Unsupervised Geometry-Aware Representation for 3D Human Pose Estimation

Helge Rhodin[3−2692−0801], Mathieu Salzmann[2−8347−8637], and Pascal Fua[2−6702−9970]

CVLab, EPFL, Lausanne, Switzerland
{helge.rhodin,mathieu.salzmann,pascal.fua}@epfl.ch

Abstract. Modern 3D human pose estimation techniques rely on deep networks, which require large amounts of training data. While weakly-supervised methods require less supervision, by utilizing 2D poses or multi-view imagery without annotations, they still need a sufficiently large set of samples with 3D annotations for learning to succeed.
In this paper, we propose to overcome this problem by learning a geometry-aware body representation from multi-view images without annotations. To this end, we use an encoder-decoder that predicts an image from one viewpoint given an image from another viewpoint. Because this representation encodes 3D geometry, using it in a semi-supervised setting makes it easier to learn a mapping from it to 3D human pose. As evidenced by our experiments, our approach significantly outperforms fully-supervised methods given the same amount of labeled data, and improves over other semi-supervised methods while using as little as 1% of the labeled data.

Keywords: 3D reconstruction, semi-supervised training, representation learning, monocular human pose reconstruction.

1 Introduction

Most current monocular solutions to 3D human pose estimation rely on methods based on convolutional neural networks (CNNs). With networks becoming ever more sophisticated, the main bottleneck now is the availability of sufficiently large training datasets, which typically require a large annotation effort. While such an effort might be practical for a handful of subjects and specific motions such as walking or running, covering the whole range of human body shapes, appearances, and poses is infeasible.

Weakly-supervised methods that reduce the amount of annotation required to achieve a desired level of performance are therefore valuable. For example, methods based on articulated 3D skeletons can be trained not only with actual 3D annotations but also using 2D annotations [21, 54] and multi-view footage [25, 47]. Some methods dispense with 2D annotations altogether and instead exploit multi-view geometry in sequences acquired by synchronized cameras [31, 55]. However, these methods still require a good enough 3D training set to initialize

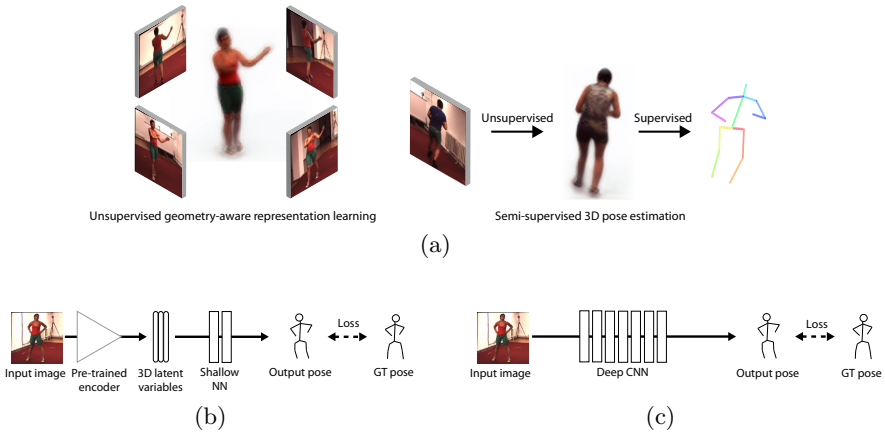


Fig. 1: Approach. (a) During training, we first learn a geometry-aware representation using unlabeled multi-view images. We then use a small amount of supervision to learn a mapping from our representation to actual 3D poses, which only requires a shallow network and therefore a limited amount of supervision. **(b)** At run-time, we compute the latent representation of the test image and feed it to the shallow network to compute the pose. **(c)** By contrast, most state-of-the-art approaches train a network to regress directly from the images to the 3D poses, which requires a much deeper network and therefore more training data.

the learning process, which sets limits on the absolute gain that can be achieved from using unlabeled examples.

In this paper, we propose to use images of the same person taken from multiple views to learn a latent representation that, as shown on the left side of Fig. 1(a), captures the 3D geometry of the human body. Learning this representation does not require any 2D or 3D pose annotation. Instead, we train an encoder-decoder to predict an image seen from one viewpoint from an image captured from a different one. As sketched on the right side of Fig. 1(a), we can then learn to predict a 3D pose from this latent representation in a supervised manner. The crux of our approach, however, is that because our latent representation already captures 3D geometry, the mapping to 3D pose is much simpler and can be learned using much fewer examples than existing methods that rely on multi-view supervision [31, 55], and more generally most state-of-the-art methods that attempt to regress directly from the image to the 3D pose.

As can be seen in Fig. 1, our latent representation resembles a volumetric 3D shape. While such shapes can be obtained from silhouettes [50, 45], body outlines are typically difficult to extract from natural images. By contrast, learning our representation does not require any silhouette information. Furthermore, at test time, it can be obtained from a monocular view of the person. Finally, it can also be used for novel view synthesis (NVS) and outperforms existing encoder-decoder algorithms [36, 37, 23] qualitatively on natural images.

Our contribution is therefore a latent variable body model that can be learned without 2D or 3D annotations, encodes both 3D pose and appearance, and can be integrated into semi-supervised approaches to reduce the required amount of supervised training data. We demonstrate this on the well-known Human3.6Million [13] dataset and show that our method drastically outperforms fully supervised methods in 3D pose reconstruction accuracy when only few labeled examples are available.

2 Related work

In the following, we first review the literature on semi-supervised approaches to monocular 3D human pose estimation, which is most closely related to our goal. We then discuss approaches that, like us, make use of geometric representations, both in and out of the context of human pose estimation, and finally briefly review the novel view synthesis literature that has inspired us.

Semi-supervised human pose estimation. While most current human pose estimation methods [25, 54, 24, 42, 27, 20, 22, 33, 38] are fully supervised, relying on large training sets annotated with ground-truth 3D positions coming from multi-view motion capture systems [21, 12], several methods have recently been proposed to limit the requirement for labeled data. In this context, foreground and background augmentation [30, 32] and the use of synthetic datasets [2, 48] focus on increasing the training set size. Unfortunately, these methods do not generalize well to new motions, apparels, and environments that are different from the simulated data. Since larger and less constrained datasets for 2D pose estimation exist, they have been used for transfer learning [47, 22] and to provide re-projection constraints [54]. Furthermore, given multiple views of the same person, 3D pose can be triangulated from 2D detections [25, 14] and a 2D pose network can be trained to be view-consistent after bootstrapping from annotations. Nevertheless, these methods still require 2D annotation in images capturing the target motion and appearance. By contrast, the methods of [31, 55] exploit multi-view geometry in sequences acquired by synchronized cameras, thus removing the need for 2D annotations. However, in practice, they still require a large enough 3D training set to initialize and constrain the learning process. We will show that our geometry-aware latent representation learned from multi-view imagery but without annotations allows us to train a 3D pose estimation network using much less labeled data.

