Neural Scene Decomposition for Multi-Person Motion Capture

Helge Rhodin   Victor Constantin   Isinsu Katircioglu   Mathieu Salzmann   Pascal Fua
CVLab, EPFL, Lausanne, Switzerland
helge.rhodin@epfl.ch

Abstract

Learning general image representations has proven key to the success of many computer vision tasks. For example, many approaches to image understanding problems rely on deep networks that were initially trained on ImageNet, mostly because the learned features are a valuable starting point to learn from limited labeled data. However, when it comes to 3D motion capture of multiple people, these features are only of limited use.

In this paper, we therefore propose an approach to learning features that are useful for this purpose. To this end, we introduce a self-supervised approach to learning what we call a neural scene decomposition (NSD) that can be exploited for 3D pose estimation. NSD comprises three layers of abstraction to represent human subjects: spatial layout in terms of bounding-boxes and relative depth; a 2D shape representation in terms of an instance segmentation mask; and subject-specific appearance and 3D pose information. By exploiting self-supervision coming from multiview data, our NSD model can be trained end-to-end without any 2D or 3D supervision. In contrast to previous approaches, it works for multiple persons and full-frame images. Because it encodes 3D geometry, NSD can then be effectively leveraged to train a 3D pose estimation network from small amounts of annotated data.

1. Introduction

Most state-of-the-art approaches to 3D pose estimation use a deep network to regress from the image either directly to 3D joint locations or to 2D ones, which are then lifted to 3D using another deep network. In either case, this requires large amounts of training data that may be hard to obtain, especially when attempting to model non-standard motions.

In other areas of computer vision, such as image classification and object detection, this has been handled by using a large, generic, annotated database to train networks to produce features that generalize well to new tasks. These features can then be fed to other, task-specific deep nets, which can be trained using far less labeled data. AlexNet [28] and VGG [59] have proved to be remarkably successful at this, resulting in many striking advances.

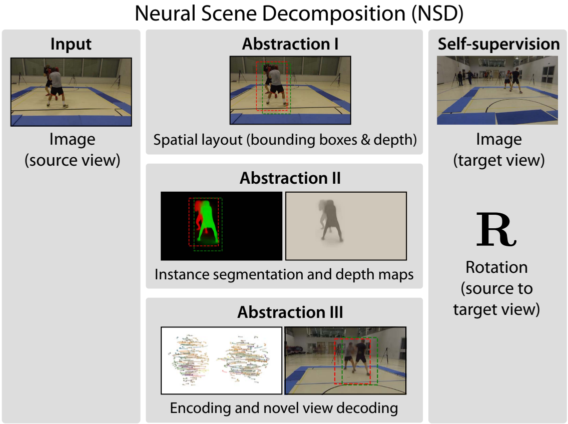

Neural Scene Decomposition (NSD)
Input
Image
(source view)
Abstraction I
Spatial layout (bounding boxes & depth)
Abstraction II
Instance segmentation and depth maps
Abstraction III
Encoding and novel view decoding
Self-supervision
Image
(target view)
R
Rotation
(source to
target view)

Figure 1. **Neural Scene Decomposition** disentangles an image into foreground and background, subject bounding boxes, depth, instance segmentation, and latent encodings in a fully self-supervised manner using a second view and its relative view transformation for training.

Our goal is to enable a similar gain for 3D human pose estimation. A major challenge is that there is no large, generic, annotated database equivalent to those used to train AlexNet and VGG that can be used to learn our new representation. For example, Human 3.6M [21] only features a limited range of motions and appearances, even though it is one of the largest publicly available human motion databases. Thus, only limited supervision has to suffice.

As a step towards our ultimate goal, we therefore introduce a new scene and body representation that facilitates the training of 3D pose estimators, even when only little annotated data is available. To this end, we train a neural network that infers a compositional scene representation that comprises three levels of abstraction. We will refer to it as *Neural Scene Decomposition* (NSD). As shown in Fig. 1, the first one captures the spatial layout in terms of bounding boxes and relative depth; the second is a pixel-wise instance segmentation of the body; the third is a geometry-aware hidden space that encodes the 3D pose, shape and appearance independently. Compared to existing solutions,

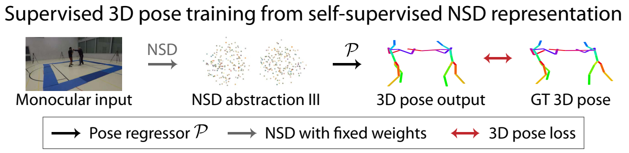


Figure 2. **3D pose estimation.** Pose is regressed from the NSD latent representation that is inferred from the image. During training, the regressor $\mathcal{P}$ requires far less supervision than if it had to regress directly from the image.

NSD enables us to deal with full-frame input and multiple people. As bounding boxes may overlap, it is crucial to also infer a depth ordering. The key to instantiating this representation is to use multi-view data at training time for self-supervision. This does not require image annotations, only knowledge of the number of people in the scene and camera calibration, which is much easier to obtain.

Our contribution is therefore a powerful representation that lets us train a 3D pose estimation network for multiple people using comparatively little training data, as shown in Fig. 2. The network can then be deployed in scenes containing several people potentially occluding each other while requiring neither bounding boxes nor even detailed knowledge of their location or scale. This is made possible though the new concept of Bidirectional Novel View Synthesis (Bi-NVS) and is in stark contrast to other approaches based on classical Novel View Synthesis (NVS). These are designed to work with only a single subject in an image crop so that the whole frame is filled [50] or require two or more views not only at training time but also at inference time [13].

Our neural network code and the new boxing dataset will be made available upon request for research purposes.

2. Related work

Existing human pose estimation datasets are either large scale but limited to studio conditions, where annotation can be automated using marker-less multiview solutions [57, 36, 21, 23], simulated [8, 77], or generic but small [5, 51] because manual annotation [33] is cumbersome. Multi-person pose datasets are even more difficult to find. Training sets are usually synthesized from single person 3D pose [37] or multi-person 2D pose [53] datasets; real ones are tiny and meant for evaluation only [37, 78]. In practice, this data bottleneck starkly limits the applicability of deep learning-based single [42, 69, 46, 39, 35, 38, 52, 43, 91, 65, 61] and multi-person [37, 53, 85] 3D pose estimation methods. In this section, we review recent approaches to addressing this limitation, in particular those that are most related to ours and exploit unlabeled images for representation learning.

Weak Pose Supervision. There are many tasks for which labeling is easier than for full 3D pose capture. This has been exploited via transfer learning [38], cross-modal variational [60] and adversarial [83] learning both 2D and 3D pose estimation; minimizing the re-projection error of 3D poses to 2D labels in single [91, 29, 31] and multiple views [23, 43]; annotating the joint depth order instead of the absolute position [45, 41]; re-projection to silhouettes [62, 75, 24, 44, 76].

Closer to us, in [58], a 2D pose detector is iteratively refined by imposing view consistency in a massive multi-view studio. A similar approach is pursued in the wild in [89, 51]. While effective, these approaches remain strongly supervised as their performance is closely tied to that of the regressors used for bootstrapping.

In short, all of these methods reduce the required amount of annotations but still need a lot. Furthermore, the process has to be repeated for new kinds of motion, with potentially different target keypoint locations and appearances. Boxing and skiing are examples of this because they involve motions different enough from standard ones to require full re-training. We build upon these methods to further reduce the annotation effort.

Learning to Reconstruct. If multiple views of the same object are available, geometry alone can suffice to infer 3D shape. By building on traditional model-based multi-view reconstruction techniques, networks have been optimized to predict a 3D shape from monocular input that fulfills stereo [16, 89], visual hull [81, 25, 71, 47], and photometric re-projection constraints [73]. Even single-view training is possible if the observed shape distribution can be captured prior to reconstruction [93, 15]. The focus of these methods is on rigid objects and they do not apply to dynamic and articulated human pose. Furthermore, many of them require silhouettes as input, which are difficult to automatically extract from natural scenes. We address both of these aspects.

Representation Learning. Completely unsupervised methods have been extensively researched for representation learning. For instance autoencoders have long been used to learn compact image representations [4]. Well-structured data can also be leveraged to learn disentangled representations, using GANs [9, 70] or variational autoencoders [19]. In general, the image features learned in this manner are rarely relevant to 3D reconstruction.

Such relevance can be induced by hand-crafting a parameteric rendering function that replaces the decoder in the autoencoder setup [66, 3], or by training either the encoder [30, 17, 74, 55, 27] or the decoder [11, 12] on structured datasets. To encode geometry explicitly, the methods of [68, 67] map to and from spherical mesh representations without supervision and that of [87] selects 2D keypoints to provide a latent encoding. However, these methods have only been applied to well-constrained problems, such as face modeling, and do not provide the hierarchical 3D decomposition we require.

Most similar to our approach are methods using camera pose estimation [84] and NVS [50, 13] as auxiliary tasks for geometry-aware representation learning. In particular, it was shown in [13] that reinforcement learning of 3D grasping converges much faster when using NVS features instead of raw images. This, however, was demonstrated only in simulation. In [50], NVS is applied in natural scenes for human pose estimation, using a geometry-aware representation based on transforming autoencoders [20, 10, 80]. This approach, however, is restricted to images of single humans with tight ground-truth bounding box annotations used at training *and* test time. Here, we introduce a hierarchical scene decomposition that allows us to deal with images depicting multiple subjects, without requiring any other information than the multiview images and the camera poses during training, and only single view images at test time.

