

Supplementary Material

Unsupervised Cross-Modal Alignment for Multi-Person 3D Pose Estimation

The supplementary material is organized as follows:

- Section 1: Adversarial Auto-Encoder- Pose representations and training
- Section 2: Architecture and implementation details
- Section 3: Artificial poses- Sampling and analysis
- Section 4: Additional results on 3DPW dataset and 2D pose estimation
- Section 5: Limitations of the proposed framework

Table 1. Notation Table.

	Symbol	Description
Pose Repr.	p_g	3D pose in global coordinate system
	p_r	3D pose in root-relative coordinate system
	p_c	Canonical 3D pose representation
	p_l	3D pose in local parent relative coordinate system
Network	\mathcal{E}, \mathcal{F}	Frozen 2D pose estimation network
	\mathcal{G}	Encodes HM-PAF to intermediate representation
	\mathcal{H}	Learns neural representation
	Φ, Ψ	Adversarial Auto-Encoder
	$Disc$	Pose Discriminator used to train AAE
Transform-ations	FK	Forward Kinematics
	\mathcal{T}_R	Rigid rotation operation on canonical pose
	\mathcal{T}_G	Translation in global 3D space
	\mathcal{T}_L	Canonical pose to local pose transformation
	\mathcal{T}_K	Camera weak perspective projection
Representation (space and samples)	\mathcal{I}	Image space
	\mathcal{V}	Intermediate representation space
	\mathcal{P}	3D space (of multi-person pose)
	\mathcal{K}	2D space (of multi-person pose)
	m_{syn}	Synthetic HM-PAF representation for 2D pose
	r_x, r_y	Root (pelvis joint) location
	s, \tilde{s}	Neural representation
	\hat{k}_p, \hat{k}_q	Student and Teacher 2D pose predictions respectively
	P, \hat{P}	Multi-person 3D pose GT and prediction
	v, \tilde{v}	A sample in \mathcal{V} space
Others	DoF	Degrees of Freedom
	\mathcal{D}_{syn}	Synthetic Dataset
	θ, γ	Angle parameters in spherical coordinate system

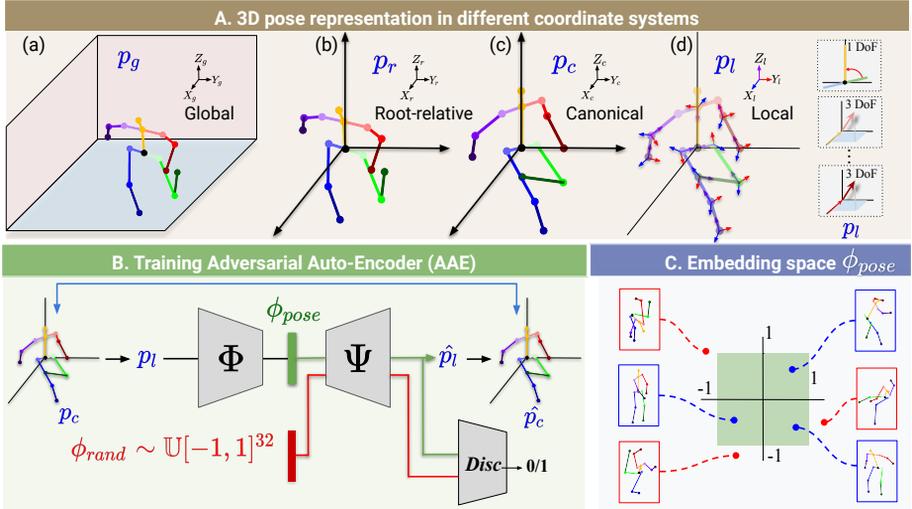


Fig. 1. **A.** 3D pose representation in 4 different coordinate systems- (a) Global, (b) Root-relative, (c) Canonical and (d) Local. On the right, DoFs are shown for certain joints. Right-hip joint has only one DoF in local coordinate system. **B.** Training framework for AAE. **C.** The AAE trained with single-person pose datasets decodes a plausible pose when sampled in $\mathcal{U}[-1, 1]^{32}$. blue box: plausible pose, red box: implausible pose

1 Adversarial Auto-Encoder (AAE)

We train an AAE to learn single-person pose embedding. The proposed framework for training the AAE using encoder Φ , decoder Ψ and adversarial discriminator $Disc$ is shown in Fig. 1B. The main motivation behind learning the single-person pose embedding is to disentangle enforcement of structural plausibility constraints for 3D human pose in the subsequent final task of multi-person pose estimation. This parameterization of 3D pose embedding not only guarantees generation of anthropomorphically plausible pose, but also follows the structural constraints [1] such as joint angle limits, limb interpretation restrictions, etc.