Geometry-aware representations. Multi-view imagery has long been used to derive volumetric representations of 3D human pose from silhouettes, for example by carving out the empty space. This approach can be used in conjunction with learning-based methods [44], by defining constraints based on perspective view rays [45, 15], orthographic projections [50], or learned projections [29]. It can even be extended to the single-view training-scenario if the distribution of the observed shape can be inferred prior to reconstruction [56, 8]. The main drawback of these methods, however, is that accurate silhouettes are difficult to automatically extract in natural scenes, which limits their applicability.

Another approach to encoding geometry relies on a renderer that generates images from a 3D representation [9, 35, 16, 52] and can function as a decoder in an autoencoder setup [1, 39]. For simple renderers, the rendering function can even be learned [5, 6] and act as an encoder. When put together, such learned encoders and decoders have been used for unsupervised learning, both with GANs [3, 43, 46] and without them [17]. In [41, 40], a CNN was trained to map to and from spherical mesh representations without supervision. While these methods also effectively learn a geometry-aware representation based on images, they have only been applied to well-constrained problems, such as face modeling. As such, it is unclear how they would generalize to the much larger degree of variability of 3D human poses.

Novel view synthesis. Our approach borrows ideas from the novel view synthesis literature, which is devoted to the task of creating realistic images from previously unseen viewpoints. Most recent techniques rely on encoder-decoder architectures, where the latent code is augmented with view change information, such as yaw angle, and the decoder learns to reconstruct the encoded image from a new perspective [36, 37]. Large view changes are difficult. They have been achieved by relying on a recurrent network that performs incremental rotation steps [51]. Optical flow information [23, 53] and depth maps [7] have been used to further improve the results. While the above-mentioned techniques were demonstrated on simple objects, methods dedicated to generating images of humans have been proposed. However, most of these methods use additional information as input, such as part-segmentations [18] and 2D poses [19]. Here, we build on the approaches of [4, 49] that have been designed to handle large viewpoint changes. We describe these methods and our extensions in more detail in Section 3.

3 Unsupervised Geometry-Aware Latent Representation

Our goal is to design a latent representation $\mathbf{L}$ that encodes 3D pose, along with shape and appearance information, and can be learned without any 2D or 3D pose annotations. To achieve this, we propose to make use of sequences of images acquired from multiple synchronized and calibrated cameras. To be useful, such footage requires care during the setup and acquisition process. However, the amount of effort involved is negligible compared to what is needed to annotate tens of thousands of 2D or 3D poses.

For $\mathbf{L}$ to be practical, it must be easy to decode into its individual components. To this end, we learn from the images separate representations for the body's 3D pose and geometry, its appearance, and that of the background. We will refer to them as $\mathbf{L}^{\text{3D}}$, $\mathbf{L}^{\text{app}}$, and $\mathbf{B}$, respectively.

Let us assume that we are given a set, $\mathcal{U} = \{(\mathbf{I}_t^i, \mathbf{I}_t^j)\}_{t=1}^{N_u}$, of N_u image pairs without annotations, where the i and j superscripts refer to the cameras used to capture the images, and the subscript t to the acquisition time. Let $\mathbf{R}^{i \to j}$ be the rotation matrix from the coordinate system of camera i to that of camera j. We now turn to the learning of the individual components of $\mathbf{L}$.

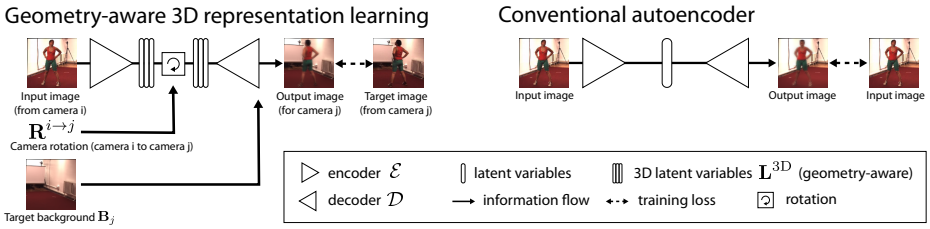


Fig. 2: Representation learning. We learn a representation that encodes geometry and thereby 3D pose information in an unsupervised manner. Our method (Left) extends a conventional auto encoder (Right) with a 3D latent space, rotation operation, and background fusion module. The 3D rotation enforces explicit encoding of 3D information. The background fusion enables application to natural images.

Learning to encode multi-view geometry. For individual images, autoencoders such as the one shown on the right side of Fig. 2 have become standard tools to learn latent representations in unsupervised settings. Let such an autoencoder comprise an encoder $\mathcal{E}_{\theta_e}$ and a decoder $\mathcal{D}_{\theta_d}$, where θ_e and θ_d are the weights controlling their behaviors. For image representation purposes, an autoencoder can be used to encode an image $\mathbf{I}$ into a latent representation $\mathbf{L} = \mathcal{E}_{\theta_e}(\mathbf{I})$, which can then be decoded into a reconstructed image $\hat{\mathbf{I}} = \mathcal{D}_{\theta_d}(\mathbf{L})$. θ_e and θ_d are learned by minimizing $\|\mathbf{I} - \hat{\mathbf{I}}\|^2$ on average over a training set $\mathcal{U}$.

