Learning Dense Visual Descriptors using Image Augmentations for Robot Manipulation Tasks

Anonymous Author(s)
Affiliation
Address
email

Abstract: We propose a self-supervised training approach for learning view-invariant dense visual descriptors using image augmentations. Unlike existing works, which often require complex datasets, such as registered RGBD sequences, we train on an unordered set of RGB images. This allows for learning from a single camera view, e.g., in an existing robotic cell with a fix-mounted camera. We create synthetic views and dense pixel correspondences using data augmentations. We find our descriptors are competitive to the existing methods, despite the simpler data recording and setup requirements. We show that training on synthetic correspondences provides descriptor consistency across a broad range of camera views. We compare against training with geometric correspondence from multiple views and provide ablation studies. We also show a robotic bin-picking experiment using descriptors learned from a fix-mounted camera for defining grasp preferences.

Keywords: self-supervised learning, computer vision, representation learning, bin-picking

1 Introduction

Scene and object understanding is essential for robot manipulation tasks, including assembly or bin picking. Often, the representation of choice is task-specific segmentation or pose estimation, trained in a supervised manner with labeled data. Labeling, however, is expensive and time-consuming, which is why self-supervised learning of dense visual descriptors has recently gained substantial attention in the robotics community, inspired by the works of Schmidt et al. [1] and Florence et al. [2].

Dense Object Nets (DONs) proposed by Florence et al. [2] learn dense visual descriptors of objects fully self-supervised in a robotic environment. The learned descriptors are view-invariant, show potential for within-class generalization and they naturally apply to non-rigid objects. The dense descriptor representation can be flexibly used for various downstream robotic tasks, such as, grasping (Florence et al. [2], Kupcsik et al. [3], Adrian et al. [4]), rope manipulation (Sundaresan et al. [5]) and learning control (Manuelli et al. [6], Florence et al. [7]).

Self-supervised training of DONs, however, relies on pixel correspondences across multiple camera views provided by a registered RGBD image sequence, which requires accurate camera calibration and pose recording. Furthermore, pixel correspondence tends to be inaccurate with inexpensive depth cameras, even with data preprocessing. Finally, data collection is constrained by robot kinematics and the need for an expert setting up and supervising the procedure.

Figure 1: Bin-picking setup with a single, fix-mounted overhead camera.

Submitted to the 6th Conference on Robot Learning (CoRL 2022). Do not distribute.

In this paper we relax these assumptions fundamentally and instead of a complex setup with a single, robot-mounted moving camera, or multiple static ones, we solely rely on an unordered set of RGB images to learn object descriptors, for example, recorded by a single fixed camera. In our work, instead of relying on multi-view, *geometric correspondence*, we use augmentations of single images to obtain alternative views and *synthetic correspondence*. This idea was already explored in computer vision by Thewlis et al. [8] and Novotny et al. [9], in the context of learning geometrically consistent pixel-level descriptors across multiple object classes. While our approach does not require a robot, depth camera, or scene registration, we show that it still achieves competitive performance in terms of keypoint tracking accuracy compared to a network trained with geometric correspondence. Our approach can easily be adopted to existing industrial setups with fix-mounted cameras, or with cameras too heavy to be mounted on a robotic arm, without additional engineering effort. We show such a robotic bin-picking setup in Fig. 1, with an overhead, fix-mounted camera.

Our contributions are as follows: (i) we combine the approach of learning visual descriptors from single images by Novotny et al. [9] with the image augmentation technique and loss formulation by Chen et al. [10], (ii) we compare learning with synthetic and geometric correspondence on keypoint tracking accuracy, and finally (iii) we show a robotic bin-picking experiment where human preference on grasp configuration is encoded with dense visual descriptors, with the constraint of using a single, fixed-mounted camera.

2 Related work

In the following, we review recent work on self-supervised dense visual descriptor learning for robotic manipulation in more detail. We also discuss related work on self-supervised representation learning and learning from a set of single images, which are core concepts in our work.

