2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Cycle-Correspondence Loss: Learning Dense View-Invariant Visual Features from Unlabeled and Unordered RGB Images

David B. Adrian[1,2], Andras Gabor Kupcsik[1], Markus Spies[1], Heiko Neumann[2]

***Abstract*— Robot manipulation relying on learned object-centric descriptors became popular in recent years. Visual descriptors can easily describe manipulation task objectives, they can be learned efficiently using self-supervision, and they can encode actuated and even non-rigid objects. However, learning robust, view-invariant keypoints in a self-supervised approach requires a meticulous data collection approach involving precise calibration and expert supervision. In this paper we introduce *Cycle-Correspondence Loss* (CCL) for view-invariant dense descriptor learning, which adopts the concept of cycle-consistency, enabling a simple data collection pipeline and training on unpaired RGB camera views. The key idea is to autonomously detect valid pixel correspondences by attempting to use a prediction over a new image to predict the original pixel in the original image, while scaling error terms based on the estimated confidence. Our evaluation shows that we outperform other self-supervised RGB-only methods, and approach performance of supervised methods, both with respect to keypoint tracking as well as for a robot grasping downstream task.**

I. Introduction

Dense visual descriptors have proven to be a flexible, easy to learn, and easy to use object representation for robot manipulation in recent years. They show potential for class-level object generalization [1], they can describe non-rigid objects [2], and they can be seamlessly applied for state-representation for control [3]–[5]. A dense descriptor network maps an RGB image of size $3 \times H \times W$ to a descriptor space image of size $D \times H \times W$, where D is the user-defined descriptor dimension.

Training a dense descriptor network, such as a Dense Object Net (DON) [1], relies on multiple views of the same object(s) and dense pixel correspondences computed from 3D geometry [1], [6]. Alternatively, RGB image augmentations can generate alternative views of the same image, while keeping track of pixel correspondences [7]–[9]. Training is commonly achieved, e.g., via contrastive [10], [11] or probabilistic [4] losses.

Utilizing pixel correspondences computed by 3D geometry naturally encodes physically distinct views of the same object(s), thus encouraging truly view-invariant descriptors. However, this requires a registered RGB-D dataset [1] or trained NeRF [12], which is often laborious due to camera calibration, hardware setup, and data logging. This is exactly the problem the RGB image augmentation approaches [7], [9], [13] aim to solve: they only require an unordered set of RGB images depicting the object(s), which can be recorded even with a smartphone. However, the learned descriptors cannot handle excessive camera view changes [9], and thus, they are not always view-invariant, which limits their applicability. In this work our aim is to combine the best of both worlds. Firstly, we wish to keep the simple data collection approach, that is, relying only on a set of unordered RGB images showing the objects. Secondly, we aim to improve the view-invariance of the descriptors, making them more robust to camera view changes or extreme object positions.

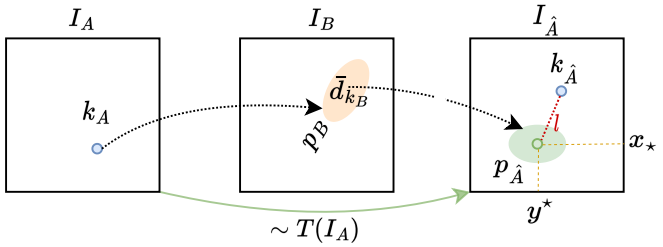

Fig. 1: Overview of the cycle-correspondence loss. $\boldsymbol{I}_A$ and $\boldsymbol{I}_{\hat{A}}$ denote versions of the same image, both related through a random image transformation $\sim T$. $\boldsymbol{I}_B$ is a randomly sampled image that exhibits partial content overlap with $\boldsymbol{I}_A$. We establish a correspondence cycle by randomly sampling location $\boldsymbol{k}_A$ on $\boldsymbol{I}_A$, computing a matching distribution p_B over $\boldsymbol{I}_B$ which we utilize to predict $\boldsymbol{k}_{\hat{A}}$ on $\boldsymbol{I}_{\hat{A}}$. As location $\boldsymbol{k}_{\hat{A}}$ is known through the augmentation, we can optimize the prediction error l to improve the model. We utilize the predicted distributions to scale individual error terms l by the associated uncertainty, effectively dealing with sampled $\boldsymbol{k}_A$ that have no valid correspondence in $\boldsymbol{I}_B$.

To this end, we introduce the Cycle-Correspondence Loss (CCL), a self-supervised loss for dense visual feature models using only unlabled, random pairs of RGB images. The core idea, based on *cycle-consistency*, is that for an image pair (I_A, I_B), given unique descriptors in image I_A and I_B, any correctly predicted keypoint location in image I_B can in turn be used to predict the original point in image I_A, completing a *cycle* of correspondence predictions, see Fig. 1 for a visual overview. The model is able to learn by itself to detect valid correspondences, without relying on ground-truth correspondence annotations, by estimating uncertainties and scaling contribution of error terms accordingly. The only assumption is that the sampled training image pairs at least partially depict the same content with unique object instances. This still allows for random object arrangements, varying backgrounds, and scene conditions. Our loss is generally applicable, and can thus also be used with existing annotations, sim-to-real data generation, and other methods.