3. Method

Our goal is to learn a high-level scene representation that is optimized for 3D human pose estimation tasks, that is, detecting people and recovering their pose from *single* images. We refer to this as Neural Scene Decomposition (NSD). To create this NSD, we rely at training time on Novel View Synthesis (NVS) using multiple views and enforcing consistency among the results generated from different views.

Fig. 1 summarizes our approach. Given a scene containing N people, we want to find N corresponding bounding boxes $(\mathbf{b}_i)_{i=1}^N$, segmentation masks $(\mathbf{S}_i)_{i=1}^N$, depth plane estimates $(z_i)_{i=1}^N$, and latent representation $([\mathbf{L}_i^{\text{app}}, \mathbf{L}_i^{\text{3D}}])_{i=1}^N$ where $\mathbf{L}_i^{\text{app}}$ is a vector representing appearance and $\mathbf{L}_i^{\text{3D}}$ a matrix encoding geometry. Our challenge then becomes training a deep network to instantiate this scene decomposition from images in a completely self-supervised fashion. This means training without bounding boxes, human pose estimates, depth, or instance segmentation labels.

To meet this challenge, we ground NSD on standard deep architectures for supervised object detection and representation learning [31, 72] and NVS [50], and add new network layers and objective functions to enable self-supervision. In the remainder of this section, we first summarize NVS. We then show how we go from there to NSD, first for a single person and then for multiple. We provide more implementation details in the supplementary material.

3.1. Novel View Synthesis

Given two images, $(\mathbf{I}_v, \mathbf{I}_{v'})$, of the same scene taken from different viewpoints, NVS seeks to synthesize from $\mathbf{I}_v$ a novel view $\mathcal{F}(\mathbf{I}_v, \mathbf{R}_{v,v'}, \mathbf{t}_{v,v'})$ that is as close as possible to $\mathbf{I}_{v'}$, where $\mathbf{R}_{v,v'}$ and $\mathbf{t}_{v,v'}$ are the rotation matrix and translation vector defining the camera motion from v to v'. This is typically done by minimizing

$$L(\mathcal{F}(\mathbf{I}_v, \mathbf{R}_{v,v'}, \mathbf{t}_{v,v'}), \mathbf{I}_{v'}) \ , \qquad (1)$$

where L is an appropriate image-difference metric, such as the L^2 norm. This requires static and calibrated cameras, which much less labor intensive to setup than precisely annotating many images with bounding boxes, 3D poses, depth ordering, and instance segmentation. This is one of the main attractions of using NVS for training purposes.

Previous NVS approaches focused merely on rigid objects [63, 64, 82, 40, 90, 14, 10, 80]. Methods that synthesize human pose and appearance have used clean silhouettes and portrait images [88, 92] and intermediate 2D and 3D pose estimates to localize the person's body parts [34, 32, 86, 56, 1, 79, 26]. We rely on the approach of [50] that focuses on representation learning and uses an encoding-decoding architecture without needing human pose supervision. Its encoder $\mathcal{E}(\cdot)$ maps the input image $\mathbf{I}_v$ to an appearance vector $\mathbf{L}_v^{\text{app}}$ and a 3D point cloud $\mathbf{L}_v^{\text{3D}}$ that represents geometry. In the rest of the paper we will refer to the pair $[\mathbf{L}_v^{\text{app}}, \mathbf{L}_v^{\text{3D}}]$ as the *latent representation* of $\mathbf{I}_v$. A novel view is then obtained by rotating $\mathbf{L}_v^{\text{3D}}$ by $\mathbf{R}_{v,v'}$ and then running the decoder $\mathcal{D}(\cdot)$ on the rotated cloud and original appearance vector, that is, computing $\mathcal{D}(\mathbf{R}_{v,v'}\mathbf{L}_v^{\text{3D}}, \mathbf{L}_v^{\text{app}})$.

This NVS formulation assumes that subjects are portrayed individually and at the same scale in each image, which makes it possible to ignore the translation $\mathbf{t}_{v,v'}$ but precludes real-world application where scale may vary significantly. In practice, this is achieved by exploiting the ground-truth bounding box around each subject at both training and test time.

To overcome this limitation, we propose to complement the latent representations produced by this NVS-based scene decomposition with all the information required to deal with multiple people appearing at different scales in the multi-view images. We therefore introduce a novel architecture that we first describe in the case where there is only one person and then in the multi-person scenario.

3.2. NSD with a Single Subject

Existing NVS solutions require scale and position normalization because changes in object scale and translations along the camera optical axis can compensate each other under perspective projection. In particular, a person's absolute height can be predicted from an image only with uncertainty [18]. Hence, it is geometrically impossible to predict the size and position in a novel view.

To alleviate this problem and to attain the sought NSD, we introduce an explicit detection and localization step, along with the notion of bidirectional NVS, that allows us to mix the information extracted from two views in the NVS process. Our complete framework is outlined in Fig. 3. We now describe each one of these components individually, assuming there is only one person in the scene.

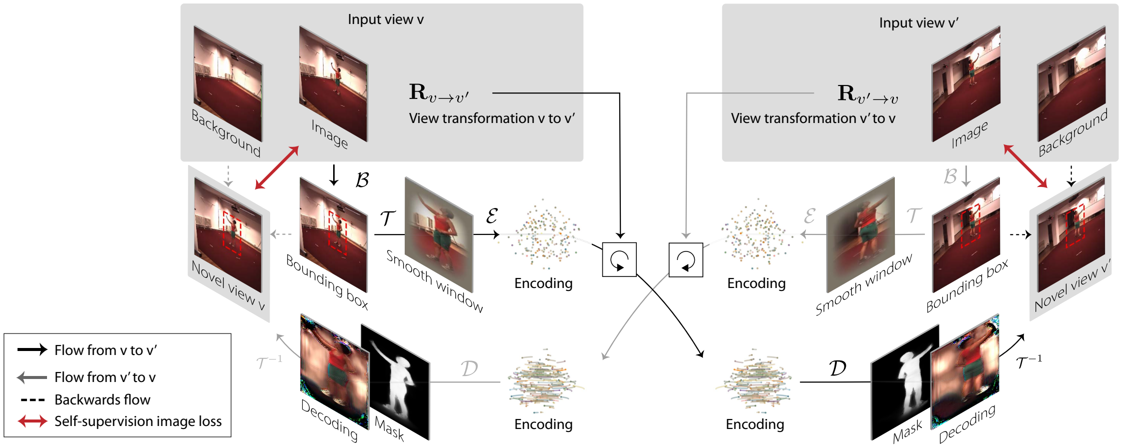


Figure 3. **Bidirectional NVS** jointly predicts a novel view v' from v and v from v', mixing object location and scale estimates between the two directions. This overcomes scale ambiguities in classical NVS, which predicts v' from v without backwards flow from v'.

Subject Detection and Localization. To estimate the position and observed size of a single subject whose height and 3D location are initially unknown, we run a detector network $\mathcal{B}$ on the input image $\mathbf{I}_v$. Let $\mathbf{b}_v = \mathcal{B}(\mathbf{I}_v)$ denote the resulting bounding box that tightly contains the subject. We use it to define the spatial transformer (ST) network, $\mathcal{T}$, that returns $\bar{\mathbf{I}}_v = \mathcal{T}(\mathbf{I}_v, \mathbf{b}_v)$, an image window of fixed size in which the person is centered. As both detection and windowing are performed by neural networks, this process is end-to-end differentiable.

Bidirectional NVS. The simplest way to use the detections described above would be to obtain them in two views $\mathbf{I}_v$ and $\mathbf{I}_{v'}$ and apply the NVS strategy of Section 3.1 to the corresponding windows $\bar{\mathbf{I}}_v$ and $\bar{\mathbf{I}}_{v'}$, that is, aim to approximate $\bar{\mathbf{I}}_{v'}$ as $\mathcal{F}(\bar{\mathbf{I}}_v, \mathbf{R}_{v,v'})$. This, however, would provide very little supervisory signal to the detection process and may result in trivial solutions where the detector focuses on background regions that are easy to match. To prevent this, we propose to reconstruct the *entire* image $\mathbf{I}_{v'}$ instead of just the window $\bar{\mathbf{I}}_{v'}$. This requires mixing the representations of the two views v and v', because generating the entire image $\mathbf{I}_{v'}$ from the window $\bar{\mathbf{I}}_v$ requires knowing the background and where to insert the transformed version of this window. Therefore, we estimate background images and simultaneously approximate $\mathbf{I}_{v'}$ given $\mathbf{I}_v$ and $\mathbf{I}_v$ given $\mathbf{I}_{v'}$.

