

# The Treachery of Images: Bayesian Scene Keypoints for Deep Policy Learning in Robotic Manipulation

Jan Ole von Hartz<sup>1</sup>, Eugenio Chisari<sup>1</sup>, Tim Welschehold<sup>1</sup>, Wolfram Burgard<sup>2</sup>,  
Joschka Boedecker<sup>1</sup> and Abhinav Valada<sup>1</sup>

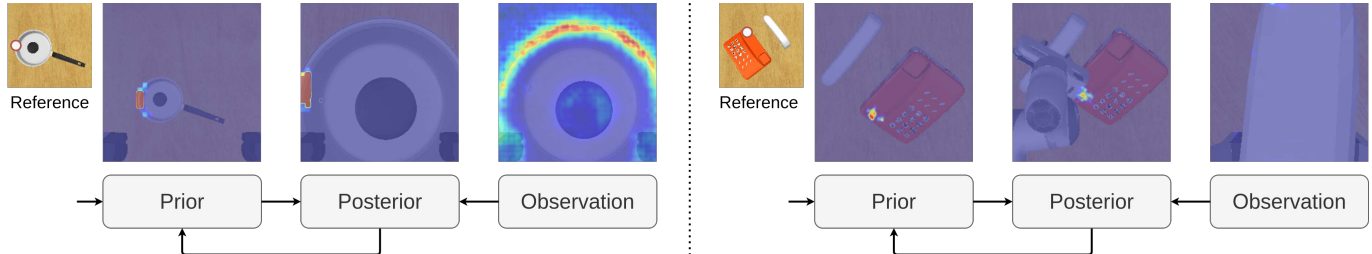


Fig. 1: Individual camera observations are often ambiguous. For example, from the observation on the left, the rotation of the saucepan cannot be uniquely inferred. When tracking object keypoints, this leads to statistically multimodal localization hypotheses. We overcome this problem by considering the image in context. We find likely correspondences across image scales and then use spatial or temporal context to resolve the ambiguities. Our model further detects when a keypoint is likely not observed, enabling our approach to track occluded objects and objects outside the current field of view as shown on the right.

**Abstract**—In policy learning for robotic manipulation, sample efficiency is of paramount importance. Thus, learning and extracting more compact representations from camera observations is a promising avenue. However, current methods often assume full observability of the scene and struggle with scale invariance. In many tasks and settings, this assumption does not hold as objects in the scene are often occluded or lie outside the field of view of the camera, rendering the camera observation ambiguous with regard to their location. To tackle this problem, we present BASK, a Bayesian approach to tracking scale-invariant keypoints over time. Our approach successfully resolves inherent ambiguities in images, enabling keypoint tracking on symmetrical objects and occluded and out-of-view objects. We employ our method to learn challenging multi-object robot manipulation tasks from wrist camera observations and demonstrate superior utility for policy learning compared to other representation learning techniques. Furthermore, we show outstanding robustness towards disturbances such as clutter, occlusions, and noisy depth measurements, as well as generalization to unseen objects both in simulation and real-world robotic experiments.

## I. INTRODUCTION

Over the last decade, policy learning methods that learn their own visual representations end-to-end have become exceedingly popular [1]–[3]. These methods are helpful in robotics, where we rarely have access to ground truth scene features but have to learn from raw camera observations instead. However, learning environment features from scratch can be prohibitively expensive, with current approaches often requiring large amounts of training data. Using pretrained visual models improves the efficacy of policy learning in robot manipulation [4]–[6]. Nevertheless, current approaches still fail on complex tasks, and no representation has been able to fully close the gap to models learning on ground truth scene features.

<sup>1</sup>Department of Computer Science, University of Freiburg, Germany.

<sup>2</sup>Department of Engineering, University of Technology Nuremberg.

This work was funded by the BrainLinks-BrainTools center of the University of Freiburg, the Carl Zeiss Foundation with the ReScaLe project, and by an academic grant from NVIDIA. (Corresponding author: Jan Ole von Hartz, hartzj@cs.uni-freiburg.de)

This is primarily due to three fundamental problems. First, efficient expert cognition is goal-directed [7]. Thus, a pretraining objective that requires semantic image understanding is required for better downstream policy success [8]. Second, state-of-the-art representation learning methods find it challenging to find corresponding visual features across different image scales [9]. Consequently, it is difficult to employ them with wrist-mounted cameras, whereas such cameras are widely available in real-world settings and enable many robotic tasks [10], [11]. Finally, images are treated in isolation and scene context is neglected. Although images are highly ambiguous, for instance, due to object symmetries and occlusions. These ambiguities can only be resolved in temporal or spatial context, as shown in Fig. 1. Similarly, any representation that we derive from an ambiguous image is prone also to be ambiguous. Hence, to generate an unambiguous image representation, the integration of context is essential. Moreover, due to their neglect of context, current methods cannot represent occluded objects or objects outside the field of view which further limits their applicability.

To address these problems, we present **Bayesian Scene Keypoints (BASK)**, a novel approach that focuses on representing the underlying scene, instead of representing each image in isolation. We address the first problem by extracting goal-directed information from each image via localizing 3D scene keypoints using a semantic encoder network and adding depth information. To address the second problem, we pay special attention to corresponding visual features across scales while training the network. Finally, we integrate our localization hypotheses across time and camera views to resolve multimodalities and represent temporarily unobserved objects. We propose an approach to integrating observations based on the Bayes filter, which makes minimal and transparent assumptions.

We extensively evaluate the efficacy of our approach for learning challenging multi-object manipulation policies. We compare against a suite of other representation methods using RL Bench [12], a standard benchmark of manipulation tasks involving everyday objects. We establish superior efficacy for

policy learning and improved localization accuracy, especially when learning from wrist camera observations. Further confirming the efficacy of our method in real-world experiments, we find our approach to transfer significantly better to the complexities of real-world perception than other methods. We further observe zero-shot transfer to cluttered scenes and previously unseen objects and environments. Enabling efficient policy learning from wrist camera observations, our method frees policy learning approaches from the confines of a lab where object tracking systems and camera arrays are available. Thus, it enables a plethora of applications such as mobile manipulation and the deployment of robots in environments where overhead cameras viewing the entire workspace are not available.

