Unsupervised 3D Keypoint Discovery with Multi-View Geometry

Sina Honari\textsuperscript{1} \hspace{1em} Chen Zhao\textsuperscript{2} \hspace{1em} Mathieu Salzmann\textsuperscript{2} \hspace{1em} Pascal Fua\textsuperscript{2}

\textsuperscript{1} Samsung AI Center Toronto \hspace{1em} \textsuperscript{2} Computer Vision Lab, EPFL, Switzerland

Abstract

Analyzing and training 3D body posture models depend heavily on the availability of joint labels that are commonly acquired through laborious manual annotation of body joints or via marker-based joint localization using carefully curated markers and capturing systems. However, such annotations are not always available, especially for people performing unusual activities. In this paper, we propose an algorithm that learns to discover 3D keypoints on human bodies from multiple-view images without any supervision or labels other than the constraints multi-view geometry provides. To ensure that the discovered 3D keypoints are meaningful, they are re-projected to each view to estimate the person’s mask that the model itself has initially estimated without supervision. Our approach discovers more interpretable and accurate 3D keypoints compared to other state-of-the-art unsupervised approaches on Human3.6M and MPI-INF-3DHP benchmark datasets.

1. Introduction

Human body joint annotations have been commonly leveraged to analyze body posture and kinematics. The 3D labels are obtained either through marker-based motion capturing \cite{17, 39}, wearable inertial-based tracking \cite{29, 31}, or marker-less capturing systems \cite{8, 24, 33}. These systems require considerable time, financial resources, manual labor, and sometimes specific capturing setup and clothes to obtain labels. Hence, there has been growing interest in unsupervised methods \cite{1, 2, 12, 13, 15, 17, 19}, which alleviate or reduce dependence on annotations. One way to achieve this is to discover unsupervised keypoints, distinct from pose labels, whose location and correlation are learned without any supervision by the model. One advantage of such a system is the discovery of meaningful 3D keypoints without any annotations. Another advantage is the reduction of the labeled annotations when the final goal is to map the discovered keypoints to the target pose of interest such as joint locations. In this line of research, some self-supervised methods learn to find 2D keypoints \cite{20, 21, 30, 38, 42, 43, 47, 49} but do not propose a way to extend them to 3D.

Figure 1. Multi-View Geometry for Unsupervised 3D Keypoint Discovery. Our approach finds unsupervised 2D keypoints in each view and then uses multi-view geometry to construct 3D keypoints. These discovered keypoints (observed above), whose location is learned without any supervision, can be later mapped to the final pose of interest (e.g. the joint locations).

In this work, we aim to learn in a completely unsupervised fashion to discover 3D keypoints in multi-view image setups, such as the one depicted in Fig. 1. We assume the cameras to be calibrated and have access to a rough estimate of the background image of the scene but do not require 2D or 3D annotations, mask labels, or pre-trained models. Some approaches \cite{3, 34, 41} also extract unsupervised 3D keypoints from images. However, they have only been demonstrated on rigid toy datasets with controllable dynamics and limited variations. By contrast, we aim at real datasets, featuring people performing diverse activities.

In this paper, given a multi-view setup, our model predicts the masks of the foreground subject in each view and then uses multi-view geometry to construct 3D keypoints. These discovered keypoints (observed above), whose location is learned without any supervision, can be later mapped to the final pose of interest (e.g. the joint locations).
in each view to reconstruct the subject’s mask that the model has itself predicted. The training is completely unsupervised. No labels such as 2D or 3D annotations or masks are used during training.

Our complete pipeline is depicted by Figure 2. This framework guarantees that the discovered 3D keypoints satisfy multi-view projection properties while correlating strongly with the 3D posture of the subject. The 3D keypoints can then easily be mapped to the target 3D pose using a simple linear or a multi-layer perceptron network. In short, we propose a real-world unsupervised 3D keypoints discovery method that can be applied directly to uncurated images. We show that our approach outperforms state-of-the-art unsupervised learning methods on Human3.6M and MPI-INF-3DHP benchmark datasets.

2. Related Work

The objective of unsupervised learning models [1, 2, 12, 13, 15, 37] is to pretrain neural networks to be later fine-tuned on the tasks of interest or to train models to extract salient features. These features are then used for downstream applications with a small amount of data using a simple mapping, such as linear adaptation [1, 2, 12, 13], or a two hidden layer multi-layer perceptron (MLP) [15, 35, 37] among others. In this section, we briefly review unsupervised models for keypoint detection and pose estimation. In this paper, we make a distinction between keypoint, found by a model in an unsupervised manner, and pose, corresponding to dataset labels such as joint locations.

Unsupervised 2D Keypoint Discovery. There have been multiple approaches [20, 30, 38, 42, 43, 47, 49] that propose unsupervised 2D keypoint discovery. However, the extension of these models to 3D keypoint estimation has not been explored. Moreover, these methods usually rely on center-cropped subject images, while our approach can directly work on uncurated images.

Unsupervised 3D Pose Models. On the other hand, some approaches learn unsupervised 3D poses from 2D poses using cross-views re-projection and matching [6, 18, 25, 28, 36, 45], or temporal consistency [16, 44, 48]. Some papers instead learn a prior over the 2D pose distribution and enforce consistency across single frame lifting, rotation, and projection [4, 46]. However, all of these approaches rely on either 2D labels or pre-trained 2D pose estimation models, which still depend on a considerable number of annotations. In this paper, we aim at finding unsupervised keypoints directly from images, without relying on any labels or pre-trained models.