Formally, given the bounding boxes and spatial transformer introduced above, applying the encoder $\mathcal{E}$ of Section 3.1 to both image windows $\bar{\mathbf{I}}_v = \mathcal{T}(\mathbf{I}_v, \mathbf{b}_v)$ and $\bar{\mathbf{I}}_{v'} = \mathcal{T}(\mathbf{I}'_v, \mathbf{b}_{v'})$ returns the latent representations $[\mathbf{L}^{\text{app}}_v, \mathbf{L}^{\text{3D}}_v]$ and $[\mathbf{L}^{\text{app}}_{v'}, \mathbf{L}^{\text{3D}}_{v'}]$, one per view. We can then invoke the decoder $\mathcal{D}$ of Section 3.1 to reconstruct the entire images as

$$\begin{aligned}\hat{\mathbf{I}}_v &= \mathcal{T}^{-1}(\mathcal{D}(\mathbf{L}^{\text{app}}_{v'}, \mathbf{R}_{v',v}\mathbf{L}^{\text{3D}}_{v'}), \mathbf{b}_v) \ , \qquad (2)\\ \hat{\mathbf{I}}_{v'} &= \mathcal{T}^{-1}(\mathcal{D}(\mathbf{L}^{\text{app}}_{v}, \mathbf{R}_{v,v'}\mathbf{L}^{\text{3D}}_{v}), \mathbf{b}_{v'}) \ .\end{aligned}$$

Intuitively, the reconstruction $\hat{\mathbf{I}}_v$ of view v is obtained by taking the pose seen in v', rotating it to view v, applying the appearance in view v' to it, and reversing the spatial transformation obtained from view v. Equivalently, $\hat{\mathbf{I}}_{v'}$ is reconstructed from v, with the roles of v and v' exchanged. As such, the two reconstructions exchange parts of their decomposition, which creates a bidirectional synthesis.

The final ingredient is to blend in the target view background. To make this easier, we assume the cameras to be static and compute background images $\mathbf{B}_v$ and $\mathbf{B}_{v'}$ by taking the median pixel value across all frames in views v and v', respectively. For each view, we then learn to produce a segmentation mask $\bar{\mathbf{S}}_v$ as an additional output channel of the decoder $\mathcal{D}$. Since this mask corresponds to the image window $\bar{\mathbf{I}}_v$, we apply the inverse spatial transformer to obtain a mask $\hat{\mathbf{S}}_{v'}$ corresponding to the full image. We then use these segmentation masks to blend the reconstructed images $\hat{\mathbf{I}}_v$ and $\hat{\mathbf{I}}_{v'}$ of Eq. 2 with the corresponding backgrounds $\mathbf{B}_v$ and $\mathbf{B}_{v'}$ to produce the final reconstructions

$$\begin{aligned}\mathcal{F}_{\mathbf{I}_v}(\mathbf{I}_{v'}, \mathbf{R}_{v',v}) &= \hat{\mathbf{S}}_v\hat{\mathbf{I}}_v + (1-\hat{\mathbf{S}}_v)\mathbf{B}_v\\ \mathcal{F}_{\mathbf{I}_{v'}}(\mathbf{I}_v, \mathbf{R}_{v,v'}) &= \hat{\mathbf{S}}_{v'}\hat{\mathbf{I}}_{v'} + (1-\hat{\mathbf{S}}_{v'})\mathbf{B}_{v'} \ . \qquad (3)\end{aligned}$$

While our approach to blending is similar in spirit to that of [1], it does not require supervised 2D pose estimation. It also differs from that of [50] where the background composition is formulated as a sum without explicit masks. The generated segmentation masks allows NSD to operate on images with complex background at test time and equips it with a shape abstraction layer.

3.3. NSD with Multiple Subjects

The approach of Section 3.2 assumes that there is a single subject in the field of view. We now extend it to the case where there are a fixed number of $N > 1$ subjects of varying stature. To this end, we first generalize the detector $\mathcal{B}$ to produce N bounding boxes $(\mathbf{b}_{v,i})_{i=1}^{N} = \mathcal{B}(I_v)$, instead of only

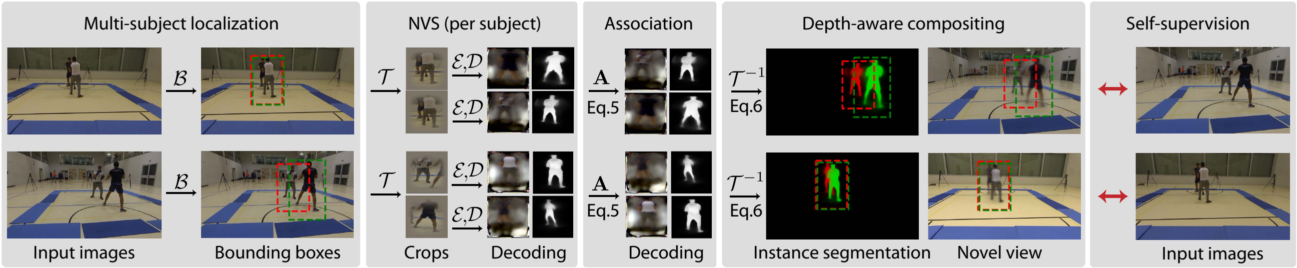


Figure 4. **Multi-person Bi-NVS**. Multiple subjects are detected in each input image and their encoding and decoding is processed separately, akin to single person Bi-NVS. Key is the association of multiple persons across views by Eq. 5 and their composition by Eq. 7.

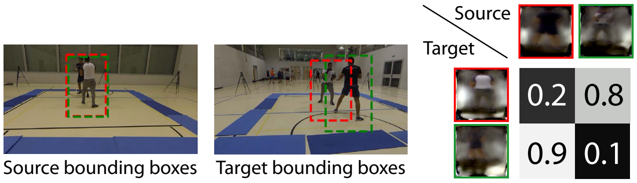


Figure 5. **Identity association.** In this example, the light subject is detected once as the first and once as the second subject, here visualized by red and green boxes. To match subjects across views, we build a similarity matrix from their respective appearance encodings, as shown on the right.

one. NVS is then applied in parallel for each detection as shown in Fig. 4. This yields tuples of latent codes $(\mathbf{L}^{\text{app}}_{v,i})^N_{i=1}$ and $(\mathbf{L}^{\text{3D}}_{v,i})^N_{i=1}$, transformed windows $(\bar{\mathbf{I}}_{v',i})^N_{i=1}$, and corresponding segmentation masks $(\bar{\mathbf{S}}_{v',i})^N_{i=1}$. The only step that is truly modified with respect to the single subject case is the compositing of Eq. 3 that must now account for potential occlusions. This requires the following two extensions.

Appearance-based view association. Objects in the source and target views are detected independently. To implement the bidirectional NVS of Section 3.1, we need to establish correspondences between bounding boxes in views v and v'. Doing so solely on the basis of geometry would result in mismatches due to depth ambiguities. To prevent this, we perform an appearance-based matching. As shown in Fig. 5, it relies on the fact that the appearance latent vector $\mathbf{L}^{\text{app}}_{v,i}$ of object i in view v should be the same same as $\mathbf{L}^{\text{app}}_{v',j}$ in view v' when i and j correspond to the same person in views v and v'. We therefore build the similarity matrix $\mathbf{M}$ whose elements are the cosine distances

$$\mathbf{M}_{j,i} = \frac{\mathbf{L}^{\text{app}}_{v,i} \cdot \mathbf{L}^{\text{app}}_{v',j}}{||\mathbf{L}^{\text{app}}_{v,i}||\ ||\mathbf{L}^{\text{app}}_{v',j}||}, \tag{4}$$

where $\cdot$ is the dot product. In practice, we found that using only the first 16 out of 128 latent variables of the $\mathbf{L}^{\text{app}}_{v,i}$s in this operation to leave room to encode commonalities between different subjects in $\mathbf{L}^{\text{app}}$ for the NVS task while still allowing for distinctive similarity matrices for the purpose of association. Ideally, subject i in view v should be matched to the subject j^* in view v' for which $\mathbf{M}_{j,i}$ is maximized with respect to j. To make this operation differentiable, we apply a row-wise softmax of the scaled similarity matrix $\beta\mathbf{M}$, with $\beta = 10$ to promote sharp distinctions. We use the resulting $N \times N$ association matrix $\mathbf{A}$ to re-order the transformed windows and segmentation masks as

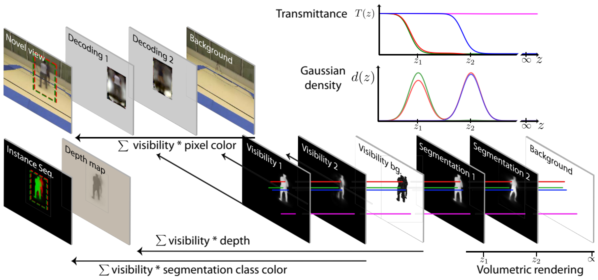


Figure 6. **A visual breakup of Eq. 7.** The novel view is the sum of decoding layers and background, weighted by their respective visibility maps. Similarly, the segmentation mask and depth map is computed from weighted color and depth values. Visibility is computed through volumetric rendering. We plotted the computation for four pixels marked in red, green blue and magenta. Each person forms a translucent layer with Gaussian density in depth direction (see lower plot), so that transmittance decays smoothly at each layer and proportionally to the segmentation mask (top plot).

$$\begin{aligned}(\bar{\mathbf{I}}_{v',j})^N_{j=1} &\leftarrow \mathbf{A}(\bar{\mathbf{I}}_{v',i})^N_{i=1}\ , \\ (\bar{\mathbf{S}}_{v',j})^N_{j=1} &\leftarrow \mathbf{A}(\bar{\mathbf{S}}_{v',i})^N_{i=1}\ .\end{aligned} \tag{5}$$

This weighted permutation is differentiable and, hence, enables end-to-end training.