In summary, our main contributions are:

- 1) We propose a new framework for representation learning, interpreting it as *scene* representation instead of image representation. In this framework, we develop a Bayesian approach to resolving the inherent ambiguities of images.
- 2) We train encoder networks to generate semantic descriptions of multi-object scenes, respecting scale variances and occlusions, e.g. due to moving cameras, and leverage our Bayes filter to resolve the resulting ambiguities.
- 3) We rigorously evaluate the ability of our approach to overcome the problems outlined above, as well as its efficacy for policy learning, both in simulation and real-world robotic experiments.
- 4) We make the code and models publicly available at <http://bask.cs.uni-freiburg.de>.

The supplementary material can be found at <https://arxiv.org/abs/2305.04718>.

## II. RELATED WORK

Our goal is to generate compact representations from camera observations suited for efficient policy learning. To be applicable for wrist camera observations, these should be scale and occlusion invariant and be able to represent objects that are temporarily occluded or outside the field of view. Existing representation learning methods range from implicit methods, neurally compressing the image [13], [14] to methods explicitly expressing the poses of scene objects [15], [16]. Our keypoints-based approach lies in the middle of this spectrum, thus avoiding the major drawbacks of either family of methods.

*Implicit methods:* A common approach to compress camera observations is training a neural network with a bottleneck on image reconstruction, such as using Variational Auto-Encoders (VAEs). Instances of this approach are  $\beta$ -VAE [13], which can produce disentangled representations, and MONet [14], which partitions the image into several *slots* first, thus separating objects. Similarly, Transporter [17] is trained to reconstruct a source image from a target image via transporting local features. These representations have been shown to enable more efficient policy learning on a set of robotic manipulation tasks [4]. Making few assumptions, they are flexibly applicable. However, they are not as effective for downstream policy learning as ground truth scene features [4] or object keypoints [8].

*Explicit methods:* Pose estimation methods [16], [18], [19] represent the pose of the relevant scene objects explicitly.

However, they typically need a ground truth 3D model of the object [18], [19], they are not applicable to deformable objects [18], [19] and do not work well in the presence of occlusions. Recent methods that do not require any CAD models or object-specific training [20], [21], still require a pre-recorded scan of the object of interest for inference and are only applicable to rigid objects. More importantly, in Sec. S.3 in the supplementary material we show that they do not exhibit the needed scale-invariance for learning from a wrist camera.

*Keypoints:* Keypoints are pixel- or 3D coordinates tracking task-relevant object parts. KETO [22] predicts single keypoints from 3D object point clouds, whereas Neural Descriptor Fields [23] encode the full object point cloud. However, extracting full object point clouds is challenging and requires an array of cameras surrounding the scene, limiting the applicability of the method. Keypoints can be learned end-to-end from camera observations in RL [24] or using multi-view consistency [25]. In their seminal work, Florence *et al.* [8], [26] generate keypoints by training Dense Object Nets (DON) in a self-supervised manner. DONs can generalize between object class instances [26] and are applicable to deformable objects [8]. Training DONs to contrast between different single-object scenes, enables deployment in a multi-object setting [27]. However, this approach requires single-object scans of all objects and handles occlusions poorly. In contrast, we propose to directly train on multi-object scenes, making data collection much faster, avoiding computational overhead and including occlusions in the training data. Moreover, in densely cluttered scenes, the object mask generation needed for DON training can be skipped [9], [28] However, on less cluttered scenes this approach runs the risk of sampling too many background pixels in pretraining, thus compromising correspondence quality.

A major open problem in keypoint detection is visual correspondence across image scales. DONs fail to find accurate correspondences for vertical camera translations of less than 10 cm [9], rendering wrist cameras less effective for policy learning. Prior work circumvents this problem by corresponding from a fixed height [8], [9], [26]–[28]. Similarly, prior work does not comprehensively address the problem of occlusion [8], [26]. We demonstrate that DONs can be trained to be invariant to the image scale and occlusions, while distinguishing between multiple objects. Specifically, we find multi-model hypotheses to emerge which we resolve using Bayesian inference.

*Visual Tracking:* Ample methods have been developed to disambiguate local visual features within the context of a single image [29]–[31]. In contrast, we consider images that are inherently ambiguous due to occlusions or otherwise missing scene context and leverage context to resolve the ambiguities. Correlation filters can be used to track visual features across time [32], but they cannot utilize 3D information and the semantic generalization of DONs. Moreover, their tracker needs to be specifically trained, whereas our Bayes filter works off-the-shelf. Similarly, Bayesian approaches for tracking visual features across videos [33], [34] do not use 3D information and do not generalize across object instances. To the best of our knowledge, we are the first to leverage the potential of Bayes Filter in representation learning for robotic manipulation.

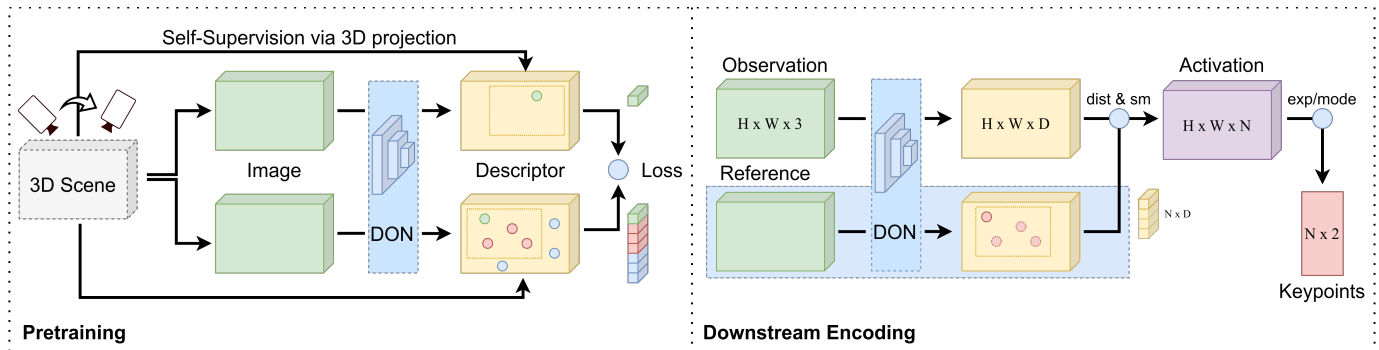


Fig. 2: Pretraining and keypoint generation processes of dense object nets. The encoder is pretrained using a self-supervised pixel-wise contrastive loss which optimizes the descriptor distance on scans of static scenes. During downstream policy learning, the descriptor of the current observation is compared to a previously selected set of reference descriptors and the pixel coordinates of the respective most likely match are used as the location of the keypoint.