Dense visual descriptors in robotic manipulation. Inspired by the work of Schmidt et al. [1], Florence et al. [2] proposed self-supervised training of dense visual descriptors by and for robotic manipulation. Their approach was later adopted by Florence et al. [7] to learn from multi-view correspondence in dynamic scenes and showed an application for policy learning. Sundaresan et al. [5] applied the descriptor space representation to learn challenging rope manipulation in simulations. Applying the dense descriptor representation to learn model predictive controllers was shown by Manuelli et al. [6]. Vecerik et al. [11] use multi-view consistency for keypoint detection and show an application for reinforcement learning.

Another line of work investigated improved training strategies of dense visual descriptors. There are multiple contributions focusing on learning multi-object and multi-class descriptors by Yang et al. [12], Hadjivelichkov and Kanoulas [13] and Adrian et al. [4]. The work by Kupcsik et al. [3] exploits known object geometry to compute optimal descriptor embeddings. Finally, Yen-Chen et al. [14] utilize NeRF to generate dense correspondence datasets from RGB images. This alleviates problems with noisy depth data and proves especially helpful for thin and reflective objects. Our proposed solution also relies only on RGB images, but without the need for registration, thus allowing for an unordered RGB set, or for images taken by a single, fix-mounted camera.

Self-supervised descriptor learning from single images. An intuitive way to generate geometric correspondence is to estimate the optical flow of subsequent frames from a video. Deekshith et al. [15] adopted this technique to train DONs using contrastive learning. Thewlis et al. [8] proposed to use optical flow from videos, or image augmentations to embed pixels of objects in view-invariant coordinate frames. Novotny et al. [9] adopted this method for pretraining of geometry-oriented tasks, such as object specific part detection in images. Zhang and Maire [16] propose to learn pixel-wise descriptors from single images with augmentations by using hierarchical visual grouping of image patches based on contour. Our work follows the image augmentation technique of single RGB images to generate alternative views and synthetic correspondence.

Self-supervised visual representations learning. Instead of training on large supervised datasets, self-supervised methods have become a popular way to obtain visual representations, which can be fine-tuned to specific downstream tasks. A recent and very successful approach using contrastive learning is SimCLR by Chen et al. [10]. It aims to maximize agreement between two augmented versions of the same image, while considering all other images in the batch as negative samples. Grill et al. [17] proposed BYOL (Bootstrap your own latent) that, in comparison to contrastive methods, does not rely on the sampling of negatives. With Barlow Twins Zbontar et al. [18] also

forgo negative samples by optimizing the cross-correlation matrix between embeddings from two augmented versions of the same image to be close to identity. Our approach is most similar to SimCLR as we employ the same loss formulation, but with the important difference that our batch is constituted by individual pixel descriptors instead of full image embeddings.

3 Method

In this section we discuss our proposed training approach using image augmentations. We first give an overview of the whole training pipeline, then discuss image augmentation techniques and finally present the loss formulation and dataset requirements. For an illustration of the training pipeline we refer to Fig. 2.

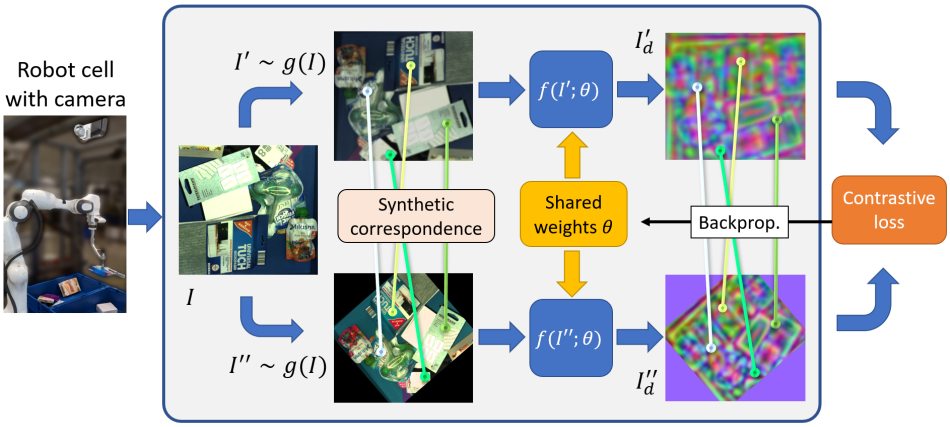


Figure 2: Illustration of the training pipeline. For every RGB image I in a batch, we sample a pair of augmented images I' and I'' while keeping track of the pixel correspondences. Then, we evaluate the RGB images with the trained fully-convolutional network $f(\cdot;\theta)$ with trainable parameters θ. Finally, we compute the contrastive loss and backpropagate the error using the embedded images I'_d and I''_d along with the correspondence information.

