Unsupervised Domain Adaptation for 3D Keypoint Estimation via View Consistency

Xingyi Zhou[1], Arjun Karpur[1], Chuang Gan[2], Linjie Luo[3], and Qixing Huang[1(✉)]

[1] The University of Texas at Austin, Austin, USA
{zhouxy,akarpur,huangqx}@cs.utexas.edu
[2] MIT-IBM Watson AI Lab, Cambridge, USA
ganchuang1990@gmail.com
[3] Snap Inc., Los Angeles, USA
linjie.luo@snap.com

Abstract. In this paper, we introduce a novel unsupervised domain adaptation technique for the task of 3D keypoint prediction from a single depth scan or image. Our key idea is to utilize the fact that predictions from different views of the same or similar objects should be consistent with each other. Such view consistency can provide effective regularization for keypoint prediction on unlabeled instances. In addition, we introduce a geometric alignment term to regularize predictions in the target domain. The resulting loss function can be effectively optimized via alternating minimization. We demonstrate the effectiveness of our approach on real datasets and present experimental results showing that our approach is superior to state-of-the-art general-purpose domain adaptation techniques.

Keywords: 3D keypoint estimation · Multi-view consistency
Domain adaptation · Unsupervised learning

1 Introduction

A new era has arrived with the proliferation of depth-equipped sensors in all kinds of form factors, ranging from wearables and mobile phones to on-vehicle scanners. This ever-increasing amount of depth scans is a valuable resource that remains largely untapped, however, due to a lack of techniques capable of efficiently processing, representing, and understanding them.

3D keypoints, which can be inferred from depth scans, are a compact yet semantically rich representation of 3D objects that have proven effective for many

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-01258-8_9) contains supplementary material, which is available to authorized users.

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11216, pp. 141–157, 2018.
https://doi.org/10.1007/978-3-030-01258-8_9

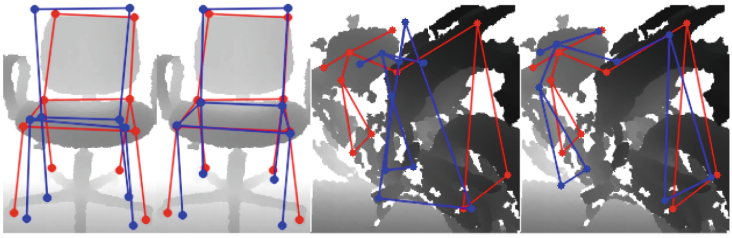

Fig. 1. Our approach improves 3D keypoint prediction results from single depth scans of the Redwood dataset [3]. For each pair: **(Left)** without domain adaptation, the pre-trained keypoint predictor from simulated examples failed to predict accurate 3D keypoints (blue). **(Right)** 3D keypoint predictions (blue) after domain adaptation are significantly improved. Note that the ground-truth keypoints are shown in red for comparison. (Color figure online)

tasks, including reconstruction [10], object segmentation and recognition [17], as well as pose estimation [33]. Despite the wide availability of depth scans of various object categories [3], there is a lack of corresponding 3D keypoint annotations, which are necessary to train reliable keypoint predictors in a supervised approach. This is partially due to the fact that depth scans are inherently partial views of the underlying objects, making it difficult to annotate the object parts occluded from view. One could automate the annotation process by leveraging the "fused" models created using the depth scans, but most depth-fusion methods are susceptible to scanning noise and cascading errors when depth scans are captured at scale [3] (Fig. 1).

In this paper, our goal is to predict 3D keypoints of an underlying object from a single raw depth scan. To train a reliable 3D keypoint predictor, we generate a large dataset of simulated depth scans using large-scale 3D model repositories such as ShapeNet [2] and ModelNet [38]. The 3D keypoint annotations on the 3D models from these repositories can naturally carry over to the simulated depth scans for effective supervised training. A large gap exists, however, between the simulated and real depth scan domains. Particularly, 3D models from repositories are generally designed with interactive tools, inevitably resulting in inaccurate geometries with varying scales. Furthermore, the real depth scans contain noticeable measurement noise and background objects, and the class distributions of 3D models from the repositories and those from real depth scans may be quite different.

To close the gap between the source domain of simulated depth scans and the target domain of real depth scans, we introduce a novel approach for unsupervised domain adaptation of 3D keypoint prediction. Our approach is motivated by the special spatial properties of the 3D keypoints and the relationship between the keypoint distributions of the source and target domains.

First, keypoint predictions from different views of the same 3D model should be consistent with each other up to a pose transformation. This allows us to formulate a *view-consistency* regularization to propagate a good prediction, e.g.

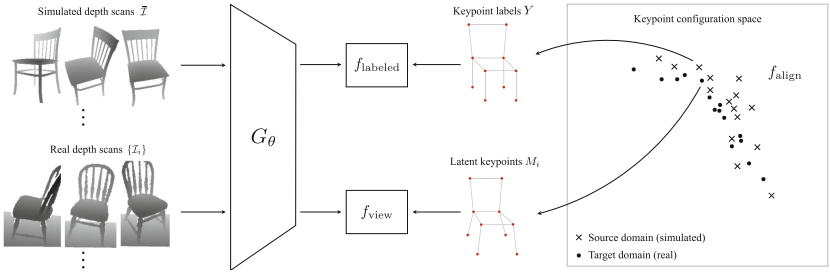

Fig. 2. Approach overview. We train an end-to-end 3D keypoint prediction network G_θ from labeled simulated depth scans $\bar{\mathcal{I}}$ of 3D models and unlabeled and unaligned real depth scans $\{\mathcal{I}_i\}$ of real world objects.

from a well-posed view where the prediction is more accurately adapted, to a challenging view with less accurate adaptation. To this end, we introduce a latent keypoint configuration to fuse the keypoint predictions from different views of the same object. Additionally, we introduce a pose-invariant metric to compare the keypoint predictions, which allows us to leverage depth scans without camera pose calibration for training (Fig. 2).

