# Learning 3D Global Human Motion Estimation from Unpaired, Disjoint Datasets

Julian Habekost[1]
julian.habekost@ed.ac.uk
Takaaki Shiratori[2]
tshiratori@fb.com
Yuting Ye[2]
yuting.ye@fb.com
Taku Komura[1]
tkomura@ed.ac.uk

[1] School of Informatics
University of Edinburgh
Edinburgh, UK

[2] Facebook Reality Labs, USA

**Abstract**

We propose a novel method to compute both the local and global 3D motion of the human body from a 2D monocular video. Our approach only uses unpaired sets of 2D keypoints from target videos and 3D motion capture data for training. The estimation target video dataset is assumed to lack any ground truth and thus our supervision signal comes from motion datasets that are fully disjoint from the target datasets. For each time step, a temporal convolutional generator configures the human pose in the global space to satisfy both a reprojection loss and an adversarial loss. The translational and rotational global motion is then derived and converted into the egocentric representation in a differentiable manner for adversarial learning. We compare our system to state-of-the-art architectures that use the Human3.6M dataset for paired training, and demonstrate comparable precision even though our system is never trained on the ground truth Human3.6M 3D motion capture data. Due to its unpaired and disjoint nature in the training data, our system can be trained on a large set of videos and 3D motion capture data, which can considerably expand the domain of the applicable motion data types.

# 1 Introduction

There is a growing demand in 3D human pose estimation from monocular videos for 3D motion capture, surveillance, autonomous driving, and motion analysis. By predicting the 3D pose of the subject, it will be easier to understand the context and predict the future status.

One of the main challenges of predicting 3D motion from videos using machine learning is that they require ground-truth training data, where the 2D monocular videos and the corresponding ground-truth 3D motion data are given. This significantly limits the availability of the training data as most high quality motion capture data do not come with corresponding 2D monocular videos if at all, let alone in a natural setting. Also, due to the lack of a large dataset with both modalities, such supervised training tends to overfit to the training

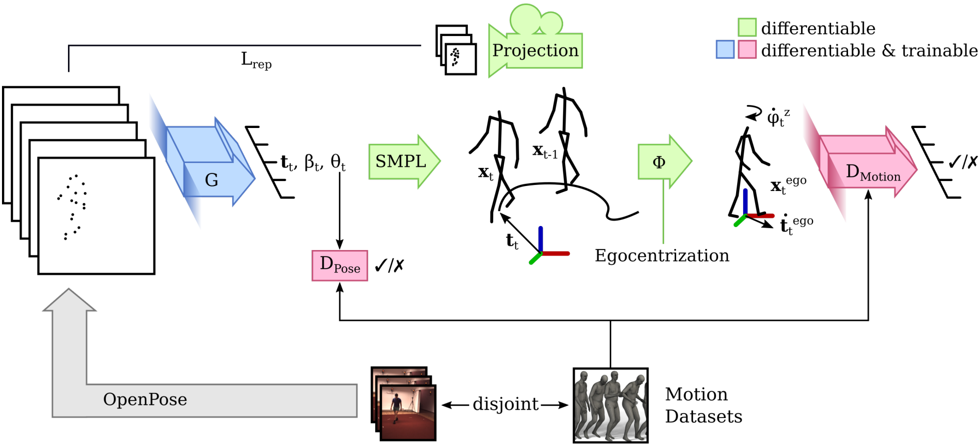


Figure 1: Overview of our system to show how disjoint motions datasets from the target videos can be used to estimate the subject's global motion. The temporal convolutional generator $G$ estimates the global SMPL parameters within the camera space to compute the reprojection loss. For adversarial training based on motion feasibility, the motion is made independent from the camera via our differentiable egocentrization.

data, where the motion types are limited and the conditions of the 2D videos are rather biased. This results in poor performance when testing in-the-wild videos that contain arbitrary motion and background.

Another issue with existing methods using monocular videos is that they mostly compute only the local motion, where the root of the body is fixed, but not the global body motion, which describes how the body is translating in the world. This can be a limitation for applications such as 3D motion capture and detailed motion analysis.