[1] Bosch Center for Artificial Intelligence, Renningen, Germany, firstname(s).lastname@de.bosch.com
[2] Institute of Neural Information Processing, Ulm University, Ulm, Germany, firstname.lastname@uni-ulm.de

II. Related Work

Keypoint detection from RGB images in robot learning and control has been extensively researched in recent years. *Sparse keypoint techniques* provide a discrete set of task-relevant keypoint locations in image plane or in camera coordinates. Early methods exploited autoencoders to reconstruct images with the bottleneck as keypoints [14] or keypoint distributions [15], and learned meaningful keypoints for solving ATARI games [16]. Following similar ideas, [17] proposes to learn object-category level keypoints from 3D models in a fully self-supervised way. More relevant to robot manipulation, the KeyPose method proposes to learn sparse keypoints for transparent objects using stereo RGB cameras [18]. Manuelli et al. showed that keypoint representation can be efficiently used to solve robot manipulation tasks [3]. Some of the most promising results with sparse keypoints for robot manipulation using human annotation and self-supervision was shown by Vecerik et al. in [19], [20].

Dense keypoint methods predict a single descriptor vector for every pixel of the RGB image. Florence et al. [1] proposed Dense Object Nets (DON) for fully autonomous object-centric dense descriptor learning. This work inspired a variety of follow up research, such as, applications for behavior cloning [4], learning model predictive controllers [5] and even rope manipulation [2]. Other works focused, e.g., on better generalization for multiple object classes [6] or class aware descriptors [21]. It has also been shown how to improve the original work [1] with alternative losses and training regimes [22], [23] and how to avoid costly preprocessing [23]. Recently, Yen-Chen et al. [12] applied NeRFs to learn DON from registered RGB scenes.

There has been another line of work focusing on learning dense descriptors from RGB images only, without the costly data collection and preprocessing. In the computer vision community image augmentations have been proposed to generate alternative views of the same image and use self-supervision for learning [7], [8]. [9] applied similar techniques to the robotics domain and showed that view-invariance of such descriptors are limited. SuperPoint is a pretrained method that uses a keypoint location heatmap and a dense descriptors head to provide robust keypoint locations [13]. Deekshith et al. showed that optical flow from video can also be used to learn dense descriptors [24]. It is also possible to implicitly train a dense descriptor model through autonomous grasp interactions [25], however, this requires a large amount of grasp interactions to do so. Another recent, but promising line of research investigates the usage of large pre-trained vision transformer models [26], [27] as provider of off-the-shelf features [28]. For example, Hadjivelichkov et al. [21] already demonstrated their usability to obtain one-shot affordance regions for robotic manipulation.

Our work builds on the idea of cycle-consistency, a well-established concept that is used, e.g, in CycleGAN [29] for image-to-image translation, for temporal correspondence learning in [30], or correspondence learning via 3D CAD models in [31]. WarpC [32] and PWarpC [33] utilize cycle-consistency to predict dense flows across two unpaired images and an augmented version that induces a known warp. Due to the close relation to our model, we explicitly discuss differences to these two models in more detail in Sec. III-D.

III. Method

In the following, we first outline our notation and preliminary concepts, followed by introducing CCL, see Fig. 1, and important considerations to be taken when using it.

A. Preliminaries

Let $\boldsymbol{I}_A, \boldsymbol{I}_B \in \mathbb{R}^{3\times H\times W}$ be two images, where H and W denote the height and width. We assume that there exists a non-empty subset of pixels in image $\boldsymbol{I}_A$ that have corresponding pixels in image $\boldsymbol{I}_B$. We refer to a single pixel in this subset as keypoint and denote it for $\boldsymbol{I}_A$ as $\boldsymbol{k}_A = (x_A, y_A)$ and the corresponding pixel on image $\boldsymbol{I}_B$ as $\boldsymbol{k}_B = (x_B, y_B)$.

View-Invariant Dense Descriptors. Let $f_\theta(\cdot)$ be a dense descriptor model that maps each pixel in an image $\boldsymbol{I}$ onto a D-dimensional latent space yielding a dense descriptor image $\boldsymbol{D} \in \mathbb{R}^{D\times H\times W}$. Let $\boldsymbol{D}_A, \boldsymbol{D}_B$ denote the descriptor images of $\boldsymbol{I}_A, \boldsymbol{I}_B$, and $\boldsymbol{d}_{\boldsymbol{k}_A} = \boldsymbol{D}_A[x_A, y_A] \in \mathbb{R}^D$ the associated descriptor of $\boldsymbol{k}_A$, and respectively $\boldsymbol{d}_{\boldsymbol{k}_B}$ for $\boldsymbol{k}_B$. The goal is to learn parameters θ such that $f_\theta(\cdot)$ will assign non-trivial, unique descriptors to two corresponding pixels, such that $\boldsymbol{d}_{\boldsymbol{k}_A} \approx \boldsymbol{d}_{\boldsymbol{k}_B}$, implying view-invariance, for example, with respect to scale, rotation, background, etc.

Probabilistic Keypoint Heatmaps. We can easily predict the location of a keypoint, given its descriptor, in a new image by finding the closest descriptor in latent space in $\boldsymbol{D}_B$ with respect to $\boldsymbol{d}_{\boldsymbol{k}_A}$. While this is sufficient for inference, one obtains the (x, y)-coordinates in a non-differentiable fashion, making it inadequate for training. Instead, we compute a distance heatmap $\boldsymbol{H}^{\boldsymbol{k}_A\to B}$ over $\boldsymbol{D}_B$ by taking the pairwise distances between $\boldsymbol{d}_{\boldsymbol{k}_A}$ and every descriptor of $\boldsymbol{D}_B$, such that

$$\boldsymbol{H}_{xy}^{\boldsymbol{k}_A\to B} = \Delta(\boldsymbol{d}_{\boldsymbol{k}_A}, \boldsymbol{D}_{B_{xy}}), \tag{1}$$