Reasoning about Depth. After re-ordering the transformed windows and segmentation masks, the reconstructed image for view v can in principle be obtained as

$$\mathcal{F}_{\mathbf{I}_v}((\mathbf{I}_{v',i}), \mathbf{R}_{v',v}) = \left(\sum_{i=1}^{N} \hat{\mathbf{S}}_{v,i}\hat{\mathbf{I}}_{v,i}\right) + \left(1 - \sum_{i=1}^{N} \hat{\mathbf{S}}_{v,i}\right)\mathbf{B}_v\ , \tag{6}$$

where $\hat{\mathbf{I}}_{v,i}$ is the initial reconstruction for view v and subject i, computed independently for each person via Eq. 2. In short, this combines the foreground region of every subject with the overall background. This strategy, however, does not account for overlapping subjects; depth is ignored when computing the intensity of a pixel that is covered by two foreground masks.

To address this, we extend the detector $\mathcal{B}$ to predict a depth value $z_{v,i}$ in addition to the bounding boxes. We then compute a visibility map for each subject based on the depth values of all subjects and their segmentation masks. To this end, we use the occlusion model introduced in [49, 48] that approximates solid surfaces with Gaussian densities to attain differentiability. This model relies on the transmittance to depth z, given in our case by $\mathbf{T}(z) = \exp(-\sum_i \mathbf{S}_{v,i}(\text{erf}(z_{v,i} - z) + 1))$. Given this transmittance, the visibility of subject i is then defined as $\mathbf{T}(z_{v,i})\mathbf{S}_{v,i}$. These visibility maps form the instance segmentation masks, and we obtain depth maps by weighting each by $z_{v,i}$. This process is depicted in Fig. 6. Altogether, this lets us re-write the reconstruction of image $\mathbf{I}_v$ as

$$\mathcal{F}_{\mathbf{I}_v}((\mathbf{I}_{v',i}), \mathbf{R}_{v',v}) = \left(\sum_i \mathbf{T}(z_{v,i})\mathbf{S}_{v,i}\hat{\mathbf{I}}_{v,i} \right) Z + \mathbf{T}(\infty)\mathbf{B}_v \ , \tag{7}$$

where $Z = \frac{1 - \mathbf{T}(\infty)}{\sum_j \mathbf{T}(z_{v,j})\mathbf{S}_{v',i}}$ is a normalization term. More details on this occlusion model are provided in the supplementary material.

If at all, depth order in NVS has been handled through depth maps [63] and by introducing a discrete number of equally spaced depth layers [14], but none of these address the inherent scale ambiguity as done here with Bi-NVS. Closely related is the unsupervised person detection and segmentation method proposed in [2], which localizes and matches persons across views through a grid of candidate positions on the ground plane.

In short, we train a combined detetection-encoding-decoding network to individually detect, order, and model foreground objects, that is, the objects visible from all views and not contained in the static background.

3.4. NSD Training

NSD is trained in a fully self-supervised fashion to carry out Bi-NVS as described in Section 3.2. We perform gradient descent on batches containing pairs of images taken from two or more available views at random. Since no labels for intermediate supervision are available and $\mathcal{B}$, $\mathcal{E}$ and $\mathcal{D}$ are deep neural networks, we found end-to-end training to be unreliable and rely on the following. To counteract, we introduce focal spatial transformers (explained in the supplemental document) and the following priors.

Using Weak Priors. Without guidance, the detector converged to a fixed location on an easy to memorize background patch. To push the optimization process towards exploring detection positions on the whole image, we add a loss term that penalizes the squared deviation of the average bounding box position across a batch from the image center. Note that this is different from penalizing the position of each detection independently, which would lead to a strong bias towards the center. Instead, it assumes a Gaussian prior on the average person position, which is fulfilled not only when the subjects are normally distributed around the center, but, by the central limit theorem, also when they are uniformly distributed. We build independent averages for the N detection windows, which avoids trivial solutions.

Similarly, we introduce a scale prior that encourages the average detection size to be close to 0.4 times the total image size and favors an aspect ratio of 1.5. As for position, this prior is weak and would be fulfilled if sizes vary uniformly from 0.1 to 0.7. Both terms are given a small weight of 0.1 to reduce the introduced bias.

4. Evaluation

In this section, we evaluate NSD for the tasks of multi-people detection, 3D pose estimation, and novel view synthesis. First, we show that, in single-person scenarios, our method delivers similar accuracy compared to existing self-supervised approaches, even though they require ground-truth bounding box annotations whereas we do *not*. Second, we use a boxing scenario that stumps state-of-the-art algorithms to demonstrate that our method can effectively handle closely interacting people. Finally, we provide results on scenes containing three people. Additional scene decomposition and re-composition results are given in the supplementary material.

4.1. Baselines

We refer to our method as **Ours** and compare it against:

- **LCR**-H36M and **LCR**-ITW. They are both versions of **LCR**++ [53], which is the current state of the art in multi-person 3D pose estimation. The first is trained on Human3.6M (H36M) and the second on in-the-wild 2D and 3D datasets.
- **Resnet-I** and **Resnet-$\bar{\mathbf{I}}$**. Two baselines that use a **Resnet** [38], whose architecture is similar to the one we use, to regress directly from the image to the 3D pose. **Resnet-I** runs on the whole image $\mathbf{I}$ whereas **Resnet-$\bar{\mathbf{I}}$** runs on the cropped one $\bar{\mathbf{I}}$ that NSD returns.
- **Auto-encoder**. A baseline that uses the same spatial transformer and encoder-decoder as we do but learns an image auto-encoding instead of NVS in **Ours**.

In the above list, we distinguish between baselines **Resnet-I**, **Resnet-$\bar{\mathbf{I}}$**, and **Auto-encoder** that we implemented ourselves and the recently published method **LCR**. The latter has been discussed in Section 2. We have used publicly available code to run **LCR** on our data.

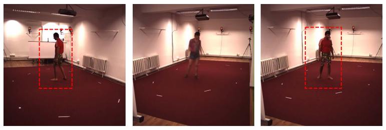


Figure 7. **Novel view synthesis.** The images on the left and right were taken at the same time by two different cameras. The dotted lines denote the NSD bounding box. The image in the middle was synthesized from the image on the left with the subject retaining his original appearance, with shorts instead of long pants, but being shown in the pose of the one on the right.

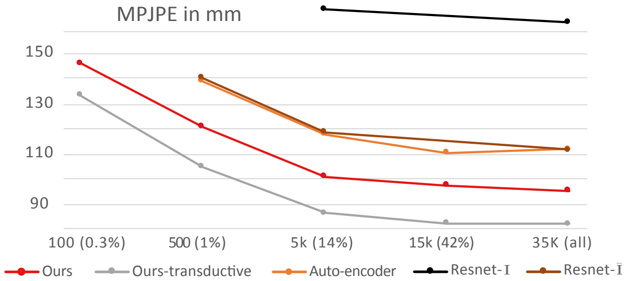


Figure 8. **Accuracy of single-person NSD.** We plot the MPJPE on the PoseTrack validation set as a function of the number of training samples used to train $\mathcal{P}$.

4.2. Supervised Training

Recall from Section 3.4 that we learn our NSD representation in a completely self-supervised way, as shown on the left side of Fig. 2. This being done, we can feed an image to the encoder $\mathcal{E}$ that yields a representation in terms of one or more bounding boxes, along with the corresponding segmentation masks, and latent representations. As our central goal is to demonstrate the usefulness of this representation for 3D pose estimation using comparably little annotated data, we then use varying amounts of such data to train a new network $\mathcal{P}$ that regresses from the representation to the 3D pose. The right side of Fig. 2 depicts this process.

At inference time on an image $\mathbf{I}$, we therefore compute $\mathcal{E}(\mathbf{I})$ and run the decoder $\mathcal{P}$ on each resulting bounding box and corresponding latent representation. Because the learned representation is rich, we can use a simple two-layer fully-connected network for $\mathcal{P}$.

4.3. Single-Person Reconstruction

We test single-person NSD on the PoseTrack2018 challenge of the well known H36M [21] dataset. The images were recorded in a four-camera studio and the task is to estimate 17 3D joint locations relative to the subject's hip. Accuracy is usually measured in terms of the mean per joint position error (MPJPE) expressed in mm. To compare against [51] and [50], we also report the N-MPJPE, that is, the MPJPE after rigidly aligning the prediction to the ground truth in the least squares sense.

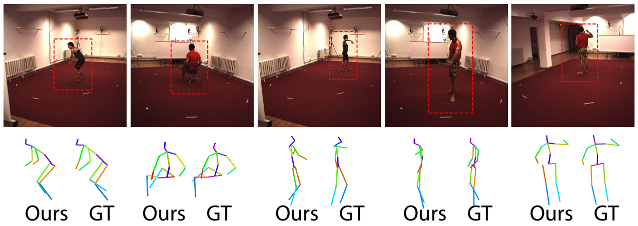


Figure 9. **Pose estimation** using only 15% of the training labels to train $\mathcal{P}$. Top row: Images with detected bounding box. Bottom row: Recovered and ground-truth poses shown side by side.