Unsupervised Pose-Relevant Features. The works in [7, 15, 35, 37] apply unsupervised learning directly to input images to extract pose-relevant information. In particular, the approaches in [7, 15] apply temporal learning to extract time-variant information, while the approaches in [35, 37] leverage multi-view geometry by applying latent features rotation from one view to another. These latent features cannot be visualized as keypoints, as they live in uninterpretable latent layers. Contrary to these approaches, we apply a full-projection matrix including both intrinsics and the translation component of the extrinsics. This allows directly projecting the estimated 3D keypoints back to the images.
using complete projection matrices, which contributes to further structure into the latent space, hence yielding interpretable 3D keypoints that can be visualized and correlate strongly with the subject’s pose. In practice, this intrinsic structure contributes to higher pose accuracy.

Kundu et al. [26, 27] also propose unsupervised 3D pose estimation models through pose and shape disentanglement and a spatial transformation map [27] or a part-based body model [26]. The proposed framework requires access to a known shape model and body kinematic priors. Contrary to these approaches, we do not resort to such priors and learn directly from images with a much simpler pipeline.

**Unsupervised 3D Keypoint Discovery.** Some approaches extract unsupervised 3D keypoints from 3D point clouds [5, 9] or a 3D shape [22]. Contrary to these approaches, we aim at extracting unsupervised 3D keypoints from only input images. On the other hand, the approaches in [3, 34, 41] learn 3D keypoints from input images using a multi-view learning framework. In particular, Chen et al. [3] learn latent 3D keypoints for control in reinforcement-learning problems. Noguchi et al. [34] estimate keypoints by applying a volumetric rendering using signed-distance-functions to reconstruct all views. This creates a complex setup that requires at least 5 cameras for training. Our model, on the other hand, is much simpler to train, with fewer loss terms, and can even train with 2 cameras.

Closest to our approach is [41], which learns unsupervised 3D keypoints on rigid objects. While this approach also leverages multi-view geometry, similar to [3, 34] it has been only evaluated on toy datasets, with the possibility of rendering many views and limited diversity in appearance, pose, and depth values. As we will show later, this approach is prone to limitations when dealing with real-world data. Moreover, none of these approaches handle foreground subject extraction, as they use toy datasets with no or simple backgrounds, with [34, 41] requiring ground-truth masks. Sun et al. [40] also propose an unsupervised spatio-temporal multi-view keypoint discovery for real-world data. The proposed model builds a 3D volume and applies multiple projections and re-projection in-between 2D and 3D spaces before constructing an edge graph on keypoints to constrain the joint lengths. Our approach has a much simpler pipeline and does not require temporal information to discover its keypoints, while obtaining higher accuracy in practice.

**3. Method**

Our goal is to train a network in an unsupervised fashion to discover 3D keypoints in multi-view calibrated images. To ensure the keypoints are meaningful, we need to guarantee they encode information about only the foreground subject rather than the background. To this end, the model should first detect the subject and then encode the foreground information. This is achieved by reconstructing the input image through the prediction of a mask that separates the foreground from the background, hence allowing to encode only the foreground information.

Our network first encodes the foreground subject and uses the resulting image features to discover potential 2D keypoints in each view. The 2D keypoints from different views are then triangulated using calibration data to create potential 3D keypoints expressed in world coordinates. These 3D keypoints are the features of interest in our model. Without any backward loop, there is no guarantee that they are meaningful since there are no keypoint labels. Hence, they are re-projected to each view to obtain 2D view-dependent keypoints, which are then passed to a mask decoder that reconstructs the foreground subject mask. This process is depicted in Figure 2. In the following sections, we describe the pipeline components in more detail.

---

**3.1. View-Dependent Feature Extraction**

In this section, we describe how detection, encoding, and decoding components are used to extract view-dependent information and the foreground mask. Given an input image $I$, a spatial transformer network (STN) [19] $S$ is used to extract four parameters; two of which specify the scale $s^x$, $s^y$ and the other two the center of the bounding box $u^x$, $u^y$, yielding $S(I) = (s^x, s^y, u^x, u^y)$. A patch $p$ is then cropped using the bounding box coordinates and is then passed to an encoder $E$ yielding $E(p) = \Xi$.

A decoder $D$ then takes the encoded features and outputs an RGB image patch $D_p$ together with a foreground mask patch $M_p$, which can be written as

$$D(\Xi) = (D_p, M_p).$$

Finally, the inverse operation of the spatial transformer network (STN) is applied to put the patches back into the full-image resolution, yielding $S^{-1}(D_p, M_p) = (D, M)$. Decoded image $D$ is then merged with the background image $B$ using the predicted mask $M$ to reconstruct the input image $I$. This operation is equal to

$$I = M \times D + (1 - M) \times B,$$

where $\times$ indicates Hadamard product. This process is depicted in Figure 2. The input image reconstruction is crucial to capture information only about the foreground subject into the latent features $\Xi$ in addition to estimating the foreground subject mask $M_p$. Note that all components including foreground subject detection and mask prediction are unsupervised. No bounding box or mask labels are used. They are all trained through input image reconstruction.