To leverage multi-view geometry, we take our inspiration from Novel View Synthesis methods [36, 37, 4, 49, 11] that rely on training encoder-decoders on multiple views of the same object, such as a car or a chair. Let $(\mathbf{I}_t^i, \mathbf{I}_t^j) \in \mathcal{U}$ be two images taken from different viewpoints but at the same time t. Since we are given the rotation matrix $\mathbf{R}^{i\to j}$ connecting the two viewpoints, we could feed this information as an additional input to the encoder and decoder and train them to encode $\mathbf{I}_t^i$ and resynthesize $\mathbf{I}_t^j$, as in [36, 37]. Then, novel views of the object could be rendered by varying the rotation parameter $\mathbf{R}^{i\to j}$. However, this does not force the latent representation to encode 3D information explicitly. To this end, we model the latent representation $\mathbf{L}^{3D} \in \mathbb{R}^{3\times N}$ as a set of N points in 3D space by designing the encoder $\mathcal{E}_{\theta_e}$ and decoder $\mathcal{D}_{\theta_e}$ so that they have a three channel output and input, respectively, as shown on the left side of Fig. 2. This enables us to model the view-change as a proper 3D rotation by matrix multiplication of the encoder output by the rotation matrix before using it as input to the decoder. Formally, the output of the resulting autoencoder $\mathcal{A}_{\theta_e,\theta_d}$ can be written as

$$\mathcal{A}_{\theta_e,\theta_d}(\mathbf{I}_t^i, \mathbf{R}^{i\to j}) = \mathcal{D}_{\theta_d}(\mathbf{R}^{i\to j}\mathbf{L}_{i,t}^{3D}), \text{ with } \mathbf{L}_{i,t}^{3D} = \mathcal{E}_{\theta_e}(\mathbf{I}_t^i)\ , \tag{1}$$

and the weights θ_d and θ_e are optimized to minimize $\|\mathcal{A}_{\theta_e,\theta_d}(\mathbf{I}_t^i, \mathbf{R}^{i\to j}) - \mathbf{I}_t^j\|$ over the training set $\mathcal{U}$. In this setup, which was also used in [4, 49] and is inspired by [11], the decoder $\mathcal{D}$ does not need to learn how to rotate the input to a new view but only how to decode the 3D latent vector $\mathbf{L}^{3D}$. This means that the encoder is forced to map to a proper 3D latent space, that is, one that can still be decoded

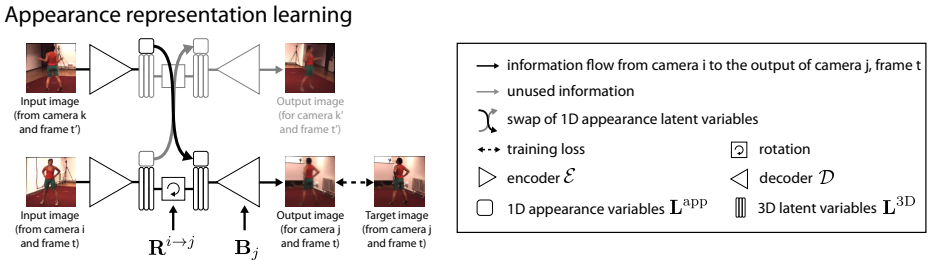


Fig. 3: Appearance representation learning. To encode subject identity, we split the latent space into a 3D geometry part and an appearance part. The latter is not rotated, but swapped between two time frames t and t' depicting the same subject, so as to enforce it not to contain geometric information.

by $\mathcal{D}$ after an arbitrary rotation. However, while $\mathbf{L}^{\text{3D}}$ now encodes multi-view geometry, it also encodes the background and the person's appearance. Our goal now is to isolate them from $\mathbf{L}^{\text{3D}}$ and to create two new vectors $\mathbf{B}$ and $\mathbf{L}^{\text{app}}$ that encode the latter two so that $\mathbf{L}^{\text{3D}}$ only represents geometry and 3D pose.

Factoring out the background. Let us assume that we can construct background images $\mathbf{B}_j$, for example by taking the median of all the images taken from a given viewpoint j. To factor them out, we introduce in the decoder a direct connection to the target background $\mathbf{B}_j$, as shown in Fig. 2. More specifically, we concatenate the background image with the output of the decoder and use an additional 1×1 convolutional layer to synthesize the decoded image. This frees the rest of the network from having to learn about the background and ensures that the $\mathbf{L}^{\text{3D}}$ vector we learn does not contain information about it anymore.

Factoring out appearance. To separate appearance from geometry in our latent representation, we break up the output of the encoder $\mathcal{E}$ into two separate vectors $\mathbf{L}^{\text{3D}}$ and $\mathbf{L}^{\text{app}}$ that should describe pose and appearance, respectively. To enforce this separation, we train simultaneously on two frames $\mathbf{I}_t$ and $\mathbf{I}_{t'}$ depicting the same subject at different times, t and t', as depicted in Fig. 3. While the decoder uses $\mathbf{L}^{\text{3D}}_t$ and $\mathbf{L}^{\text{3D}}_{t'}$, as before, it swaps $\mathbf{L}^{\text{app}}_t$ and $\mathbf{L}^{\text{app}}_{t'}$. In other words, the decoder uses $\mathbf{L}^{\text{3D}}_t$ and $\mathbf{L}^{\text{app}}_{t'}$ to resynthesize frame t and $\mathbf{L}^{\text{3D}}_{t'}$ and $\mathbf{L}^{\text{app}}_t$ for frame t'. Assuming that the person's appearance does not change drastically between t and t' and that differences in the images are caused by 3D pose changes, this results in $\mathbf{L}^{\text{3D}}$ encoding pose while $\mathbf{L}^{\text{app}}$ encodes appearance.

In practice, the encoder $\mathcal{E}$ has two outputs, that is, $\mathcal{E}_{\theta_e} : \mathbf{I}^i_t \to (\mathbf{L}^{\text{3D}}_{i,t}, \mathbf{L}^{\text{app}}_{i,t})$ and the decoder $\mathcal{D}_{\theta_d}$ accepts these plus the background as inputs, after swapping appearance and rotating the geometric representation for two views i and j. We therefore write the output of our encoder-decoder as

$$\mathcal{A}_{\theta_e,\theta_d}(\mathbf{I}^i_t, \mathbf{R}^{i\to j},\ \mathbf{L}^{\text{app}}_{k,t'},\ \mathbf{B}_j) = \mathcal{D}_{\theta_d}(\mathbf{R}^{i\to j}\mathbf{L}^{\text{3D}}_{i,t},\ \mathbf{L}^{\text{app}}_{k,t'},\ \mathbf{B}_j)\ . \tag{2}$$

The viewpoint k can be arbitrary. The critical point is that it was acquired at time $t' \neq t$ such that the poses at t and t' are uncorrelated. Thus, only

time-invariant appearance features are encoded into $\mathbf{L}^{\text{app}}$. A similar exchange of information has been performed before in [28] for analogy transformations. It is related to works that separate facial identity, pose and illumination [17, 51, 26].