### III. TECHNICAL APPROACH

We aim to generate 3D keypoints as an efficient representation for downstream policy learning. To be applicable to wrist camera observations, these keypoints need to be scale and occlusion invariant. Furthermore, they should be able to track multiple relevant scene objects and represent objects that are temporally occluded or outside the camera’s field of view. Our approach, **Bayesian Scene Keypoints (BASK)**, is two-pronged. First, we find semantic correspondences between images. To this end, we train Dense Object Nets (DON) directly on multi-object scenes. We compute the localization hypotheses for a keypoint by comparing the corresponding reference descriptor to the descriptor image generated by the DON. Ambiguous images lead to multimodal hypotheses. We then integrate these hypotheses using the Bayes filter to resolve ambiguities.

#### A. Learning Semantic Correspondence

To extract keypoints from camera observations, we train a DON in a self-supervised manner [26] and use the generated embeddings for downstream keypoint generation [8]. We then adapt these techniques to the multi-object case and describe how to achieve invariance towards scale, rotation, and occlusions.

1) *Dense-Correspondence Pretraining*: By moving an RGB-D camera in a static scene and tracking the camera pose, we reconstruct the 3D representation of that scene using volumetric reconstruction. After filtering out background points, we project the object point cloud back onto the image plane to generate object masks for all images along the trajectory. For a given pixel position in one image in the trajectory, we find the corresponding pixel position in another image of the same trajectory via simple 3D projections, using the respective camera pose and calibration matrix.

Using this technique for finding correspondences between pairs of images, we train an encoder network  $e_\eta: \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^{H \times W \times D}$ , mapping an RGB image to a  $D$ -dimensional descriptor, to minimize the descriptor distance between corresponding points while enforcing at least a margin  $M$  between non-corresponding points. Specifically, for a given pair of images  $I_a, I_b$ , we sample a set of  $m$  pixel locations  $U_a$  from the object mask of  $I_a$  and compute the set of corresponding pixel positions  $U_b$  in  $I_b$ . Additionally, for each point  $u_a \in U_a$  we sample a set of  $n$  non-corresponding points  $N_{u_a}$  from both  $I_b$ ’s object mask and the background. Let  $e_\eta(I_a)_{u_a}$  denote

the value of descriptor image  $e_\eta(I_a)$  at position  $u_a$ . We then compute the loss for the encoder  $e_\eta$  as

$$\mathcal{L}(I_a, I_b) = \sum_{u_a, u_b \in U_a, U_b} \left( \frac{\|e_\eta(I_a)_{u_a} - e_\eta(I_b)_{u_b}\|^2}{m} + \sum_{u_c \in N_{u_a}} \frac{\max(0, M - \|e_\eta(I_a)_{u_a} - e_\eta(I_b)_{u_c}\|^2)}{n} \right). \quad (1)$$

We found improved correspondence quality by enforcing a larger margin  $M_{bg}$  for background non-matches than for the foreground non-matches  $M_{fg}$ . Fig. 2 illustrates this approach.

2) *Multi-Object Tasks*: To extend DONs to multi-object tasks, we directly train on multi-object scenes such that the data is fast to collect and already contains occlusions. We again employ volumetric reconstruction and split the resulting point cloud using density-based clustering [35]. Projecting these object-wise point clouds back onto the camera planes yields consistent object masks for the trajectory. Furthermore, when working with a multi-camera setup, we can generate masks that respect occlusions, e.g. caused by the robot arm, further diversifying the training data. During one step of pretraining, we sample one of the object masks and treat the other objects as part of the background. This ensures that the model learns to distinguish the different objects.

3) *Invariances*: DONs struggle to generalize to camera perspectives outside the training distribution, especially for vertical camera movements [9]. This is not limited to cases where the change in perspective removes necessary scene context but to changes in distance between object and camera in general.

a) *Scale*: We hypothesize that this is due to CNNs being biased towards texture [36], [37], with textures transferring badly between image scales due to image rasterization occurring at a fixed resolution. Hence, correspondences across scales can only emerge at a higher semantic feature level. However, scale invariance is a critical property to be able to use DONs on wrist camera observations. By training the network on multiple image scales, we teach it to generate a more scale-invariant and semantically meaningful representation. Moreover, in contrast to previous work [8], we automatically scale the fraction of masked non-matches by the relative size of the object mask in the image to better account for scale differences.

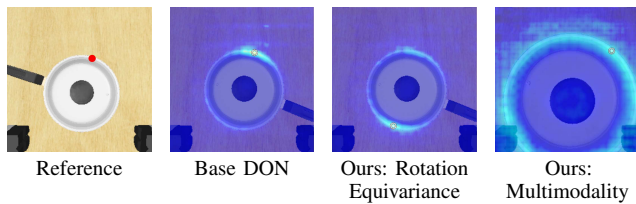


Fig. 3: Rotation equivariance and multimodal correspondence distributions. The first image shows the reference image and reference location. The remaining images are overlaid with a heatmap indicating their correspondence likelihood. In contrast to the base DON, our version effectively addresses both rotation equivariance and scale invariance, with multimodal hypotheses emerging when the spatial context is insufficient for unique correspondence.

*b) Rotation:* Furthermore, many objects are partially symmetrical. This makes rotation equivariance of the descriptor image an important property to generate consistent keypoints. Similar to scale-invariance, achieving rotation equivariance requires higher semantic features and only emerges late in the training process. Consequently, for partially symmetrical objects, the network needs to integrate information across the full image to resolve local symmetry, as shown in Fig. 3. Adding random rotations in training further helps the network generate more rotation equivariant descriptors.

*c) Occlusion:* We find that larger descriptor dimensions and a deeper network enable training the encoder on more perspectives without loss in quality. Thus, while previous work [8] uses the ResNet-34 model and descriptor size of 16, we use the ResNet-101 and descriptor size of 64. To improve training on large descriptors, we normalize the descriptor distances by the square root of the descriptor dimension. Moreover, during pretraining, we add aggressive crops of random size up to half of the dimension of the image. We found that these enable the network to generalize better and to improve its robustness towards occlusions, especially when the size of the crops is randomized as well. These adaptations enable us to train the DON to generate more semantically meaningful descriptors. As we detail in Sec. IV-C, our DON even shows zero-shot transfer to unseen object instances and task environments. Even more importantly, our DON consequently produces multimodal hypothesis distributions if the visual context is insufficient for unique localization, as shown in Fig. 3. This allows us to find likely correspondences across scales and then to resolve the multimodality of the hypothesis using context, as we detail in the subsequent section.