---

1The background image is obtained by taking a per-pixel median of every frame in the video sequence.
3.2. Unsupervised 3D Keypoint Discovery

The latent information in each view $\Xi^\theta$, $\theta \in \{1, \ldots, V\}$, with $V$ being the total number of views, is passed to a 2D keypoint encoder $\psi$ that outputs $N$ channels, one for each potential keypoint $n$. This yields

$$\psi(\Xi) = \{C_n : n = 1, \ldots, N\},$$

where we refer to the set $\{C_n\}_{n=1}^N$ by $C$. Soft-argmax operation [14] is then applied to each channel $C_n$ to obtain a potential 2D keypoint $\mathbf{x}_n = (u, v)_n$. This can be written as

$$\begin{bmatrix} u_n \\ v_n \end{bmatrix} = \sum_{i,j} \text{softmax}(C_n)_{i,j} \begin{bmatrix} i \\ j \end{bmatrix},$$

where the standard softmax operation is applied to each channel $C_n$ to obtain a probability distribution, with $i, j$ indicating the row and column coordinates of the channel. In summary, this process obtains a weighted average of the coordinates, where the weights are given by the softmax probability map.

Given an intrinsic matrix $K_I$ specifying the projection from the camera coordinates to the pixel coordinates on image $I$ and an extrinsic matrix $E$ giving the change of coordinate system from camera to world coordinates, the camera projection matrix on the full image resolution can be written as $\Pi_I = K_I E$.

Having the scale parameter $(s^x, s^y)$ and the top-left coordinates $(b^x, b^y)$ of the bounding box estimated by the spatial transformer network $\mathcal{S}$ in Section 3.1, the intrinsic matrix of the full-resolution image

$$K_I = \begin{bmatrix} f^x & 0 & c^x \\ 0 & f^y & c^y \\ 0 & 0 & 1 \end{bmatrix},$$

is updated to correspond to the detected patch $p$ by using

$$K_p = \begin{bmatrix} s^x f^x & 0 & s^x(c^x - b^x) \\ 0 & s^y f^y & s^y(c^y - b^y) \\ 0 & 0 & 1 \end{bmatrix}.$$

This yields the updated projection matrix of the detected patch $\Pi_p = K_p E$.

For each keypoint $n \in \{1, \ldots, N\}$, its corresponding 2D locations $\{\mathbf{x}_n^\theta\}_{\theta=1}^{\nu}$ and projection matrices $\{\Pi_p^\theta\}_{\theta=1}^{\nu}$ from all $V$ views are then passed to a triangulation operation $\Delta$ to output potential 3D keypoint $\mathbf{X}_n = (x, y, z)_n$ by

$$\Delta(\{\mathbf{x}_n^\theta\}_{\theta=1}^{\nu}, \{\Pi_p^\theta\}_{\theta=1}^{\nu}) = \mathbf{X}_n.$$

We use direct linear transform [10] for triangulation, which is differentiable and allows back-propagation through the keypoints. Note that this triangulation operation is applied separately for each keypoint. The semantic consistency of the keypoints across all views is enforced through multi-view geometry, as the $n$-th keypoint from different views are mapped together.

The potential 3D keypoints $\{\mathbf{X}_n^\theta\}_{n=1}^N$ are the features of interest that we will eventually use from the network’s prediction. The obtained keypoints so far are not guaranteed to contain any useful information as they are unsupervised. We project them to the foreground subject mask to constrain the keypoints in order to make them correspond to the subject of interest. To do so, each 3D keypoint $\mathbf{X}_n$ is first re-projected to each view $\theta$ by using $\Pi_p^\theta$, this yields

$$\mathbf{x}_n^\theta = (\hat{u}, \hat{v})_n^\theta = \Pi_p^\theta(\mathbf{X}_n).$$

The set of re-projected 2D keypoints $\{\mathbf{x}_n^\theta\}_{n=1}^N$ in each view is then passed to a mask-decoder $\phi$ that estimates the mask of the foreground subject $\mathbf{M}_p$, which is equivalent to

$$\mathbf{M}_p^\theta = \phi(\{\mathbf{x}_n^\theta\}_{n=1}^N).$$

This creates a full pipeline validating the discovered 3D keypoints that correspond to the foreground subject and satisfy the multi-view constraints. The semantic information of keypoints is learned automatically by the model without supervision.

3.3. Unsupervised Training Losses

To train the unsupervised keypoint discovery model we use the following losses. We drop the view-dependent notations $\theta$, as the following losses apply to all views.

**Image Reconstruction Loss.** The reconstructed image $\hat{I}$, described in Section 3.1, should match the input image $I$ using two losses; an RGB pixel-wise loss, and a perceptual loss using ImageNet features [15, 23, 37]; comparing the extracted features of the first 3 layers of the ResNet18 [11]. This can be written as

$$\mathcal{L}_{\text{reconst}} = \|I - \hat{I}\|_2^2 + \beta \sum_{l=1}^{3} \|\text{Res}_l(I) - \text{Res}_l(\hat{I})\|_2^2.$$

**Mask Reconstruction Loss.** The estimated mask patch $\mathbf{M}_p$ in the image-reconstruction module of Section 3.1, is taken as target for the mask $\mathbf{M}_p$ predicted from discovered keypoints in Section 3.2 to minimize

$$\mathcal{L}_{\text{mask}} = \|\hat{\mathbf{M}}_p - \mathbf{M}_p\|_2^2.$$

In order to further enforce the predicted keypoints lie in the foreground mask $\mathbf{M}_p$ and enhance the location of the detected bounding box, the two following losses are also used.
Coverage Loss. Each projected keypoint \( \hat{x}_n \) should be only on the foreground subject mask \( M_p \) and not the background. To enforce this, a Gaussian heatmap \( H_n \) is first generated by taking its mean to be the keypoint \( \hat{x}_n \). This map is then normalized to have a probability map that sums to one, which equals to

\[
\bar{H}_n = H_n / \sum_{i,j} \left( H_{i,j}^n \right).
\]