3.1 Dense Descriptor Training with Synthetic Correspondence

Inspired by the work of Novotny et al. [9] we rely on training on an unordered set of images and use image augmentations to arrive at alternative views of each image. First, we sample a minibatch of N RGB images from the training data set consisting of independent RGB images. For every image I in the minibatch we sample two augmented views $I' \sim g(I)$ and $I'' \sim g(I)$ by applying randomized augmentations $g : \mathbb{R}^{H\times W\times 3} \mapsto \mathbb{R}^{H\times W\times 3}$ (described in more detail below). A learned fully-convolutional network [19] model $f(\cdot;\theta)$, $f : \mathbb{R}^{H\times W\times 3} \mapsto \mathbb{R}^{H\times W\times D}$ maps the augmented images I' and I'' to their descriptor space embeddings I'_d and I''_d. The user defined parameter $D \in \mathbb{N}^+$ controls the resolution of the descriptor space.

By keeping track of the origin of pixels before the augmentations, we sample pairs of pixel locations between I' and I'' that have the same origin in I. We refer to these as *synthetic* correspondences, emphasizing the use of synthetic image augmentations as opposed to *geometric* correspondences coming from the 3D geometry of multiple camera views, as in [1, 2]. The descriptor values at the sampled pixel correspondence locations serve as positive pairs for contrastive learning.

3.2 Image Augmentations

For robotic applications we require descriptors that are invariant to translations, rotations and perspective changes of the objects, as well as changes of lighting conditions. In vanilla DONs training, cf. Florence et al. [2], this is achieved by recording a diverse training set of registered RGBD image sequences, which contain sufficient variance in camera and object poses, and to some extent in lighting conditions. In our work, we achieve a similar effect purely by imposing data augmentations on single RGB images from an unordered set.

(a) Original (b) Affine (c) Perspective (d) Resize&Crop (e) Color Jitter (f) Combined

Figure 3: Visualization of the augmentations utilized for synthetic view training. Figure 3f shows the combination of all augmentations as used in practice for training a synthetic view descriptor model. While each augmentation is guaranteed to be applied, the individual parameters, e.g., scale of distortion, angle of rotation, crop size and location, etc. are still randomly selected each iteration.

We carefully select augmentations, which reflect the desired invariance properties stated above, as follows: *affine transformation* induce rotations and scale changes (zoom out), *perspective distortions* mimic view changes, *resize&crop* implies scale changes (zoom in), and lastly, *color jitter* affects brightness, contrast, hue, and saturation of the image, hence the lighting conditions of the scene. We show an illustration of these augmentations in Fig. 3. We utilize *torchvision library* [20] of Pytorch as reference implementations.

Adrian et al. [4] already demonstrated the impact on improved performance for training of DONs on geometric datasets. In our work, each augmentation is not only helpful, but relevant to the ability of the model to successfully learn an invariant descriptor space. See Sec. 4.3 for an ablation on the respective impact of each augmentation on the overall performance.

3.3 Loss Function

Following Chen et al. [10] we adopt the NT-Xent loss, for learning the dense descriptor representations. For a pair of corresponding descriptors $\{d_i, d_j\}$, obtained from images I' and I'', we define their individual loss term as

$$l_{i,j} = -\log \frac{\exp(\text{dist}\,(d_i, d_j)\,/\tau)}{\sum_{k=1;k\neq i}^{2M} \exp(\text{dist}\,(d_i, d_k)\,/\tau)}, \tag{1}$$

where we choose the metric $\text{dist}\,(d_i, d_j)$ to be the cosine similarity between descriptors d_i and d_j. The total loss is given as the mean over all individual loss terms, cf. Chen et al. [10].