*4) Keypoint Inference:* For policy learning, we sample one frame from the set of training trajectories and feed it through the DON to select reference descriptors from. In a GUI, we select the reference positions that we wish to track by clicking on them. The descriptors at these reference positions serve as the *reference descriptors*. We encode a camera observation by feeding it through the frozen encoder and computing the Euclidean distance map between each of the reference descriptors and the image embedding. Subsequently, we employ a softmax function on the negative distance map to yield an activation map, interpreted as the probability of each pixel location corresponding to the reference position, i.e.  $a_{I,r} = \sigma(\|e_\eta(I) - r\|)$  for an image  $I$  and reference descriptor  $r$ . To reduce the effect of background noise, we add a *temperature*  $\alpha > 1$  to the softmax, as shown in Fig. 4.

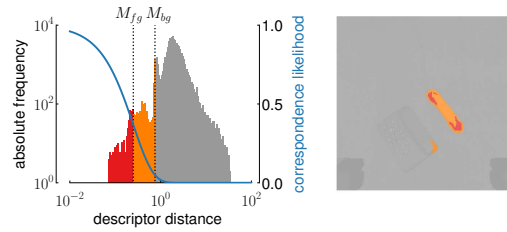


Fig. 4: Our correspondence likelihood model  $p(z) = e^{-4z}$  overlaid on a log-log histogram of an example distance map, next to a binned version of the map.  $M_{fg}$  and  $M_{bg}$  denote the margin parameters for foreground and background non-matches, respectively. As desired, the correspondence likelihood sharply declines for background pixels.

For an unambiguous image that generates a unimodal correspondence map, we can use the expectation of this distribution  $\mathbb{E}[a_{I,r}]$  as the keypoint location of  $r$  in  $I$ . By adding the associated depth measurement, we obtain a 3D keypoint. In contrast, for an ambiguous image, it is first necessary to collapse the multimodality of the hypothesis distribution, as we discuss in the next section.

### B. Bayesian Scene Keypoints

Our goal is to integrate the keypoint localization hypotheses from successive camera observations to resolve ambiguities and reduce noise. We propose an approach based on the Bayes filter which has four main advantages for our purpose. First, it works with any number of observations, whereas *learning* a sequence model would again require additional training data. Second, it is easy to interpret and transparent in its assumptions, which renders it easy to debug, extend, and adapt to new situations. For example, to integrate an additional modality such as LiDAR, all we need to formulate is a measurement model for it. Third, in contrast to sequence models such as an LSTM [38], the Bayes filter keeps the spatial structure of the hypothesis space at all times, which constitutes a powerful inductive bias. Finally, as no additional training is needed, representation learning and policy learning can be entirely decoupled. Shortening the length of backpropagation paths drastically improves computational efficiency. We verify these advantages in Sec. S.4 in the supplementary material.

We first summarize the basic form of the Bayes filter, before applying it to the scene representation by formulating the appropriate motion and measurement models. We develop two Bayes filters: a simple discrete filter for single-camera setups and a more powerful particle filter that can represent unobserved hypotheses and is suited for multi-camera setups.

*1) Background: Bayes Filter:* For a time step  $t$ , let  $x_t \in \mathbb{R}^d$  be the state of the environment at that time,  $u_t$  the action taken by the agent and  $z_t$  the measurement made. For brevity, let  $z_{a:b} := z_a, \dots, z_b$ . To calculate the posterior  $P(x_t | z_{1:t}, u_{1:t})$  of the state given all past measurements and actions, we use Bayes' rule. As per the usual convention,  $\eta$  denotes the normalization component of Bayes' rule, here  $\eta_t := p(z_t | z_{1:t-1}, u_{1:t})^{-1}$ . With the Markov assumption, we get:

$$p(x_t | z_{1:t}, u_{1:t}) = \eta_t \cdot p(z_t | x_t) \cdot \int p(x_t | x_{t-1}, u_t) \cdot p(x_{t-1} | z_{1:t-1}, u_{1:t-1}) dx_{t-1}, \quad (2)$$

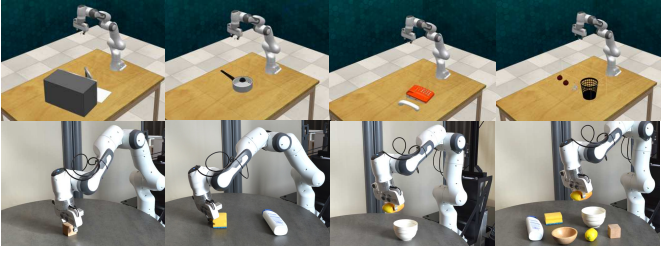


Fig. 5: RL Bench Tasks: CloseMicrowave, TakeLidOffSaucepan, PhoneOnBase, PutRubbishInBin. Real-world tasks: PickUp, PushToGoal, PickAndPlace, and PickAndPlace with clutter.

where  $P(z_t | x_t)$  is referred to as the measurement model and  $P(x_t | x_{t-1}, u_t)$  as the motion model. However, the integration in Eq. (2) is not always computationally feasible. Instead, in many applications, approximations of the posterior have been successfully utilized. These include the Kalman filters that can express Gaussian beliefs and the particle filter for arbitrary hypotheses.

2) *Application*: We propose two models. For a single-camera setup, a discrete filter can be used by defining the camera’s pixel space as the hypothesis space ( $d = 2$ ). Here, the main objective is to resolve the multimodality of the hypothesis distributions. For a multi-camera setup, as well as to better handle occlusions and objects outside the current view frustum, we propose a particle filter. Here, the hypothesis space is given by the world coordinate space ( $d = 3$ ). Whereas in the discrete filter, each pixel in the camera’s pixel space is a hypothesis, in the particle filter the hypothesis distribution is defined by a set of particles. Each particle has an associated 3D location and weight that together express the filter’s posterior. In both filters, the location and weight of the hypotheses are updated using the motion and measurement model respectively. After each update, we normalize the hypothesis likelihoods and return the weighted average of all hypotheses.