where Δ is some distance function, e.g., ℓ_2-norm, or derived from a similarity measure, such as cosine similarity. We assume cosine similarity and normalized descriptors in the following. We obtain a probability distribution $P(x, y \mid \boldsymbol{d}_{\boldsymbol{k}_A}, \boldsymbol{D}_B)$ by applying a temperature-scaled softmax function, such that

$$P(x, y \mid \boldsymbol{d}_{\boldsymbol{k}_A}, \boldsymbol{D}_B) = \frac{\exp(H_{xy}^{\boldsymbol{k}_A\to B}/\tau)}{\sum_{i=1}^{H}\sum_{j=1}^{W}\exp(H_{ij}^{\boldsymbol{k}_A\to B}/\tau)}, \tag{2}$$

where τ is the temperature. By interpreting the expected values of the marginal distributions as coordinates, we derive $\boldsymbol{k}_B^\star = (x^\star, y^\star)$ as

$$x^\star = \mu_x = \textstyle\sum_{i=1}^{H} i \cdot \sum_{j=1}^{W} P(i, j \mid \boldsymbol{d}_{\boldsymbol{k}_A}, \boldsymbol{D}_B), \tag{3}$$

$$y^\star = \mu_y = \textstyle\sum_{j=1}^{W} j \cdot \sum_{i=1}^{H} P(i, j \mid \boldsymbol{d}_{\boldsymbol{k}_A}, \boldsymbol{D}_B). \tag{4}$$

The variances σ_x^2, σ_y^2 follow naturally. Conveniently, this formulation is differentiable. If ground-truth annotations $\boldsymbol{k}_B^{(i)}$

exist, for example, in the case of pixelwise correspondences from 3D geometry, it is straight-forward to directly optimize the prediction error via the spatial expectation above, for example, with the loss function

$$\mathcal{L}_{\text{distributional},A\to B} = \sum_i^N \|\boldsymbol{k}_B^{\star\,(i)} - \boldsymbol{k}_B^{(i)}\|_2, \tag{5}$$

where N is the number of sampled keypoints in $\boldsymbol{I}_A$. The loss $\mathcal{L}_{\text{distributional}}$ was previously introduced in a more general form in [22]. A version relying on KL-divergence has also been proposed, see e.g., [19].

B. Cycle-Correspondence Loss

We now extend the above concept into a fully self-supervised training regime, when no ground-truth annotation $\boldsymbol{k}_B^{(i)}$ is given and we can not directly define an error to optimize as in Eq. (5). For sake of explanation, we temporarily assume the constraint that any $\boldsymbol{k}_A^{(i)}$ sampled has exactly one corresponding pixel $\boldsymbol{k}_B^{(i)}$ in $\boldsymbol{I}_B$, albeit unknown. Given this assumption, we know that if the prediction $\boldsymbol{k}_B^{\star\,(i)}$ for $\boldsymbol{I}_A \to \boldsymbol{I}_B$ is correct, then the associated descriptor $\boldsymbol{d}_{\boldsymbol{k}_B^{\star\,(i)}}$ should yield a prediction $\boldsymbol{k}_A^{\star\,(i)}$ from $\boldsymbol{I}_B \to \boldsymbol{I}_A$, such that $\boldsymbol{k}_A^{(i)} \equiv \boldsymbol{k}_A^{\star\,(i)}$ holds. This effectively completes a cycle of correspondence matching. Since $\boldsymbol{k}_A^{(i)}$ is known, we can directly measure the prediction error, allowing us to define an error term for keypoint i as

$$l_i = \|\boldsymbol{k}_A^{\star\,(i)} - \boldsymbol{k}_A^{(i)}\|_2. \tag{6}$$

See Fig. 1 for a visualization.

C. Implementation

Although the loss is conceptually easy to formulate we now outline practical considerations that need to be taken into account for a successful implementation.

1) Prevention of Short-Cut Learning: To ensure the network will not ignore $\boldsymbol{I}_B$ and short-cut learn an identity mapping, we generate a copy $\boldsymbol{I}_{\hat{A}}$ of input image $\boldsymbol{I}_A$ and augment each separately. As common in self-supervised training [1], [9], [11], [23], [34], [35], we apply a variety of augmentations to our input images. In particular, we follow the selection presented in [23] by using affine transformations (rotation, scale), perspective distortion, and color jitter - the latter primarily for brightness and contrast augmentations. We also know $\boldsymbol{k}_{\hat{A}}^{(i)}$, that is the location of $\boldsymbol{k}_A^{(i)}$ in $\boldsymbol{I}_{\hat{A}}$, as the applied mapping is known, allowing us to redefine l_i in Eq. (6) as

$$l_i = \|\boldsymbol{k}_{\hat{A}}^{\star\,(i)} - \boldsymbol{k}_{\hat{A}}^{(i)}\|_2. \tag{7}$$

2) Expected Descriptor and Keypoint Prediction: In order to obtain $\boldsymbol{d}_{\boldsymbol{k}_B^{\star\,(i)}}$ in a differentiable fashion, we extend the concept of the spatial expectation, see Eq. (3, 4), to compute the *expected descriptor*, that is

$$\bar{\boldsymbol{d}}_{\boldsymbol{k}_B} = \sum_{i=1}^H \sum_{j=1}^W \boldsymbol{D}_{B_{ij}} \cdot P(i,j \mid \boldsymbol{d}_{\boldsymbol{k}_A}, \boldsymbol{D}_B). \tag{8}$$

If the descriptors are normalized, one should additionally normalize $\bar{\boldsymbol{d}}_{\boldsymbol{k}_B}$, which we implicitly assume to be the case. This allows us to define $P(x,y \mid \bar{\boldsymbol{d}}_{\boldsymbol{k}_B}, \boldsymbol{D}_{\hat{A}})$ via $\bar{\boldsymbol{d}}_{\boldsymbol{k}_B}$ and determine $\boldsymbol{k}_{\hat{A}}^{\star\,(i)}$ using the spatial expectation.

