trained policies on Kinova-specific and IIWA-specific data to evaluate few-shot policy transfer across robots.

4.3 Evaluation setup

Baselines. We integrate and compare ICon with two variants of the Diffusion Policy [5]: (i) **Diff-C**, a CNN-based variant that performs well on most manipulation tasks with minimal need for hyperparameter tuning; and (ii) **Diff-T**, a transformer-based variant shown to be particularly effective for complex manipulation tasks involving frequent action changes. We refer to our methods as **ICon-Diff-C** and **ICon-Diff-T**, respectively. Additionally, we compare against Crossway Diffusion [25], which shares the same backbone as Diff-C but incorporates an auxiliary reconstruction loss to improve representation learning. For brevity, we refer to it as **Crossway-Diff-C**.

Policy rollout. Before each rollout, the simulated environment is randomly initialized using a predefined seed that is consistent across all learning algorithms. At each step, instead of relying solely on the current observation to predict the next action, the policy receives the past T_o observations from the environment and predicts the next T_a actions, of which only the first T_a' are executed in the scene. In practice, we find it crucial to apply *Temporal Ensemble* [46] to the predicted action sequences to ensure smoother control and mitigate action jitters.

Evaluation methodology. We report success rates for each learning algorithm and manipulation task. Results are averaged over 3 training seeds and 50 different environment initial conditions (150 episodes in total), with standard deviations computed across the 3 training seeds. A task is considered successful if and only if the reward returned by the simulated environment changes from 0 to 1. In addition, each task has a predefined maximum number of rollout steps; if the robotic agent fails to complete the task within this limit, the episode is deemed a failure.

Table 1: Performance comparison of different algorithms on the RLBench benchmark. We present success rates for 5 algorithms across 5 tasks in the format of (mean) $\pm$ (standard deviation), as described in Section 4.3.

	Diff-C	Diff-T	Crossway-Diff-C	ICon-Diff-C	ICon-Diff-T
Close Microwave	0.040 ± 0.016	0.993 ± 0.009	0.033 ± 0.019	0.153 ± 0.034	**1.000**
Close Drawer	0.713 ± 0.034	0.893 ± 0.025	0.667 ± 0.041	0.713 ± 0.050	**0.913 ± 0.047**
Take Lid off Saucepan	0.033 ± 0.019	0.280 ± 0.075	0.073 ± 0.025	0.113 ± 0.050	**0.413 ± 0.151**
Open Box	0.087 ± 0.074	0.113 ± 0.090	0.047 ± 0.066	**0.300 ± 0.043**	0.127 ± 0.019
Put Rubbish in Bin	0.000	0.033 ± 0.025	0.000	0.000	**0.093 ± 0.082**

Table 2: Performance comparison of different algorithms on the Robosuite benchmark. Success rates are reported for 3 algorithms across 3 tasks in the same format as in Table 1.

	Diff-C	Crossway-Diff-C	ICon-Diff-C
Lift	0.527 ± 0.104	**0.573 ± 0.100**	0.487 ± 0.057
Door	0.860 ± 0.028	0.827 ± 0.082	**0.887 ± 0.034**
Stack	0.160 ± 0.016	0.067 ± 0.025	**0.220 ± 0.016**

4.4 Performance improvements

As shown in Table 1, diffusion policies coupled with ICon consistently outperform or match the baselines across all 5 tasks in the RLBench simulated environments. Notably, ICon-Diff-C achieves absolute improvements of 21.3% and 11.3% over Diff-C in the *Open Box* and *Close Microwave* tasks, respectively. In another articulated object manipulation task *Close Drawer*, the positive effects of incorporating ICon are less pronounced, but ICon-augmented policies still perform on par with or better than the baselines. In contrast, Crossway-Diff-C underperforms Diff-C and ICon-Diff-C across all three articulated object manipulation tasks. In the *Take Lid off Saucepan* task, ICon-Diff-C and Crossway-Diff-C both exhibit higher success rates than Diff-C, with ICon-Diff-C showing more substantial improvements. Likewise, ICon-Diff-T surpasses Diff-T with an absolute improvement of 13.3%. In the long-horizon *Put Rubbish in Bin* task, all CNN-based diffusion policies fail to succeed, whereas ICon-Diff-T remains better than Diff-T.