In this paper, we propose a novel method to compute both the local and global 3D motion of the human body from a 2D monocular video in an fully unpaired manner with disjoint datasets for estimation target and supervision signal. The overview of our system is shown in Fig. 1. Our temporal convolutional motion generator receives a set of 2D body joint positions from an off-the-shelf 2D pose detector as input, and outputs the 3D joint positions as well as the root translation and orientation in the world space. Our system uses an unpaired, disjoint set of 2D keypoint detections and 3D motion capture data for training. Our 2D and 3D datasets come from completely separate sources where no 3D ground truth of the 2D data are included in the training set. The system is trained by a reprojection loss and an adversarial loss. To the best of our knowledge, this is the first method to estimate global motion from 2D keypoint detections on monocular videos using unpaired and disjoint datasets for the estimation target and supervision signals.

Our results are evaluated quantitatively through an ablation study and qualitatively via accompanied videos. We also compare our method with supervised architectures that require a pair of 2D videos and 3D motion capture data, trained on Human3.6M. Our results are comparable to supervised methods in terms of accuracy, even though our system is never trained on the ground truth Human3.6M motion capture data.

To evaluate our system with the Human3.6M dataset, we retarget its motion data onto the SMPL [15] skeleton. To map the dataset in a consistent manner, we propose a motion retargeting framework where we optimize the meta-parameters of the target skeleton based on all the motion capture data by all the subjects jointly.

The contribution of this paper is summarized as follows:

- a novel architecture for estimation of the 3D pose of a human subject including the global root motion from a monocular video using unpaired and disjoint datasets,
- a comprehensive evaluation of our framework in comparison to existing state-of-the-art approaches, and
- a motion retargeting framework that optimizes the meta-parameters of the target skeleton based on the entire Human3.6M dataset.

# 2 Related Works

In this section, we first review techniques on 2D human pose estimation from images. We next review techniques on 3D pose estimation from images, and finally those that make use of temporal coherence to predict the 2D/3D human motions from videos.

## 2.1 2D Pose Estimation

Human pose estimation from videos is a classic computer vision problem [1]. Most successful classical approaches are based on the Deformable Part Models (DPM) [6], where the system recognizes each joint based on the features and their connectivity with adjacent joints. Hand-crafted features such as HOG, edges, color histograms etc are used for detecting the joints.

The performance of the pose estimators have improved since the usage of deep learning techniques. The model by Toshev *et al.* [28] significantly outperforms classical approaches based on hand-crafted features. Jain *et al.* [10] apply convolutional neural networks for computing the heatmaps for joints and then use the global position prior to compute their final positions. Newell *et al.* [22] use multiple resolutions of the image to increase the precision of the local pose estimation. Cao *et al.* [3] improve the body pose estimation using the Part Affinity Field that predicts the connection between joints.

## 2.2 3D Pose Estimation

There is an increasing interest in the prediction of 3D human pose from 2D images. Martinez *et al.* [18] regress 2d detections from [22] to 3D poses with a simple neural network. Zhou *et al.* [30] expand [22] by a depth regression and formulate geometric losses for semi-supervised learning, enabling combined training on data with 3D and 2D GT only. Mehta *et al.* [19] introduce a green screen dataset with exchangeable backgrounds for better generalization. They also optimize for reprojection in camera space to obtain global poses assuming the camera stays fixed. Kanazawa *et al.* [11] estimate both 3D pose and shape based on SMPL [15], a parametric model of human shape and pose. A concurrent work by Pavlakos *et al.* [23] completes the same task in a similar manner.

These approaches predict human poses from individual images, and thus suffer from jerkiness when applied to videos. In addition, due to depth ambiguity from monocular views, they can only predict root relative poses but not the global motion in the world space over time. We call these root relative human poses *local poses*.

Methods that consider temporal coherence can remove the jerkiness of the joints when blindly applying per-frame pose-estimators to video frames. Tekin *et al.* [27] use motion compensation for 3D pose prediction but the poses still are root relative. Mehta *et al.* [20] filter poses over consecutive frames and optimize for reprojection in camera space. Their method therefore results in global motion sequences with translation as long as the camera is fixed. Dabral *et al.* [6] extend [30] by a supervised temporal pose refinement for root relative poses. Hossain and Little [9] and Pavllo *et al.* [24] regress sequences of 2D poses to sequences of 3D (root relative) poses using LSTM and TCN, respectively. Xu *et al.* [29] reconstruct pose, mesh, and cloth in the global space but needs an initialization per subject, where the subject's standing pose must be scanned by a video from different directions. Peng *et al.* [25] learn 3d pose sequences in the world space from videos as a byproduct from learning reinforcement policies within a physical environment. In contrast to our work, their method needs retraining for every video. Kanazawa *et al.* [12] propose a system to predict the 3D motion of the person from a single image. Kocabas *et al.* [13] use a temporal discriminator to learn smooth movements of the body. Arnab [2] use bundle adjustment of a sequence of poses based on consecutive single frame estimations.