Combined optimization. To train $\mathcal{A}$ with sequences featuring several people and backgrounds, we randomly select mini-batches of Z triplets $(\mathbf{I}_t^i, \mathbf{I}_t^j, \mathbf{I}_{t'}^k)$ in $\mathcal{U}$, with $t \neq t'$, from individual sequences. In other words, all three views feature the same person. The first two are taken at the same time but from different viewpoints. The third is taken at a different time and from an arbitrary viewpoint k. For each such mini-batch, we compute the loss

$$E_{\theta_d,\theta_e} = \frac{1}{Z} \sum_{\substack{\mathbf{I}_t^i, \mathbf{I}_t^j, \mathbf{I}_{t'}^k \in \mathcal{U} \\ t \neq t'}} \left\| \mathcal{A}_{\theta_e,\theta_d}(\mathbf{I}_t^i, \mathbf{R}^{i \to j}, \mathbf{L}_{k,t'}^{\text{app}},\ \mathbf{B}_j) - \mathbf{I}_t^j \right\| , \tag{3}$$

where $\mathbf{L}_{k,t'} = (\mathbf{L}_{k,t'}^{\text{3D}}, \mathbf{L}_{k,t'}^{\text{app}})$ is the output of encoder $\mathcal{E}_{\theta_e}$ applied to image $\mathbf{I}_{t'}^k$, $\mathbf{B}_j$ is the background in view j, and $\mathbf{R}^{i \to j}$ denotes the rotation from view i to view j. Note that we apply $\mathcal{E}$ twice, to obtain $\mathbf{L}_{i,t}^{\text{3D}}$ and $\mathbf{L}_{k,t'}^{\text{app}}$ in Eq. 3 while ignoring $\mathbf{L}_{i,t}^{\text{app}}$ and $\mathbf{L}_{k,t'}^{\text{3D}}$ with the swap discussed above.

At training time, we minimize a total loss that is the sum of the pixel-wise error E_{θ_d,θ_e} of Eq. 3 and a second term obtained by first applying a Resnet with 18 layers trained on ImageNet on the output and target image and then computing the feature difference after the second block level, as previously done with VGG by [23]. All individual pixel and feature differences are averaged and their influence is balanced by weighting the feature loss by two. We experiment with L1 and L2 norms. The L1 norm in combination with the additional feature term allows for crisper decodings and improved pose reconstruction.

Translation and Augmentation. Object scale and translation in depth direction are inherently ambiguous for monocular reconstruction and NVS. To make our model invariant instead of ambiguous to these effects, we use the crop information provided in the training datasets. We compute the rotation between two views with respect to the crop center instead of the image center and shear the cropped image so that it appears as if it were taken from a virtual camera pointing in the crop direction. With the human in the same position and scale, these crops remove the need to model object and camera translation. We also apply random in-plane rotations to increase view diversity. As a result, $\mathbf{R}^{i \to j}$ and $\mathbf{B}_j$ depend on time t, but we neglect this in our notation for readability.

4 3D Human Pose Estimation

Recall that our ultimate goal is to infer the 3D pose of a person from a monocular image. Since $\mathbf{L}^{\text{3D}}$ can be rotated and used to generate novel views, we are already part way there. Being a $3 \times N$ matrix, it can be understood as a set of N 3D points, but these do not have any semantic meaning. However, in most practical

applications, one has to infer a pre-defined representation, such as a skeleton with K major human body joints, encoded as a vector $\mathbf{p} \in \mathbb{R}^{3K}$.

To instantiate such a representation, we need a mapping $\mathcal{F} : \mathbf{L}^{\text{3D}} \rightarrow \mathbb{R}^{3K}$, which can be thought as a different decoder that reconstructs 3D poses instead of images. To learn it, we rely on supervision. However, as we will see in the results section, the necessary amount of human annotations is much smaller than what would have been required to learn the mapping directly from the images, as in many other recent approaches to human pose estimation.

Let $\mathcal{L} = \{(\mathbf{I}_t, \mathbf{p}_t)\}_{t=1}^{N_s}$ be a small set of N_s labeled examples made of image pairs and corresponding ground-truth 3D poses. We model $\mathcal{F}$ as a deep network with parameters θ_f. We train it by minimizing the objective function

$$E_{\theta_f} = \frac{1}{N_s} \sum_{t=1}^{N_s} \|\mathcal{F}_{\theta_f}(\mathbf{L}_t^{\text{3D}}) - \mathbf{p}_t\| \ , \ \text{with } (\mathbf{L}_t^{\text{3D}}, \cdot) = \mathcal{E}_{\theta_e}(\mathbf{I}_t) \ . \tag{4}$$

Because our latent representation $\mathbf{L}^{\text{3D}}$ already encodes human 3D pose and shape, $\mathcal{F}$ can be implemented as a simple two-layer fully-connected neural network. Together with the encoder-decoder introduced in Section 3, which is trained in an unsupervised manner, they form the semi-supervised setup depicted by Fig. 1(b). In other words, our unsupervised representation does a lot of the hard-work in the difficult task of lifting the image to a 3D representation, which makes the final mapping comparatively easy.

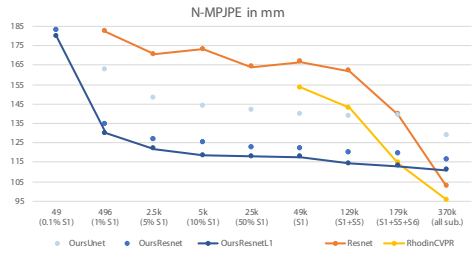
5 Evaluation

In this section, we first evaluate our approach on the task of 3D human pose estimation, which is our main target application, and show that our representation enables us to use far less annotated training data than state-of-the-art approaches to achieve better accuracy. We then evaluate the quality of our latent space itself and show that it does indeed encode geometry, appearance, and background separately.

Dataset. We use the well-known Human3.6M (H36M) [12] dataset. It is recorded in a calibrated multi-view studio and ground-truth human poses are available for all frames. This makes it easy to compare different levels of supervision, unsupervised, semi-supervised, or fully supervised. As in previous approaches [54, 31, 42, 27, 22], we use the bounding boxes provided with the dataset to crop images.

5.1 Semi-Supervised Human Pose Estimation

Our main focus is semi-supervised human pose estimation. We now demonstrate that, as shown in Fig. 4, recent state-of-the-art methods can do better than us when large amounts of annotated training data are available. However, as we use fewer and fewer of these annotations, the accuracy of the baselines suffers greatly whereas ours does not, which confers a significant advantage in situations where annotations are hard to obtain. We now explain in detail how the graphs of Fig. 4 were produced and further discuss their meaning.