The coverage loss in Eq. (13) then maximizes the linear projection of \( \bar{H}_n \) onto \( M_p \) for each keypoint \( \hat{x}_n \). This ensures the probability mass in \( H_n \) falls into locations of the mask \( M_p \) that correspond to the foreground subject. This can be written as

\[
L_{\text{coverage}} = \frac{1}{n} \sum_{n=1}^{N} \left| 1 - \bar{H}_n \odot M_p \right|,
\]

where \( \odot \) indicates the dot product.

Bounding Box Centering Loss. To ensure the detected bounding box completely covers the foreground subject, we enforce the center of the predicted keypoints on the subject to be equal to the center of the detected bounding box. Considering \( u = (u^x, u^y) \) to be the center of the bounding box predicted by \( S \), as described in Section 3.1, and the set of re-projected keypoints on the subject to be \( \{\hat{x}_n\}_{n=1}^{N} \), the loss then equals to

\[
L_{\text{centering}} = |u - \frac{1}{N} \sum_{n=1}^{N} \hat{x}_n|.
\]

This loss together with \( L_{\text{reconst}} \) of Eq. (10) force the predicted bounding box to completely cover the keypoints and hence completely detect the subject. The total loss for training the model is then equal to

\[
L_{\text{unsup}} = L_{\text{reconst}} + \gamma L_{\text{mask}} + \delta L_{\text{coverage}} + \eta L_{\text{centering}}.
\]

3.4. Single-View Keypoint Estimation

The procedure explained in Sections 3.2 and 3.3 trains an unsupervised multi-view keypoint discovery model. Once this network is trained, one can take the discovered multi-view 3D keypoints \( X = \{X_n\}_{n=1}^{N} \) as targets to train an unsupervised single-view 3D keypoint estimation model. Since the 2D keypoints \( x = \{x_n\}_{n=1}^{N} \), are found in each view independently, it is only needed to lift them to 3D. Hence, we train a simple residual MLP model [32], which maps \( x \) to the set of discovered 3D keypoint targets \( X \). The single-view 3D lifting network \( \psi \) is trained using mean-squared error between the predicted and target 3D keypoints:

\[
L_{\text{SV}} = \| \psi(x) - X \|_2^2.
\]

3.5. Pose Estimation

Once the unsupervised model is trained, 3D keypoints are extracted using the procedures explained in Sections 3.2 and 3.4, for multi-view and single-view setups respectively. To verify the quality of the discovered unsupervised 3D keypoints \( X = \{X_n\}_{n=1}^{N} \), we map them to the pose labels \( Y = \{Y_n\}_{n=1}^{N} \), specifying body-joint locations, using a simple linear or two hidden-layer multi-layer perceptron (MLP) \( \theta \) by minimizing

\[
L_{\text{pose}} = \| \theta(X) - Y \|_2^2.
\]

Note that the pose labels are only used to evaluate the quality of the discovered 3D keypoints through this mapping. They are not used for training the unsupervised 3D keypoint model, described in Sections 3.1 to 3.4.

4. Experiments

In this section, we first present the evaluation datasets and metrics. Then, we compare our model quantitatively with other unsupervised approaches and present ablation studies.

4.1. Datasets

We use the following 3D human pose benchmark datasets.

Human3.6M [17] (H36M). As in [15, 37], we use subjects 1, 5, 6, 7, and 8 to train the unsupervised keypoint estimation model. This yields 308,760 training samples. Subjects 9 and 11 are used for evaluation, sub-sampled every 10 frames, and applied uniformly over video frames. This yields 53,720 test images. We show qualitative results of the discovered unsupervised 2D and 3D keypoints in Fig. 3.

MPI-INF-3DHP [33] (3DHP). We use subjects 1 to 6 for training and subjects 7 and 8 for evaluation. We take only frames where the person is the foreground subject. This yields 59,952 frames for training and 7,312 frames for evaluation. We show qualitative results of the unsupervised keypoint predictions in Fig. 4.

4.2. Comparison to the State-of-The-Art Models

The qualitative results of Figs. 3 and 4 indicate that our approach captures 3D posture effectively, as the keypoints remain consistent across views. To quantify this, we compare against other unsupervised 3D and 2D approaches as well as to a pre-trained ImageNet feature extractor. We report 3D pose errors using mean per joint position error (MPJPE), the normalized N-MPJPE, and the procrustes aligned P-MPJPE between predicted and ground-truth 3D poses.

Unsupervised 3D Approaches. We compare against state-of-the-art unsupervised approaches on the H36M

\footnote{We leave out frames with the actor on a chair as it occludes the foreground subject.}
Figure 3. 2D and 3D keypoints found by a 32-keypoint prediction model on H36M. The 2D keypoints are consistent across views and the 3D keypoints capture the posture of the person, which indicates that they correlate with the person’s pose.

Figure 4. 2D and 3D keypoints found by 16- (left) and 32- (right) keypoint prediction models on 3DHP. As in Fig. 3, the 2D keypoints are consistent across views and the 3D keypoints capture the posture of the person.