As displayed in Table 2, ICon-Diff-C outperforms both Diff-C and Crossway-Diff-C across all tasks except *Lift*, where it performs slightly worse than the baseline methods. Note that *Lift* involves one of the simplest environments among all tasks, which may limit ICon-based policies to learn meaningful representations from full-scene images. In the *Open Door* task, Diff-C underperforms ICon-Diff-C but outperforms Crossway-Diff-C, aligning with earlier experimental results on articulated object manipulation tasks in the RLBench environments. Finally, in the *Stack* task, ICon-Diff-C surpasses both Diff-C and Crossway-Diff-C with improvements of 6.0% and 15.3%, respectively. Overall, integrating ICon into diffusion policies leads to improved performance across most manipulation tasks.

Table 3: Results of few-shot policy transfer across different robots on the Robosuite benchmark. Policies are transferred from a source robot to a target robot, with task success rates reported for each robot and learning algorithm. Success rates are displayed following the same format as in Table 1 and Table 2.

Task	Source Robot		Target Robot			
	Franka (Default Gripper)		Kinova (Robotiq85)		IIWA (Robotiq140)	
	Diff-C	ICon-Diff-C	Diff-C	ICon-Diff-C	Diff-C	ICon-Diff-C
Lift	**0.527 ± 0.104**	0.487 ± 0.057	0.233 ± 0.066	**0.260 ± 0.102**	**0.060 ± 0.016**	0.040 ± 0.028
Stack	0.160 ± 0.016	**0.220 ± 0.016**	0.007 ± 0.009	**0.053 ± 0.025**	0.007 ± 0.009	**0.047 ± 0.025**

4.5 Transferability across robots

Here, we evaluate the transferability of ICon-augmented policies across 3 robots from the Robosuite benchmark, where variations come from both robotic arms (Franka, Kinova, IIWA) and grippers (Franka Default Gripper, Robotiq85, Robotiq140). We initially pre-train policies on data collected from a source robot, and then fine-tune them using a smaller dataset collected from a target robot. Results in Table 3 show that ICon enhances the performance of the base policy across all three robots in the *Stack* task. In the *Lift* task, it is noteworthy that even if ICon-Diff-C slightly underperforms the baseline on the source robot, it can still yields improved performance when transferred to a different robot, such as the Kinova. We also find that polices are more effectively transferred to the Kinova robot than to the IIWA robot, which we believe is because of the appearance similarity between the Kinova and the source Franka robot.

4.6 Training stability

A key strength of ICon is to maintain good training stability during end-to-end policy learning. For a quantitative measure, we train each policy for an equal number of epochs with checkpoints saved every 50 epochs, and report the average of the top-10 success rates as well as the overall maximum success rate for the *Open Door* task. Results are visualized in Figure 4, with dark and light colors representing maximum and average success rates, respectively. The accompanying percentages stand for the relative drop from the maximum to the average performance. We see that when maximum performances are comparable, Crossway-Diff-C exhibits the largest gap between maximum and average success rates, indicating that the auxiliary reconstruction loss hinders the training stability of the base policy. In contrast, ICon-Diff-C shows superior training stability by maintaining a relatively higher average performance throughout the training process. This suggests that ICon enables the base policy to learn more robust and consistent behaviors from pixel observations.