3) Handling Keypoints Without Correspondences: By training on unordered RGB images, objects may or may not be present, backgrounds change, or occlusion occurs. Hence, we must now relax the above assumption that every $\boldsymbol{k}_A^{(i)}$ has a correspondence in $\boldsymbol{I}_B$. Clearly, l_i for some $\boldsymbol{k}_A^{(i)}$ without correspondence violates the underlying assumption of the cycle-consistency and calculated gradients might be completely counter-productive. At the same time, as $\bar{\boldsymbol{d}}_{\boldsymbol{k}_B}$ is essentially a weighted sum of those descriptors in $\boldsymbol{I}_B$ most similar to $\boldsymbol{d}_{\boldsymbol{k}_A}$, the model could in practice still find a path from $\boldsymbol{k}_A^{(i)}$ to $\boldsymbol{k}_{\hat{A}}^{\star\,(i)}$, even without a correspondence. This short-cut learning should be prevented.

We mitigate these issues by exploiting the previously determined probability distributions through two distinct mechanisms. For both we first compute the summed variances $X_i = \chi_{\hat{A},i} + \chi_{B,i}$, with $\chi_{\cdot,i} = \sigma_{x,i}^2 + \sigma_{y,i}^2$, for the i-th keypoint predictions over images $\boldsymbol{I}_B$ and $\boldsymbol{I}_{\hat{A}}$. Intuitively, we assume that χ_i is small, if a unique correspondence exists and the model is confident. If no correspondence exists, or the model is not confident, χ_i should be large. See Fig. 2 for a visualization of this emergent behaviour in our CCL trained model. For the first method we determine the q-quantile, e.g., $q = 35\%$, over the N summed variances $\{X_i\}_{i=0}^N$. This gives us the $q\%$ most reliably detected points and we discard all other points from optimization. For the second method, we modify Eq. (7), by scaling the contribution of each error with respect to the associated uncertainty, giving us the final loss

$$\mathcal{L}_{\text{cycle}} = \sum_i^N \frac{1}{1+X_i} \|\boldsymbol{k}_{\hat{A}}^{\star\,(i)} - \boldsymbol{k}_{\hat{A}}^{(i)}\|_2, \tag{9}$$

where we add 1 in the denominator to prevent the term from growing prohibitively large if some X_i is smaller than 1. Importantly, we detach the calculated variances from the computational graph and do not back-propagate gradients, else the model will simply learn to make predictions with low confidence instead of solving the prediction task.

4) Pretraining & Model Initialization: Although the model can be successfully trained from scratch, one can efficiently initialize by first performing a self-supervised pre-training akin to [9]. Here we directly match keypoints between $\boldsymbol{I}_A$ and $\boldsymbol{I}_{\hat{A}}$ by defining $P(x,y \mid \boldsymbol{d}_{\boldsymbol{k}_A}, \boldsymbol{D}_{\hat{A}})$ and re-utilizing the distributional loss, such that

$$\mathcal{L}_{\text{identical}} = \mathcal{L}_{\text{distributional},A\to\hat{A}}, \tag{10}$$

where descriptors are learned from correspondences generated synthetically via two augmented views, and each sampled $\boldsymbol{k}_A^{(i)}$ is guaranteed to be valid. A combined loss $\mathcal{L} = \mathcal{L}_{\text{cycle}} + \lambda\mathcal{L}_{\text{identical}}$ is also explored in the experiments.

D. Relation to WarpC

We note that *WarpC* [32] and its probabilistic extension *PWarpC* [33] both utilize the notion of cycle-consistency in the context of dense matching. Our work shares the same abstract concept of optimizing across unpaired images through completing some cycle, an idea also popularized in other contexts, e.g., in CycleGAN [29]. However, critical aspects differentiate our approaches. Firstly, our optimization

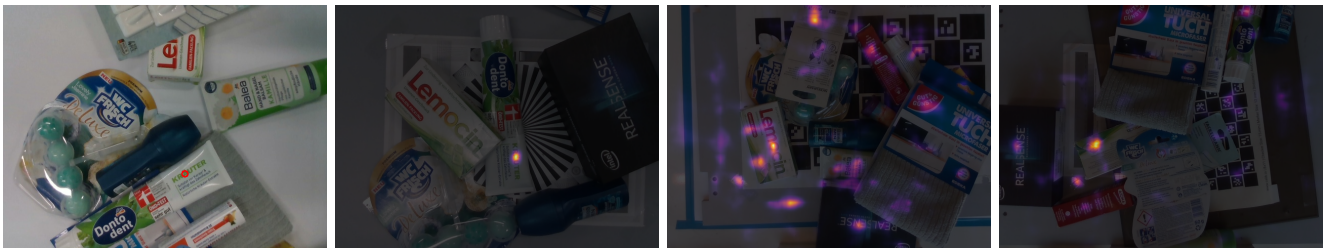

Fig. 2: Visualization of the matching uncertainty. The red circle in the left most image marks the sampled keypoint. The following test images are superimposed with the predicted distribution as heatmap. If a correspondence exists (second from left), the mass of the distribution is well localized. If no correspondence exists (middle right and right most image), the mass is spread over various areas that are the most similar in descriptor space. Viewed best in color.