(a)

Type	Method	MPJPE	NMPJPE	PMPJPE
Fully-supervised S1	**Resnet**	177.2	166.5	135.6
Semi-supervised S1	**RhodinCVPR** [31]	n/a	153.3	128.6
	OursUnet	149.5	135.9	106.4
	OursResnet	**131.7**	**122.6**	**98.2**

(b)

Fig. 4: (a) Performance as function of the number of training samples. When using all the available annotated 3D data in H36M, that is, 370,000 images, **RhodinCVPR** and **Resnet** yield a better accuracy than our approach. However, when the number of training examples drops below 180'000 the baselines' accuracy degrades significantly, whereas **OursResnet** degrades much more gracefully and our accuracy becomes significantly better. **(b)** This improvement is consistent across metrics.

Metrics. We evaluate pose prediction accuracy in terms of the mean per joint prediction error (MPJPE), and its normalized variants N-MPJPE and P-MPJPE, where poses are aligned to the ground truth in the least-square sense either in scale only or in scale, rotation and translation, respectively, before computing the MPJPE. The latter is also known as Procrustes alignment. We do this over 16 major human joints and all positions are centered at the pelvis, as in [54]. Our results are consistent across all metrics, as shown in Fig. 4 (b).

Baselines. We compare our approach against the state-of-the-art semi-supervised method of [31], which uses the same input as ours and outputs normalized poses. We will refer to it as **RhodinCVPR**. We also use the popular ResNet-based architecture [22] to regress directly from the image to the 3D pose, as shown in Fig. 1(c), we will refer to this as **Resnet**.

Note that even higher accuracies on H36M than those of **RhodinCVPR** and **Resnet** have been reported in the literature [24, 20, 54, 38, 27] but they depend both on more complex architectures and using additional information such as labeled 2D poses [22, 54, 38, 20] or semantic segmentation [27], which is not our point here. We want to show that when using *only* 3D annotations and not many of them are available, our representation still allows us to perform well.

Implementation. We base our encoder-decoder architecture on the UNet [34] network, which was used to perform a similar task in [19]. We simply remove the skip connections to force the encoding of all information into the latent spaces and reduce the number of feature channels by half.

Concretely the encoder $\mathcal{E}$ consists of four blocks of two convolutions, where each two convolutions are followed by max pooling. The resulting convolutional features are of dimension $512 \times 16 \times 16$ for an input image resolution of 128×128 pixels. These are mapped to $\mathbf{L}^{\text{app}} \in \mathbb{R}^{128}$ and $\mathbf{L}^{\text{3D}} \in \mathbb{R}^{200\times 3}$ by a single fully-connected layer followed by dropout with probability 0.3. The decoder $\mathcal{D}$ maps

$\mathbf{L}^{3D}$ to a feature map of dimension $(512-128)\times 16\times 16$ with a fully-connected layer followed by ReLU and dropout and duplicates $\mathbf{L}^{app}$ to form a spatial uniform map of size $128\times 16\times 16$. These two maps are concatenated and then reconstructed by four blocks of two convolutions, where the first convolution is preceded by bilinear interpolation and all other pairs by up-convolutions. Each convolution is followed by batch-normalization and ReLU activation functions. We also experimented with a variant in which the encoder $\mathcal{E}$ is an off-the shelf Resnet with fifty layers [10], pre-trained on ImageNet, and the decoder is the same as before. We will refer to these two versions as **OursUnet** and **OursResnet**, respectively.

The pose decoder $\mathcal{F}$ is a fully-connected network with two hidden layers of dimension 2048. The ground-truth poses in the least-squares loss of Eq. 4 are defined as root-centered 3D poses. Poses and images are normalized by their mean and standard deviation on the training set. We use mini-batches of size 32 and the Adam optimizer with learning rate 10^{-3} for optimization of θ_e, θ_d and θ_f.

Dataset splits. On H36M, we take the unlabeled set $\mathcal{U}$ used to learn our representation to be the complete training set—S1, S5, S6, S7 and S8, where SN refers to all sequences of the N^{th} subject—but without the available 3D labels. To provide the required supervision to train the shallow network of Fig. 1(b), we then define several scenarios.

- Fully supervised training with the 3D annotation of all five training subjects.
- We use all the 3D annotations for S1; S1 and S5; or S1, S5 and S6.
- We use only 50%, 10%, 5%, 1% or 0.1% of the 3D annotations for S1.

In all cases we used S9 and S11 for testing. We subsampled the test and training videos at 10ps to reduce redundancy and validation time. The resulting numbers of annotated images we used are shown along the x-axis of Fig. 4.

Comparison to the state of the art. RhodinCVPR is the only method that is designed to leverage unlabeled multi-view footage without using a supplemental 2D dataset [31]. **OursUnet** outperforms it significantly, e.g., on labeled subject S1 by 13.6 mm (8.9% relative improvement) and **OursResnetL1** even attains a gain of 35.7 mm (23.3% relative improvement). The fact that the Resnet architecture, training procedure, and dataset split is the same for our method and **RhodinCVPR** evidences that this gain is due to our new way of exploiting the unlabeled examples, thus showing the effectiveness of learning a geometry-aware latent representation in an unsupervised manner.

Discussion and Ablation Study. As shown in Fig. 4, when more than 300,000 annotated images are used the baselines outperform us. However, their accuracy decreases rapidly when fewer are available and our approach then starts dominating. It only loses accuracy very slowly down to 5,000 images and still performs adequately given only 500.

We used the L2 loss in Eq. 3 by default since our main goal is 3D pose estimation, not NVS quality. Interestingly, however, using the L1 loss makes reconstructions not only crisper but also 3D poses estimates more accurate. It

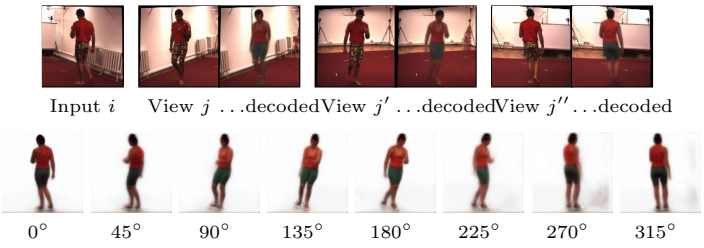


Fig. 5: Novel viewpoint synthesis. Top row. Each one of the three image pairs of image to the left of it comprise an original image acquired from a different viewpoint and the image synthesized from the input image i. **Bottom row.** We can also synthesize images for previously unseen viewpoints and remove the background.

improves pose accuracy consistently by about 5%, shown as **OursResnetL1** in Fig. 4. Unless indicated otherwise, all results are produced with the L2 metric.