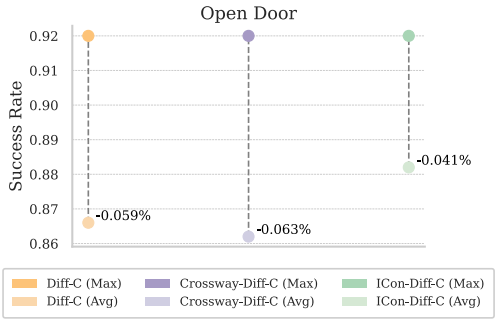


Figure 4: Comparison of training stability based on maximum and average performance during the training process.

4.7 Ablation study

We evaluate how each key component of ICon contributes to its performance through ablation experiments on the *Open Box* task. Specifically, we investigate the impact of applying ICon only at the final layer of the vision transformer, rather than fusing contrastive losses across all layers (W/o Multi-Level Contrast), as well as replacing the Farthest Point Sampling (FPS) with random sampling (W/o FPS). A summary of results is presented in Figure 5.

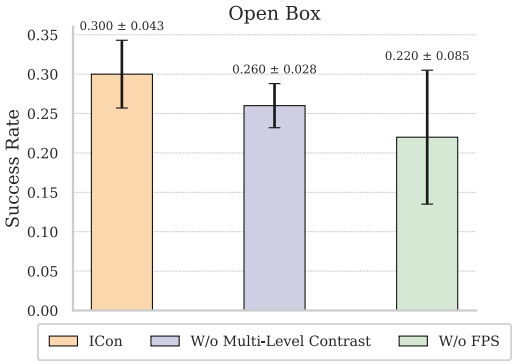


Figure 5: Summary of ablation experiments.

We observe a performance degradation when multi-level contrast is not employed, which we attribute to the insufficient disentangling of the intermediate representations from the vision transformer. A more significant performance drop occurs when random sampling is applied in place of FPS for key selection. We believe this degradation is due to the reduced expressivity of the sampled keys. Overall, FPS and multi-level contrast are crucial to the success of our method across a wide range of manipulation tasks.

5 Limitations

While our simulation experiments demonstrate that ICon improves the base policy across a variety of manipulation tasks, our work has several limitations. First, our method is compatible only with vision transformers and their variants, which restricts its applicability to other commonly used visual encoder architectures in visuomotor policy learning, such as ResNet [14]. Second, the farthest point sampling process incurs substantial computational overhead during forward propagation, making ICon inefficient for policy training on large-scale manipulation datasets. Eventually, our experiments are confined to simulation, and we have not yet evaluated our method in real-world settings due to limited hardware resources.

6 Discussion and future work

In this work, we investigate the benefits of grounding bodily awareness in visual representations and introduce ICon, a contrastive learning framework for extracting agent-centric representations from pixel observations. We demonstrate that policies augmented with ICon consistently achieve performance improvements across a diversity of manipulation tasks and can be effectively few-shot transferred across robots with different morphologies and configurations. In our future work, we plan to evaluate our method in complex real-world settings, where additional noise and distractors are present in the environments. Additionally, we hope to further enhance the learned agent-centric representations and develop more effective ones to enable zero-shot policy transfer.

References

[1] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2021.

[2] J. L. Bermúdez. Bodily awareness and self-consciousness. In Shaun Gallagher, editor, *The Oxford Handbook of the Self*, pages 157–179. Oxford University Press, 2011.

[3] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, Montreal, Canada, 2021.

[4] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, pages 1597–1607, Vienna, Austria, 2020.

[5] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, 2023.

[6] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 539–546, San Diego, CA, USA, 2005.

[7] S. Dasari and A. Gupta. Transformers for one-shot visual imitation. In *Proceedings of the 5th Conference on Robot Learning*, pages 2071–2084, London, UK, 2021.

[8] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[9] L. Fu, H. Huang, G. Datta, L. Y. Chen, W. C. Panitch, F. Liu, H. Li, and K. Goldberg. In-context imitation learning via next-token prediction. *arXiv preprint arXiv:2408.15980*, 2024.

[10] J. J. Gibson. The senses considered as perceptual systems. 1966.