We train our system purely with unpaired 2D and 3D motion capture data while also predicting the global motion. Among the previous works, majority of the works only predict the local poses of the subjects (except [20, 25, 29]). Also, although some works such as [11, 12, 13] claim unpaired training, their system are co-trained by paired data such as the Human3.6M training videos. This makes our work unique from existing works; we compare our system with existing methods later in this paper and show that it performs comparably even though it is trained in a fully unpaired fashion.

# 3 Methodology

We regress global 3D motion from 2D keypoint detections by learning a mapping to satisfy a motion critic, a pose critic, and a reprojection loss based on a known camera. We assume the camera is static and its intrinsics and extrinsics are known in advance. We use the off-the-shelf detector OpenPose [4] for obtaining keypoint estimates on test videos. A temporal convolutional generator learns a mapping from a sequence of 2D detections to a sequence of body shape, body pose and translation parameters for the differentiable SMPL human body model [15] (see section 3.1). In order to reduce the dimensional complexity of global motion, we then express it from the perspective of the subject (what we call *egocentrization*) in a differentiable way (see section 3.2). The global joint positions can be projected into 2D with known camera parameters for a reprojection loss (see section 3.3). A temporal convolutional discriminator is trained to distinguish the generated and real motion in a motion database [17] (see section 3.4). Further we use a pose discriminator which only sees the SMPL pose angle parameters (see section 3.5). Note that the system never sees any video's 3D ground truth data. We also propose a heuristic for obtaining SMPL parameterized motion sequences from skeletal motion data (see section 3.7). This function is useful when marker data is not available and MoSh [14] cannot be used.

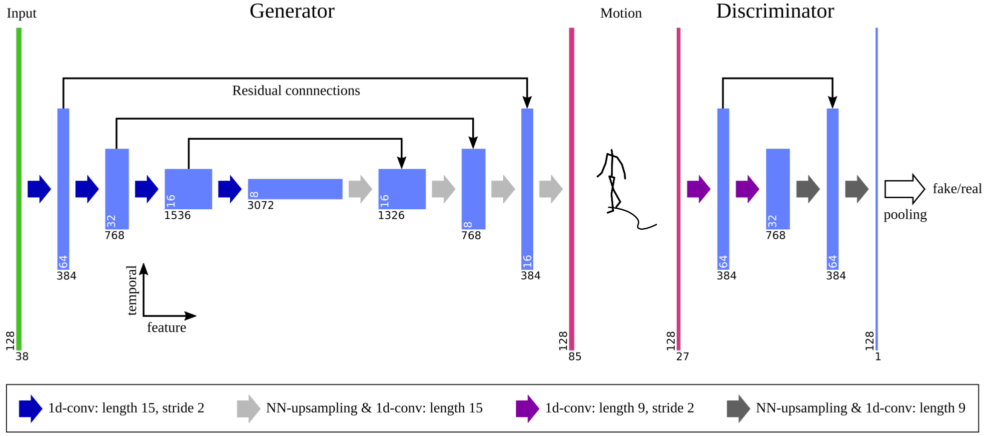


Figure 2: The generator (left) and discriminator (right) structures of our system. Note that the generator's output motion is processed through our differentiable egocentrization before it is inputted to the discriminator, which is not depicted here.

## 3.1 3D Global Motion Generator for SMPL Sequences

The temporal convolutional generator $G$ (see Fig. 1) takes a sequence of 2D body keypoint detections $\mathbf{X}_t^{2d} \in \mathbb{R}^{19\times 2}$ as an input and maps it to a sequence of SMPL [15] parameters $\{\beta, \theta, \mathbf{t}\}_t$ including the body shape PCA basis $\beta \in \mathbb{R}^{10}$, the pose angles $\theta \in \mathbb{R}^{24\times 3}$ in rotation vector representation and the root global translation $\mathbf{t} \in \mathbb{R}^3$ in meters. The body rotation is hereby included in $\theta$ as the first joint's rotation.