target is defined differently. (P)WarpC implements the cycle concept by densely estimating the known ground-truth warp W between $\boldsymbol{I}_A$ and $\boldsymbol{I}_{\hat{A}}$, induced by augmentations, by *independently* predicting flow F_{AB} between $\boldsymbol{I}_A$ and $\boldsymbol{I}_B$ and $F_{B\hat{A}}$ between $\boldsymbol{I}_B$ and $\boldsymbol{I}_{\hat{A}}$, such that $W \approx F_{AB} + F_{B\hat{A}}$. In contrast, CCL operates on a small subset of pixels. For each we probabilistically estimate a descriptor over $\boldsymbol{I}_B$, which is directly used to infer a prediction over $\boldsymbol{I}_{\hat{A}}$, making the prediction over $\boldsymbol{I}_{\hat{A}}$ dependent on the prediction over $\boldsymbol{I}_B$.

Secondly, we differ from WarpC and PWarpC when it comes to discarding unmatchable pixels from optimization. WarpC uses the current error between predicted flows and W to compute a visibilty mask (cf. Eq. 9, [32]). PWarpC instead uses predicted confidence values to discard the $q\%$ most unrealiable points (cf. Eq. 9, [33]) like our first method. In our work, we additionally scale the individual contribution of remaining error terms based on their associated confidence. As we show in Sec.IV-F, this considerably improves our models performance, while making the exact choice of q less sensitive.

IV. Experiments

We now discuss the methods and data of the evaluation setup. This is followed by experimental results for the standard keypoint prediction accuracy task. We then present a 6D grasp pose prediction experiment using a parallel gripper and conclude with an ablation study.

A. Method Comparison

We compare our loss primarily against task-agnostic methods for obtaining dense visual features. We do, however, not review and compare against methodologies that focus on the data-generation side, such as sim-to-real DONs [36] or NeRF-supervised DONs [12], nor methods that utilize already trained dense descriptor networks, such as [35], as these can be combined with the presented CCL. Table I summarizes all methods alongside their respective evaluation results. The column *(weakly) supervised* indicates which method requires, e.g., pixel-level masks or class labels.

The first set of methods we compare against are DON-like [1] models. (i) a model trained using augmented versions of a single image and extraction of synthetic correspondences (Identical View) according to [9] that utilizes only unordered RGB images like our method. (ii) maskless multi-object scenes (MO-maskless) following [23]. This is a specialized version of vanilla DONs utilizing ground-truth geometric correspondences extracted from RGBD sequences. Finally, a fully supervised baseline: (iii) DON trained using synthetically composed collages of real image crops of objects (MO Collage Scenes) from many camera views, allowing for construction of object occlusions and other advanced compositions. This method uses both object-level masks and ground-truth geometric correspondences based on 3D scene reconstructions. This yields a strong baseline setup to test impact of different levels of data complexity. We trained all the variants using the distributional loss proposed in [22].

We also compare against DINOv2 [27], a recent large-scale unsupervised trained vision approach. We extract dense features using the authors provided code from last intermediate layer as it provided the best results out-of-the-box.

As closely related work, we also compare against WarpC [32] and PWarpC [33], both however specialized on dense matching via flow prediction. These models are intended for dense geometric and semantic matching and seem to work best on images with large overlap or a single central object.

In addition to the vanilla version of CCL, we also train a variant in combination with (Identical View), which shares the same data requirements. Here we simply use $(\boldsymbol{I}_A, \boldsymbol{I}_{\hat{A}})$ as input to $\mathcal{L}_{\text{identical}}$ (Eq. 10), while CCL is trained as previously. We combine both losses as $\mathcal{L} = \mathcal{L}_{\text{cycle}} + \lambda\mathcal{L}_{\text{identical}}$, where we found $\lambda = 0.1$ to perform well.

B. Datasets

We collected data of 12 objects in total to train and evaluate on, including challenging objects with transparent plastic, reflective, or black surfaces. We provide method-specific training datasets described below, however, each method is compared against the same test dataset, which is described in more detail in Sec. IV-D.

3D Reconstructed/RGBD-Datasets: We followed the same protocol as in [1] to collect RGBD-sequences using a wrist-mounted camera on a robot arm. By means of 3D reconstruction, masks and geometric correspondences can be extracted. This collection consists of 20 RGBD-sequences for training and five for validation. Each sequence has around 480 frames. This amount of sequences ensures that each object is seen from all sides and overall enough variety of scene and camera configurations that, e.g., also

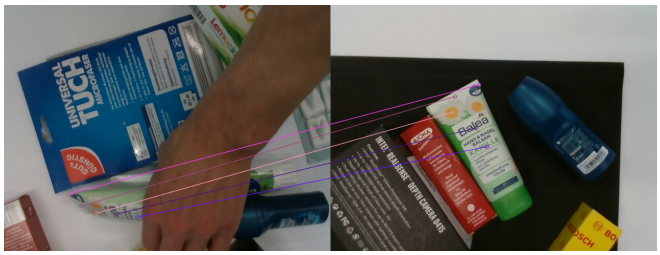

Fig. 3: Example of hand-annotated, cross-scene keypoint matching test image pair. Occlusion, background changes, strong view-point or object pose changes are induced.

includes occlusions, is captured. We trained *MO-maskless*, *MO Collage Scenes*, and WarpC on this dataset. Although not relying on the annotations, WarpC would fail to train on the dataset discussed next.