To better evaluate different aspects of our approach, we use the **OursUnet** version to conduct an ablation study whose results we report in Table 1. In short, not separating the background and appearance latent spaces reduces N-MPJPE by 14 mm and P-MPJPE by more than 12 mm. Using two hidden layers in $\mathcal{F}$ instead of one increases accuracy by 12 mm. The loss term based on ResNet-18 features not only leads to crisper NVS results but also improves pose estimation by 9 mm. Using bilinear upsampling instead of deconvolution for all decoding layers reduces performance by 4 mm. The largest decrease in accuracy by far, 46.1mm, occurs when we use our standard **OursUnet** architecture but *without* our geometry-aware 3D latent space. It appears in the last line of the table on the left and strongly suggests that using our latent representation has more impact than tweaking the architecture in various ways.

5.2 Evaluating the Latent Representation Qualitatively

We now turn to evaluating the quality of our latent representation as such with a number of experiments on **OursUnet**. We show that geometry can be separated from appearance and background and that this improves results. The quality of the synthesized images is best seen in the supplemental videos.

Method	N-MPJPE	P-MPJPE
OursUnet*	**145.6**	**112.2**
OursUnet*, w/o appearance space, as in [4, 49]	159.0	117.1
OursUnet*, w/o background handling, as in [4, 49]	159.6	124.6
OursUnet*, w/o 3D latent space, as in [36, 37]	191.7	139.0

Method	N-MPJPE	P-MPJPE
OursUnet*	**145.6**	**112.2**
OursUnet*, bilinear upsampling	149.2	114.1
OursUnet*, w/o ImgNet loss	154.1	118.7
OursUnet*, $\mathcal{F}$ with 1 hidden layer	157.4	121.9

* no rotation augmentation. Errors are reported in mm.

Table 1: Ablation study, using S1 for semi-supervised training. The extensions to the NVS methods [36, 37] and [4, 49] as well as further model choices improve accuracy.

Fig. 6: Ablation study. First row. Without background handling, as used in [4, 49], the synthesized foreground pose appears fuzzy. **Second row.** Without a geometry-aware latent space, as used by [36, 37], results are inaccurate and blurred in new views. **Third row. OursResnet** captures pose and appearance accurately, but contours are still blurred. **Fourth row. OursResnetL1** produces crisper and more accurate results.

Novel View Synthesis. Recall from Section 3 that $\mathcal{E}$ encodes the image into variables $\mathbf{L}^{\text{3D}}$ and $\mathbf{L}^{\text{app}}$, which are meant to represent geometry and appearance, respectively. To check that this is indeed the case, we multiply $\mathbf{L}^{\text{3D}}$ by different rotation matrices $\mathbf{R}$ and feed the result along with the original $\mathbf{L}^{\text{app}}$ to $\mathcal{D}$. Fig. 5 depicts such synthesized novel views.

For comparison purposes, in Fig. 6, we synthesize rotated images without using our geometry-aware latent space, that is, as in [37]. The resulting images are far blurrier than those of **OursResnet**. Fig. 6 further shows that results degrade without the background handling, that is, as in [4, 49, 11]. Using the L1 instead of L1 loss further improves reconstruction quality. Test subjects wear clothes that differ in color and shape from those seen in the training data. As a result, the geometry in the synthesized images remains correct, but the appearance ends up being a mixture of training appearances that approximates the unseen appearance. Arguably, using more than the five subjects that appear in the training set should result in a better encoding of appearance, which is something we plan to investigate in future work.

Appearance and background switching. Let $\mathbf{I}_j$ and $\mathbf{I}_g$ be two images of subjects j and g and $(\mathbf{L}_j^{\text{3D}}, \mathbf{L}_j^{\text{app}}, \mathbf{B}_j) = \mathcal{E}(\mathbf{I}_j)$ and $(\mathbf{L}_g^{\text{3D}}, \mathbf{L}_g^{\text{app}}, \mathbf{B}_g) = \mathcal{E}(\mathbf{I}_g)$ their encodings. Re-encoding using $\mathbf{L}^{\text{3D}}$ of one and $\mathbf{L}^{\text{app}}$ of the other yields results such as those depicted by Fig. 7. Note that the appearance of one is correctly transferred to the pose of the other while the geometry remains intact under rotation. This method could be used to generate additional training data, by changing the appearance of an existing multi-view sequence to synthesize images of the same motion being performed by multiple actors.

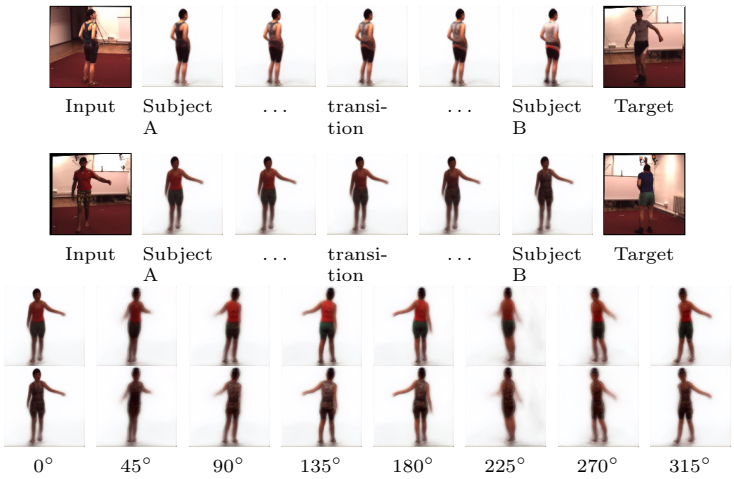


Fig. 7: Appearance separation. Top two rows. The same pose can be decoded to different identities by blending the appearance latent vectors. In the first row, both subjects appear in the training set. In the second row, they are from the test set. **Bottom two rows.** We generate rotated views of the test subject and its transferred appearance, to demonstrate that appearance can be changed without affecting 3D pose.

Fig. 8: Background separation. The background is handled separately from the foreground an can be chosen arbitrary at decoding time. From left to right, input image, decoded on the input background, on a novel view, on a white, and on a picture. The first row features someone from the training set and the second row from the test set.

Similarly, we can switch backgrounds instead of appearances before decoding the latent vectors, as shown in Fig. 8. In one case, we make the background white and in the other we use a natural scene. In the first case, dark patches are visible below the subject, evidently modeling shadowing effects that were learned implicitly. In the second case, the green trees tend to be rendered as orange because our training scenes were mostly reddish—problem a larger training database would almost certainly cure.

5.3 Generalization and Limitations

To analyze the scalability of the unsupervised training we tested using only four out of the five unsupervised training subjects. The additional subject improves OurResnet drastically by 16 N-MPJPE. This indicates that training is not yet

Fig. 9: Foreground objects are reconstructed too, if seen in training and testing.