3) *Motion Model*: For the discrete filter, the hypothesis space is the camera’s pixel space. Thus, at each time step, we correct for the movement of the camera by projecting the hypothesis distribution from the previous pixel space onto the current one. For the particle filter which is anchored in the world frame, this is not necessary. Meanwhile, in both filters, the movements of scene objects are modeled using a random walk with a fixed magnitude.

$$\bar{x}_t = x_{t-1} + m_r, \quad m_r \sim \mathcal{N}(0, \sigma_r^2) \quad (3)$$

Additionally, for the particle filter, we use the observation that the movements of scene objects are correlated with the movements of the robot gripper. This allows us to reduce the required magnitude of the random walk, by stochastically applying the gripper motion  $G_t$  to each particle as

$$\bar{x}_t = x_{t-1} + m_r + m_w \cdot G_t, \quad m_w \sim \mathcal{B}(p_w) \quad (4)$$

where  $\mathcal{N}$  denotes a Gaussian and  $\mathcal{B}$  a Bernoulli distribution.  $\sigma_r^2, p_w$  are hyperparameters.

4) *Measurement Model*: For the discrete filter, we use the correspondence model of the DON. Recall that this is given by the softmax  $\sigma(z_t)$  over the current correspondence map as

$$x_t = \bar{x}_t \odot \sigma(z_t). \quad (5)$$

In contrast, for the particle filter our measurement model consists of three components: a correspondence likelihood model  $p_c$ , a depth likelihood model  $p_d$ , and an occlusion model  $p_o$ . As any particle might not be observed in an observation, we cannot use the softmax over the current observation as the correspondence model. Instead, we use the underlying exponential function but omit the normalization component

$$p_c(z_t | x_t) = e^{-\alpha \cdot z_t}, \quad (6)$$

where the temperature  $\alpha$  is a hyperparameter that allows us to tweak the measurement model for the expected value range, as shown in Fig. 4.

To distinguish between particles with identical pixel coordinates but differing depths, we leverage an RGB-D camera. We then factorize the measurement model into pixel-correspondence likelihood and depth likelihood. To this end, we assume conditional independence of the depth measurement  $z_t^d$  and correspondence map  $z_t^v$ , i.e.  $z_t^v \perp\!\!\!\perp z_t^d | x_t$ :

$$p(z_t | x_t) = p(z_t^d, z_t^v | x_t) = p_d(z_t^d | x_t) \cdot p_c(z_t^v | x_t) \quad (7)$$

The depth-likelihood  $p_d$  of a particle is then given by the noise model of the depth sensor. We assume a zero-mean Gaussian, although more complex noise models can be used as well. So far, our measurement model is only suited for *observed* particles, not for particles outside the view frustum and occluded particles. To detect occlusions, we again use the depth model and compare the expected depth of a particle  $x_t^d$  with the measured depth  $z_t^d$  at the particle’s position. Additionally, we can leverage the assumption that an occlusion has only occurred if the depth difference  $z_t^d - x_t^d$  is larger than some margin  $\epsilon$ . This margin  $\epsilon \geq 0$  allows us to encode additional prior knowledge, leading to more robust occlusion detection. For example, an occlusion caused by the robot or another object should lead to a depth difference of several centimeters. In practice  $\epsilon = 5$  cm works well, however,  $\epsilon = 0$  cm can be used to detect arbitrary occlusions as well. Let  $p_o(z_t^d | x_t^d)$  denote the particle’s occlusion likelihood. Then:

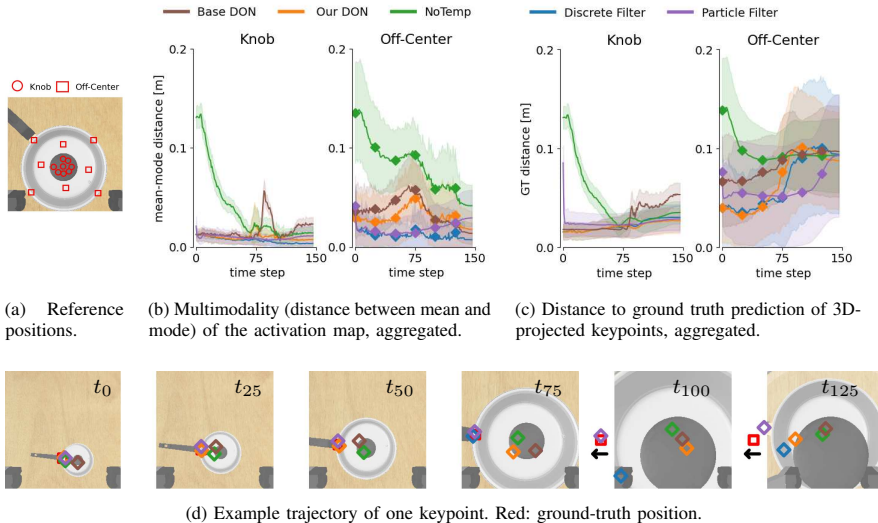
$$p_o(z_t^d | x_t^d) = 1 - p(z_t^d < x_t^d - \epsilon) \quad (8)$$

$$= 1 - \int_{-\infty}^{z_t^d} p_d(y | x_t^d - \epsilon) dy. \quad (9)$$

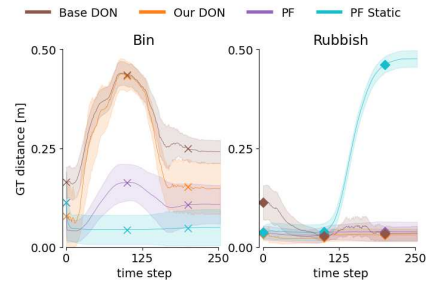
In the case of occlusion, neither the descriptor  $z_t^v$  nor the depth measurement  $z_t^d$  is informative of the particle and we update the particle’s likelihood using a constant value  $\tau \in (0, 1)$ . The same holds true for particles outside the view frustum, which we detect by projecting their position onto the camera plane. Let  $p_f(z_t) := \mathbf{1}_{\text{fov}}(z_t)$  indicate whether the particle lies *inside* the view frustum. Then, we combine all parts of our measurement model as

$$p(z_t | x_t) = p_o(z_t^d | x_t^d) \cdot p_f(z_t) \cdot \tau + (1 - p_f(z_t)) \cdot \tau + (1 - p_o(z_t^d | x_t^d)) \cdot p_f(z_t) \cdot p_d(z_t^d | x_t) \cdot p_c(z_t^v | x_t). \quad (10)$$

5) *Initialization and Resampling*: We initialize the discrete filter using a uniform prior. For the particle filter, we sample from the first correspondence likelihood map to ensure representativeness. To avoid particle impoverishment, we use stratified resampling and resample as soon as the estimated fraction of effective particles  $\hat{N}_{\text{eff}} = (\sum_i w_i^2)^{-1}$  drops below



(a) Reference positions. (b) Multimodality (distance between mean and mode) of the activation map, aggregated. (c) Distance to ground truth prediction of 3D-projected keypoints, aggregated.