a) View-invariant Canonical 3D Pose Representation. Let p_g be a 3D pose in the global coordinate system, as shown in Fig. 1A(a). The root-relative 3D pose p_r (origin of coordinate system is located at root joint) as shown in Fig. 1A(b) is obtained after subtracting human pelvis location (*a.k.a* root) from p_g . Then, the rigid transformation on p_r , disentangles the root-relative pose into view invariant canonical pose p_c . Let us consider a plane passing through the neck, left-hip and right-hip joints. Let \hat{n} be a normal to this plane. In the canonical coordinate system, which is defined by axes X_c, Y_c and Z_c in Fig. 1A(c), the vector \hat{n} is canonically aligned with +ve X axis. This alignment makes the canonical pose p_c view-invariant. Note that, the root-relative pose p_r can be recovered from p_c by performing a simple rigid transformation described by the corresponding rotation matrix. The rotation matrix itself can be described with Euler angles used to rotate p_r to form p_c .

b) Local 3D pose representation. Inline with [10], the forward kinematic formulation expresses each body joint with respect to its parent joint. In the local coordinate system for each joint (see Fig. 1A(d)), the kinematic 3D structure of the human skeleton can be studied by capturing the limitations of joint movements relative to the corresponding parent joint. Further, every parent-child limb is assigned a fixed bone length. For example, the bone-length of the limb connecting the left-shoulder and left-elbow is fixed for all poses. A 3D pose expressed using this kinematic formulation is termed as local pose p_l and is shown in Fig. 1A(d). As p_l is obtained from p_c , it is both view-invariant and bone-length scale invariant. The local pose coordinate system X_l , Y_l and Z_l is defined as follows: Each joint (except neck, pelvis, left-hip and right-hip) is expressed with respect to its parent joint, or in other words, the origin of the coordinate system is fixed at the parent joint. The coordinate axes are obtained by performing Gram-Schmidt orthogonalization of a vector joining parent-child and a normal \hat{n} to the plane spanning neck, left-hip and right-hip joints. The transformation from canonical pose p_c to local pose p_l is given as $\mathcal{T}_L : p_c \rightarrow p_l$.

c) Training AAE. The architecture of AAE (see Fig. 1B) is based on a kinematic tree of limb-connections mentioned in [3]. The pose embedding ϕ_{pose} is 32 dimensional vector and obtained through *tanh* nonlinearity. We choose to train an AAE with an aim to learn pose embedding in continuous manner. This generative approach allows us to uniformly sample any random vector as $\phi \sim \mathcal{U}[-1, 1]^{32}$ and predicts an anthropomorphically plausible human pose when decoded through Ψ . The plausible and implausible pose pattern obtained after sampling pose embedding is shown in Fig. 1C. We employ discriminator *Disc* to distinguish between real pose embedding ϕ_{real} and pose embedding sampled through $\phi_{rand} \sim \mathcal{U}[-1, 1]^{32}$. In order to enforce learning of an one-to-one mapping in a generative adversarial setup, we add cyclic reconstruction loss on both canonical pose p_c and pose embedding ϕ_{pose} as follows:

$$\mathcal{L}_{cyc} = |p_c - \hat{p}_c| + |\phi_{pose} - \hat{\phi}_{pose}| \quad (1)$$

Where, $\hat{p}_c = FK \circ \Psi \circ \Phi \circ \mathcal{T}_L(p_c)$, $\hat{\phi}_{pose} = \Phi \circ \Psi(\phi_{pose})$, FK: $p_l \rightarrow p_c$ and $\mathcal{T}_L : p_c \rightarrow p_l$. We train encoder Φ using \mathcal{L}_{cyc} and decoder Ψ using $\mathcal{L}_{cyc} + \mathcal{L}_{adv}$ inline with [4].

2 Architecture

In this section, we describe network architectures of $\mathcal{E}, \mathcal{F}, \mathcal{H}, \mathcal{H}', \mathcal{G}$.

Module \mathcal{E} : We use a pre-trained model of Cao *et al.* [2] as a teacher model as shown in Fig. 2. The teacher model uses VGG19 backbone, followed by separate branches of fully convolutional layers for heatmap and PAF. The *concat* operation concatenates the outputs of these branches into an output of shape $28 \times 28 \times 1024$.