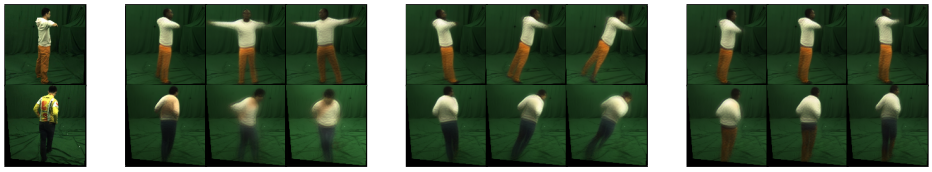


Fig. 10: Generalization on 3DHP. Our NVS solution generalizes well to the different camera placements in 3DHP, allowing for yaw, pitch and roll transformations.

saturated and much higher accuracies seem possible by leveraging huge unsupervised sets. These, be it indoors or outdoors, are relatively easy to obtain.

In the data we used, some of the images contain a chair on which the subject sits. Interestingly, as shown in Fig. 9, the chair appearance and 3D position is faithfully reconstructed by our method. This suggests that it is not specific to human poses and can generalize to rigid objects as well as multiple object classes. In future work, we intend to apply it to such more generic problems.

We further tested our method on the MPI-INF-3DHP (3DHP) [21] dataset, which features more diverse clothing and viewpoints, such as low-hanging and ceiling cameras, and is therefore well suited to probe extreme conditions for NVS. Without changing any parameter, **OursResnet** is able to synthesis view transformations in roll, yaw and pitch, as shown in Fig. 10. On H36M, pitch transformation could not be learned due to the solely chest-height training views.

6 Conclusion

We have introduced an approach to learning a geometry-aware representation of the human body in an unsupervised manner, given only multi-view imagery. Our experiments have shown that this representation is effective both as an intermediate one for 3D pose estimation and for novel view synthesis. For pose estimation, our semi-supervised approach performs much better than state-of-the-art methods when only very little annotated data is available. In future work, we will extend its range by learning an equivalent latent representation for much larger multi-view datasets but still in an unsupervised manner.

Acknowledgment. This work was supported in part by a Microsoft Joint Research Project.

References

1. Bas, A., Huber, P., Smith, W., Awais, M., Kittler, J.: 3D Morphable Models as Spatial Transformer Networks. arXiv Preprint (2017)
2. Chen, W., Wang, H., Li, Y., Su, H., Wang, Z., Tu, C., Lischinski, D., Cohen-or, D., Chen, B.: Synthesizing Training Images for Boosting Human 3D Pose Estimation. In: 3DV (2016)
3. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. In: Advances in Neural Information Processing Systems. pp. 2172–2180 (2016)
4. Cohen, T., Welling, M.: Transformation Properties of Learned Visual Representations. arXiv Preprint (2014)
5. Dosovitskiy, A., Springenberg, J., Brox, T.: Learning to Generate Chairs with Convolutional Neural Networks. In: Conference on Computer Vision and Pattern Recognition (2015)
6. Dosovitskiy, A., Springenberg, J., Tatarchenko, M., Brox, T.: Learning to Generate Chairs, Tables and Cars with Convolutional Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(4), 692–705 (2017)
7. Flynn, J., Neulander, I., Philbin, J., Snavely, N.: Deepstereo: Learning to Predict New Views from the World's Imagery. In: Conference on Computer Vision and Pattern Recognition. pp. 5515–5524 (2016)
8. Gadelha, M., Maji, S., Wang, R.: 3d shape induction from 2d views of multiple objects. arXiv preprint arXiv:1612.05872 (2016)
9. Grant, E., Kohli, P., van M. Gerven: Deep Disentangled Representations for Volumetric Reconstruction. In: European Conference on Computer Vision. pp. 266–279 (2016)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: CVPR. pp. 770–778 (2016)
11. Hinton, G., Krizhevsky, A., Wang, S.: Transforming Auto-Encoders. In: International Conference on Artificial Neural Networks. pp. 44–51 (2011)
12. Ionescu, C., Carreira, J., Sminchisescu, C.: Iterated Second-Order Label Sensitive Pooling for 3D Human Pose Estimation. In: Conference on Computer Vision and Pattern Recognition (2014)
13. Ionescu, C., Papava, I., Olaru, V., Sminchisescu, C.: Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. IEEE Transactions on Pattern Analysis and Machine Intelligence (2014)
14. Joo, H., Liu, H., Tan, L., Gui, L., Nabbe, B., Matthews, I., Kanade, T., Nobuhara, S., Sheikh, Y.: Panoptic Studio: A Massively Multiview System for Social Motion Capture. In: International Conference on Computer Vision (2015)
15. Kar, A., Häne, C., Malik, J.: Learning a multi-view stereo machine. In: Advances in Neural Information Processing Systems. pp. 364–375 (2017)
16. Kim, H., Zollhöfer, M., Tewari, A., Thies, J., Richardt, C., Theobalt, C.: Inversefacenet: Deep Single-Shot Inverse Face Rendering from a Single Image. arXiv Preprint (2017)
17. Kulkarni, T.D., Whitney, W., Kohli, P., Tenenbaum, J.B.: Deep Convolutional Inverse Graphics Network. In: arXiv (2015)
18. Lassner, C., Pons-Moll, G., Gehler, P.: A Generative Model of People in Clothing. arXiv Preprint (2017)