(a) Distance to ground truth prediction of 3D-projected keypoints, aggregated across keypoints and trajectories. (b) Example trajectory of one keypoint per object.

Fig. 6: Correspondence quality for symmetrical objects. Thick lines and shaded areas indicate mean and standard deviation across trajectories and keypoints. As predicted, the unfiltered DONs produce a multimodal hypothesis for the off-center positions, when the spatial context is removed ( $t_{75}$ ), leading to a steep increase in localization error. Both the discrete and particle filter resolve these multi-modalities. While the discrete filter, too, starts to fail when the ground truth position moves outside the field of view ( $t_{100}$ ), the particle filter does not but only decreases in accuracy later due to the transparency of the lid ( $t_{125}$ ). Both also show the effectiveness of using a temperature in the softmax for reducing the influence of background noise, as introduced in Sec. III-A4. Subfig. d illustrates how the unmodified base DON (brown) fails to reliably find correspondences across the trajectory.

Fig. 7: Correspondence accuracy for unobserved and moving objects. While the unfiltered DON fails to accurately localize the rubbish bin ( $\times$ ) outside the camera’s fov ( $t_{100}$ ), the particle filter does not. Injecting noise into the filter can reduce correspondence quality (purple), but the filter corrects when the bin comes back into view ( $t_{200}$ ). In contrast, not doing so, disallows the model to track the moving piece of rubbish ( $\diamond$ , cyan).

a fixed threshold. As some objects might not be visible in the first observation, we randomly resample particles from each observation with some small probability. This technique allows the filter to quickly correct once the object becomes visible without impairing localization quality.

#### IV. EXPERIMENTAL EVALUATIONS

We perform experiments in simulation using RL Bench [12], a suite of manipulation tasks using everyday objects, and in the real world using the Franka Emika robot arm. In both cases, the objects are placed randomly in the scene. We evaluate on a challenging and representative subset of tasks, showing important *visual* challenges in robotic manipulation. In *CloseMicrowave*, the policy is confronted with an articulated object, whereas in *TakeLidOffSaucepan* there is high object symmetry and transparency of the lid. *PhoneOnBase* is a difficult multi-object task that requires careful alignment of the gripper, and *PutRubbishInBin* introduces visual clutter. The latter two tasks further introduce occlusions and the need to track multiple objects. We pre-train the dense correspondence for all tasks as described in Sec. III-A1.

##### A. Correspondence Accuracy

To verify that BASK is indeed able to resolve object symmetry and track objects across occlusions, we record 14 task demonstrations for *TakeLidOffSaucepan* and 70 demonstrations for *PutRubbishInBin* using a wrist camera. In *TakeLidOffSaucepan*, we track 8 keypoints on the knob of the lid and 8 that are placed off-center. In *PutRubbishInBin* we track 8 keypoints each on the static bin and the moving piece of rubbish. The prediction accuracy

is given by the Euclidean distance between the 3D-projected keypoints and the respective ground truth position. Results are aggregated across keypoints and trajectories. Furthermore, the distance between the expectation of the posterior density and its global mode (or  $\text{argmax}$ ) serves as an indicator for a multimodal hypothesis distribution. We compare BASK and our improved DON to the original DON [8], but for fairness use the same network architecture (ResNet-101), descriptor dimension (64), and softmax temperature, as well as normalizing the descriptor distance, as described in Sec. III-A3. Furthermore, we compare against a variant of our DON without temperature.

Fig. 6 presents the results on *TakeLidOffSaucepan* and shows the emergence of a multimodal hypothesis for the DON as scene context is removed from the observation. This is associated with a drop in accuracy which the discrete filter avoids. Furthermore, the particle filter even stays accurate as the ground truth locations move outside the camera’s field of view. Fig. 7 shows the accuracy for *PutRubbishInBin*. This plot highlights the particle filter’s ability to recall the location of unobserved objects over long stretches of time, all the while tracking moving objects. Without our improvements, the base DON fails to reliably track keypoints across scale differences and occlusions. Fig. 7b in particular highlights how the base DON yields poor keypoint predictions in the presence of multiple objects, which our model effectively handles.

##### B. Policy Learning in Simulation

We train a 2-layer LSTM via behavioral cloning. The action space is given by the change in the robot’s end-effector pose and the observation space is given by the respective visual representation, concatenated with the robot’s end-effector pose in the world frame. For the particle filter, we add zero-mean Gaussian noise with  $\sigma = 0.02$  to the predicted keypoint

TABLE I: Success rates of the learned policies in simulation for different representation learning methods. Mean, standard dev. across three training seeds.

Camera		Wrist-Only				+ Overhead	
Method	Task	Microwave	Lid	Phone	Rubbish	Phone	Rubbish
	<b>Baselines</b>	CNN	0.72 ± 0.09	0.93 ± 0.04	0.62 ± 0.03	0.39 ± 0.26	0.47 ± 0.12
$\beta$ -VAE [13]		0.81 ± 0.03	0.82 ± 0.04	0.03 ± 0.03	0.04 ± 0.00	0.06 ± 0.06	0.07 ± 0.01
Transporter [17]		0.75 ± 0.04	0.98 ± 0.04	0.54 ± 0.08	0.43 ± 0.04	0.67 ± 0.02	0.46 ± 0.12
MONet [14]		0.82 ± 0.01	0.90 ± 0.12	0.61 ± 0.02	0.70 ± 0.10	0.72 ± 0.02	0.73 ± 0.05
Keypoints [8]		0.72 ± 0.13	0.35 ± 0.14	0.04 ± 0.05	0.04 ± 0.03	0.10 ± 0.02	0.07 ± 0.02
<b>Ours</b>	<i>Keypoints</i>	<b>0.93 ± 0.02</b>	0.92 ± 0.02	<b>0.78 ± 0.02</b>	0.00 ± 0.00	<b>0.78 ± 0.03</b>	<b>0.92 ± 0.02</b>
	<i>BASK: Discrete Filter</i>	<b>0.93 ± 0.02</b>	<b>0.99 ± 0.01</b>	0.59 ± 0.03	0.79 ± 0.03	0.60 ± 0.03	0.79 ± 0.08
	<i>BASK: Particle Filter</i>	<u>0.92 ± 0.02</u>	<u>0.98 ± 0.01</u>	<u>0.77 ± 0.03</u>	<b>0.92 ± 0.01</b>	<u>0.77 ± 0.03</u>	<b>0.92 ± 0.01</b>
<b>Baseline</b>	Ground Truth Keypoints	0.90 ± 0.00	0.95 ± 0.01	0.82 ± 0.01	0.95 ± 0.02	0.82 ± 0.01	0.95 ± 0.02