Note that this loss, together with our correspondence sampling, does not distinguish background from objects, nor does it explicitly address multiple object classes and instances, as opposed to the approach by Florence et al. [2]. Instead, we follow the method of Adrian et al. [4] and sample correspondences uniformly in image plane and assume that every pixel is unique. This method is tailored to datasets depicting densely packed scenes with single object instances, for example, a heap of objects in a bin-picking scenario. The learned descriptor space does not imply semantic information on object classes or background, but still provides consistent keypoint detection and robust tracking performance, which is essential for downstream tasks.

4 Comparison of Training with Synthetic and Geometric Correspondence

In this section we show an in-depth comparison between training with geometric and synthetic correspondence. We also investigate the invariance of descriptors obtained from a network trained with synthetic correspondence with respect to object-camera relative transformations.

In all our evaluations we utilize a pretrained ResNet-34 with 8-stride output as used by Florence et al. [2], which yields an upsampled output matching the resolution of the input. Note that each descriptor d is normalized since we utilize the cosine similarity in the loss formulation (see Section 3.3). In the supplementary material we give a brief introduction into the baseline training method we use in our comparisons relying on geometric correspondence by Adrian et al. [4].

We recorded a dataset consisting of a set of registered RGBD sequences, to enable comparison between both approaches. Despite the availability for registered image pairs, the synthetic view training only uses single RGB images for both training and validation. However, the camera poses help with the evaluation as they allow us to generate ground truth pixel matches across any two

images of the same static scene without the need for manual labeling. The dataset consists of eight scenes with various object configurations, with every scene containing only one instance per object, and every object is visible to some extent in every frame. The scenes are recorded with a robot wrist mounted camera while the robot arm follows a predefined trajectory keeping the objects in view.

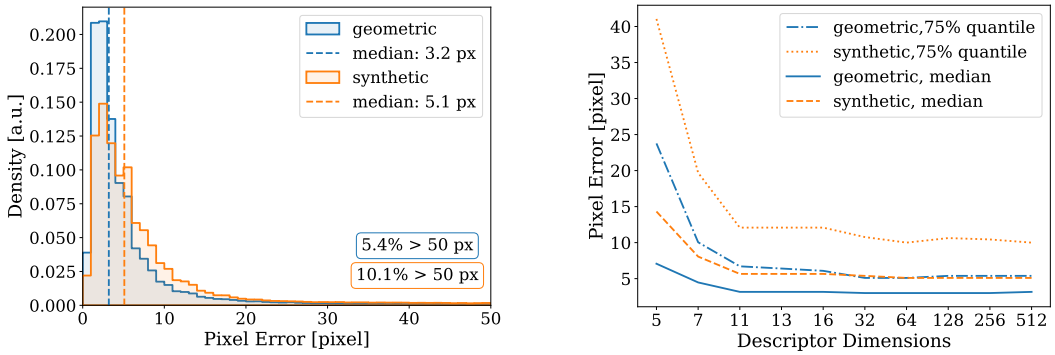
Both approaches are evaluated on the same ground-truth image pairs and correspondences. For robustness, we perform a k-fold cross-validation, that is, each scene from our total dataset was once used as test set. One scene is chosen as validation set, with the remaining 6 scenes used for training. The averaged results are reported. We use the same loss function and training parameters for both approaches, with the exception that augmentations are chosen with 50% probability for the geometric training, as it yields better results. For synthetic training, each augmentation is always used. Further training details are listed in supplementary materials.

4.1 Keypoint Tracking Performance

Given a common dataset, measuring keypoint tracking accuracy provides a task-agnostic comparison between both training methods. We define a keypoint as a location in image plane $k_i = (u, v)$. Each keypoint is associated with a unique descriptor $d \in \mathbb{R}^D$ and the keypoint tracking problem is defined as finding the pixel location k_i^* closest to d in the descriptor image I_d such that $k_i^* = \arg\min_{k_i} \text{dist}(I_d(k_i), d)$.

Evaluation. From the test set we sample 1000 image pairs $\{A, B\}$ representing alternative views of the same scene. For each image pair we sample 200 keypoints $\{k_i^A, d_A\}$, where $d_A = f(A; \theta)(k_i^A)$ is located on one of the objects in image A. We also recorded keypoint locations in image B, $\{k_i^B\}$, such that every pair $\{k_i^A, k_i^B\}$ unprojects to the same location in world coordinates. Then, we solve the keypoint tracking problem for every descriptor d_A in image B and record the pixel error $e = \|k_i^B - k_i^*\|_2$. The network parameters θ are either obtained by geometric [4] or synthetic correspondence training proposed in Sec. 3.1.