[11] J. J. Gibson. A theory of direct visual perception. *Vision and Mind: selected readings in the philosophy of perception*, pages 77–90, 2002.

[12] K. Gmelin, S. Bahl, R. Mendonca, and D. Pathak. Efficient rl via disentangled environment and agent representations. *arXiv preprint arXiv:2309.02435*, 2023.

[13] M. Gutmann and A. Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, pages 297–304, Sardinia, Italy, 2010.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, Las Vegas, NV, USA, 2016.

[15] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, Seattle, WA, USA, 2020.

[16] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th Conference on Neural Information Processing Systems*, pages 6840–6851, Vancouver, Canada, 2020.

[17] E. S. Hu, K. Huang, O. Rybkin, and D. Jayaraman. Know thyself: Transferable visual control policies through robot-awareness. *arXiv preprint arXiv:2107.09047*, 2021.

[18] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison. Rlbench: The robot learning benchmark and learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020.

[19] S. Karamcheti, S. Nair, A. S. Chen, T. Kollar, C. Finn, D. Sadigh, and P. Liang. Language-driven representation learning for robotics. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, 2023.

[20] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W. Lo, P. Dollár, and R. Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, Paris, France, 2023.

[21] I. Kostrikov, D. Yarats, and R. Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2020.

[22] M. Laskin, A. Srinivas, and P. Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning*, pages 5639–5650, Vienna, Austria, 2020.

[23] S. Lee, Y. Wang, H. Etukuru, H. J. Kim, N. M. M. Shafiullah, and L. Pinto. Behavior generation with latent actions. In *Proceedings of the 41st International Conference on Machine Learning*, pages 26991–27008, Vienna, Austria, 2024.

[24] S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016.

[25] X. Li, V. Belagali, J. Shang, and M. S. Ryoo. Crossway diffusion: Improving diffusion-based visuomotor policy via self-supervised learning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 16841–16849, Yokohama, Japan, 2024.

[26] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. *arXiv preprint arXiv:2108.03298*, 2021.

[27] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta. R3m: A universal visual representation for robot manipulation. In *Proceedings of the 6th Conference on Robot Learning*, pages 892–909, Auckland, New Zealand, 2022.

[28] A. V. D. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[29] A. Pore, R. Muradore, and D. Dall'Alba. Dear: Disentangled environment and agent representations for reinforcement learning without reconstruction. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 650–655, Abu Dhabi, UAE, 2024.

[30] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 5099–5108, Long Beach, CA, USA, 2017.

[31] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International conference on machine learning*, pages 8748–8763, 2021.

[32] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell. Real-world robot learning with masked visual pre-training. In *Proceedings of the 7th Conference on Robot Learning*, pages 416–426, Atlanta, GA, USA, 2023.

[33] N. Ravi, V. Gabeur, Y. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[34] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, Munich, Germany, 2015.

[35] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, Boston, MA, USA, 2015.

[36] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and Google Brain. Time-contrastive networks: Self-supervised learning from video. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1134–1141, Brisbane, Australia, 2018.

[37] N. M. Shafiullah, Z. Cui, A. A. Altanzaya, and L. Pinto. Behavior transformers: Cloning k modes with one stone. In *Proceedings of the 36th Conference on Neural Information Processing Systems*, pages 22955–22968, New Orleans, LA, USA, 2022.

[38] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, page 1857–1865, Red Hook, NY, USA, 2016.

[39] G. Soter, A. Conn, H. Hauser, and J. Rossiter. Bodily aware soft robots: integration of proprioceptive and exteroceptive sensors. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2448–2453, Brisbane, Australia, 2018.

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010, Long Beach, CA, USA, 2017.

[41] X. Wang, K. Zhao, R. Zhang, S. Ding, Y. Wang, and W. Shen. Contrastmask: Contrastive learning to segment every thing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11604–11613, New Orleans, LA, USA, 2022.