Second, despite the distinctive differences between the source and target domains, their 3D keypoint distributions are highly correlated. However, naively aligning the 3D keypoint distributions between the two domains is sub-optimal since the occurrences of the same type of objects differ. To address this challenge, we propose a *geometric alignment* regularization that is insensitive to varying densities of the objects in order to align the keypoint distributions of the two domains. We make use of the target domain's latent keypoint configurations from view consistency regularization to compute the geometric alignment with the source domain. Note that since possible keypoint configurations lie on a manifold with much lower dimension over the ambient space, the geometric alignment can provide effective regularization.

Our final formulation combines a standard supervised loss on the source domain with the two unsupervised regularization losses on view-consistency and geometric alignment. Our formulation can be easily optimized via alternating minimization and admits a simple strategy for variable initialization.

We evaluate the proposed approach on unsupervised domain adaptation from ModelNet [38] to rendered depth scans from the synthesized ShapeNet [2] 3D model dataset, and to real depth scans from the Redwood Object Scans [3] and 3DCNN Depth Scans [22] datasets. Experimental results demonstrate that our approach can effectively reduce the domain gap between the online 3D model repositories and the real depth scans with background noise. Our approach is significantly better than without domain adaptation and is superior to general-purpose domain adaptation techniques such as ADDA [35]. We also provide ablation studies to justify the design choice of each component of our approach. Code is available at https://github.com/xingyizhou/3DKeypoints-DA.

2 Related Works

Keypoint Detection. Keypoint detection from a single RGB or RGB-D image is a fundamental task in computer vision. We refer to [7,19,25,45] for some recent advances on this topic. While most techniques focus on developing novel neural network architectures for this task, fewer works focus on addressing the issue of domain shifts between the training data and testing data, e.g., the setting described in this paper. In [45], the authors introduce a domain adaptation technique for 3D human pose estimation in the wild. Additionally for human pose estimation, [7] proposes to align the source and target label distributions using a GAN loss. We opt to use an alternate metric that offers more flexibility in addressing domain shifts. Similarly to our method, [25] also leverages the consistency across multiple views to boost the supervision on the target domain. However, the output of this approach is computed directly from the initial predictions from the source domain. In contrast, our approach only uses the initial predictions to initialize final predictions. Moreover, we utilize a latent configuration for synchronizing the predictions from multiple views, which avoids performing pair-wise analysis.

Multi-view Supervision. RGB and RGB-D video sequences essentially consist of different views of the same underlying 3D environment. In the literature, people have utilized such weak supervision for various tasks such as 3D reconstruction, novel view synthesis and 2D keypoint prediction, e.g.,[15,25,34,40,43]. Our work differs from most works in the sense that we do not assume that relative poses between cameras are known. Instead, we introduce a pose invariant metric to compare keypoint configurations. Concurrent to our work, Helge et al. [23] also introduced a similar viewpoint consistency term for un-supervised 3D human pose estimation. However, the multi-view data for articulated object is still hard to obtain. On the contrary, we use viewpoint consistency for rigid objects, where the views are free from videos.

Supervision from Big 3D Data. Thanks the availability of annotated big 3D data such as ModelNet [38] and ShapeNet [2], people have leveraged synthetic data generated from 3D models for various tasks, including image classification [38], object recognition [21,26,27], semantic segmentation [42], object reconstruction [4,28,32], pose estimation [29] and novel-view synthesis [30,44]. The fundamental challenge of these approaches is that there are domain shifts between synthetic data and real RGB or RGB-D images. Most existing works focus on improving the simulation process to close this gap. In contrast, we focus on developing an unsupervised loss for domain adaptation.

Domain Adaptation. Domain adaptation [1,8,9,12,16,18,20,24,36,39,41] for various visual recognition tasks is an active research area in computer vision, and our problem falls into the general category of domain adaptation. It is beyond the

scope of this paper to provide a comprehensive review of the literature, however we refer to a recent survey [6] on this topic. A common strategy for unsupervised domain adaptation is to align the output distributions between source and target domains, e.g., either through explicit domain-wise maps or through use of a GAN. In contrast, our regularizations are tailored for the particular problem we consider, i.e., view-consistency and domain shifts caused by varying densities.

3 Problem Statement

We study the problem of predicting complete 3D keypoints of an underlying object from a single image or depth scan. We assume the input consists of a labeled dataset $\overline{\mathcal{I}}$ and an unlabeled dataset $\mathcal{I}$. Moreover, the unlabeled dataset is comprised of N subsets $\mathcal{I}_i, 1 \leq i \leq N$, where each subset collects depth scans/images of the same object from different views. Such data naturally arises from RGB-D or RGB video sequences.

Each instance $I \in \overline{\mathcal{I}}$ in the labeled dataset possesses a ground-truth label $Y(I) \in \mathbb{R}^{3\times d}$, which is a matrix that collects the coordinates of the ordered keypoints in its columns. Without losing generality, we assume that the 3D local coordinate system of I is chosen so that the centroid of $Y(I)$ is at the origin:

$$Y(I)\mathbf{1} = 0. \tag{1}$$

It is expected that the source domain of the labeled dataset and the target domain of the unlabeled dataset are different (e.g., the source domain consists of synthetic images/scans but the target domain consists of real images/scans). Our goal is to train a neural network $G_\theta : \mathbb{R}^{m\times n} \rightarrow \mathbb{R}^{3\times d}$ that takes an image from the target domain as input and outputs the predicted keypoints by leveraging both the labeled dataset $\overline{\mathcal{I}}$ and unlabeled subsets $\mathcal{I}_i, 1 \leq i \leq N$. We define this problem as unsupervised domain adaptation for 3D keypoint prediction.