Results. Fig. 4a shows the distribution of pixel errors for the two training approaches. The median of the corresponding distribution is highlighted as a dashed line. With a difference of only 1.9px in median pixel error, the synthetic correspondence performs competitively to the geometric training. The percentage of pixel errors that are larger than 50 pixels are 10.1% and 5.4% respectively, as indicated in the bottom right part of Fig. 4a.



(a) (b)

Figure 4: (a) Pixel error distributions obtained using synthetic (orange) and geometric (blue) correspondence training. (b) Median and 75% quantile of the pixel error distribution for both training approaches and different descriptor dimensions.

In Fig. 4b we compare the median and the 75% quantile of the two training methods with respect to the descriptor dimensions D. The results were obtained without k-fold cross-validation. For both training approaches the median pixel error decreases for larger descriptor dimensions. For dimensions larger than 9 the median pixel error decreases only marginally. In contrast, the 75% quantile error still increases further until $D = 64$, before improvements saturate. For additional insights into these results we refer to the supplementary material.

4.2 Invariance Tests

In the following, we wish to answer the question: how well does training with image augmentations proposed in Sec. 3.2 generalize to physical camera transformations? For this purpose, we recorded different test scenes with a wrist-mounted camera and the following specific camera movements: (i) changing camera perspective (camera tilting), (ii) translation along the camera z-axis (zooming in and out), and (iii) camera rotation along the camera z-axis. We compare the performance of a network trained with synthetic and one with geometric correspondence both with 64 descriptor dimensions and affine, perspective and resize&crop augmentations. As in the previous section we compute the 75% quantile pixel error and use 1000 keypoints per image pair, fix the base image A and only vary image B, which shows the changing camera views. The results for three different types of transformations are compiled in Fig. 5.

It can be seen that training with synthetic correspondence generalizes well to a large range of camera transformations, especially to those parallel to the object plane (Fig. 5a, Fig. 5b). These physical camera transformations are very similar to the affine and resize&crop augmentations used during training. For generalizing to the perspective transformations with angles above $45°$ degrees, as shown in Fig. 5c, the synthetic correspondence training shows a clear deterioration in performance. As the perspective changes, occluded parts become visible and vice versa. This physical transformation is not well captured by the synthetic augmentations for larger angles.