Table 1. Comparison with unsupervised 3D models on H36M (in mm). SV and MV indicate single- and multi-view models. kpts denotes number of keypoints.

| Model                  | MPJPE | N-MPJPE | P-MPJPE |
|------------------------|-------|---------|---------|
| Known Kinematic Model  |       |         | 99.2    |
| Kundu et al. [26]      |       |         |         |
| Kundu et al. [27]      | 99.2  |         |         |
| Uninterpretable latents|       |         |         |
| NVS [35]               |       |         |         |
| Honari et al. [15]     |       | 115.0   |         |
| Keypoint Discovery     |       |         | 100.3   |
| KeypointNet [41]       |       |         |         |
| SV 2 hid MLP 32 kpts   | 158.7 | 156.8   | 112.9   |
| Ours MV Linear         | 125.73| 121.04  | 89.05   |
| BKinD-3D [80]          |       |         |         |
| MV Linear 15 kpts      | 120.9 | 117.9   | 93.5    |
| Ours MV 32 kpts        | 73.8  | 72.6    | 63.0    |

We report the best performing model with 15 kpts. It works better than the 30 kpt model.

We outperform the competing methods discussed in Section 2. Most importantly, our approach does not rely on any known priors such as kinematic and shape priors, and neither suffers from the lack of interpretability of the discovered features. Moreover, it outperforms other single-view and multi-view keypoint discovery approaches.

Cross Dataset Evaluation. We take a model trained on H36M dataset and test it on 3DHP dataset to evaluate the generalization of our approach across different datasets. The result is shown in the second to the last row in Table 2.

Table 2. Comparison with unsupervised 3D models on 3DHP (in cm). All models use a 2 hidden layer MLP to map either features (top 3 models) or 48 discovered keypoints (Ours) to the labelled pose.

| Model                  | Train-Set | MPJPE | N-MPJPE | P-MPJPE |
|------------------------|-----------|-------|---------|---------|
| Uninterpretable latents|           |       |         |         |
| DrNet [7]              | 3DHP      | 22.28 | 21.55   | 14.94   |
| NSD [37]               | 3DHP      | 20.24 | 19.29   | 14.09   |
| Honari et al. [15]     | 3DHP      | 20.95 | 19.78   | 14.04   |
| Ours H36M              |           |       |         |         |
| Ours 3DHP              |           |       |         |         |

This model even outperforms other latent extraction models trained on the same test dataset.

Table 3. Comparison with 2D keypoint estimation models. All models predict 32 keypoints on 6 actions of wait, pose, greet, direct, discuss, and walk and regress a linear model from 2D keypoints to the 2D pose labels.

| Model                  | %-MSE Error |
|------------------------|-------------|
| Thewlis et al. [43]    | 7.51        |
| Zhang et al. [49]      | 4.14        |
| Schmidtke et al. [38]  | 3.31        |
| Lorenz et al. [30]     | 2.79        |
| Jakab et al. [21]      | 2.73        |
| Ours                   | 2.38        |

Unsupervised 2D models. For completeness, even though our primary goal is to discover 3D keypoints, we...
compare the discovered 2D keypoints $x$ of Eq. (4) with those produced by other unsupervised 2D approaches. Table 3 shows that we also outperform these approaches.

**Supervised Pre-Training.** As in recent unsupervised papers [1, 12, 13, 15], we compare the quality of our unsupervised keypoints to the features obtained by a fully-supervised ImageNet model, which is commonly used for transfer learning. This evaluation highlights how much the extracted features by these models are suitable for the downstream application, which in the case of this paper is 3D human pose estimation. The results are shown in Figure 5. The gap in MPJPE is over 75 mm in all cases, with our approach reducing the error by about 50%. Even in low-labeled cases, the margin is maintained. This indicates the keypoints discovered by our model are much more correlated with the target pose.

4.3. Ablation Studies

We next evaluate the impact of loss and model components.

4.3.1 Contribution of the Loss Components

We first study the contribution of each loss component in Eq. (15). The results are presented in Table 4. The first row only reconstructs the input image using Eq. (10) without any 3D keypoint estimation. In this case the latent features $\Xi$ are mapped to the 3D poses, as they are the closest latent representation to the pose. As observed in Table 4, this case has about 37 mm (corresponding to 50%) higher MPJPE error compared to our best approach in the last line.

The second row adds the mask reconstruction loss of Eq. (11). This improves the results by about 32 mm compared to the first row indicating the structure of the latent 3D keypoints has a considerable impact. In the third and fourth rows, we respectively add the coverage loss of Eq. (13) and the centering loss of Eq. (14), where the last row shows our final model with all the features activated. While these two losses have a relatively small contribution, they still help improve the accuracy. They can be removed without hurting much the accuracy to further simplify the approach.