[42] T. Xiao, I. Radosavovic, T. Darrell, and J. Malik. Masked visual pre-training for motor control. *arXiv preprint arXiv:2203.06173*, 2022.

[43] M. Xu, Z. Xu, C. Chi, M. Veloso, and S. Song. Xskill: Cross embodiment skill discovery. In *Proceedings of the 7th Conference on Robot Learning*, pages 3536–3555, Atlanta, GA, USA, 2023.

[44] D. Yarats, R. Fergus, A. Lazaric, and L. Pinto. Mastering visual continuous control: Improved data-augmented reinforcement learning. *arXiv preprint arXiv:2107.09645*, 2021.

[45] D. Yarats, A. Zhang, I. Kostrikov, B. Amos, J. Pineau, and R. Fergus. Improving sample efficiency in model-free reinforcement learning from images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10674–10681, 2021.

[46] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, 2023.

[47] J. Zhu, Y. Xia, L. Wu, J. Deng, W. Zhou, T. Qin, T. Liu, and H. Li. Masked contrastive representation learning for reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3421–3433, 2022.

[48] Y. Zhu, J. Wong, A. Mandlekar, R. Martín-Martín, A. Joshi, S. Nasiriany, and Y. Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020.

A Pseudocode

Algorithm 2 Inter-token Contrast (ICon)

Input: an RGB image $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$, an agent mask $\mathcal{M} \in \mathbb{R}^{H \times W}$, a vision transformer $\mathcal{E}(\cdot)$ with patch size P and embedding dimension D, number of agent-specific keys N_a, number of environment-specific keys N_e
Output: a contrastive loss $\mathcal{L}_{\text{ICon}}$
$[\mathcal{F}_{\text{cls}}, \mathcal{F}] \leftarrow \mathcal{E}(\mathcal{I})$
$\mathcal{F}_{\text{map}} = \{f_{k,l} \in \mathbb{R}^D\}_{k=1,l=1}^{H/P,W/P} \leftarrow \text{Reshape}(\mathcal{F})$
$\mathcal{P}_{\text{mask}} = \{p_{k,l}\}_{k=1,l=1}^{H/P,W/P} \leftarrow \text{Patchify}(\mathcal{M})$
$\mathcal{M}_{\text{token}} = \{m_{k,l} \in \{0,1\}\}_{k=1,l=1}^{H/P,W/P} \leftarrow \text{Threshold}(\mathcal{P}_{\text{mask}})$ ▷ Equation 2
$q_a,\ q_e \leftarrow \text{Average}(\mathcal{F}_{\text{map}},\ \mathcal{M}_{\text{token}}),\ \text{Average}(\mathcal{F}_{\text{map}},\ 1 - \mathcal{M}_{\text{token}})$ ▷ Equation 3
$\mathcal{K}_a,\ \mathcal{K}_e =\leftarrow \text{FPS}(\mathcal{F}_{\text{map}},\ \mathcal{M}_{\text{token}},\ N_a),\ \text{FPS}(\mathcal{F}_{\text{map}},\ 1 - \mathcal{M}_{\text{token}},\ N_e)$ ▷ Algorithm 1
$\mathcal{L}_a,\ \mathcal{L}_e \leftarrow \mathcal{L}_{\text{InfoNCE}}(q_a, \mathcal{K}_a, \mathcal{K}_e),\ \mathcal{L}_{\text{InfoNCE}}(q_e, \mathcal{K}_e, \mathcal{K}_a)$ ▷ Equation 1
$\mathcal{L}_{\text{ICon}} \leftarrow \mathcal{L}_a + \mathcal{L}_e$
return $\mathcal{L}_{\text{ICon}}$

B Implementation details

B.1 Data augmentation

Following Chi et al. [5], we apply random cropping to both RGB images and agent masks during training. The crop size is fixed at $3 \times 224 \times 224$ across all tasks. During inference, a static center crop of the same size is used.