19. Ma, L., Jia, X., Sun, Q., Schiele, B., Tuytelaars, T., Gool, L.V.: Pose Guided Person Image Generation. In: Advances in Neural Information Processing Systems. pp. 405–415 (2017)
20. Martinez, J., Hossain, R., Romero, J., Little, J.: A Simple Yet Effective Baseline for 3D Human Pose Estimation. In: International Conference on Computer Vision (2017)
21. Mehta, D., Rhodin, H., Casas, D., Fua, P., Sotnychenko, O., Xu, W., Theobalt, C.: Monocular 3D Human Pose Estimation in the Wild Using Improved CNN Supervision. In: International Conference on 3D Vision (2017)
22. Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H., Xu, W., Casas, D., Theobalt, C.: Vnect: Real-Time 3D Human Pose Estimation with a Single RGB Camera. In: ACM SIGGRAPH (2017)
23. Park, E., Yang, J., Yumer, E., Ceylan, D., Berg, A.: Transformation-Grounded Image Generation Network for Novel 3D View Synthesis. In: Conference on Computer Vision and Pattern Recognition. pp. 702–711 (2017)
24. Pavlakos, G., Zhou, X., Derpanis, K., Konstantinos, G., Daniilidis, K.: Coarse-To-Fine Volumetric Prediction for Single-Image 3D Human Pose. In: Conference on Computer Vision and Pattern Recognition (2017)
25. Pavlakos, G., Zhou, X., Konstantinos, K.D.G., Kostas, D.: Harvesting Multiple Views for Marker-Less 3D Human Pose Annotations. In: Conference on Computer Vision and Pattern Recognition (2017)
26. Peng, X., Feris, R.S., Wang, X., Metaxas, D.N.: A recurrent encoder-decoder network for sequential face alignment. In: ECCV (2016)
27. Popa, A.I., Zanfir, M., Sminchisescu, C.: Deep Multitask Architecture for Integrated 2D and 3D Human Sensing. In: Conference on Computer Vision and Pattern Recognition (2017)
28. Reed, S., Zhang, Y., Zhang, Y., Lee, H.: Deep Visual Analogy-Making. In: Advances in Neural Information Processing Systems. pp. 1252–1260 (2015)
29. Rezende, D., Eslami, S., Mohamed, S., Battaglia, P., Jaderberg, M., Heess, N.: Unsupervised Learning of 3D Structure from Images. In: Advances in Neural Information Processing Systems. pp. 4996–5004 (2016)
30. Rhodin, H., Richardt, C., Casas, D., Insafutdinov, E., Shafiei, M., Seidel, H.P., B, S., Theobalt, C.: Egocap: Egocentric Marker-Less Motion Capture with Two Fisheye Cameras. ACM SIGGRAPH Asia **35**(6) (2016)
31. Rhodin, H., Spoerri, J., Katircioglu, I., Constantin, V., Meyer, F., Moeller, E., Salzmann, M., Fua, P.: Learning Monocular 3D Human Pose Estimation from Multi-View Images. In: Conference on Computer Vision and Pattern Recognition (2018)
32. Rogez, G., Schmid, C.: Mocap Guided Data Augmentation for 3D Pose Estimation in the Wild. In: Advances in Neural Information Processing Systems (2016)
33. Rogez, G., Weinzaepfel, P., Schmid, C.: Lcr-Net: Localization-Classification-Regression for Human Pose. In: Conference on Computer Vision and Pattern Recognition (2017)
34. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Conference on Medical Image Computing and Computer Assisted Intervention (2015)
35. Shu, Z., Yumer, E., Hadap, S., Sunkavalli, K., Shechtman, E., Samaras, D.: Neural Face Editing with Intrinsic Image Disentangling. In: Conference on Computer Vision and Pattern Recognition (2017)
36. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Single-view to multi-view: Reconstructing unseen views with a convolutional network. CoRR abs/1511.06702 **1**, 2 (2015)

37. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Multi-view 3d models from single images with a convolutional network. In: European Conference on Computer Vision. pp. 322–337. Springer (2016)
38. Tekin, B., Márquez-neila, P., Salzmann, M., Fua, P.: Learning to Fuse 2D and 3D Image Cues for Monocular Body Pose Estimation. In: International Conference on Computer Vision (2017)
39. Tewari, A., Zollhöfer, M., Kim, H., Garrido, P., Bernard, F., Pérez, P., Theobalt, C.: Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. International Conference on Computer Vision (2017)
40. Thewlis, J., Bilen, H., Vedaldi, A.: Unsupervised learning of object frames by dense equivariant image labelling. In: Advances in Neural Information Processing Systems. pp. 844–855 (2017)
41. Thewlis, J., Bilen, H., Vedaldi, A.: Unsupervised learning of object landmarks by factorized spatial embeddings. In: International Conference on Computer Vision (2017)
42. Tome, D., Russell, C., Agapito, L.: Lifting from the Deep: Convolutional 3D Pose Estimation from a Single Image. In: arXiv preprint, arXiv:1701.00295 (2017)
43. Tran, L., Yin, X., Liu, X.: Disentangled representation learning gan for pose-invariant face recognition. In: CVPR. vol. 3, p. 7 (2017)
44. Tulsiani, S., Efros, A., Malik, J.: Multi-View Consistency as Supervisory Signal for Learning Shape and Pose Prediction. arXiv Preprint (2018)
45. Tulsiani, S., Zhou, T., Efros, A., Malik, J.: Multi-view supervision for single-view reconstruction via differentiable ray consistency. In: Conference on Computer Vision and Pattern Recognition. vol. 1, p. 3 (2017)
46. Tung, H.Y., Harley, A., 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). vol. 2 (2017)
47. Tung, H.Y., Tung, H.W., Yumer, E., Fragkiadaki, K.: Self-supervised learning of motion capture. In: Advances in Neural Information Processing Systems. pp. 5242–5252 (2017)
48. Varol, G., Romero, J., Martin, X., Mahmood, N., Black, M., Laptev, I., Schmid, C.: Learning from synthetic humans. In: Conference on Computer Vision and Pattern Recognition (2017)
49. Worrall, D., Garbin, S., Turmukhambetov, D., Brostow, G.: Interpretable transformations with encoder-decoder networks. In: International Conference on Computer Vision. vol. 4 (2017)
50. Yan, X., Yang, J., Yumer, E., Guo, Y., Lee, H.: Perspective Transformer Nets: Learning Single-View 3D Object Reconstruction Without 3D Supervision. In: Advances in Neural Information Processing Systems. pp. 1696–1704 (2016)
51. Yang, J., Reed, S., Yang, M.H., Lee, H.: Weakly-supervised disentangling with recurrent transformations for 3d view synthesis. In: Advances in Neural Information Processing Systems. pp. 1099–1107 (2015)
52. Zhao, B., Wu, X., Cheng, Z.Q., Liu, H., Feng, J.: Multi-view image generation from a single-view. arXiv preprint arXiv:1704.04886 (2017)
53. Zhou, T., Tulsiani, S., Sun, W., Malik, J., Efros, A.: View synthesis by appearance flow. In: European conference on computer vision. pp. 286–301. Springer (2016)
54. Zhou, X., Huang, Q., Sun, X., Xue, X., We, Y.: Weakly-Supervised Transfer for 3D Human Pose Estimation in the Wild. arXiv Preprint (2017)

55. Zhou, X., Karpur, A., Gan, C., Luo, L., Huang, Q.: Unsupervised domain adaptation for 3d keypoint prediction from a single depth scan. arXiv preprint arXiv:1712.05765 (2017)
56. Zhu, J.Y., Park, T., Isola, P., Efros, A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. arXiv preprint arXiv:1703.10593 (2017)