We learn our NSD representation from the training sequences featuring five different subjects. We evaluate on the validation sequences that feature two different subjects. In Fig. 7, we use one image pair from the validation set to show that NSD localizes and scale normalizes a subject well enough for resynthesis in a different view. We provide additional examples in the supplementary material.

$\mathcal{P}$ is learned on subsets of the complete training set. In Fig. 8, we plot the MPJPE as a function of the amount of labeled training data we used for supervised training, as described in Section 4.2. In practice, the smaller training sets are obtained by regularly sub-sampling the dedicated 35k examples. Direct regression from the full-frame image as in **Resnet** is very inaccurate. Using the NSD bounding boxes as in **Resnet-$\bar{\text{I}}$** and **Auto-encoder** significantly improves performance. Using our complete model further improves accuracy by exploiting the learned high level abstraction. It remains accurate when using as few as 1% of the available labels. Fig. 9 depicts predictions obtained when $\mathcal{P}$ has been trained using less than 15% of the available labels.

Among the semi-supervised methods, **Ours** is more than 15mm more accurate than **Auto-encoder**. The results reported by [51] and [50] are not directly comparable, since their evaluation is on a non-standard training and test sets of H36M and they use ground truth bounding boxes. Nevertheless, their reported N-MPJPE are higher than ours throughout, for example 153.3 and 117.6 for 15k supervised labels while we obtain 91. This confirms that our approach can handle full-frame input without loosing accuracy.

To demonstrate that our approach benefits from additional multi-view data *without* additional annotations, we retrained the encoder $\mathcal{E}$ using not only the training data but also the PoseTrack challenge test data for which the ground-truth poses are not available to us. Furthermore, our approach can also be used in a transductive manner, by additionally incorporating the images used during evaluation without the corresponding annotations at training time. We refer to these two strategies as **Ours**-extended and **Ours**-transductive, respectively. As can be seen in Fig. 7, they both increase accuracy. More specifically, when using only 500 pose labels, the error reduces by 5mm with the former and another 10mm with the latter, as shown in Fig. 10,

Of course, many existing methods attain a higher accu-

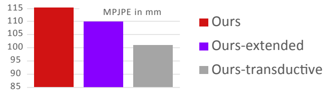


Figure 10. **Varying the unlabeled training set size.** Using more samples improves accuracy, particularly in the transductive case, where examples come from the test distribution.

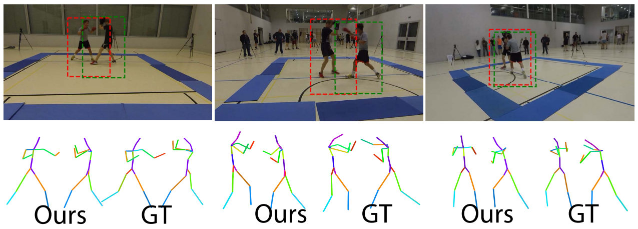


Figure 11. **Estimating the poses of two boxers.** Top row: Images with one detected bounding box per boxer. Bottom row: Recovered and ground-truth poses shown side by side.

Method	MPJPE in mm	NMPJPE in mm	NMPJPE* in mm	Detection rate
Ours	**125.4**	**99.7**	**97.8**	**99.8** %
LCR-ITW	155.6	154.37	122.7	79.7 %
LCR-H36M	240.9	238.5	171.7	37.6 %
Resnet-Ī	196.0	194.8	182.2	98.9 %

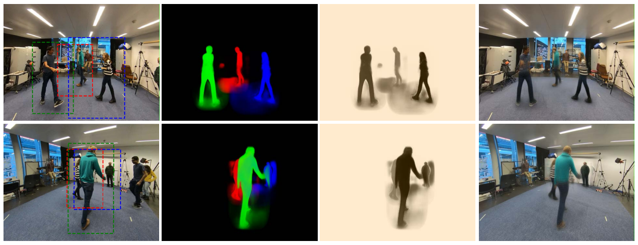
Figure 12. **Accuracy of two-person NSD on the boxing dataset**, as average over all detected persons. NMPJPE* is a version of NMPJPE that accounts for LCR's different skeleton dimensions. It normalize predictions before error computation with the 17×17 linear map that aligns prediction and GT in the least squares sense.

racy than **Ours** by using all the annotated data and adding to it either synthetic data or additional 2D pose datasets for stronger supervision. While legitimate under the PoseTrack challenge rules, it goes against our aim to reduce the required amount of labeling. For example, **LCR**-H36M reports an accuracy of 49.4mm, but this has required creating an additional training dataset of 557,000 synthetic images to supplement the real ones. Without it, the original **LCR** [52] achieves accuracies that are very close to those of **Ours**—ranging from 75.8 to 127.1 depending on the motion—when using full supervision. However, the strength of **Ours** is that its accuracy only decreases very slowly when reducing the amount of annotated data being used.

4.4. Two-Person Reconstruction

To test the performance of NSD when two people are interacting, we introduce a new boxing dataset that comprises 8 sequences with sparring fights between 11 different boxers. We used a semi-automated motion capture system [6] to annotate 6 of these sequences, of which we set aside 4 for supervised training of $\mathcal{P}$ and 2 for testing purposes. We then use the remaining 2 in combination with the annotated training sequences for self-supervised NSD learning.

Fig. 11 depicts 3 different images and the recovered 3D poses for each boxer, which are accurate in spite of the strong occlusions. In Fig. 12, we compare our results to those of **LCR**-H36M, **LCR**-ITW, and **Resnet-Ī**. We clearly outperform all three. While LCR is trained on another dataset, which precludes a direct comparison, this demonstrates the importance of domain specific training and NSD's ability to learn a depth ordering and occlusions.



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

Figure 13. **Three-person NSD training.** (a) The three detected bounding boxes. (b) Segmentation masks. (c) Depth ordering, where darker pixels are closer. (d) Re-synthesized images.

4.5. Multi-Person Reconstruction

Our formalism is designed to handle a pre-defined yet arbitrary number of people. To test this, we captured a 10 minute five-camera sequence featuring 6 people interacting in groups of three and used it to train a 3-people NSD representation, still in a fully self-supervised way. Fig. 13 depict the NSD representation of two images of that sequence, along with the image re-synthesized using it. Note that, in both cases, there are only three people in the re-synthesized image, which makes sense in this case.

5. Conclusion

We have proposed a multi-view self-supervision approach to training a network to produce a hierarchical scene representation that is tailored for 3D human pose capture, yet general enough to be employed for other reconstruction tasks. It includes 3 levels of abstraction, spatial layout (bounding box and relative depth), instance segmentation (masks), and body representation (latent vectors that encode appearance and pose). Given that representation, very little annotated data suffices to train a secondary network to map it to a full 3D pose. The trained network can then operate without being given *a priori* locations for the people. It can compute their poses in parallel, even when they overlap.

In this work, we have limited ourselves to a few people in the scene. It serves well to our primary application domain of sports performance analysis, which demands high accuracy but where the number of athletes is known in advance. In future work, we will extend this to a larger and unknown number of people.

Acknowledgments. This work was supported by the Swiss Innovation Agency and by a Microsoft Joint Research Project.

Appendix

In this document, we supply additional implementation details, provide an ablation study, introduce focal spatial transformers, and explain the differentiable occlusion model in depth.

A. Implementation Details

Staged Training. Multi-person NSD requires to train $\mathcal{B}$, $\mathcal{E}$ and $\mathcal{D}$ in staged fashion. First, we train all three networks without depth reasoning. Second, we re-initialize $\mathcal{E}$ and $\mathcal{D}$ to random values and incorporate the depth output of the detector. In practice, we found that the first stage correctly localizes the subjects but inconsistently matches them across the views. The second stage corrects the person associations. $\mathcal{P}$ is trained in a third stage, keeping $\mathcal{B}$ and $\mathcal{E}$ fixed.

Network architecture and hyperparameter. We use 18- and 50-layer residual networks for $\mathcal{B}$ and $\mathcal{E}$, respectively. They were pre-trained on ImageNet. For $\mathcal{D}$ we employ a U-Net architecture [54] with 32, 64, 128, 256 feature channels in each stage. Following [50], we define $\mathcal{P}$ as a simple fully connected network with two layers of 1024 features and dropout with probability 0.5. All training stages are optimized for 200k iterations with Adam and a learning rate of 1e-3 for $\mathcal{B}$ and $\mathcal{E}$, and of 1e-4 for $\mathcal{D}$ and $\mathcal{P}$. We use an input image resolution of 910px×512px and a batch size of 16 for the boxing dataset, 480px×360px and a batch size of 12 for the three-person dataset, and 500px×500px and a batch size of 32 for H36M. The loss $L(\cdot)$ in Eq. 1 of the main document is implemented as the combination of a simple image loss on the pixel intensity and a perceptual loss on ResNet-ImageNet features. Both losses use the L^2 distance and the perceptual loss is weighted by a factor two. $\mathcal{P}$ is optimized with the mean squared error (MSE) loss.

The input images are whitened and the segmentation masks $\hat{S}$ are normalized to the range [0,1] before foreground-background blending.

Implementation. We use the PyTorch platform for NN training. To deal with the increased memory throughput due to using full-frame images instead of pre-processed crops, we use the NVVL loader [7]. It loads videos in compressed format and decodes them efficiently on the GPU.