$G$ is a temporal convolutional neural network (see Fig. 2, left) inspired by the U-Net [26] architecture. It consists of *down sampling blocks*, *up sampling blocks* and *keep blocks*. A down sampling block halves the temporal width by a 1d convolution layer with stride two while also doubling the amount of channels. The up sampling block uses nearest neighbor upsampling to double the temporal width before a 1d convolution layer halves the channel size. The keep block is a 1d convolution that keeps the input's and output's temporal width and channel size the same. All blocks also feature a ReLU [21] after the convolution except when a block is used as the last output layer. To assemble the network $N_d$ down sampling blocks are followed by $N_k$ keep blocks which in turn are followed by $N_u = N_d$ up sampling blocks. Also, the $n$th down sampling block is connected to the $(N_U - n)$th up sampling block via residual connections. We chose $N_d = N_u = 3, N_k = 1$, a filter length 15, a temporal window length of 128 frames at 10fps and a channel size of 512 after the first down sampling block for this generator.

## 3.2 Differentiable Motion Egocentrization

In order to construct a useful global motion representation for adversarial motion learning we convert the global motion $\{\beta, \theta, \mathbf{t}\}_t$ into the subject's egocentric coordinate system, which we call egocentrization ($\Phi$ in Fig. 1). The motion stems from the fact that for example a straight walk is the same motion regardless from which to which point and at which angle it is performed. By transforming the representation from the world coordinates into the subject's egocentric coordinate system, which is a process that we call egocentrization, the motion becomes invariant to arbitrary rotations and translations while keeping the temporal path within the sequence intact. This reduces the dimensional complexity without losing relevant information. The original global translation and rotation can be obtained by forward integration.

While similar representations have been used in data preprocessing pipelines for character animation modelling [8] we need to construct it as a differentiable function of an SMPL [15] parameter sequence to use it online for training. We call this differentiable egocentrization function $\Phi$. First we need the SMPL accompanied default world space 3D joint position computation module, here denoted $J$: $\mathbf{x}^w = J(\beta, \theta, \mathbf{t})$, where $\mathbf{x}^w \in \mathbb{R}^{24\times3}$ are the position of the joints in the world space. $\Phi$ further maps $\mathbf{x}^w$ to joint positions in egocentric space ($\mathbf{x}^{ego}$), the velocity in egocentric space ($\dot{\mathbf{t}}^{ego}$) and the angular velocity around the z-axis ($\dot{\varphi}_z$), with z being the up axis. For this we set the world space joints $\mathbf{x}^w$ with its hips to the origin $\mathbf{x}_0 = \mathbf{x}^w - \mathbf{t}_{x,y}$ on the horizontal plane using $\mathbf{t}_{x,y}$, the global translation $\mathbf{t}$ with the z-component set to zero. Let $\mathbf{R}$ be the rotation matrix that rotates $\mathbf{x}_0$ around the z-axis so that the subject looks in $+x$ direction and the hips are aligned with the y-axis. This gives

$$\mathbf{x}^{ego} = (\mathbf{R}\mathbf{x}_0^\intercal)^\intercal \tag{1}$$

$$\dot{\mathbf{t}}^{ego} = \mathbf{R}(\mathbf{t}_t - \mathbf{t}_{t-1}) \tag{2}$$

$$\dot{\varphi}_z = -(\mathbf{R}_{t-1}^\intercal \mathbf{R}_t)_{i=1,j=1}. \tag{3}$$

All these operations are differentiable so we can use Tensorflow's autograd feature to obtain the gradients.

## 3.3 Reprojection Loss

The reprojection loss $L_{rep}$ aligns the generated global poses $\{\beta, \theta, \mathbf{t}\}_t$ with the input 2D keypoint detections $\mathbf{X}_t^{2d}$ by projecting them with given camera parameters. Because the default SMPL skeleton does not match OpenPose's underlying assumed 3D skeleton, we follow the common practice of using a learned regression from the vertices of the SMPL output mesh to the target skeleton. Here we denote $V(\beta, \theta, \mathbf{t})$ as the differentiable SMPL function to calculate the 3D position of the body mesh vertices. $J_{OP}(V(\beta, \theta, \mathbf{t})) \in \mathbb{R}^{19\times3}$ maps the vertices further to the OpenPose skeleton. The camera projection $P(\cdot)$ then calculates the 2D projections given the camera's matrix $\mathbf{M} \in \mathbb{R}^{4\times4}$, focal length $f \in \mathbb{R}^2$ and center point $c \in \mathbb{R}^2$. Hence the reprojection loss is

$$L_{rep} = \sum_t^{N_t} |P(J_{OP}(V(G(\mathbf{X}_{1,\dots,N_t}^{2d})_t)), \mathbf{M}, f, c) - \mathbf{X}_t^{2d}|. \tag{4}$$