locations as we observe this to improve policy learning over using the low variance keypoint predictions directly. We provide the policy model with 14 demonstrations for the single-object tasks, using a wrist-mounted camera with  $256 \times 256$  pixels. For the multi-object tasks, we provide 140 demonstrations and use a stereo setup of overhead and a wrist camera with identical resolutions. We train all the policies for 10,000 gradient steps on sub-trajectories consisting of 30 consecutive time steps [39] and on three random seeds. We then evaluate all the policies in the respective task environments for 200 episodes. We compare our method against a suite of other representation learning methods introduced in Sec. II, as well as a ground truth keypoints model. Additionally, we compare against an end-to-end trained 3-layer CNN with one downsampling layer and ELU activations as a baseline.

Tab. I shows the policy success rates. The basic keypoints [8] fail on these tasks due to insufficient scale- and occlusion-invariance. In contrast, our improved keypoints model outperforms the other representations across all the tasks and achieves a performance close to the ground truth model. For the single-object tasks, the discrete filter, strictly improves over the bare keypoints model, even outperforming the ground truth model. When only a few demonstrations are available, a small amount of noise in the representation leads to a more robust policy. For example, on *TakeLidOffSaucepan*, adding iid zero-mean Gaussian noise with  $\sigma = 0.05$  to the ground truth keypoints model’s prediction leads to improved policy success of  $0.99 \pm 0.01$ . On the multi-object tasks, data is more plentiful, with the ground-truth model matching the success rate of the human demonstrator.

Our improved keypoints model performs close to optimally as long as all relevant scene details are visible. However, when learning from a wrist camera only, the model completely fails on *PutRubbishInBin* due to visual clutter. The particle filter matches the keypoints model’s performance in all cases (up to statistical certainty) and improves over it in a number of important cases. Crucially, it is not affected by the clutter in *PutRubbishInBin* when learning from a wrist camera. Moreover, it is as effective as the discrete filter when confronted with the symmetrical lid and it is more sample efficient and robust towards observation dropouts, as shown in Sec. S.2 in the supplementary material.

TABLE II: Success rates of the learned policies in real-world experiments.

		PickUp	PushToGoal	PickAndPlace
<b>Baselines</b>	CNN	0.04	0.38	0.00
	MONet [14]	0.08	0.44	0.00
<b>Ours</b>	<i>Keypoints</i>	0.40	0.68	0.34
	<i>BASK</i>	<b>1.00</b>	<b>0.88</b>	<b>0.74</b>

### C. Real Robot Policy Learning

We perform real-world policy learning experiments on a Franka Emika robot arm with a wrist-mounted Intel Realsense D405 camera. We compare the performance of our improved DON, BASK with the particle filter, a CNN baseline, and MONet [14] as the so far strongest competitor. We design three tasks: *PickUp*, *PushToGoal*, and *PickAndPlace*, depicted in Fig. 5. We collect 35 demonstrations for *PickUp*, 50 demonstrations for *PushToGoal*, and 100 demonstrations for *PickAndPlace*. All policies are trained for 15,000 gradient steps and evaluated for 50 episodes.

The challenges in real-world data clearly show in Tab. II. Although MONet performs well in simulation, where idealized graphics allow it to easily segment objects by color, it is not effective in real-world experiments. We provide qualitative insights in Sec. S.1 in the supplementary material. In contrast, our improved keypoints model performs strongly on the *PushToGoal* task. Nevertheless, BASK outperforms it substantially on multi-object tasks. Furthermore, BASK stabilizes localization at close ranges, significantly improving policy success on the *PickUp* task as well.

As the supplementary videos show, BASK is robust towards visual clutter and manual intervention in the scene such as moving task objects. Both these qualities emerge without additional training. Even more, we observe zero-shot generalization of the policy to previously unseen object, as reported in related work [8], and to unseen task environments.

## V. CONCLUSIONS

In this work, we introduce Bayesian Scene Keypoints (BASK) as a novel representation for robotic manipulation. BASK overcomes the inherent limitations of current representation learning methods with respect to scale invariance and ambiguities, e.g. due to occlusions and a limited field of view. It allows for efficient policy learning in multi-object scenes, especially when learning from wrist camera observations on a

real robot. Moreover, it facilitates robustness towards visual clutter and disturbances, as well as effectively generalizing to unseen objects. Thus, it opens up a plethora of applications such as learning from wrist cameras, as well as flexible deployment in homes and in mobile manipulation.

Our Bayesian framework is agnostic towards the representation learning method itself as well as the policy learning approach. Hence, it can be employed for alternative representations and pretraining schemes as well as novel policy learning methods. Finally, using the particle spread as a certainty measure for the localization of a keypoint opens up further research directions toward Bayesian policies and active camera control for optimizing the localization certainty.