Table 4. Ablation Study on Losses. Each column shows the impact of adding a loss term in Eq. (15). The last row shows our model with all features activated. All models use a 2-hidden layer MLP to map the latent features (1st row) or the discovered keypoints (2nd to 4th rows) to the 3D pose labels. The results are presented in millimeters (mm) on H36M dataset, with a lower value indicating a lower error.

| $L_{\text{rec}}$ | $L_{\text{mask}}$ | $L_{\text{coverage}}$ | $L_{\text{centering}}$ | MPJPE | N-MPJPE | P-MPJPE |
|-----------------|-----------------|-----------------|-----------------|-------|--------|--------|
| ✓               | X               | X               | ✓               | 111.8 | 107.6  | 79.7   |
| ✓               | ✓               | X               | ✓               | 79.9  | 78.6   | 67.0   |
| ✓               | ✓               | ✓               | ✓               | 78.3  | 77.0   | 65.6   |
| ✓               | ✓               | ✓               | ✓               | 73.8  | 72.6   | 63.0   |

4.3.2 Keypoint, Label and Pose-Model Complexity

We study here the impact of the size and complexity of model components, in addition to analyzing performance under different amounts of labeled data. One key question is how many unsupervised keypoints should the model predict and how much it impacts the results. Figure 6 shows the results for 3 different numbers of keypoints:

- 17: number of keypoints equal to the labeled 3D joints.
- 32: a number in-between the two bounds.
- 200: number of keypoints matching prior works [15, 37].

As observed in Figure 6, increasing the number of discovered keypoints increases accuracy. However, the gap between 17 and 32 is bigger than the one between 32 and 200 keypoints. Hence, we use 32 keypoints by default on H36M dataset to avoid adding much complexity. On 3DHP dataset we use 48 keypoints. Check Supplementary for details.

The complexity of the pose estimator model inversely correlates with the interpretability of the discovered keypoints. The more the complexity of the required pose model, the more the keypoints lie in an unrelated latent space, which requires more complex adaptation to map them to the actual pose of the subject. Our results in Figure 6 show that while a simple 2-hidden layer MLP obtains more accurate results, a simpler linear adaptation of the keypoints to the pose labels can still obtain decent results. This further shows the robustness of the unsupervised keypoints and indicates they correlates strongly with the target pose.

4.3.3 Number of Views

In order to analyze how many views are required for a robust pose estimation, we train models with 2, 3, and 4 views. The results are reported in Table 5. As expected, higher accuracy is obtained with more views, since the triangulation obtains more robust 3D keypoints. However, 3 views are enough to obtain robust results, with a clear gap with 2 views and a small gap with 4 views. For a fair comparison with previous baselines[40, 41] that use 4 views, we use the same number of views by default in all experiments.

4.3.4 Target Mask

Our model reconstructs the mask that is estimated by the model itself in the image-reconstruction module. The quality of the obtained mask directly impacts the accuracy of the model as this is the target for the keypoint discovery components. If one has access to the ground-truth mask, how much the results would change? To answer this question we train two variants, one using the full pipeline of Section 3 with all elements of Eq. (15), and another by omitting $L_{\text{rec}}$ from

---

*The dimensionality of the extracted latent features in these methods, which serves as the input dimensionality to the pose model, is 600, which equals to 200 3D keypoints.*
Figure 5. Comparison against the latent features of the pre-trained ImageNet model on H36M. The three plots from left to right show depict MPJPE, NMPJPE, and PMPJPE (in mm) for different percentage of labeled 3D data. Both models use a ResNet50 encoder [11] and leverage a 2-hidden layer MLP to regress either the 3D keypoints (Ours) or the latent features—features before the classification layer in ImageNet—to the target 3D pose.

Figure 6. Analysis of the impact of the number of discovered keypoints, pose model complexity, and the amount of labelled data on H36M dataset. The three plots show results on MPJPE, NMPJPE, and PMPJPE (in mm). Each plot depicts the impact of trained models using 17, 32, and 200 unsupervised keypoints. For each model, two pose-regressors are trained; one using a linear model (the dashed line) and another using a 2-hidden layer MLP (the solid line). The X axes shows the percentage of 3D labels.

Table 5. Evaluation on the number of views used for triangulation (in mm). In all cases 32 unsupervised 3D keypoints are predicted by the models. All models use a 2-hidden layer MLP for the pose regressor model. Results are reported on H36M dataset.

| Number of Views | Pose Model | MPJPE | N-MPJPE | P-MPJPE |
|-----------------|------------|-------|---------|---------|
| 2               | 2-hid MLP  | 103.21| 100.7   | 81.6    |
| 3               | 2-hid MLP  | 77.7  | 75.7    | 64.0    |
| 4               | 2-hid MLP  | 73.8  | 72.6    | 63.0    |

Table 6. Impact of different target masks (in cm). 1st row: a mask is obtained by reducing background from the input image and normalizing it to [0, 1]. 2nd row: the ground-truth mask is used. 3rd row: our model’s predicted mask is used to train key-point discovery model. All models predict 48 keypoints on the 3DHP dataset and use a 2-hidden layer MLP for pose regression.

| Mask Used             | Pose Model | MPJPE | N-MPJPE | P-MPJPE |
|-----------------------|------------|-------|---------|---------|
| IMG - BG              | 2-hid MLP  | 17.48 | 16.93   | 12.64   |
| Ground-truth          | 2-hid MLP  | 14.57 | 14.21   | 11.52   |
| Predicted (Ours)      | 2-hid MLP  | 14.37 | 13.98   | 10.99   |

5. Conclusion and Limitations

In this paper, we proposed an unsupervised technique that leverages the full projection properties of a multi-view system to discover 3D keypoints by reconstructing the foreground subject’s mask. In doing so, our model finds keypoints that are consistent across different views and correspond to the foreground subject. We show its application on real-world data featuring articulating human bodies.