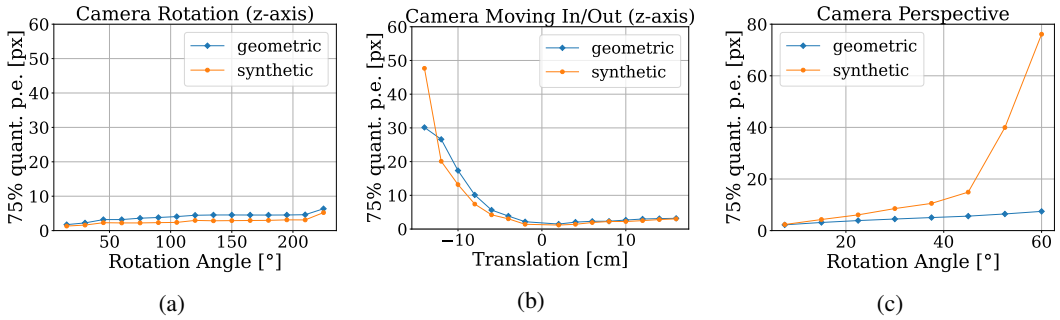


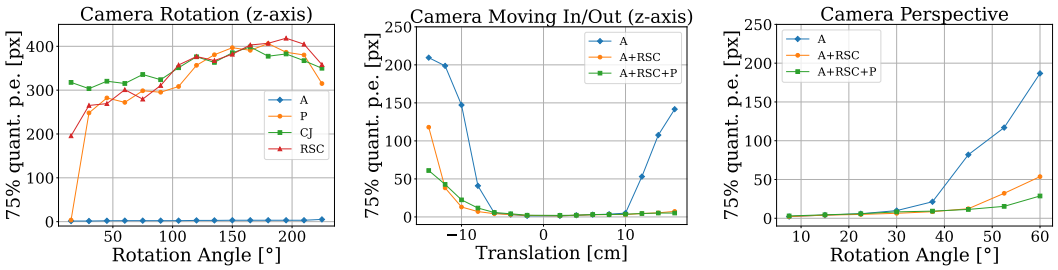
Figure 5: 75% quantile pixel error between two different images for the geometric (blue) and synthetic (orange) correspondence training approach for three different isolated camera movements between those images in varying magnitude: (a) camera rotates around z-axis, (b) camera moving closer and further away from the objects, (c) camera moving on a sphere in x-direction around the objects, facing the objects. Note the scale of the y-axes.

4.3 Augmentations

Complementing the findings in section 4.2 we study the influence of different augmentations for the synthetic correspondence training on the ability of the network to generalize to different camera transformations. Fig. 6 shows the 75% quantile of the pixel error distribution for the synthetic correspondence training with different augmentations. We find that *affine transformations* are most critical, as only a model trained with it shows invariance to rotations, see Fig. 6a. This matches the expectations as standard CNN are by default not invariant to rotations. Nevertheless, we find that both *resize&crop* and *perspective distortion* both further improve the performance of just affine transformations. In particular, for camera movements that induce perspective distortions and scale changes, see Fig. 6c and Fig. 6b, the error decreases considerably. Lastly, we find that *color jitter* further reduces the overall mean pixel error from 19.4 to 17.1 pixel. The improvement appears modest, but we note that our test dataset was recorded at the same time as train and validation, and the lighting conditions of the scene were not explicitly altered. For a complete table of all the combinations of augmentations, see the supplementary material.

5 Grasping Experiment with Fix-mounted Camera

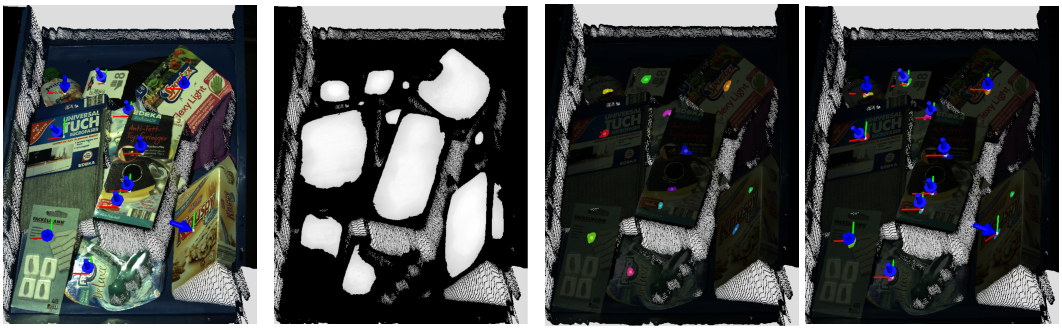
We demonstrate a robotic bin-picking experiment that relies on dense visual descriptors for defining grasp preferences. We use a 7-DoF Franka Emika Panda arm with a suction gripper mounted on the



(a) (b) (c)

Figure 6: 75% quantile pixel error for different combinations of augmentation. **A**ffine, **P**erspective, **C**olor **J**itter, **Re**size & **C**rop, with respect to different tasks: (a) camera rotates around z-axis, (b) camera moving closer and further away from the objects, (c) camera moving on a sphere in x-direction around the objects, facing the objects. Note the scale of the y-axes.

end-effector, see Fig. 1. Our setup uses a fix-mounted Zivid One+ camera above the bin in a robotic cell. In the experiment we consider picking ten different types of objects from one bin. However, using a suction gripper the objects are difficult to grasp at certain locations: some have cutouts on the packaging, transparent or foldable parts, and uneven surfaces (see supplementary material). In our experiments, a purely model-free grasp pose generator often predicts poses at these challenging parts of objects, ultimately reducing picking performance.