## 3.4 Motion Discriminator

The motion discriminator $D_{Motion}$ (see Fig. 1) compares the generated egocentric motion with real motion examples. $D_{Motion}$ is a convolutional neural network with a similar structure as the generator $G$ (see Fig. 2, right). It uses a shorter filter length of nine, only one down, up sampling and keep block and 384 channels. This yields a shorter receptive field of about three seconds at ten fps. We use the loss and gradient penalty described in [7]:

$$L_{D_M} = \sum_t [D_{Motion}(\Phi(\mathbf{x}_{1,\dots,N_t}^{real}))_t - D_{Motion}(\Phi(J(G(\mathbf{X}_{1,\dots,N_t}^{2d}))))_t] + c_{pen}^{Motion} R_{pen}^{Motion}, \tag{5}$$

where $R_{pen}^{Motion}$ is the gradient penalty and $c_{pen}^{Pose} = 10$.

### 3.5 Pose Discriminator

We use a time independent pose discriminator $D_{Pose}$ (see Fig. 1) on all 23 relative joint angles in a generated $\theta_t$. Hence we ignore the first angle in $\theta_t$, which is the pose's absolute rotation angle. Note that this allows the generator to freely rotate the subject globally and avoid discontinuities that are found in datasets that do not use rotations above $2\pi$. The real pose examples are drawn from AMASS dataset [17]. $D_{Pose}$ is a feed forward network with two hidden layers with dimensionality 512. Like for $D_{Motion}$ we use the loss and gradient penalty described in [7]:

$$L_{D_P} = \sum_t [D_{Pose}(\mathbf{x}_t^{real}) - D_{Pose}(J(G(\mathbf{X}_t^{2d})))] + c_{pen}^{Pose} R_{pen}^{Pose}, \tag{6}$$

where $R_{pen}^{Pose}$ is the gradient penalty and $c_{pen}^{Motion} = 0.0001$.

### 3.6 Loss Composition and Training

The generator $G$ and the discriminators $D_{Motion}, D_{Pose}$ are trained with adversarial training. We use the following loss for the generator:

$$L_G = \sum_t [D_{Motion}(\Phi(J(G(\mathbf{X}_{1,\dots,N_t}^{2d}))))_t + D_{Pose}(J(G(\mathbf{X}_t^{2d})))] + c_{rep} L_{rep}. \tag{7}$$

where $c_{rep} = 10000$ and $c_\beta = 100$. We use Adam [16] batched stochastic gradient descent with a batch size of 32. One iteration trains $D_{Motion}$ five times, $D_{Pose}$ three times and $G$ one time. We train for 25k iterations. We use the AMASS [17] motion dataset, which itself is a compilation of multiple motion datasets reparameterized as SMPL sequences.

### 3.7 Motion Retargeting by Optimizing Meta Parameters

For our supervised comparison experiment we need the Human3.6M 3D ground truth as a SMPL parameter sequence. The SMPL skeleton differs from the Human3.6M skeleton and thus we need to retarget the Human3.6M dataset to the SMPL skeleton. [1]

Given a set of motion sequences with $N_S$ subjects we subsample the dataset to a reasonable size so that it completely fits on a single GPU (we use 1fps for Human3.6M). For each of the SMPL joint, we define sets of joints in the source skeleton: Each SMPL joint $j$ is assigned the most similar source skeleton joint and all neighboring joints $n_1^j, \dots, n_{Nj}^j$. We optimize the convex combination of these joints to have the same position as the corresponding SMPL joint. Hence given a set of linear coefficients $C = \{c_{ij} | \forall ij\}$ the cost function for a single pose is:

$$L_{single}(C, \beta_S, \theta) = \sum_j^{24} \left\| J(\beta_S, \theta)_j - \sum_{i=1}^{N_j} c_{ij} \mathbf{v}'_{n_i^j} \right\|_2 + \left\| 1 - \sum_{i=1}^{N_j} c_{ij} \right\|_2, \tag{8}$$

where $\mathbf{v}'_a$ is the position of a joint $a$ in the source skeleton. The coefficients $c_{ij}$ are shared among all bodies and poses. Thus for all body shapes $B = \{\beta_1, \dots, \beta_{N_S}\}$ and poses $\Theta =$

[1] Previously [11, 12, 13] this conversion was achieved with marker based MoSh[14]. The resulting data has since been removed from the internet and the official Human3.6M dataset does not feature the marker data.