Unordered RGB: For training CCL an unordered collection of images is sufficient, for example, collected from a single, top-down view of a fixed camera. We recorded 513 images, all from the same camera view, but altering the object arrangement in each frame. To simplify this process, we obtained these by recording a continuous video stream, where an operator shuffles the objects and removes his hands from the camera view every other frame for a brief moment. Duplicate static frames and blurry frames, e.g., where the operator hands are moving, can be trivially removed using common image processing tools. We note that many of the frames, however, still contain the operator's hands, which we found did not hamper training success. This way a complete training set was recorded in 5 minutes including processing. This strongly contrasts the geometric datasets required for Dense Object Nets, which can take hours as they require multiple recordings per object, each taking several minutes, followed by 3D reconstructions and potentially manual mask generation. We also trained PWarpC on this dataset as it yielded better results than on the above.

C. Training Details

Similar to prior work [1], [9], [23], [37], we use a pretrained ResNet [38] with an output stride of 8 and upsampling to match the input resolution, specifically a ResNet-50. All input images are ImageNet normalized using $\mu = [0.485, 0.456, 0.406]$ and $\sigma = [0.229, 0.224, 0.225]$. To increase efficiency we train using 16-bit precision using PyTorch [39]. For the CCL trained model, we use the upsampled descriptor images only for evaluation, but for training the low resolution descriptor images are used. This yields a descriptor image $\hat{\boldsymbol{D}} \in \mathbb{R}^{D \times \frac{H}{8} \times \frac{W}{8}}$, making the pairwise distance calculation very efficient. We train with a batch size of 4 images and 2000 batches per epoch. We sampled $N = 500$ keypoint candidates per image pair or triplet. The embedding size has been set to $D = 64$, following [9], [35], and we use $\tau = 0.03$, chosen by grid search. Main results are reported for $q = 35\%$. We use AdamW [40] as optimizer with a fixed learning rate $lr = 3\text{e}{-5}$. Models trained with CCL have been initialized with the final checkpoint of the model obtained using identical view training [9].

D. Keypoint Prediction Accuracy

Although the descriptors are task-agnostic, we follow a range of prior work [1], [12], [23], [35], [36], [41] and evaluate how well keypoints can be matched across image pairs. However, unlike some of aforementioned works we do not test using 3D reconstructed RGBD sequences for ground-truth annotation, as it limits testing the object poses and scene configurations of image pairs from static scenes.

Instead, we compiled a test dataset of 80 images, each depicting different scenes and object placements, and hand-annotated keypoints for each image and object. In total 9124 image pairs, each featuring an object annotation consisting of around 10 keypoints on average. Half the keypoints are located close to or on the object boundaries, the other half *inside* the object. This requires models to be robust to background changes and not calculate descriptors based on the background or close-by located objects. Each image exhibits a different subset of objects, background changes, occlusion, or other scene composition factors, such as lighting conditions, see Fig. 3 for an example. This ensures that methods are robustly tested for their ability to generate descriptors that are view- and scene-invariant.

The results are compiled in Table I. We find that *MO Collage Scenes* out-performs all methods, while relying on pixel-level masks and ground-truth geometric correspondences and thus having the highest data complexity. *CCL* and *MO-maskless* perform comparably, with the latter scoring higher on PCK@$\{3, 5, 10\}$ and CCL on AUC and normalized mean pixel error. The combination *CCL+Identical View* improves the results even further.

WarpC and *PWarpC* seem to struggle on our data. When tested on image pairs from the same scene, that is same background and object arrangement but varied camera poses, they perform well. However, when large parts of the images can not be matched and objects are subject to strong pose variations, as in our test set, the dense flow prediction is breaking down. The pretrained *DINOv2* model is not able to make accurate predictions under strong camera perspectives. Although we found it can re-identify objects, it does not precisely locate positions. This is partially also due to the large down-sampling factor of 14.

In summary, our proposed method outperforms all methods that do not rely on ground-truth geometric correspondences and approaches performance of the fully supervised MO Collage Scenes, despite being trained on only a small, but highly varied, unlabeled RGB-only dataset.

E. Oriented Grasping Experiment

We compared the best performing methods on a 6D grasp pose prediction task using a parallel gripper as done in related work [12], [23]. To fairly compare the methods, we first recorded a single top-down view of each test object on a plain white background. We define an axis along which we want to grasp by manually annotating two pixels per object and extracting the respective descriptors using each

TABLE I: Evaluation results for keypoint prediction. Methods requiring masks or, e.g., class labels (supervision) are marked. Metrics are percentage of correct keypoints (PCK@k), area-under-curve for PCK@k for $k \in [1..50]$, and normalized mean pixel error. Standard deviation is denoted by the preceding $\pm$ symbol. Arrows ↑ and ↓ indicate if higher or lower is better.