B.2 Model architecture

The policy networks used in this work are built upon the Diffusion Policy [5]. We keep the overall model architecture unchanged except for the visual encoder, where we replace the ResNet [14] with a Vision Transformer (ViT) [8]. To save computing resources, we employ ViT-S with a patch size of 16 and an input image size of 224 as the visual encoder for our policy network.

B.3 Environment Setup

Details of the environment setup for RLBench and Robosuite are provided in Table 4. Note that in RLBench, robot proprioception includes arm joint positions, end-effector poses, and gripper status, whereas in Robosuite, robot proprioception consists of end-effector poses and gripper joint positions.

Table 4: Summary of task environments. **Objs**: number of objects in the scene; **Views**: number of viewpoints; **Img-Size**: image size; **P-D**: robot proprioception dimension; **A-D**: action dimension; **Controller**: robotic arm controller; **Steps**: maximum number of rollout steps.

	Objs	Views	Img-Size	P-D	A-D	Controller	Steps
Close Microwave	1	2	$3 \times 256 \times 256$	14	7	IK Pose	150
Close Drawer	1	2	$3 \times 256 \times 256$	14	7	IK Pose	200
Open Box	1	2	$3 \times 256 \times 256$	14	7	IK Pose	200
Take Lid off Saucepan	2	2	$3 \times 256 \times 256$	14	7	IK Pose	200
Put Rubbish in Bin	4	2	$3 \times 256 \times 256$	14	7	IK Pose	300
Lift	1	2	$3 \times 256 \times 256$	9	7	OSC Pose	200
Door	1	2	$3 \times 256 \times 256$	9	7	OSC Pose	300
Stack	2	2	$3 \times 256 \times 256$	9	7	OSC Pose	300

B.4 Training

We train our policy networks, ICon-Diff-C and ICon-Diff-T, using 3 training seeds (0, 42, and 100) and a batch size of 64. For each task, all policies are trained for 600 epochs on a single Nvidia GeForce RTX 3090 GPU, while in cross-robot transfer settings, the pre-trained policies are fine-tuned on the target robotic data for an additional 300 epochs. All other training configurations follow the settings described in the original codebase of Diffusion Policy [5]. The corresponding training time is summarized in Table 5.

Table 5: Training time measured in GPU hours for each task.

	ICon-Diff-C	ICon-Diff-T
Close Microwave	9.46	9.57
Close Drawer	11.50	13.02
Open Box	17.28	17.47
Take Lid off Saucepan	8.69	12.13
Put Rubbish in Bin	15.06	15.88
Lift	9.11	-
Door	16.48	-
Stack	11.76	-

C Visualization of learned representations

After training the vision transformer end-to-end with the policy network from scratch, we visualize the attention maps from the final layer of the vision transformer across several tasks. As shown in Figure 6, unlike the dispersed attention patterns exhibited by the baseline method, our contrastive learning approach encourages the vision transformer to focus on the agent's body rather than the entire scene. This confirms that the learned representations are agent-centric and carry body-relevant information about the robotic agent.