(a) (b) (c) (d)

Figure 7: Illustration of the grasping experiment. (a) The RGB images with the predicted grasp poses. (b) The graspable areas generated using RGB and depth images. (c) The predicted keypoint heatmaps on graspable areas. (d) The predicted grasp poses overlayed on the heatmaps.

We hypothesize that introducing human domain knowledge to aid grasp pose selection will lead to higher grasp success rates. Human domain knowledge is considered by highlighting parts of the RGB image where grasp poses are preferred. To do so, we show a small number of RGB images depicting the objects in different configurations in the bin and ask the human to click on pixel locations corresponding to a preferred grasp location. We track these descriptors to generate a *preference heatmap*, as shown in Fig. 7c. To generate the final set of grasp pose candidates we intersect the grasp preference heatmap with the detected graspable areas identified by the model-free grasp detector (Fig. 7b). For the example image the final set of poses are shown in Fig. 7a and Fig. 7d. Note that the descriptor network is trained purely from the set of RGB images of the bin, including the objects in random configurations, recorded with the overhead camera. We require a single instance of the objects to be present in the bin. For more details on the experiment setup and heatmap generation we refer to the supplementary material.

5.1 Quantitative Evaluation

We evaluate the benefit of the proposed method for robotic bin picking based on a set of sixteen manually annotated scenes. The scene images include the ten known objects, but are not contained in the training set. For each image, we manually annotated object instances, pixel-wise graspable

areas, and those areas that correspond to selected descriptors. In total, the evaluation dataset contains 131 graspable objects of which 103 have visible descriptor spots.

Table 1 shows a summary of the results. We compare our descriptor-based grasping approach with a purely model-free approach that directly uses the graspable areas without accounting for descriptors. *Success Rate* denotes the number of successful grasps compared to all grasps attempted. *Descriptor Success* denotes the number of grasps at those spots marked as desired grasping points compared to all grasps attempted. *Object Hits* denotes the percentage of all graspable objects for which at least one feasible grasp pose has been found irrespective of descriptors, including objects without visible descriptor spots. *Descriptor Hits* denotes the percentage of objects for which a grasping pose has been found at the respective descriptor spot, only including objects with visible descriptors.

Table 1: Evaluation results on individual, annotated scene images similar to Fig. 7.

	Success Rate	Descriptor Success	Object Hits	Descriptor Hits
Preference (ours)	98.9%	91.1%	63.4%	78.6%
Baseline	79.9%	50.4%	78.6%	68.0%

It can be concluded from Table 1 that using the proposed method to encode grasp preferences is effective as it significantly increases the amount of grasps at desired spots from 50.4% to 91.1%. Due to the challenging object geometries, this helps to raise the overall grasp success rate from 79.9% to 98.9%. The downside, however, is that the proposed method finds grasp poses only for a smaller amount of objects, 63.4% instead of 78.6%. Still, this includes objects that have no desired grasping spot visible. In some applications it can be the desired behavior to not propose a grasp pose for objects if they cannot be grasped at the preferred location. In consequence, when only considering which preferred grasping spots have been covered by grasp poses, our method manages to find grasps for 78.6% instead of 68.0% of the visible spots.

6 Limitations

In the following we share more insights on the limitations of our method and discuss future research.
Variance in camera poses. As seen in Fig. 5, the synthetic training ensures that descriptors are stable within a limited margin of camera transformations relative to the object. For example, in Fig. 5c, for viewing angles steeper than $45°$, accuracy deteriorates. This imposes a limit on descriptor consistency across images with large differences in object poses.
Changing Environment. A change in environment (background, lighting) between training and inference time may have a negative influence on keypoint tracking performance. It is expected that augmentations such as color jitter and, if masks are available, background randomization can mitigate these effects. We will investigate these aspects in future research.
Generalization to unseen objects. Although Florence et al. [2] demonstrated capabilities to generalize intra-class instances, our work focuses on instance specific descriptors. Given our outlined training setup, it works best on known objects from the training set.
Object Edges. We observe worse descriptor consistency across images for keypoints located at the edges of objects or close to parts that are occluded by other objects. In setups where object masks are present background randomization could reduce this effect.