Method	(Weakly) Supervised	PCK@ 3 ↑	PCK@ 5 ↑	PCK@ 10 ↑	PCK@ 25 ↑	PCK@ 50 ↑	AUC@ [1..50] ↑	Norm. Mean Pixel Error ↓
DINOv2 (b/14) pretrained [27]	–	.019 ± .135	.05 ± .217	.148 ± .356	.368 ± .482	.564 ± .496	.151 ± .103	.110 ± .137
WarpC [32]	–	.04 ± .196	.059 ± .236	.082 ± .274	.121 ± .326	.173 ± .378	.043 ± .109	.247 ± .173
Identical View [9]	–	.042 ± .202	.100 ± .299	.236 ± .425	.442 ± .497	.594 ± .491	.177 ± .117	.109 ± .145
CCL (ours)	–	.100 ± .300	.222 ± .416	.438 ± .496	.664 ± .472	.775 ± .418	.266 ± .133	.070 ± .129
CCL (ours) + Identical View [9]	–	**.124 ± .329**	**.261 ± .439**	**.481 ± .500**	**.690 ± .462**	**.793 ± .405**	**.277 ± .137**	**.064 ± .122**
PWarpC [32]	✓	.004 ± .067	.013 ± .113	.045 ± .208	.165 ± .371	.310 ± .463	.066 ± .089	.185 ± .165
MO-maskless [23] [22]	✓	.130 ± .337	.273 ± .445	.476 ± .499	.644 ± .479	.741 ± .438	.264 ± .133	.071 ± .124
MO Collage Scenes [1], [22]	✓	**.140 ± .347**	**.289 ± .453**	**.516 ± .500**	**.700 ± .458**	**.799 ± .401**	**.286 ± .130**	**.056 ± .110**

TABLE II: **Grasping Experiment Success Rate.**

Method / Loss	Success Rate
Identical View [9]	41.4%
MO (maskless) [23]	44.8%
MO Collage Scenes [1], [22]	77.6%
CCL (ours)	70.7%

model. We tested on six out of 12 training objects, as some would require a suction gripper. We defined two alternative axis definitions per object, one with keypoints close to the object edges and one with locations further inside. The latter is beneficial for methods trained without masks, like [23], where descriptors are stable inside objects, but not close to the edges. We test on cluttered scenes, where objects are placed on a heap with frequent background changes, including reflective surfaces and materials of similar color as the target object. The current target object is always visible and graspable, but its placement might still induce strong perspective distortions w.r.t. the annotation image. Each grasp configuration is tested on five different scene configurations, for a total of 30 grasps per network. All networks are tested on the same scenes, which we accurately restore after each grasp attempt.

The results are compiled in Table II. Unsurprisingly, the model trained on collage scenes has the most successful grasp attempts. This model behaves less sensitive to changes in background due to strong background randomization and modeling of occlusion during training. In comparison, the models (Identical View) and MO-maskless struggle, as they tend to integrate information from the background being trained on image pairs, where both images are from the same scene, as can be visualized by VisualBackProb [42]. In contrast, CCL, which is trained exclusively on RGB images showing different views, appears to learn more robustly encoded view and scene-invariant features, similar to the network trained on synthetically generated views.

F. Ablation: Impact of Quantile Drop and Variance Scaling

We proposed two mechanisms to handle sampled keypoints candidates without correspondence in Sec. III-C.3. To isolate their respective impact, different settings of q were evaluated, with and without variance scaling. See Fig. 4 for

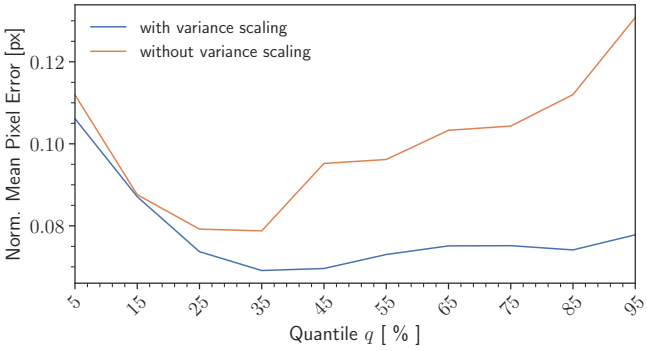

Fig. 4: Evaluation of prediction accuracy for different quantile q and variance scaling.

results. Clearly, having a smaller quantile q helps prune bad samples efficiently. However, particularly variance scaling leads to considerably better result overall while making the choice of the quantile q much less sensitive.

V. Limitations

Despite the flexibility of the self-supervised formulation, some limitations need to be considered. Firstly, the loss trains most effectively if both views have many valid pixel correspondences. Although we demonstrated variance scaling and using a lower quantile threshold as remedy, we recommend to record data, e.g., as proposed. Secondly, good performance of the loss will not necessarily imply good performance on downstream tasks, as our self-supervised loss is task agnostic. Hence, validation directly on a downstream task or using a small labeled dataset for validation can prove helpful.

VI. Conclusions

We presented a novel, self-supervised loss that allows to train complex dense visual feature extractors for object understanding in robotic manipulation using unordered collection of RGB images. We effectively combine the benefits of pixel correspondence via alternative views and a simple data collection pipeline. While there is still room for improvement, we could show highly competitive performance w.r.t. methods trained on registered RGBD scenes. We plan to explore more advanced architectures, e.g., vision transformers, and methods for match cost calculation using self-attention in future work.

REFERENCES

[1] 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.

[2] 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*, pp. 9411–9418, 2020.

[3] L. Manuelli, W. Gao, P. R. Florence, and R. Tedrake, "KPAM: keypoint affordances for category-level robotic manipulation," in *Robotics Research - The 19th International Symposium ISRR 2019, Hanoi, Vietnam, October 6-10, 2019*, ser. Springer Proceedings in Advanced Robotics, vol. 20. Springer, 2019, pp. 132–157. [Online]. Available: https://doi.org/10.1007/978-3-030-95459-8_9

[4] 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, 2020.

[5] 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.

[6] 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*, vol. 6, no. 2, pp. 4109–4116, 2021.