## REFERENCES

- [1] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.
- [2] E. Chisari, T. Welschhold, J. Boedecker, W. Burgard, and A. Valada, "Correct me if i am wrong: Interactive learning for robotic manipulation," *IEEE Robotics and Automation Letters*, 2022.
- [3] W. Burgard, A. Valada, N. Radwan, T. Naseer, J. Zhang, J. Vertens, O. Mees, A. Eitel, and G. Oliveira, "Perspectives on deep multimodel robot learning," in *Robotics Research: The 18th International Symposium ISRR*. Springer, 2020, pp. 17–24.
- [4] M. Wulfmeier, A. Byravan, T. Hertweck, I. Higgins, A. Gupta, T. Kulkarni, M. Reynolds, D. Teplyashin, R. Hafner, T. Lampe, *et al.*, "Representation matters: Improving perception and exploration for robotics," in *Int. Conf. on Robotics and Automation*, 2021, pp. 6512–6519.
- [5] F. Schmalstieg, D. Honerkamp, T. Welschhold, and A. Valada, "Learning long-horizon robot exploration strategies for multi-object search in continuous action spaces," in *Robotics Research*, 2023, pp. 52–66.
- [6] A. Younes, D. Honerkamp, T. Welschhold, and A. Valada, "Catch me if you hear me: Audio-visual navigation in complex unmapped environments with moving sounds," *IEEE Robotics and Automation Letters*, 2023.
- [7] T. Drew, M. L.-H. Võ, and J. M. Wolfe, "The invisible gorilla strikes again: Sustained inattention blindness in expert observers," *Psychological science*, vol. 24, no. 9, pp. 1848–1853, 2013.
- [8] P. Florence, L. Manuelli, and R. Tedrake, "Self-supervised correspondence in visuomotor policy learning," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 492–499, 2019.
- [9] C. Graf, D. B. Adrian, J. Weil, M. Gabriel, P. Schillinger, M. Spies, H. Neumann, and A. Kupcsik, "Learning dense visual descriptors using image augmentations for robot manipulation tasks," *arXiv preprint arXiv:2209.05213*, 2022.
- [10] K. Hsu, M. J. Kim, R. Rafailov, J. Wu, and C. Finn, "Vision-based manipulators need to also see from their hands," in *Int. Conf. on Learning Representations*, 2022.
- [11] D. Honerkamp, T. Welschhold, and A. Valada, "N2m2: Learning navigation for arbitrary mobile manipulation motions in unseen and dynamic environments," *arXiv preprint arXiv:2206.08737*, 2022.
- [12] S. James, Z. Ma, D. Rovick Arrojo, and A. J. Davison, "Rlbench: The robot learning benchmark & learning environment," *IEEE Robotics and Automation Letters*, 2020.
- [13] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *Int. Conf. on Learning Representations*, 2017.
- [14] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner, "Monet: Unsupervised scene decomposition and representation," *arXiv preprint arXiv:1901.11390*, 2019.
- [15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.
- [16] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2019, pp. 3343–3352.
- [17] T. D. Kulkarni, A. Gupta, C. Ionescu, S. Borgeaud, M. Reynolds, A. Zisserman, and V. Mnih, "Unsupervised learning of object keypoints for perception and control," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [18] X. Deng, A. Mousavian, Y. Xiang, F. Xia, T. Bretl, and D. Fox, "Poserbpf: A rao-blackwellized particle filter for6d object pose estimation," in *Robotics: Science and Systems*, 2019.
- [19] N. A. Piga, F. Bottarel, C. Fantacci, G. Vezzani, U. Pattacini, and L. Natale, "Maskukf: An instance segmentation aided unscented kalman filter for 6d object pose and velocity tracking," *Frontiers in Robotics and AI*, vol. 8, p. 38, 2021.
- [20] J. Sun, Z. Wang, S. Zhang, X. He, H. Zhao, G. Zhang, and X. Zhou, "Onepose: One-shot object pose estimation without CAD models," *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2022.
- [21] Y. Liu, Y. Wen, S. Peng, C. Lin, X. Long, T. Komura, and W. Wang, "Gen6d: Generalizable model-free 6-dof object pose estimation from rgb images," in *Europ. Conf. on Computer Vision*, 2022.
- [22] Z. Qin, K. Fang, Y. Zhu, L. Fei-Fei, and S. Savarese, "Keto: Learning keypoint representations for tool manipulation," in *Int. Conf. on Robotics and Automation*, 2020, pp. 7278–7285.
- [23] A. Simeonov, Y. Du, A. Tagliasacchi, J. B. Tenenbaum, A. Rodriguez, P. Agrawal, and V. Sitzmann, "Neural descriptor fields: Se (3)-equivariant object representations for manipulation," in *Int. Conf. on Robotics and Automation*, 2022, pp. 6394–6400.
- [24] B. Chen, P. Abbeel, and D. Pathak, "Unsupervised learning of visual 3d keypoints for control," in *Int. Conf. on Machine Learning*, 2021, pp. 1539–1549.
- [25] M. Vecerik, J.-B. Regli, O. Sushkov, D. Barker, R. Pevcecviute, T. Rothörl, R. Hadsell, L. Agapito, and J. Scholz, "S3k: Self-supervised semantic keypoints for robotic manipulation via multi-view consistency," in *Conference on Robot Learning*. PMLR, 2021, pp. 449–460.
- [26] P. Florence, L. Manuelli, and R. Tedrake, "Dense object nets: Learning dense visual object descriptors by and for robotic manipulation," *Conf. on Robot Learning*, 2018.
- [27] D. Hadjiveličkov and D. Kanoulas, "Fully self-supervised class awareness in dense object descriptors," in *Conf. on Robot Learning*, 2022, pp. 1522–1531.
- [28] D. B. Adrian, A. G. Kupcsik, M. Spies, and H. Neumann, "Efficient and robust training of dense object nets for multi-object robot manipulation," in *Int. Conf. on Robotics and Automation*, 2022, pp. 1562–1568.
- [29] B. Han, C. Yang, R. Duraiswami, and L. Davis, "Bayesian filtering and integral image for visual tracking," in *Workshop on Image Analysis for Multimedia Interactive Services*, 2005.
- [30] N. M. Oliver, B. Rosario, and A. P. Pentland, "A bayesian computer vision system for modeling human interactions," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 831–843, 2000.
- [31] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2020, pp. 4938–4947.
- [32] C. Ma, X. Yang, C. Zhang, and M.-H. Yang, "Long-term correlation tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5388–5396.
- [33] T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environment," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 2. IEEE, 2004, pp. II–II.
- [34] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 1269–1276.
- [35] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *kdd*, vol. 96, 1996, pp. 226–231.
- [36] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness," *arXiv preprint arXiv:1811.12231*, 2018.
- [37] M. M. Naseer, K. Ranasinghe, S. H. Khan, M. Hayat, F. Shahbaz Khan, and M.-H. Yang, "Intriguing properties of vision transformers," *Advances in Neural Information Processing Systems*, vol. 34, 2021.
- [38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [39] 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," in *Conf. on Robot Learning*, 2021.