Note that we do not assume the underlying cameras of each unlabeled subset are calibrated, or in other words, the relative transformations between different views of the same object are not required. Although it is possible to align the depth scans to obtain relative transformations, we found that such alignments are not always reliable in the presence of scanning discontinuities where little overlaps between consecutive scans are available. In contrast, our formulation treats relative camera poses as latent variables, which are optimized together with the network parameters.

4 Approach

In this section, we describe our detailed approach to unsupervised domain adaptation for 3D keypoint prediction. We first introduce a pose-invariant distance metric to compare keypoint configurations in Sect. 4.1. This allows us to compare the predictions in different views without knowing the relative transformations for uncalibrated datasets. We then present the formulation of our approach in Sect. 4.2. Finally, we discuss our optimization strategy in Sect. 4.3.

4.1 Pose-Invariant Distance Metric

The pose-invariant distance metric compares two keypoint configurations $X, Y \in \mathbb{R}^{3\times d}$ described in different coordinate systems. Since the mean of each keypoint configuration is zero, we introduce a latent rotation R to account for the underlying relative transformation:

$$r(X,Y) = \min_{R\in SO(3)} \|RX - Y\|_{\mathcal{F}}^2, \tag{2}$$

where $\|\cdot\|_{\mathcal{F}}$ denotes the matrix Frobenius Norm. It is clear that $r(X,Y)$ is independent of the coordinate systems associated with X and Y, making it particularly suitable for comparing predictions from uncalibrated views and aligning the source domain and the target domain.

In the following, we discuss a few key properties of $r(X,Y)$ that will be used extensively in our approach. First of all, both $r(X,Y)$ and the gradient of $r(X,Y)$ with respect to each of its argument admit closed-form expressions. These are summarized in the following two propositions.

Proposition 1. *$r(X,Y)$ admits the following analytic expression:*

$$r(X,Y) = \|X\|_{\mathcal{F}}^2 + \|Y\|_{\mathcal{F}}^2 - 2\cdot \mathit{trace}\big(R\cdot(XY^T)\big)$$

where R is derived from the singular value decomposition (or SVD) of $YX^T = U\Sigma V^T$:

$$R = U\mathrm{diag}(1,1,s)V^T, \quad s = \mathit{sign}(\det(XY^T)). \tag{3}$$

Proof: See [13]. □

Proposition 2. *The gradient of $r(X,Y)$ with respect to X is given by*

$$\frac{\partial r}{\partial X}(X,Y) = 2(X - R^TY),$$

where R is given by Eq. (3).

Proof: Please refer to the supplemental material. □

Our optimization procedure also frequently involves the following optimization problem that computes the weighted average $X^\star$ of a set of keypoint configurations $Y_i, 1 \le i \le n$ in the quotient space $\mathbb{R}^{3\times d}/SO(3)$:

$$X^\star = \operatorname*{argmin}_{X\in\mathbb{R}^{3\times d}} \sum_{i=1}^{n} c_i r(X,Y_i) = \operatorname*{argmin}_{X\in\mathbb{R}^{3\times d}} \sum_{i=1}^{n} c_i \min_{R_i\in SO(3)} \|X - R_i^TY_i\|_{\mathcal{F}}^2, \tag{4}$$

where $c_i, 1 \le i \le n$ are constants. Although Eq. (4) does not admit a closed-form solution, it can be easily optimized via alternating minimization. Specifically, when X is fixed, each R_i can be computed independently using Proposition 1. When the R_i latent variables are fixed, X is simply given by the mean of $R_i^TY_i$, i.e., $X = \frac{1}{\sum c_i}\sum_{i=1}^{n} c_iR_i^TY_i$. To make the solution unique, we always set $R_1 = I_3$.

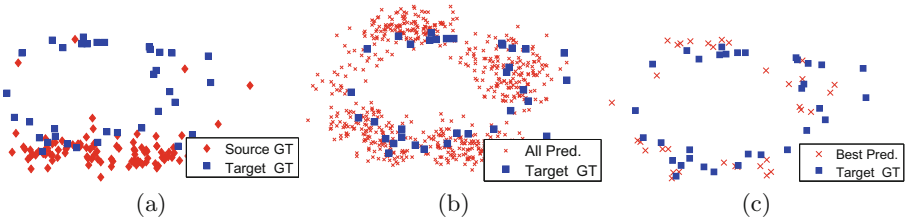

Fig. 3. Latent Distribution and View Selection. This figure provides visualizations of label distributions and view selection for initializing the latent configurations from ModelNet (source domain) to Redwood (target domain) on the Chair category. All visualizations are done by 2D projections using the first two principal components. (a) Label distributions of the source and target domains. (b) Visualizations of all predictions from different views. (c) Visualizations of the best prediction from each object.

4.2 Formulation

To train the keypoint prediction network $G_\theta(\cdot)$, we introduce three loss terms, namely, a labeled term f_{labeled}, a view-consistency term f_{view} and a geometric alignment term f_{align}.