Inverse spatial transformers The inverse spatial transformer maps from the small bounding box crop to full-frame. To handle regions without source pixels, we use the Pytorch grid sample function with padding. Zero padding is used for the segmentation and border padding for the decoded image. The resulting partiallyfilled but full-frame images are completed by blending those regions where the segmentation mask is 0 with the background via Eq. 3 and Eq. 6. (main paper).

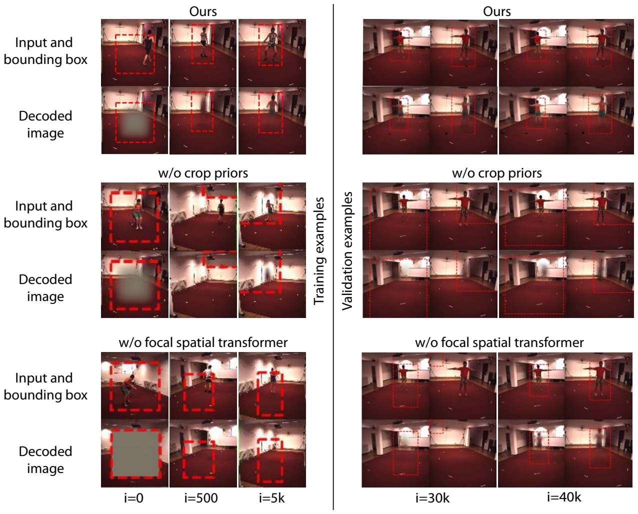


Figure 14. **Training progress** with respect to the number of performed gradient steps. Top row: With our full model the detector learns to localize the persons quickly (left) and Bi-NVS gives already reasonable reconstructions on the validation set after 30k iterations. Central row: Without weak priors, training gets trapped in a local minima. Bottom row: With classical spatial transformers the detector converges much slower and, hence, also the Bi-NVS takes longer to converge and remains blurry after 40k iterations

B. Introducing Focal Spatial Transformers

Spatial transformers [22] make image transformations differentiable. We use them for localizing the subjects and noticed that their unsupervised training slows down convergence. To counteract, we propose to encourage them to focus on the subject, which should be at the center of the crop. We therefore define a smooth mask $\mathbf{G}$, which we model as a bump function $\mathbf{G} = \exp\left(1 - \frac{1}{1-\sqrt{x^4+y^4}}\right)$ with C^∞ smoothness on the compact crop window. We then post-multiply the spatial transformer operation with $\mathbf{G}$, which yields the focal spatial transformer (FST)

$$\tilde{\mathcal{T}}(\mathbf{b}, \mathbf{I}) = \mathcal{T}(\mathbf{b}, \mathbf{I})\mathbf{G}. \tag{8}$$

Conversely, we use a pre-multiplication for the inverse unwarping operation, that is, $\tilde{\mathcal{T}}^{-1}(\mathbf{b}, \bar{\mathbf{I}}) = \mathcal{T}^{-1}(\mathbf{b}, \mathbf{G}\bar{\mathbf{I}})$.

C. Ablation Study

All of our model components contribute to the success of NSD. Not using weak priors leads to divergent training, as shown in the second row of Fig. 14. Convergence is significantly slower without using focal spatial transformers, as

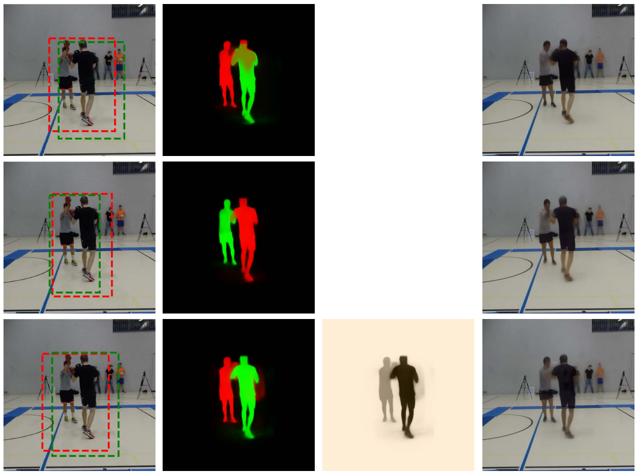


Figure 15. **Ablation study.** Top row: Without NVS task, that means using **Auto-encoder**, partially overlapping subjects merge. Central row: Without depth handling, overlapping parts blend in the segmentation mask. Bottom row: NSD yields clear instance segmentation masks and depth map.

shown in the third row of Fig. 14. Fig. 15 highlights the importance of depth handling and Bi-NVS. The top row shows that simple image encoding separates the subjects roughly, but leads to bleeding between the subject instances. Using Bi-NVS without depth information creates two separate masks, but partial occlusions, in this example at the arms, are not resolved. By contrast, our full model produces clear separation together with depth maps.

Furthermore, we evaluate the influence of using predicted and ground truth (GT) bounding boxes on pose estimation. Unfortunately, the popular Human80K subset of H36M provides only cropped images and the recent PoseTrack challenge only full-frame input without 2D pose or bounding box annotation. To nevertheless compare the algorithm on the same dataset, we use the unofficial protocol from [51]. We train self-supervised on all five training subjects of H36M and supervised on Subject 1 only. Training **Ours** with GT bounding boxes instead of the ones produced by $\mathcal{B}$ leads to a reduction in pose estimation error from 145.3 to 124.4 N-MPJPE. Such a shift is expected, since the bounding box location provides additional albeit unrealistic cues at test time.

D. Differentiable Occlusion Model

Our goal is an occlusion model that is smooth and thereby differentiable in the depth ordering of objects. It should also be physically correct in that a partial occluder is as much visible as it reduces the visibility of the further occluded objects—the visibility of all objects must sum to one.

While occlusion and dis-occlusion of solid objects is generally non-differentiable, these properties can be attained by smoothing the scene to be partially translucent and modeling physical light transport in a participating medium without scattering. In the following, we review the model used in [49, 48] that approximates the scene with a set of Gaussian densities and explain our simplifications to it.

In contrast to previous work that approximated arbitrary scenes consisting of multiple objects by hundreds of Gaussians, we assume that people are sufficiently separated and model each with a single depth plane, see Fig. 6 in the main document. For the sake of smoothness, we make each plane partially translucent with a smooth Gaussian density in the z direction. To model the complex shape of humans, we consider different densities for each pixel. In practice, we use the generated segmentation masks to perform a kind of alpha blending.

We model how light travels in the z direction along a view ray. In the following, we consider a single pixel and apply this model to each of the pixels, with varying opacity for each layer and pixel. Beer-Lambert law states that the transmittance function from a point s to the observer at $-\infty$ in a participating medium decays exponentially with the traversed density, that is,

$$T(s) = \exp\left(-\int_{-\infty}^{s} d(z)dz\right) , \tag{9}$$

with $d(z)$ the density at point z and assuming an orthographic projection with the observer at $-\infty$. Using a Gaussian density with means $(\mu_q)_q$ and $(\sigma_q)_q$ has the advantage that this integral can be written in closed form as

$$\begin{aligned} \int_{-\infty}^{s} d(z) &= \int_{-\infty}^{s} \sum_q c_q G_q(z; \sigma_q, \mu_q) dz \\ &= \int_{-\infty}^{s} \sum_q c_q \exp(-\frac{(z-\mu_q)^2}{2\sigma_q^2}) dz \\ &= \sum_q \frac{\sigma_q c_q \sqrt{\pi}}{\sqrt{2}} \left(\text{erf}\left(\frac{s-\mu_q}{\sqrt{2}\sigma_q}\right) + 1\right) , \end{aligned} \tag{10}$$

in terms of the error function $\text{erf}(s) = \frac{2}{\pi} \int_0^s \exp(-z^2)dz$.

In our case, we simplify this equation by assuming fixed Gaussian widths $\sigma = \frac{1}{\sqrt{2}}$ and magnitude $c_q = \frac{c'_q 2}{\sqrt{\pi}}$. This yields

$$\begin{aligned} T(s) &= \exp\left(-\sum_q \frac{c_q\sqrt{\pi}}{2} \left(\text{erf}(s-\mu_q)+1\right)\right) \\ &= \exp\left(-\sum_q c'_q \left(\text{erf}(s-\mu_q)+1\right)\right) . \end{aligned} \tag{11}$$

In this model, an object occludes as much as it is visible. The object's visibility $V(s)$ at position s is proportional to the transmittance and the density of the object at s, that is,

$$V_q(s) = d(s)T(s) \ . \tag{12}$$

For the background, which is assumed to be ∞ distant, we use the simplified model of [48] expressed as

$$\begin{aligned} T(\infty) &= \exp(-\sum_q c'_q \left(\operatorname{erf}(\infty - \mu_q) + 1\right)) \\ &= \exp(-\sum_q c'_q \left(2\right)) \ . \end{aligned} \tag{13}$$

and the visibility of the background plane is equal to that remaining fraction $T(\infty)$. The individual depth planes have been diffused in z direction. To capture the entire visibility of one in relation to the other potentially intersecting depth layers, one has to integrate the point-wise visibility of the diffused density

$$V_q = \int_{-\infty}^{\infty} V_q(s) ds \ . \tag{14}$$

This integral cannot be computed in terms of simple functions and was approximated by regular sampling in [49]. Here we approximate it with a single sample at the Gaussian's position μ_q, that is,

$$\begin{aligned} V_q &\approx V_q(s) S_q \frac{2}{\sqrt{pi}} ds \\ &\propto V_q(s) S_q \ . \end{aligned} \tag{15}$$

The 'lost' energy due to this drastic approximation can be inferred from $1-T(\infty)$, for which we have an analytic solution. Assuming that the error is equally distributed across all Gaussians, we simply re-weight the visibility of each Gaussian by

$$Z = \frac{1 - \mathbf{T}(\infty)}{\sum_j \mathbf{T}(z_{v,j}) \mathbf{S}_{v',i}} \ , \tag{16}$$

so that their sum with the background visibility is exactly one. These simplifications ensure computational efficiency while maintaining smoothness and differentiability.