There are multiple directions that our approach can be improved. Our approach works on images with a single foreground subject. The existence of multiple foreground entities can occlude the mask, leading to extraction of keypoints that lie on all of them. Another limitation is the dependence on an estimation of the background image to help learn the mask of the foreground subject. While as we show in Section 4.3.4, our model does not need to rely on ground-truth masks and does a better job than background subtraction from the input image, it requires a background image in the pipeline. Moreover, our keypoints can jitter on consecutive frames of a video. Leveraging a temporal constraint can enhance the consistency of the obtained keypoints across time. We leave these extensions to future work.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
References

[1] H. Bao, L. Dong, S. Piao, and F. Wei. BEiT: BERT Pre-Training of Image Transformers. In International Conference on Learning Representations, 2022. 1, 2, 7

[2] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging Properties in Self-Supervised Vision Transformers. In International Conference on Computer Vision, 2021. 1, 2

[3] B. Chen, P. Abbeel, and D. Pathak. Unsupervised Learning of Visual 3D Keypoints for Control. In ICML, pages 1539–1549. PMLR, 2021. 1, 3

[4] C. Chen, A. Tyagi, A. Agrawal, D. Drover, R. MV, S. Stojanov, and J. M. Rehg. Unsupervised 3D Pose Estimation with Geometric Self-Supervision. In Conference on Computer Vision and Pattern Recognition, 2019. 2

[5] N. Chen, L. Liu, Z. Cui, R. Chen, D. Ceylan, Ch. Tu, and W. Wang. Unsupervised learning of intrinsic structural representation points. In CVPR, pages 9121–9130, 2020. 3

[6] X. Chen, K. Lin, W. Liu, C. Qian, and L. Lin. Weakly-Supervised Discovery of Geometry-Aware Representation for 3D Human Pose Estimation. In Conference on Computer Vision and Pattern Recognition, 2019. 2

[7] Emily L Denton et al. Unsupervised Learning of Disentangled Representations from Video. In Advances in Neural Information Processing Systems, pages 4414–4423, 2017. 2, 6

[8] A. Elhayek, E. de Aguiar, A. Jain, J. Thompson, L. Pishchulina, M. Andriluka, C. Bregler, B. Schiele, and C. Theobalt. Marconi—convnet-based marker-less motion capture in outdoor and indoor scenes. PAMI, 39(3):501–514, 2017. 1

[9] C. Fernandez-Labrador, A. Chhatkuli, D. P. Paudel, J. J. Guerrero, C. Demonceaux, and L. V. Gool. Unsupervised learning of category-specific symmetric 3d keypoints from point sets. In ECCV, pages 546–563, 2020. 3

[10] R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, 2000. 4

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In Conference on Computer Vision and Pattern Recognition, pages 770–778, 2016. 4, 8, 1

[12] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In Conference on Computer Vision and Pattern Recognition, pages 9729–9738, 2020. 1, 2, 7

[13] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. Masked Autoencoders Are Scalable Vision Learners. In Conference on Computer Vision and Pattern Recognition, pages 16000–16009, 2022. 1, 2, 7

[14] S. Honari, P. Molchanov, S. Tyree, P. Vincent, C. Pal, and J. Kautz. Improving Landmark Localization with Semi-Supervised Learning. In Conference on Computer Vision and Pattern Recognition, pages 1546–1555, 2018. 4

[15] S. Honari, V. Constantin, H. Rhodin, M. Salzmann, and P. Fua. Temporal Representation Learning on Monocular Videos for 3D Human Pose Estimation. PAMI, (01):1–12, 2022. 1, 2, 4, 5, 6, 7

[16] Xiaodan Hu and Narendra Ahuja. Unsupervised 3D Pose Estimation for Hierarchical Dance Video Recognition. In ICCV, pages 11015–11024, 2021. 2

[17] C. Ionescu, I. Papava, V. Olaru, and C. Sminchisescu. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014. 1, 5

[18] U. Iqbal, P. Molchanov, and J. Kautz. Weakly-Supervised 3D Human Pose Learning via Multi-View Images in the Wild. In Conference on Computer Vision and Pattern Recognition, 2020. 2

[19] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial Transformer Networks. In Advances in Neural Information Processing Systems, pages 2017–2025, 2015. 3

[20] T. Jakab, A. Gupta, H. Bilen, and A. Vedaldi. Unsupervised Learning of Object Landmarks through Conditional Image Generation. In Advances in Neural Information Processing Systems, 2018. 1, 2

[21] T. Jakab, A. Gupta, H. Bilen, and A. Vedaldi. Self-Supervised Learning of Interpretable Keypoints from Unlabelled Videos. In Conference on Computer Vision and Pattern Recognition, pages 8787–8797, 2020. 1, 2, 6

[22] T. Jakab, R. Tucker, A. Makadia, J. Wu, N. Snavely, and A. Kanazawa. Keypointdeformer: Unsupervised 3d keypoint discovery for shape control. In CVPR, pages 12783–12792, 2021. 3

[23] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In European Conference on Computer Vision, pages 694–711, 2016. 4

[24] 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. 1

[25] M. Kocabas, S. Karagoz, and E. Akbas. Self-Supervised Learning of 3D Human Pose Using Multi-View Geometry. In Conference on Computer Vision and Pattern Recognition, 2019. 2