$\{\theta_{S,t}|\forall S,t\}$ we optimize

$$L_{total}(C,B,\Theta) = \sum_{S}^{N_S}\sum_{t}^{N_t^S} L_{single}(C,\beta_S,\theta_{S,t}) \tag{9}$$

with respect to $C, B, \Theta$ using gradient descent until convergence.

# 4 Evaluation

## 4.1 Translations

We show our mean global translation error per frame in millimeters (see Table 1). Due to lack of work for fair comparison, we train a simple supervised baseline for a comparison (see section 4.1.1). Note that the baseline in this setting is quite strong and its practical use would be limited to datasets with ground truth data only. Still we are coming reasonably close.

We also show in our ablation experiments that the motion discriminator plays an important role. Without it the generator has trouble to position the subject close to the ground truth. If the pose discriminator is also removed the error again rises, but not as drastically as before. This can also be seen in the ablation results in our video.

### 4.1.1 Supervised Baseline

We train a convolutional neural network with the same architecture as the generator $G$ to supervisedly regress $\{\beta,\theta,\mathbf{t}\}_t$ from 2d keypoint detections from the same camera. The ground truth Human3.6M sequences are obtained by using the method described in section 3.7. We train on subjects 1, 5, 6, 7, 8 and report for subjects 9 and 11 as evaluation.

## 4.2 Root Relative Poses

Like previously proposed, we report the root relative pose error on subject 9 and 11 on the Human3.6m dataset. Note that our system never sees any Human3.6M 3D ground truth data. We align the estimated pose with its 3D ground truth via the root joint and report the mean per joint position error (MPJPE) in mm (see Table 2). We compare our results to a selection of previous works as well as our own supervised baseline (section 4.1.1) and our ablation experiments. We mark a method as *disjoint* in Table 2 if no dataset is used for both target and label supervision signal in the training process.

We almost reach state of the art performance with a significant conceptual disadvantage of not using any Human3.6M ground truth in training. We also use less non-pose training target data (images, videos or keypoints) than any other method except for [18], which uses the same amount.

Table 1: Translational error (mm).

| | |
|---|---|
| Supervised Baseline | 120.1 |
| Ours (only $L_{rep}$, no discriminators) | 901.1 |
| Ours (without $D_{Motion}$) | 718.3 |
| Ours | 259.3 |

Table 2: Mean per joint position error (MPJPE) in mm.

| | MPJPE | unpaired | disjoint |
|---|---|---|---|
| Martinez et. al. 2017 [18] | 62.9 | ✗ | ✗ |
| VIBE (Kocabas et. al. 2019)[13] | 65.6 | ✗ | ✗ |
| VNect (Mehta et. al. 2017)[20] | 80.5 | ✗ | ✗ |
| Pavllo et. al. 2018 [24] | 46.8 | ✗ | ✗ |
| HMR (Kanazawa et. al. 2018)[11] | 88.0 | ✗ | ✗ |
| HMR (Kanazawa et. al. 2018)[11] unpaired | 106.8 | ✓ | ✗ |
| Supervised Baseline | 72.2 | ✗ | ✗ |
| Ours (only $L_{rep}$, no discriminators) | 246.6 | ✓ | ✓ |
| Ours (without $D_{Motion}$) | 189.2 | ✓ | ✓ |
| Ours | 118.2 | ✓ | ✓ |

Our ablation shows that both discriminators have similar effect in improving the pose configuration. Note that the pose discriminator also restricts the pose angles to a feasible range similar to the discriminator in [11]. Without it joints can rotate in an unnatural way to reach poses that look valid to the motion discriminator as it is only looking at the positional configuration.