References

[1] G. Balakrishnan, A. Zhao, A. Dalca, F. Durand, and J. Guttag. Synthesizing Images of Humans in Unseen Poses. In *Conference on Computer Vision and Pattern Recognition*, 2018. 3, 4

[2] P. Baqué, F. Fleuret, and P. Fua. Deep Occlusion Reasoning for Multi-Camera Multi-Target Detection. In *International Conference on Computer Vision*, 2017. 6

[3] A. Bas, P. Huber, W. Smith, M. Awais, and J. Kittler. 3D Morphable Models as Spatial Transformer Networks. *arXiv Preprint*, 2017. 2

[4] Y. Bengio, A. Courville, and P. Vincent. Representation Learning: A Review and New Perspectives. *ArXiv e-prints*, 2012. 2

[5] M. Burenius, J. Sullivan, and S. Carlsson. 3D Pictorial Structures for Multiple View Articulated Pose Estimation. In *Conference on Computer Vision and Pattern Recognition*, 2013. 2

[6] The Captury. http://thecaptury.com. 8

[7] J. Casper, J. Barker, and B. Catanzaro. Nvvl: Nvidia video loader. https://github.com/NVIDIA/nvvl, 2018. 9

[8] W. Chen, H. Wang, Y. Li, H. Su, Z. Wang, C. Tu, D. Lischinski, D. Cohen-or, and B. Chen. Synthesizing Training Images for Boosting Human 3D Pose Estimation. In *3DV*, 2016. 2

[9] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. Infogan: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, pages 2172–2180, 2016. 2

[10] T. Cohen and M. Welling. Transformation Properties of Learned Visual Representations. *arXiv Preprint*, 2014. 3

[11] A. Dosovitskiy, J. Springenberg, and T. Brox. Learning to Generate Chairs with Convolutional Neural Networks. In *Conference on Computer Vision and Pattern Recognition*, 2015. 2

[12] A. Dosovitskiy, J. Springenberg, M. Tatarchenko, and T. Brox. Learning to Generate Chairs, Tables and Cars with Convolutional Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):692–705, 2017. 2

[13] A. Eslami, D. Rezende, F. Besse, F. Viola, A. Morcos, M. Garnelo, A. Ruderman, A. Rusu, I. Danihelka, K. Gregor, D. Reichert, L. Buesing, T. Weber, O. Vinyals, D. Rosenbaum, N. Rabinowitz, H. King, C. Hillier, M. Botvinick, D. Wierstra, K. Kavukcuoglu, and D. Hassabis. Neural Scene Representation and Rendering. *Science*, 360(6394):1204–1210, 2018. 2, 3

[14] J. Flynn, I. Neulander, J. Philbin, and N. Snavely. Deepstereo: Learning to Predict New Views from the World's Imagery. In *Conference on Computer Vision and Pattern Recognition*, pages 5515–5524, 2016. 3, 6

[15] M. Gadelha, S. Maji, and R. Wang. 3D Shape Induction from 2D Views of Multiple Objects. *arXiv preprint arXiv:1612.05872*, 2016. 2

[16] R. Garg, G. Carneiro, and I. Reid. Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue. In *European Conference on Computer Vision*, pages 740–756, 2016. 2

[17] E. Grant, P. Kohli, and van M. Gerven. Deep Disentangled Representations for Volumetric Reconstruction. In *European Conference on Computer Vision*, pages 266–279, 2016. 2

[18] S. Günel, H. Rhodin, and P. Fua. What Face and Body Shapes Can Tell About Height. *arXiv*, 2018. 3

[19] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. Beta-Vae: Learning Basic Visual Concepts with a Constrained Variational Framework. 2016. 2

[20] G. Hinton, A. Krizhevsky, and S. Wang. Transforming Auto-Encoders. In *International Conference on Artificial Neural Networks*, pages 44–51, 2011. 3

[21] C. Ionescu, J. Carreira, and C. Sminchisescu. Iterated Second-Order Label Sensitive Pooling for 3D Human Pose Estimation. In *Conference on Computer Vision and Pattern Recognition*, 2014. 1, 2, 7

[22] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial Transformer Networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015. 9

[23] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh. Panoptic Studio: A Massively Multiview System for Social Motion Capture. In *International Conference on Computer Vision*, 2015. 2

[24] A. Kanazawa, M. Black, D. Jacobs, and J. Malik. End-To-End Recovery of Human Shape and Pose. In *Conference on Computer Vision and Pattern Recognition*, 2018. 2

[25] A. Kar, C. Häne, and J. Malik. Learning a Multi-View Stereo Machine. In *Advances in Neural Information Processing Systems*, pages 364–375, 2017. 2

[26] H. Kim, P. Garrido, A. Tewari, W. Xu, J. Thies, M. Nießner, P. Pérez, C. Richardt, M. Zollhöfer, and C. Theobalt. Deep Video Portraits. *arXiv preprint arXiv:1805.11714*, 2018. 3

[27] H. Kim, M. Zollhöfer, A. Tewari, J. Thies, C. Richardt, and C. Theobalt. Inversefacenet: Deep Single-Shot Inverse Face Rendering from a Single Image. *arXiv Preprint*, 2017. 2

[28] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, pages 1106–1114, 2012. 1

[29] Y. Kudo, K. Ogaki, Y. Matsui, and Y. Odagiri. Unsupervised adversarial learning of 3d human pose from 2d joint locations. *CoRR*, abs/1803.08244, 2018. 2

[30] T. D. Kulkarni, W. Whitney, P. Kohli, and J. B. Tenenbaum. Deep Convolutional Inverse Graphics Network. In *arXiv*, 2015. 2

[31] A. Kundu, Y. Li, and J. Rehg. 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In *Conference on Computer Vision and Pattern Recognition*, pages 3559–3568, 2018. 2, 3

[32] C. Lassner, G. Pons-Moll, and P. Gehler. A Generative Model of People in Clothing. *arXiv Preprint*, 2017. 3

[33] C. Lassner, J. Romero, M. Kiefel, F. Bogo, M. Black, and P. Gehler. Unite the People: Closing the Loop Between 3D and 2D Human Representations. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[34] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. V. Gool. Pose Guided Person Image Generation. In *Advances in Neural Information Processing Systems*, pages 405–415, 2017. 3

[35] J. Martinez, R. Hossain, J. Romero, and J. Little. A Simple Yet Effective Baseline for 3D Human Pose Estimation. In *International Conference on Computer Vision*, 2017. 2

[36] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt. Monocular 3D Human Pose Estimation in the Wild Using Improved CNN Supervision. In *International Conference on 3D Vision*, 2017. 2

[37] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, S. Sridhar, G. Pons-moll, and C. Theobalt. Single-Shot Multi-Person 3D Pose Estimation from Monocular RGB. In *3DV*, 2018. 2

[38] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H. Seidel, W. Xu, D. Casas, and C. Theobalt. Vnect: Real-Time 3D Human Pose Estimation with a Single RGB Camera. In *ACM SIGGRAPH*, 2017. 2, 6

[39] F. Moreno-noguer. 3D Human Pose Estimation from a Single Image via Distance Matrix Regression. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[40] E. Park, J. Yang, E. Yumer, D. Ceylan, and A. Berg. Transformation-Grounded Image Generation Network for Novel 3D View Synthesis. In *Conference on Computer Vision and Pattern Recognition*, pages 702–711, 2017. 3

[41] G. Pavlakos, X. Zhou, and K. Daniilidis. Ordinal Depth Supervision for 3D Human Pose Estimation. *Conference on Computer Vision and Pattern Recognition*, 2018. 2

[42] G. Pavlakos, X. Zhou, K. Derpanis, G. Konstantinos, and K. Daniilidis. Coarse-To-Fine Volumetric Prediction for Single-Image 3D Human Pose. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[43] G. Pavlakos, X. Zhou, K. D. G. Konstantinos, and D. Kostas. Harvesting Multiple Views for Marker-Less 3D Human Pose Annotations. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[44] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis. Learning to Estimate 3D Human Pose and Shape from a Single Color Image. In *Conference on Computer Vision and Pattern Recognition*, 2018. 2

[45] G. Pons-Moll, D. F. Fleet, and B. Rosenhahn. Posebits for Monocular Human Pose Estimation. In *Conference on Computer Vision and Pattern Recognition*, 2014. 2

[46] A.-I. Popa, M. Zanfir, and C. Sminchisescu. Deep Multitask Architecture for Integrated 2D and 3D Human Sensing. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[47] D. Rezende, S. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess. Unsupervised Learning of 3D Structure from Images. In *Advances in Neural Information Processing Systems*, pages 4996–5004, 2016. 2