The labeled term f_{labeled} fits predictions on the source domain labeled dataset $\overline{\mathcal{I}}$ to the prescribed ground-truth labels. We use the regression loss under the L2-norm, which works well for 3D keypoint prediction tasks (c.f. [31,45]):

$$f_{\text{labeled}} = \frac{1}{|\overline{\mathcal{I}}|} \sum_{I \in \overline{\mathcal{I}}} \|G_\theta(I) - Y(I)\|_{\mathcal{F}}^2. \tag{5}$$

The view-consistency term f_{view} is defined on the target domain to enforce consistency between the predictions from different views of the same object. In other words, there exist pairwise rotations that transform the predictions from one view to another. A straightforward approach is to minimize $r(G_\theta(I_{ij}), G_\theta(I_{ij'}))$, where I_{ij} and $I_{ij'}$ are different views of the same object. However, we found that such approach introduces a quadratic number of terms as the number of views increases and quickly becomes intractable. Therefore, we introduce a latent configuration $M_i \in \mathbb{R}^{3\times d}$ for each unlabeled subset $\mathcal{I}_i$ that characterizes the underlying ground-truth in the canonical frame. We then define the view consistency term as:

$$f_{\text{view}} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|\mathcal{I}_i|} \sum_{I_{ij} \in \mathcal{I}_i} r(G_\theta(I_{ij}), M_i). \tag{6}$$

It is clear that minimizing f_{view} automatically aligns the predictions across different views. The key advantages of Eq. (6) over enforcing pairwise view-consistency are (i) the number of items is linear to the number of views, and (ii) as we will see immediately, the latent configurations $\{M_i\}$ allow us to easily formulate the geometric alignment term f_{align}.

The geometric alignment term f_{align} prioritizes that the latent configurations $\{M_i, 1 \leq i \leq N\}$, which characterize the predictions on the target domain, shall be consistent with ground-truth labels $\{Y_I | I \in \overline{\mathcal{I}}\}$ of the source domain. This term is conceptually similar to the idea of aligning output distributions for unsupervised domain adaptation, but our formulation is tailored to the specific problem we consider in this paper. A straightforward formulation is to use the Earth-Mover Distance between $\{M_i, 1 \leq i \leq N\}$ and $\{Y(I) | I \in \overline{\mathcal{I}}\}$, which essentially aligns the two corresponding empirical distributions. However, we found that this strategy would force the alignment of keypoint configurations that are far apart, since the repetition counts of the same sub-types of an object may be different between the source and target domains (See Fig. 3(a)). To address this issue, we propose to use the Chamfer distance for alignment:

$$f_{\text{align}} = \frac{1}{N} \sum_{i=1}^{N} \min_{I \in \overline{\mathcal{I}}} r(M_i, Y_I) + \frac{1}{|\overline{\mathcal{I}}|} \sum_{I \in \overline{\mathcal{I}}} \min_{1 \leq i \leq N} r(M_i, Y_I). \tag{7}$$

Intuitively, Eq. (7) still aligns the source and target domains, but it is insensitive to local density variations, and provides an effective way to address domain shifts.

We combine the labeled term f_{labeled}, the view-consistency term f_{view} and the geometric alignment term f_{align} into the final loss function:

$$\underset{\theta, \{M_i\}}{\text{minimize}} f_{\text{labeled}} + \lambda f_{\text{view}} + \mu f_{\text{align}}. \tag{8}$$

In our implementation, we set $\lambda = 1$ and $\mu = 0.1$.

4.3 Optimization

The major difficulty for optimizing Eq. (8) lies in the fact that the alignment term f_{align} is highly non-convex. In our experiments, we found that obtaining good initial values of the network parameters and latent variables is critical to achieving high-quality keypoint prediction network. In the following, we first introduce effective strategies to initialize the variables. We then show how to refine the variables using alternating minimization.

Network Parameter Initialization. The network parameters are initialized by pre-training on the the source domain labeled dataset, i.e.,

$$\theta^{(0)} = \min_{\theta} \sum_{I \in \overline{\mathcal{I}}} \|G_\theta(I) - Y_I\|_{\mathcal{F}}^2. \tag{9}$$

It is then optimized via standard back-propagation.

Latent Configuration Initialization. We use the predictions obtained from the initial network $G_{\theta^{(0)}}(I_{ij}), I_{ij} \in \mathcal{I}_i$ to initialize each latent variable M_i. To this end, we define a score for each prediction and set M_i as the one with the highest score. The scoring function is motivated by the fact that the latent variables

are expected to align with the source domain, we thus define an un-normalized density function:

$$p(M) = \sum_{I \in \overline{\mathcal{I}}} \exp(-\frac{r(M, Y(I))}{2\sigma^2}), \tag{10}$$

where σ is chosen as mean of $r(G_{\theta^{(0)}}(I_{ij}), Y(I))$ between the predicted configurations and their closest labeled instances. Given Eq. (10), we set

$$M_i^{(0)} = \underset{M \in \{G_{\theta^{(0)}}(I) | I \in \mathcal{I}_i\}}{\operatorname{argmax}} p(M). \tag{11}$$

As illustrated in Fig. 3(b–c), this strategy leads to initial configurations that are close to the underlying ground-truth.

Alternating Minimization. Given the initial network parameter $\theta^{(0)}$ and the initial latent configurations $M_i^{(0)}, 1 \leq i \leq N$, we then refine them by solving Eq. (8) via alternating minimization. With $M_i^{(k)}$ and $\theta^{(k)}$ we denote their values at iteration k. At each alternating minimization step, we first fix the latent variables to optimize the network parameters. This leads to computing

$$\theta^{(k+1)} = \underset{\theta}{\operatorname{argmin}} \frac{1}{|\overline{\mathcal{I}}|} \sum_{I \in \overline{\mathcal{I}}} \|G_\theta(I) - Y_I\|_{\mathcal{F}}^2 + \frac{\lambda}{N} \sum_{i=1}^{N} \frac{1}{|\mathcal{I}_i|} \sum_{I \in \mathcal{I}_i} r(G_\theta(I), M_i^{(k)}). \tag{12}$$

Utilizing Proposition 2, we apply stochastic gradient descent via back-propagation for solving Eq. (12).