# 5 Discussion

The reason the motion discriminator improves not only the global translation but also the local pose configuration error is that it enforces temporal coherence that a pose discriminator alone cannot impose. Even if all poses are enforced to be valid, the body can still float or slide over the ground freely. Also, the body shape and the pose can switch between frames in an unnatural manner. The readers are referred to the supplementary video for the visual details.

The general problem of finding the absolute distance of an object in camera space is impossible to solve without knowing its size. In our method the sizes of the subjects are learned indirectly because world contacts (like the foot on the ground) need to be hypothesized correctly by the generator in order to produce feasible motion. The ground contacts imply a scale when the ground plane location is known.

Our method is robust to dataset changes by design. Since we do not use supervision from any target dataset we provide a lower bound of what is achievable with guaranteed no target dataset specific overfitting.

A disadvantage to common methods is that for a new setting the camera needs to be calibrated and the system needs to be retrained. Once this is done similar results like reported in this paper can be expected to be achieved. For this our method only needs motion data for training. Motion data is easier available and often licensed less restrictively than datasets with video and 3D ground truth, which other methods need for their training.

To further advance our method one could investigate how to transfer information about a subjects body shape over longer temporal distances in a computational feasible way. Currently we achieve this for short temporal distances by regularizing body shape stability but this is limited in terms of information flow by the length of the receptive field. It might also be useful to extend this method to a multi-view task.

# 6 Conclusion

We propose an unpaired, disjoint system to estimate the 3D human poses from 2D keypoints extracted from the state-of-the-art 2D pose detector. To achieve this task, we propose a fully differentiable pipeline composed of egocentrization, generator, pose discriminator and motion discriminator. Our system shows comparable results to supervised frameworks, which has much more restrictions on the data that can be used for training.

As a future work, we are interested in extending our technique to predict the camera parameters in videos where the camera is dynamically moving.

# Acknowledgement

We thank the reviewers for the constructive suggestions and comments. This project is supported by Facebook.

# References

[1] Ankur Agarwal and Bill Triggs. Recovering 3d human pose from monocular images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(1):44–58, January 2006. ISSN 0162-8828.

[2] Anurag Arnab, Carl Doersch, and Andrew Zisserman. Exploiting temporal context for 3d human pose estimation in the wild. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3390–3399, 2019.

[3] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.

[4] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[5] Rishabh Dabral, Anurag Mundhada, Uday Kusupati, Safeer Afaque, Abhishek Sharma, and Arjun Jain. Learning 3d human pose from structure and motion. In *Computer Vision – ECCV 2018*, Lecture Notes in Computer Science. Springer International Publishing, 2018.

[6] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.

[7] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5767–5777. Curran Associates, Inc., 2017.

[8] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016.

[9] Mir Rayat Imtiaz Hossain and James J. Little. Exploiting temporal information for 3d human pose estimation. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part X*, pages 69–86, 2018.

[10] Arjun Jain, Jonathan Tompson, Mykhaylo Andriluka, Graham W. Taylor, and Christoph Bregler. Learning human pose estimation features with convolutional networks. In *In: ICLR.*, 2014.

[11] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Regognition (CVPR)*, 2018.

[12] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5614–5623, 2019.

[13] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. *arXiv preprint arXiv:1912.05656*, 2019.

[14] Matthew Loper, Naureen Mahmood, and Michael J Black. Mosh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics (TOG)*, 33(6):1–13, 2014.

[15] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

[16] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101, 2017.

[17] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. Amass: Archive of motion capture as surface shapes. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2019.

[18] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, 2017.

[19] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017.

[20] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics*, 36(4), July 2017.

[21] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[22] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII*, pages 483–499, 2016.

[23] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3D human pose and shape from a single color image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[24] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. *arXiv*, abs/1811.11742, 2018.

[25] Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. Sfv: Reinforcement learning of physical skills from videos. *ACM Trans. Graph.*, 37 (6), November 2018.

[26] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[27] Bugra Tekin, Artem Rozantsev, Vincent Lepetit, and Pascal Fua. Direct Prediction of 3D Body Poses from Motion Compensated Sequences. *arXiv e-prints*, art. arXiv:1511.06692, Nov 2015.

[28] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1653–1660, June 2014.

[29] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Trans. Graph.*, 37(2):27:1–27:15, May 2018. ISSN 0730-0301.

[30] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. Towards 3d human pose estimation in the wild: A weakly-supervised approach. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.