[48] H. Rhodin, N. Robertini, D. Casas, C. Richardt, H.-P. Seidel, and C. Theobalt. General Automatic Human Shape and Motion Capture Using Volumetric Contour Cues. In *European Conference on Computer Vision*, 2016. 6, 10, 11

[49] H. Rhodin, N. Robertini, C. Richardt, H.-P. Seidel, and C. Theobalt. A Versatile Scene Model with Differentiable Visibility Applied to Generative Pose Estimation. In *International Conference on Computer Vision*, December 2015. 6, 10, 11

[50] H. Rhodin, M. Salzmann, and P. Fua. Unsupervised Geometry-Aware Representation for 3D Human Pose Estimation. In *European Conference on Computer Vision*, 2018. 2, 3, 4, 7, 9

[51] H. Rhodin, J. Spoerri, I. Katircioglu, V. Constantin, F. Meyer, E. Moeller, M. Salzmann, and P. Fua. Learning Monocular 3D Human Pose Estimation from Multi-View Images. In *Conference on Computer Vision and Pattern Recognition*, 2018. 2, 7, 10

[52] G. Rogez, P. Weinzaepfel, and C. Schmid. Lcr-Net: Localization-Classification-Regression for Human Pose. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2, 8

[53] G. Rogez, P. Weinzaepfel, and C. Schmid. Lcr-Net++: Multi-Person 2D and 3D Pose Detection in Natural Images. In *arXiv preprint arXiv:1803.00455*, 2018. 2, 6

[54] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Conference on Medical Image Computing and Computer Assisted Intervention*, pages 234–241, 2015. 9

[55] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras. Neural Face Editing with Intrinsic Image Disentangling. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[56] C. Si, W. Wang, L. Wang, and T. Tan. Multistage Adversarial Losses for Pose-Based Human Image Synthesis. In *Conference on Computer Vision and Pattern Recognition*, June 2018. 3

[57] L. Sigal and M. Black. Humaneva: Synchronized Video and Motion Capture Dataset for Evaluation of Articulated Human Motion. Technical report, Department of Computer Science, Brown University, 2006. 2

[58] T. Simon, H. Joo, I. Matthews, and Y. Sheikh. Hand Keypoint Detection in Single Images Using Multiview Bootstrapping. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[59] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*, 2015. 1

[60] A. Spurr, J. Song, S. Park, and O. Hilliges. Cross-Modal Deep Variational Hand Pose Estimation. In *Conference on Computer Vision and Pattern Recognition*, pages 89–98, 2018. 2

[61] X. Sun, J. Shang, S. Liang, and Y. Wei. Compositional Human Pose Regression. In *International Conference on Computer Vision*, 2017. 2

[62] J. Tan, I. Budvytis, and R. Cipolla. Indirect Deep Structured Learning for 3D Human Body Shape and Pose Prediction. In *British Machine Vision Conference*, 2017. 2

[63] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Single-View to Multi-View: Reconstructing Unseen Views with a Convolutional Network. *CoRR abs/1511.06702*, 1:2, 2015. 3, 6

[64] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Multi-View 3D Models from Single Images with a Convolutional Network. In *European Conference on Computer Vision*, pages 322–337, 2016. 3

[65] B. Tekin, P. Marquez-neila, M. Salzmann, and P. Fua. Learning to Fuse 2D and 3D Image Cues for Monocular Body Pose Estimation. In *International Conference on Computer Vision*, 2017. 2

[66] A. Tewari, M. Zollhöfer, H. Kim, P. Garrido, F. Bernard, P. Pérez, and C. Theobalt. MoFA: Model-Based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction. *International Conference on Computer Vision*, 2017. 2

[67] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised Learning of Object Frames by Dense Equivariant Image Labelling. In *Advances in Neural Information Processing Systems*, pages 844–855, 2017. 2

[68] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised Learning of Object Landmarks by Factorized Spatial Embeddings. In *International Conference on Computer Vision*, 2017. 2

[69] D. Tome, C. Russell, and L. Agapito. Lifting from the Deep: Convolutional 3D Pose Estimation from a Single Image. In *arXiv preprint, arXiv:1701.00295*, 2017. 2

[70] L. Tran, X. Yin, and X. Liu. Disentangled Representation Learning Gan for Pose-Invariant Face Recognition. In *Conference on Computer Vision and Pattern Recognition*, page 7, 2017. 2

[71] S. Tulsiani, A. Efros, and J. Malik. Multi-View Consistency as Supervisory Signal for Learning Shape and Pose Prediction. *arXiv Preprint*, 2018. 2

[72] S. Tulsiani, S. Gupta, D. Fouhey, A. Efros, and J. Malik. Factoring shape, pose, and layout from the 2d image of a 3d scene. In *CVPR*, pages 302–310, 2018. 3

[73] S. Tulsiani, T. Zhou, A. Efros, and J. Malik. Multi-View Supervision for Single-View Reconstruction via Differentiable Ray Consistency. In *Conference on Computer Vision and Pattern Recognition*, page 3, 2017. 2

[74] H.-Y. Tung, A. Harley, W. Seto, and K. Fragkiadaki. Adversarial Inverse Graphics Networks: Learning 2D-To-3D Lifting and Image-To-Image Translation from Unpaired Supervision. In *International Conference on Computer Vision*, 2017. 2

[75] H.-Y. Tung, H.-W. Tung, E. Yumer, and K. Fragkiadaki. Self-Supervised Learning of Motion Capture. In *Advances in Neural Information Processing Systems*, pages 5242–5252, 2017. 2

[76] G. Varol, D. Ceylan, B. Russell, J. Yang, E. Yumer, I. Laptev, and C. Schmid. BodyNet: Volumetric Inference of 3D Human Body Shapes. In *European Conference on Computer Vision*, 2018. 2

[77] G. Varol, J. Romero, X. Martin, N. Mahmood, M. Black, I. Laptev, and C. Schmid. Learning from Synthetic Humans. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[78] T. von Marcard, R. Henschel, M. Black, B. Rosenhahn, and G. Pons-Moll. Recovering Accurate 3D Human Pose in the Wild Using IMUs and a Moving Camera. In *European Conference on Computer Vision*, 2018. 2

[79] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro. Video-To-Video Synthesis. In *Advances in Neural Information Processing Systems*, 2018. 3

[80] D. Worrall, S. Garbin, D. Turmukhambetov, and G. Brostow. Interpretable Transformations with Encoder-Decoder Networks. In *International Conference on Computer Vision*, 2017. 3

[81] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee. Perspective Transformer Nets: Learning Single-View 3D Object Reconstruction Without 3D Supervision. In *Advances in Neural Information Processing Systems*, pages 1696–1704, 2016. 2

[82] J. Yang, S. Reed, M.-H. Yang, and H. Lee. Weakly-Supervised Disentangling with Recurrent Transformations for 3D View Synthesis. In *Advances in Neural Information Processing Systems*, pages 1099–1107, 2015. 3

[83] W. Yang, W. Ouyang, X. Wang, J. Ren, H. Li, and X. Wang. 3D Human Pose Estimation in the Wild by Adversarial Learning. *Conference on Computer Vision and Pattern Recognition*, 2018. 2

[84] A. R. Zamir, T. Wekel, P. Agrawal, J. Malik, and S. Savarese. Generic 3D Representation via Pose Estimation and Matching. In *European Conference on Computer Vision*, 2016. 3

[85] A. Zanfir, E. Marinoiu, and C. Sminchisescu. Monocular 3D Pose and Shape Estimation of Multiple People in Natural Scenes - the Importance of Multiple Scene Constraints. In *Conference on Computer Vision and Pattern Recognition*, June 2018. 2

[86] M. Zanfir, A.-I. Popa, A. Zanfir, and C. Sminchisescu. Human Appearance Transfer. In *Conference on Computer Vision and Pattern Recognition*, pages 5391–5399, 2018. 3

[87] Y. Zhang, Y. Guo, Y. Jin, Y. Luo, Z. He, and H. Lee. Unsupervised Discovery of Object Landmarks as Structural Representations. In *Conference on Computer Vision and Pattern Recognition*, pages 2694–2703, 2018. 2

[88] B. Zhao, X. Wu, Z.-Q. Cheng, H. Liu, and J. Feng. Multi-View Image Generation from a Single-View. *arXiv preprint arXiv:1704.04886*, 2017. 3

[89] T. Zhou, M. Brown, N. Snavely, and D. Lowe. Unsupervised Learning of Depth and Ego-Motion from Video. In *Conference on Computer Vision and Pattern Recognition*, 2017. 2

[90] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. Efros. View Synthesis by Appearance Flow. In *European Conference on Computer Vision*, pages 286–301, 2016. 3

[91] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. We. Weakly-Supervised Transfer for 3D Human Pose Estimation in the Wild. *arXiv Preprint*, 2017. 2

[92] H. Zhu, H. Su, P. Wang, X. Cao, and R. Yang. View extrapolation of human body from a single image. In *Conference on Computer Vision and Pattern Recognition*, pages 4450–4459, 2018. 3

[93] R. Zhu, H. Galoogahi, C. Wang, and S. Lucey. Rethinking reprojection: Closing the loop for pose-aware shape reconstruction from a single image. *International Conference on Computer Vision*, pages 57–65, 2017. 2