We then fix the network parameters θ and optimize the latent variables $\{M_i^{(k+1)}\}$. In this case, Eq. (8) reduces to

$$\{M_i^{(k+1)}\} = \underset{\{M_i\}}{\operatorname{argmin}} \frac{\mu}{|\overline{\mathcal{I}}|} \sum_{I \in \overline{\mathcal{I}}} \min_{1 \leq i \leq N} r(M_i, Y_I)$$
$$+ \frac{1}{N} \sum_{i=1}^{N} \Big(\frac{\lambda}{|\mathcal{I}_i|} \sum_{I \in \mathcal{I}_i} r(G_{\theta^{(k)}}(I), M_i) + \mu \min_{I \in \overline{\mathcal{I}}} r(M_i, Y_I) \Big). \tag{13}$$

We again apply alternating minimization to solve Eq. (13). In particular, we fix the closest point pairs given $\{M_i^{(k)}\}$:

$$\hat{I}(i) = \underset{I \in \overline{\mathcal{I}}}{\operatorname{argmin}} r(M_i^{(k)}, Y_I), \quad \hat{i}(I) = \underset{1 \leq i \leq N}{\operatorname{argmin}} r(M_i^{(k)}, Y_I). \tag{14}$$

Given these closest pairs, we can optimize each latent configuration as

$$\underset{M_i}{\operatorname{argmin}} \frac{\mu}{|\overline{\mathcal{I}}|} \sum_{I | \hat{i}(I) = i} r(M_i, Y_I) + \frac{1}{N} \Big(\frac{\lambda}{|\mathcal{I}_i|} \sum_{I \in \mathcal{I}_i} r(G_{\theta^{(k)}}(I), M_i) + \mu r(M_i, Y_{\hat{I}(i)}) \Big). \tag{15}$$

Equation (15) admits a form of Eq. (4), and we apply the procedure described above to solve Eq. (15). In our experiments, we typically apply the inner alternating minimizations each 5 epochs for training the network parameters θ.

5 Evaluation

For experimental evaluation, we first describe the experimental setup in Sect. 5.1. We then present qualitative and quantitative results and compare our technique against baseline approaches in Sect. 5.2. We also present an ablation study to evaluate each component of our approach in Sect. 5.3. Finally, we further extend our method to 3D human pose estimation and RGB images in Sects. 5.4 and 5.5, respectively.

5.1 Experimental Setup

Dataset. Rendered depth scans of synthesized object models from the ModelNet [38] dataset serve as our source domain, and we test our domain adaptation method on three different target domains, namely: ShapeNet [2] (another synthesized 3D model dataset), the Redwood Object Scans real depth scan dataset [3], and the 3DCNN real depth scan dataset [22]. We focus our experiments on the chair, motorbike, and human classes, however we provide the most-detailed results on chairs because of their ubiquitousness across many popular 3D model and depth scan datasets. To provide keypoint labels for our source domain, we manually annotate the training samples in ModelNet with Meshlab [5]. To evaluate the accuracy of our system, we also annotate keypoints on our target domain datasets. This annotation is done by recovering each object's 3D mesh and each frame's camera pose from a depth video sequence. We only maintain frames in which all 2D projections of keypoints are within the image and keep the models with at least 20 valid frames. A summary of the four datasets used in our experiments is presented in Table 3. As a natural extension, we also test our method on the RGB images from the same Redwood dataset [3].

Data Pre-processing. We assume the camera intrinsic and object's 3D bounding box are known both in training and testing depth images solely for data pre-processing. We use the 2D projection of the 3D bounding box to crop each depth image. Additionally, the input depth images are centered by the mean depth and the depth values are normalized by the diagonal length of the 3D bounding box. Aside from the images, all keypoints are converted and evaluated in a unified coordinate system. Given a configuration, we subtract their mean and normalize by the diagonal length of the 3D bounding box.

Evaluation Protocol. Similar to [37], we measure the Average distance Error (AE) between each predicted keypoint configuration and the corresponding annotation and plot the Percentage of Correct Keypoint (PCK) with respect to a threshold for each method for detailed comparison. We also introduce a new metric, Pose-invariant Average distance Error (PAE) based on (2), for a better illustration of how our proposed method works. The AE and PAE are shown in percentage and represent the relative ratio to the diagonal length of the 3D bounding box.

Baseline Methods. We consider three baseline methods for experimental evaluation.

- **Baseline I.** We first test performance without any domain adaptation techniques, namely we directly apply the keypoint predictor trained on the source domain to the target domain. This baseline serves as a performance lower bound for accessing domain adaptation techniques.
- **Baseline II.** We implement a state-of-the-art deep unsupervised general domain adaptation technique described in [35], which encourages domain confusion by fine-tuning the feature extractor on the target domain.
- **Baseline III.** We apply supervised keypoint prediction on the target domain. To this end, we annotate 50 additional models from each domain and fine-tune Baseline I on these labeled instances. This baseline serves as a performance upper bound for accessing domain adaptation techniques.

In Table 1 we compare these baselines to our approach on the Chair dataset. In addition, we provide before/after adaptation results for motorbike and human in Table 2. We also conduct an ablation study on the Chair dataset to evaluate each component of our approach (Table 4 and Fig. 4).