7 Conclusion

In this paper we proposed a novel training method for learning dense visual descriptors based on image augmentations for robotic manipulation. The evaluation shows that overall our proposed method is competitive to the existing geometric training approach. For physical transformations like changing the camera perspective on the scene, which are harder to mimic by augmentations, the training method using geometric correspondences shows superior performance. Being aware of these limitations our proposed method is expected to perform well for setups where objects are mostly altered by translations, rotations parallel to the camera plane, or are slightly tilted. As this is often the case for random heaps of objects in a bin, our method is especially suitable for such setups that are constrained by a fix-mounted camera. Finally, we demonstrated the use of our method in a realistic grasping experiment to increase grasp success rates by human annotated grasp preferences.

References

[1] T. Schmidt, R. Newcombe, and D. Fox. Self-Supervised Visual Descriptor Learning for Dense Correspondence. *IEEE Robotics and Automation Letters*, 2(2):420–427, 2017.

[2] P. Florence, L. Manuelli, and R. Tedrake. Dense Object Nets: Learning Dense Visual Object Descriptors By and For Robotic Manipulation. *Conference on Robot Learning*, 2018.

[3] A. G. Kupcsik, M. Spies, A. Klein, M. Todescato, N. Waniek, P. Schillinger, and M. Bürger. Supervised training of dense object nets using optimal descriptors for industrial robotic applications. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6093–6100, 2021.

[4] D. B. Adrian, A. G. Kupcsik, M. Spies, and H. Neumann. Efficent and robust training of dense object nets for multi-object robot manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[5] P. Sundaresan, J. Grannen, B. Thananjeyan, A. Balakrishna, M. Laskey, K. Stone, J. E. Gonzalez, and K. Goldberg. Learning rope manipulation policies using dense object descriptors trained on synthetic depth data. *IEEE International Conference on Robotics and Automation*, pages 9411–9418, 2020.

[6] L. Manuelli, Y. Li, P. Florence, and R. Tedrake. Keypoints into the Future: Self-Supervised Correspondence in Model-Based Reinforcement Learning. *Conference on Robot Learning*, 2020.

[7] P. Florence, L. Manuelli, and R. Tedrake. Self-Supervised Correspondence in Visuomotor Policy Learning. *IEEE Robotics and Automation Letters*, 5(2):492–499, 2020.

[8] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised learning of object frames by dense equivariant image labelling. *Advances in Neural Information Processing Systems*, pages 845–856, 2017.

[9] D. Novotny, S. Albanie, D. Larlus, and A. Vedaldi. Self-Supervised Learning of Geometrically Stable Features Through Probabilistic Introspection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3637–3645, 2018.

[10] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[11] M. Vecerik, J.-B. Regli, O. Sushkov, D. Barker, R. Pevceviciute, T. Rothörl, C. Schuster, R. Hadsell, L. Agapito, and J. Scholz. S3K: Self-Supervised Semantic Keypoints for Robotic Manipulation via Multi-View Consistency. *Conference on Robot Learning*, 2020.

[12] S. Yang, W. Zhang, R. Song, J. Cheng, and Y. Li. Learning multi-object dense descriptor for autonomous goal-conditioned grasping. *IEEE Robotics and Automation Letters*, 6(2):4109–4116, 2021.

[13] D. Hadjivelichkov and D. Kanoulas. Fully Self-Supervised Class Awareness in Dense Object Descriptors. *Conference on Robot Learning*, pages 1–10, 2021.

[14] L. Yen-Chen, P. Florence, J. T. Barron, T.-Y. Lin, A. Rodriguez, and P. Isola. Nerf-supervision: Learning dense object descriptors from neural radiance fields. *IEEE International Conference on Robotics and Automation*, 2022.

[15] U. Deekshith, N. Gajjar, M. Schwarz, and S. Behnke. Visual descriptor learning from monocular video. *VISIGRAPP 2020 - Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 5:444–451, 2020.

[16] X. Zhang and M. Maire. Self-supervised visual representation learning from hierarchical grouping. *Advances in Neural Information Processing Systems*, (NeurIPS):1–12, 2020.

[17] J. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. Á. Pires, Z. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko. Bootstrap your own latent - A new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 2020.

[18] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 12310–12320. PMLR, 2021.

[19] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, 2017.

[20] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.