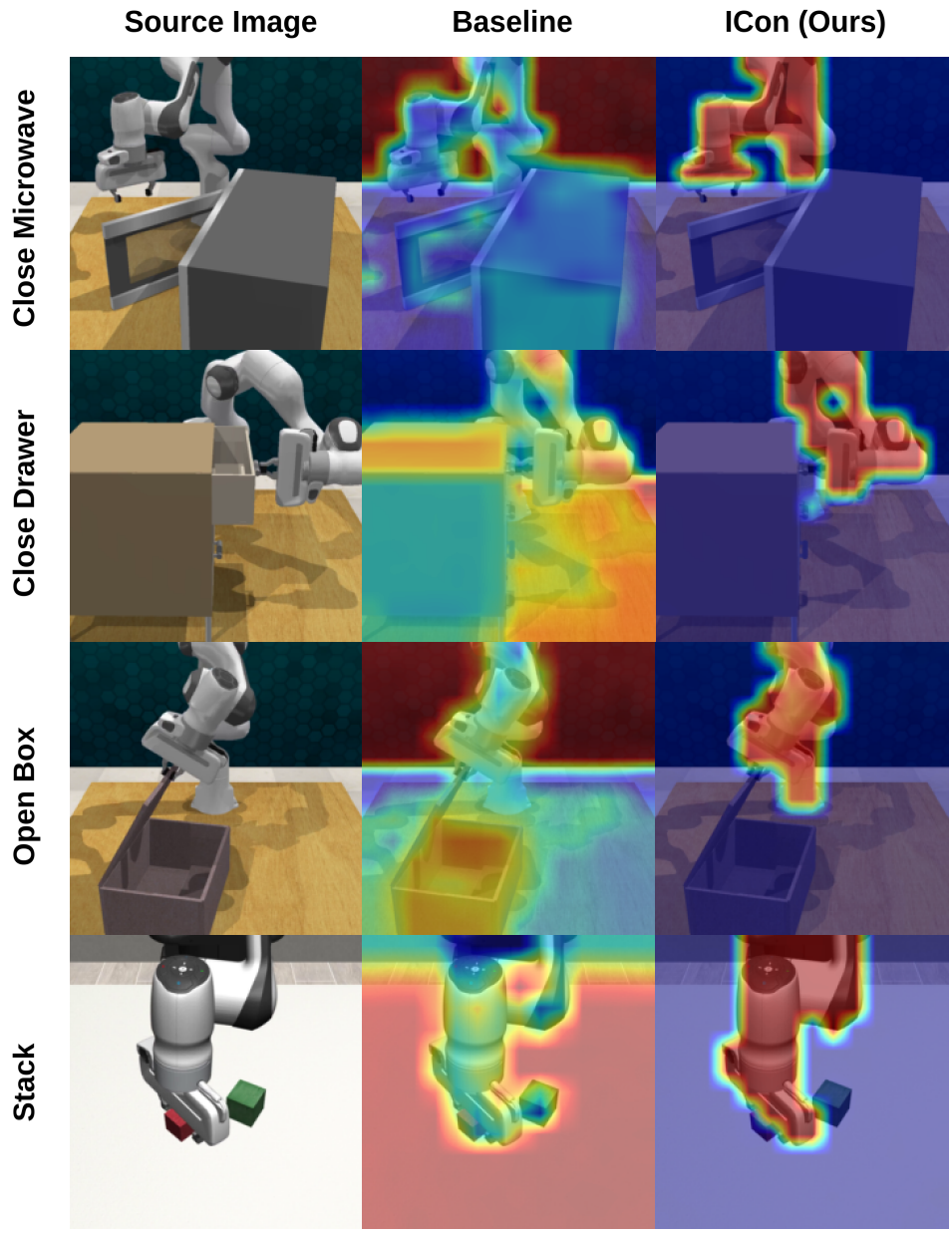


Figure 6: Visualization of representations learned by different algorithms across several tasks. For each task, we show the original image alongside the feature maps produced by different algorithms. Each feature map is computed by averaging the attention maps from all heads in the final layer of the vision transformer, with the [CLS] token as the query.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We describe our claims and contributions in Section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our work in Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We describe our experimental setup and implementation details in Section 4 and Section B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We open-source our code and data on the project website.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We specify all the training and test details in Section 4 and Section B.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We report experimental results in the format of (mean) $\pm$ (standard deviation), where the mean is computed over 3 training seeds and 50 different environment initial conditions (150 in total), and the standard deviation is computed across the 3 training seeds, as described in Section 4.3.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide information on the computer resources in Section B.4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: The work has no societal impact at the current stage.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The creators or original owners of assets, such as code, data, or models, used in the paper, are properly credited. Additionally, the license and terms of use are explicitly mentioned and properly respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide documentations for our code and data on the project website.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.