Implementation Details. We use ResNet50 [11] pre-trained on ImageNet as our keypoint prediction network G_θ. In order to fit our depth scans to the ResNet50 input (and additionally, to allow for natural extension to the RGB image domain), we duplicate the depth channel three times. The network is first trained on source domain $\overline{\mathcal{I}}$ for 120 epochs, and then fine-tuned on a specific target domain $\mathcal{I}$ for 30 epochs. The network is trained using a SGD optimizer via back-propagation, with learning rate 0.01 (dropped to 0.001 after 20 epochs), batch size 64, momentum 0.9 and weight decay 1e-4, which are all the default parameters in Resnet50 [11]. Our implementation is done in PyTorch.

5.2 Analysis of Results

Tables 1, 2 and 4, Figs. 4, and 5 present the quantitative and qualitative results of our approach.

Table 1. Results of our proposed methods tested on chairs after domain adaptation on different target domains. We show Average distance Error (AE) and Pose-Invariant Average distance Error (PAE) in percentage. For both metrics, the lower the better.

Target-Metric	Default-AE	ADDA-AE	Ours-AE	Supervised-AE	Default-PAE	ADDA-PAE	Ours-PAE	Supervised-PAE
ModelNet [38]	-	-	-	5.56	-	-	-	4.76
ShapeNet [2]	6.97	6.98	**6.60**	5.82	5.77	5.89	**5.32**	4.77
RedwoodDepth [3]	16.01	15.44	**12.76**	8.67	10.73	10.13	**8.27**	5.68
3DCNN [22]	11.61	11.81	**10.60**	6.73	8.15	8.19	**7.25**	4.98
RedwoodRGB [3]	27.59	26.16	**25.24**	11.90	13.44	12.31	**11.38**	7.67

Table 2. Quantitative results - AE

Category	Motorcycle	Human
Before adaptation	21.55%	153.39 mm
After adaptation	18.92%	135.56 mm
Supervised	16.17%	113.44 mm

Table 3. Statistics of the datasets.

Target	#Train models	#Test models	Avg #frames
ModelNet [38]	899	100	Inf
ShapeNet [2]	2500	100	Inf
Redwood [3]	200	35	150
3DCNN [22]	9	3	80

Qualitative Results. As illustrated in Fig. 5, our approach yields keypoint structures that are consistent with the underlying ground-truths. Even under significant background noise and incomplete observations, our approach leads to faithful structures. Exceptions include the case for chair types that involve swivel bases. In this case, the predicted legs may be tilted. This is expected since the annotations may become unreliable in cases when the legs do not fall directly below the seat corners.

Quantitative Assessment. As shown in Table 1, the mean deviations of our approach in the two real depth scan datasets Redwood [3] and 3DCNN [22] for the chair object class are 12.76% and 10.60% of the diagonal length of object bounding box, respectively. This translates to approximately 7–10 cm, which is fairly accurate when compared to the radius of a chair's base. Additional experiments done on the motorbike class yield similar improvements, as indicated by Table 2. For the motorbike training process, we utilize the ShapeNet dataset as our source domain and the Redwood dataset as our target domain.

Analyses of Performance Across Different Datasets. Table 1 shows that our method gives consistent performance improvements on all three target depth domains. For the synthesized dataset ShapeNet [2], which has a relatively small domain shift from the supervised training set, our unsupervised terms are still able to push error rates close to the supervised upper bound. The advantages of our proposed method can be best observed in the Redwood dataset [3], where using our full error terms leads to a 44% step towards the supervised performance upper-bound. Additionally, the improvement in 3DCNN Dataset [22] is still decent despite the very limited available models and poor depth image quality.

Analysis of Performance Gain. Our performance gains can be attributed to our network learning more plausible keypoint configuration shapes, which is supported by the fact that the improvement of AE is always close to that of PAE. This is expected because our unsupervised terms are viewpoint-invariant and focus on improving the keypoint configuration shape.

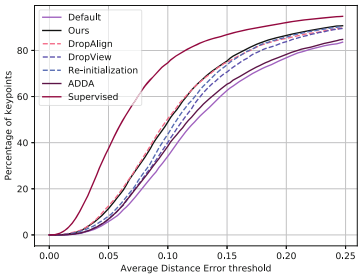

Fig. 4. Baseline & Ablation Study. Comparisons between our approach with alternative approaches on Redwood depth Dataset [3]. The Figure shows Percentage of Correct Keypoints (PCK) under a threshold.

Table 4. Chair ablation study on ShapeNet and Redwood Object Scans dataset. We show the Average distance Error (AE) in percentage for each approach, including the three baselines.

Target domain	ShapeNet(%)	Redwood depth(%)
Ours	**6.60**	**12.76**
Drop view	6.70	13.95
Drop align	6.67	12.97
Re-initialize	6.66	13.43
Default	6.97	16.01
ADDA [35]	6.98	15.44
Supervised lower bound	5.82	8.67

Comparison to ADDA [35]. Our approach is superior to the state-of-the-art unsupervised domain adaptation technique [35] in the keypoint estimation task. ADDA aims to cross the domain gap by aligning the feature distributions of the source and target domains, which is complementary to our approach's constraints on the label space. We argue that there is more structure to rely on in label space than feature space for rigid objects. Another important factor is that view consistency is not incorporated in ADDA [35].

5.3 Ablation Study

We present ablation studies to justify each component of our approach. We restrict our study to a sole object class, chair, and to the representative target domains, ShapeNet and Redwood Object Scans.