[26] J. N. Kundu, S. Seth, V. Jampani, M. Rakesh, R. V. Babu, and A. Chakraborty. Self-Supervised 3D Human Pose Estimation via Part Guided Novel Image Synthesis. In Conference on Computer Vision and Pattern Recognition, 2020. 3, 6

[27] J. N. Kundu, S. Seth, M.V. Rahul, M. Rakesh, R. V. Babu, and A. Chakraborty. Kinematic-Structure-Preserved Representation for Unsupervised 3D Human Pose Estimation. In AAAI Conference on Artificial Intelligence, pages 11312–11319, 2020. 3, 6

[28] Y. Li, K. Li, Sh. Jiang, Z. Zhang, C. Huang, and R. Xu. Geometry-Driven Self-Supervised Method for 3d Human Pose Estimation. In AAAI, pages 11442–11449, 2020. 2

[29] I. H. Lopez-Nava and A. Munoz-Melendez. Wearable inertial sensors for human motion analysis: A review. IEEE Sensors Journal, 16(22):7821–7834, 2016. 1

[30] D. Lorenz, L. Bereska, T. Milbich, and B. Ommer. Unsupervised Part-Based Disentangling of Object Shape and Ap-
pearance. In Conference on Computer Vision and Pattern Recognition, pages 10955–10964, 2019. 1, 2, 6
[31] Xiang Lu, Kaiyan Yu, Yizhai Zhang, Jingtai Yi, Jingtai Liu, and Qjie Zhao. Whole-body pose estimation in physical rider–bicycle interactions with a monocular camera and wearable gyroscopes. Journal of Dynamic Systems, Measurement, and Control, 139(7):071005, 2017. 1
[32] J. Martinez, R. Hossain, J. Romero, and J.J. Little. A Simple Yet Effective Baseline for 3D Human Pose Estimation. In International Conference on Computer Vision, 2017. 5, 1
[33] 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. 1, 5
[34] A. Noguchi, U. Iqbal, J. Tremblay, T. Harada, and O. Gallo. Watch It Move: Unsupervised Discovery of 3D Joints for Reposing of Articulated Objects. In CVPR, pages 3677–3687, 2022. 1, 3
[35] H. Rhodin, M. Salzmann, and P. Fua. Unsupervised Geometry-Aware Representation for 3D Human Pose Estimation. In European Conference on Computer Vision, 2018. 2, 6
[36] H. Rhodin, J. Spoerri, I. Katicioglu, 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
[37] H. Rhodin, V. Constantin, I. Katicioglu, M. Salzmann, and P. Fua. Neural Scene Decomposition for Human Motion Capture. In Conference on Computer Vision and Pattern Recognition, 2019. 1, 2, 4, 5, 6, 7
[38] L. Schmidtke, A. Vlontzos, S. Ellershaw, A. Lukens, T. Arichi, and B. Kainz. Unsupervised Human Pose Estimation Through Transforming Shape Templates. In CVPR, pages 2484–2494, 2021. 1, 2, 6
[39] L. Sigal, A. Balan, and M. J. Black. Humaneva: Synchronized Video and Motion Capture Dataset and Baseline Algorithm for Evaluation of Articulated Human Motion. International Journal of Computer Vision, 2010. 1
[40] J. Sun, L. Karashchuk, A. Dravid, S. Ryou, S. Fereidooni, J. C Tuthill, A. Katsaggelos, B. W Brunton, G. Gkioxari, A. Kennedy, et al. Bkind-3d: Self-supervised 3d keypoint discovery from multi-view videos. In CVPR, pages 9001–9010, 2023. 3, 6, 7
[41] S. Suwajanakorn, N. Snavely, J. Tompson, and M. Norouzi. Discovery of Latent 3D Keypoints via End-To-End Geometric Reasoning. In Advances in Neural Information Processing Systems, 2018. 1, 3, 6, 7
[42] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised Learning of Object Landmarks by Factorized Spatial Embeddings. In International Conference on Computer Vision, 2017. 1, 2
[43] J. Thewlis, S. Albanie, H. Bilen, and A. Vedaldi. Unsupervised Learning of Landmarks by Descriptor Vector Exchange. In International Conference on Computer Vision, pages 6361–6371, 2019. 1, 2, 6
[44] Sh. Tripathi, S. Ranade, A. Tyagi, and A. Agrawal. PoseNet3D: Learning Temporally Consistent 3D Human Pose via Knowledge Distillation. In 3DV, pages 311–321, 2020. 2
[45] B. Wandt, M. Rudolph, P. Zell, H. Rhodin, and B. Rosenhahn. Canonpose: Self-Supervised Monocular 3D Human Pose Estimation in the Wild. In Conference on Computer Vision and Pattern Recognition, 2021. 2
[46] B. Wandt, J. Little, and H. Rhodin. ElePose: Unsupervised 3D Human Pose Estimation by Predicting Camera Elevation and Learning Normalizing Flows on 2D Poses. In CVPR, pages 6635–6645, 2022. 2
[47] O. Wiles, A.S. Koepke, and A. Zisserman. Self-Supervised Learning of a Facial Attribute Embedding from Video. In British Machine Vision Conference, 2018. 1, 2
[48] Z. Yu, B. Ni, J. Xu, J. Wang, Ch. Zhao, and W. Zhang. Towards Alleviating the Modeling Ambiguity of Unsupervised Monocular 3D Human Pose Estimation. In ICCV, pages 8651–8660, 2021. 2
[49] 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. 1, 2, 6

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.