[7] J. Thewlis, H. Bilen, and A. Vedaldi, "Unsupervised learning of object frames by dense equivariant image labelling," *Advances in Neural Information Processing Systems*, vol. 2017-Decem, no. figure 1, pp. 845–856, 2017.

[8] 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*, pp. 3637–3645, 2018.

[9] C. Graf, D. B. Adrian, J. Weil, M. Gabriel, P. Schillinger, M. Spies, H. Neumann, and A. G. Kupcsik, "Learning dense visual descriptors using image augmentations for robot manipulation tasks," in *Conference on Robot Learning*. PMLR, 2023, pp. 871–880.

[10] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, 2006, pp. 1735–1742.

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

[12] 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.

[13] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 337–33 712.

[14] T. Jakab, A. Gupta, H. Bilen, and A. Vedaldi, "Unsupervised learning of object landmarks through conditional image generation," *Advances in Neural Information Processing Systems*, no. NeurIPS, pp. 4016–4027, 2018.

[15] Y. Zhang, Y. Guo, Y. Jin, Y. Luo, Z. He, and H. Lee, "Unsupervised Discovery of Object Landmarks as Structural Representations," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2694–2703, 2018.

[16] T. 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, no. NeurIPS, 2019.

[17] S. Suwajanakorn, N. Snavely, J. Tompson, and M. Norouzi, "Discovery of latent 3D keypoints via end-to-end geometric reasoning," *Advances in Neural Information Processing Systems*, vol. 2018-Decem, no. 1, pp. 2059–2070, 2018.

[18] X. Liu, R. Jonschkowski, A. Angelova, and K. Konolige, "KeyPose: Multi-View 3D Labeling and Keypoint Estimation for Transparent Objects," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 11 599–11 607, 2020.

[19] 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.

[20] M. Vecerík, J. Kay, R. Hadsell, L. Agapito, and J. Scholz, "Few-shot keypoint detection as task adaptation via latent embeddings," in *2022 International Conference on Robotics and Automation, ICRA 2022, Philadelphia, PA, USA, May 23-27, 2022*. IEEE, 2022, pp. 1251–1257. [Online]. Available: https://doi.org/10.1109/ICRA46639.2022.9812209

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

[22] P. R. Florence, "Dense visual learning for robot manipulation," Ph.D. dissertation, Massachusetts Institute of Technology, 2020.

[23] 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.

[24] 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*, vol. 5, pp. 444–451, 2020.

[25] E. Jang, C. Devin, V. Vanhoucke, and S. Levine, "Grasp2vec: Learning object representations from self-supervised grasping," in *2nd Annual Conference on Robot Learning, CoRL 2018, Zürich, Switzerland, 29-31 October 2018, Proceedings*, ser. Proceedings of Machine Learning Research, vol. 87. PMLR, 2018, pp. 99–112. [Online]. Available: http://proceedings.mlr.press/v87/jang18a.html

[26] 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 International Conference on Computer Vision (ICCV)*, 2021.

[27] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P. Huang, S. Li, I. Misra, M. G. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jégou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision," *CoRR*, vol. abs/2304.07193, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.07193

[28] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel, "Deep vit features as dense visual descriptors," *ECCVW What is Motion For?*, 2022.

[29] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 2242–2251. [Online]. Available: https://doi.org/10.1109/ICCV.2017.244

[30] X. Wang, A. Jabri, and A. A. Efros, "Learning correspondence from the cycle-consistency of time," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 2566–2576.

[31] T. Zhou, P. Krähenbühl, M. Aubry, Q. Huang, and A. A. Efros, "Learning dense correspondence via 3d-guided cycle consistency," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 117–126. [Online]. Available: https://doi.org/10.1109/CVPR.2016.20

[32] P. Truong, M. Danelljan, F. Yu, and L. V. Gool, "Warp consistency for unsupervised learning of dense correspondences," in *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*. IEEE, 2021, pp. 10 326–10 336. [Online]. Available: https://doi.org/10.1109/ICCV48922.2021.01018

[33] ——, "Probabilistic warp consistency for weakly-supervised semantic correspondences," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, 2022, pp. 8698–8708. [Online]. Available: https://doi.org/10.1109/CVPR52688.2022.00851

[34] 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.

[35] J. O. von Hartz, E. Chisari, T. Welschehold, W. Burgard, J. Boedecker, and A. Valada, "The treachery of images: Bayesian scene keypoints

for deep policy learning in robotic manipulation," *arXiv preprint arXiv:2305.04718*, 2023.

[36] H.-G. Cao, W. Zeng, and I.-C. Wu, "Learning sim-to-real dense object descriptors for robotic manipulation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 9501–9507.

[37] D. Hadjivelichkov and D. Kanoulas, "Fully self-supervised class awareness in dense object descriptors," in *Conference on Robot Learning, 8-11 November 2021, London, UK*, ser. Proceedings of Machine Learning Research, A. Faust, D. Hsu, and G. Neumann, Eds., vol. 164. PMLR, 2021, pp. 1522–1531. [Online]. Available: https://proceedings.mlr.press/v164/hadjivelichkov22a.html

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[39] 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*, pp. 8024–8035, 2019.

[40] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[41] 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*, vol. 35, no. 7, 2021, pp. 6093–6100.

[42] M. Bojarski, A. Choromanska, K. Choromanski, B. Firner, L. J. Ackel, U. Muller, P. Yeres, and K. Zieba, "Visualbackprop: Efficient visualization of cnns for autonomous driving," in *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*. IEEE, 2018, pp. 1–8. [Online]. Available: https://doi.org/10.1109/ICRA.2018.8461053