Dropping the View-Consistency Term. We test the effects of dropping the view-consistency term. In this case, we simply align the output from all the depth scans with annotations of the source domain. As shown in Table 4 and Fig. 4, the performance drops considerably compared to our full term, while still maintaining better performance than without adaptation. Thus, if the predictions on the majority of views are consistent with one another, the keypoint configuration obtained by averaging all the predictions can serve as a reliable guidance to correct the bad outliers.

Dropping the Alignment Term. Without output alignment, merely utilizing the view consistency term can also significantly reduce the testing error. This can be interpreted as the network updating the latent variables in a self-guided manner, based solely on the consistency between different views.

Latent Configuration Updates Versus Re-initialization. Instead of updating the latent configurations M_i by solving Eq. 15, we can apply Eq. 11 to re-initialize the latent configurations, which is also consistent with our training framework. The results is worse than updating M_i by minimizing the view-consistency term, showing an advantage of our alternating minimization schema.

5.4 Extension to Human Pose

Additionally, we perform experiments for human keypoints using the Human 3.6M dataset [14]. The Human 3.6M dataset [14] provides 3D human joint annotations for 7 subjects (5 for training and 2 for testing) from 4 different camera views. We use 3 of the 5 training subjects as supervised (source) samples and the remaining 2 training subjects as unsupervised (target) samples, trained with the proposed multi-view consistency and output alignment constraints. The result is shown in Table 2 and Fig. 5. The supervised performance upper-bound of our implementation is 113.44 mm, which approximately matches the *3D data-only* state-of-the-art [31].

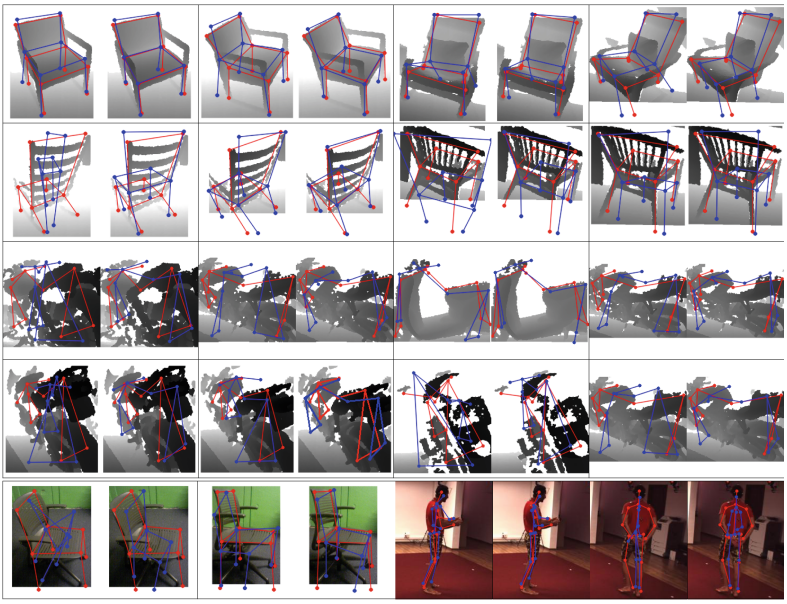

Fig. 5. Qualitative results. We compare 3D keypoint predictions (blue) before (left) and after (right) using our approach on different datasets. For each model we show 2 views. Reference ground-truth are in red (Color figure online).

5.5 Extension to RGB Images

Our approach can seamlessly be applied to keypoint estimation from RGB images. We show our preliminary results on Table 1, which indicate that our proposed method is able to reduce the AE from the baseline without domain adaptation. As shown in Fig. 5, our method helps regularize the output when the before-adaptation baseline predicts a seemingly random point set.

6 Conclusions

In this paper, we introduced an unsupervised domain adaptation approach for keypoint prediction from a single depth image. Our approach combines two task-specific regularizations, i.e., view-consistency and label distributions alignment of the source and target domains. Experimental results show that our approach is significantly better than without domain adaptation and is superior to state-of-the-art generic domain adaptation methods. Additionally, our multi-view consistency and output alignment terms makes it easier to leverage mass amounts of unlabeled 3D data for 3D tasks such as viewpoint estimation and object reconstruction.

Acknowledgement. Qixing Huang would like to acknowledge support of this research from NSF DMS-1700234, a Gift from Snap Research, and a hardware Donation from NVIDIA.

References

1. Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
2. Chang, A.X., et al.: ShapeNet: an information-rich 3D model repository. CoRR abs/1512.03012 (2015)
3. Choi, S., Zhou, Q.Y., Miller, S., Koltun, V.: A large dataset of object scans (2016). arXiv:1602.02481
4. Choy, C.B., Xu, D., Gwak, J.Y., Chen, K., Savarese, S.: 3D-R2N2: a unified approach for single and multi-view 3D object reconstruction. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 628–644. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_38
5. Cignoni, P., Callieri, M., Corsini, M., Dellepiane, M., Ganovelli, F., Ranzuglia, G.: MeshLab: an open-source mesh processing tool. In: Eurographics Italian Chapter Conference, vol. 2008, pp. 129–136 (2008)
6. Csurka, G.: Domain adaptation for visual applications: a comprehensive survey. CoRR abs/1702.05374 (2017)
7. Fish Tung, H.Y., Harley, A.W., Seto, W., Fragkiadaki, K.: Adversarial inverse graphics networks: learning 2D-to-3D lifting and image-to-image translation from unpaired supervision. In: The IEEE International Conference on Computer Vision (ICCV), October 2017
8. Gebru, T., Hoffman, J., Fei-Fei, L.: Fine-grained recognition in the wild: a multi-task domain adaptation approach. In: The IEEE International Conference on Computer Vision (ICCV), October 2017

9. Gholami, B., (Oggi) Rudovic, O., Pavlovic, V.: PUnDA: probabilistic unsupervised domain adaptation for knowledge transfer across visual categories. In: The IEEE International Conference on computer Vision (ICCV), October 2017
10. Gupta, S., Arbeláez, P.A., Girshick, R.B., Malik, J.: Aligning 3D models to RGB-D images of cluttered scenes. In: Computer Vision and Pattern Recognition (CVPR) (2015)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778 (2016)
12. Herath, S., Harandi, M., Porikli, F.: Learning an invariant hilbert space for domain adaptation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
13. Horn, B.K.P.: Closed-form solution of absolute orientation using unit quaternions. J. Opt. Soc. Am. A **4**(4), 629–642 (1987)
14. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6M: large scale datasets and predictive methods for 3D human sensing in natural environments. IEEE Trans. Pattern Anal. Mach. Intell. **36**(7), 1325–1339 (2014)
15. Kalogerakis, E., Averkiou, M., Maji, S., Chaudhuri, S.: 3D shape segmentation with projective convolutional networks. CoRR abs/1612.02808 (2016)
16. Koniusz, P., Tas, Y., Porikli, F.: Domain adaptation by mixture of alignments of second- or higher-order scatter tensors. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
17. Li, Y., Dai, A., Guibas, L., Nießner, M.: Database-assisted object retrieval for real-time 3D reconstruction. In: Computer Graphics Forum, vol. 34. Wiley Online Library (2015)
18. Maria Carlucci, F., Porzi, L., Caputo, B., Ricci, E., Rota Bulo, S.: AutoDIAL: automatic domain alignment layers. In: The IEEE International Conference on Computer Vision (ICCV), October 2017
19. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 483–499. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_29
20. Panareda Busto, P., Gall, J.: Open set domain adaptation. In: The IEEE International Conference on Computer Vision (ICCV), October 2017
21. Peng, X., Sun, B., Ali, K., Saenko, K.: Learning deep object detectors from 3D models. In: ICCV, pp. 1278–1286. IEEE Computer Society (2015)
22. Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J.: Volumetric and multi-view CNNs for object classification on 3D data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5648–5656 (2016)
23. Rhodin, H., et al.: Learning monocular 3D human pose estimation from multi-view images. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018
24. Sankaranarayanan, S., Balaji, Y., Castillo, C.D., Chellappa, R.: Generate to adapt: aligning domains using generative adversarial networks. CoRR abs/1704.01705 (2017)
25. Simon, T., Joo, H., Matthews, I., Sheikh, Y.: Hand keypoint detection in single images using multiview bootstrapping. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
26. Song, S., Xiao, J.: Sliding shapes for 3D object detection in depth images. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 634–651. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_41

27. Song, S., Xiao, J.: Deep sliding shapes for amodal 3D object detection in RGB-D images (2016)
28. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.: Semantic scene completion from a single depth image. In: Proceedings of 30th IEEE Conference on Computer Vision and Pattern Recognition (2017)
29. Su, H., Qi, C.R., Li, Y., Guibas, L.J.: Render for CNN: viewpoint estimation in images using CNNs trained with rendered 3D model views. In: The IEEE International Conference on Computer Vision (ICCV), December 2015
30. Su, H., Wang, F., Yi, E., Guibas, L.J.: 3D-assisted feature synthesis for novel views of an object. In: ICCV, pp. 2677–2685. IEEE Computer Society (2015)
31. Sun, X., Shang, J., Liang, S., Wei, Y.: Compositional human pose regression. In: The IEEE International Conference on Computer Vision (ICCV), October 2017
32. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Multi-view 3D models from single images with a convolutional network. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 322–337. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_20
33. Tulsiani, S., Malik, J.: Viewpoints and keypoints. CoRR abs/1411.6067 (2014)
34. Tulsiani, S., Zhou, T., Efros, A.A., Malik, J.: Multi-view supervision for single-view reconstruction via differentiable ray consistency. CoRR abs/1704.06254 (2017)
35. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. arXiv preprint arXiv:1702.05464 (2017)
36. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
37. Wu, J., et al.: Single image 3D interpreter network. CoRR abs/1604.08685 (2016)
38. Wu, Z., et al.: 3D ShapeNets: a deep representation for volumetric shapes. In: CVPR, pp. 1912–1920 (2015)
39. Yan, H., Ding, Y., Li, P., Wang, Q., Xu, Y., Zuo, W.: Mind the class weight bias: weighted maximum mean discrepancy for unsupervised domain adaptation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
40. Yan, X., Yang, J., Yumer, E., Guo, Y., Lee, H.: Perspective transformer nets: learning single-view 3D object reconstruction without 3D supervision. CoRR abs/1612.00814 (2016)
41. Zhang, Y., David, P., Gong, B.: Curriculum domain adaptation for semantic segmentation of urban scenes. In: The IEEE International Conference on Computer Vision (ICCV), October 2017
42. Zhang, Y., et al.: Physically-based rendering for indoor scene understanding using convolutional neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
43. Zhao, B., Wu, X., Cheng, Z., Liu, H., Feng, J.: Multi-view image generation from a single-view. CoRR abs/1704.04886 (2017)
44. Zhou, T., Tulsiani, S., Sun, W., Malik, J., Efros, A.A.: View synthesis by appearance flow. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 286–301. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_18
45. Zhou, X., Huang, Q., Sun, X., Xue, X., Wei, Y.: Towards 3D human pose estimation in the wild: a weakly-supervised approach. In: The IEEE International Conference on Computer Vision (ICCV), October 2017