Module \mathcal{F} : We use upto stage-2 of Cao *et al.* [2] as \mathcal{F} . As seen in Fig. 2, there are 8 convolutional layers in both HM and PAF branches. Each branch takes the input from the corresponding output branch of \mathcal{E} in the distillation pathway and output of \mathcal{G} in the auto-encoding pathway (Fig. 4 of the main paper).

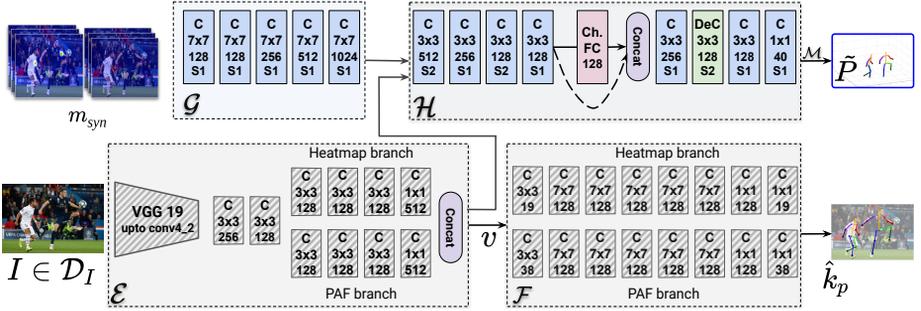


Fig. 2. ‘C’ stands for Convolutional layer. ‘Ch. FC’ stands for Channel-wise Fully Connected layer [7]. ‘DeC’ stands for Deconvolutional layer. Dashed connection indicates skip-connection. Both \mathcal{E} and \mathcal{F} are frozen while training \mathcal{G} and \mathcal{H} . S indicates stride.

Module \mathcal{G} : It consists of five 7×7 convolutional layers as shown in Fig. 2. The input m_{syn} is of $28 \times 28 \times 43$ dimension where 15 channels correspond to each of the 15 joints and 28 channels correspond to PAF representation for all limbs.

Module \mathcal{H} and \mathcal{H}' : Both \mathcal{H} and \mathcal{H}' network modules share the same architecture. These modules take an embedding v as an input and predict a tensor of shape $14 \times 14 \times 39$. Further, these modules have a Channel-wise Fully Connected layer (Ch-FC) (similar to [7]) where the layer connects all nodes of a given input channel to all nodes of corresponding output channel. In our architecture, this layer takes $7 \times 7 \times 128$ as input tensor shape and outputs tensor of the same shape. Since each of the 128 channels has a spatial dimension of 7×7 , the Ch-FC layer consists of 128 fully connected layers with 49 input nodes and 49 output nodes in each layer. The final layer of \mathcal{H} uses an activation of \tanh which ensures that the output space of \mathcal{H} results in plausible 3D pose prediction (via Ψ). All other layers in the module \mathcal{H} use Leaky ReLU activation.

2.1 Differentiable transformation operations in \mathcal{M}

Module \mathcal{M} consists of frozen 3D pose embedding decoder Ψ , forward kinematics operation (FK), pose 3D rigid transformation \mathcal{T}_R and 3D scene composition by translating multiple root-relative 3D poses \mathcal{T}_G .

a) Forward kinematics (FK) $p_l \rightarrow p_c$. Using forward kinematics, the local pose predicted by Ψ , is converted into view-invariant canonical 3D pose [1].

b) Rigid rotation transformation \mathcal{T}_R : $p_c \rightarrow p_r$. Module \mathcal{H} predicts sine and cosine angle components for 3 angle parameters (Euler angles, denoted as c) required to perform rigid rotation. Using the Euler angles, the canonical pose p_c is transformed to the root-relative pose p_r as described in Section 1.

c) Global scene composition \mathcal{T}_G : $p_r \rightarrow p_g$. Using the predicted 2D root-keypoints r_x, r_y and the depth d , the net translation of the pose is computed as a function of (r_x, r_y, d) . This translation is performed on 3D pose of each person as inferred in the neural representation (*i.e.* where a root-joint can be inferred).

2.2 Other implementation details

We develop a differentiable camera module with fixed configuration (focal lengths and center of camera are fixed based on input image size) for projecting the 3D scene. The unpaired 3D poses are normalized for keeping the bone length ratio fixed. As discussed previously, this dataset is used for training the pose decoder Ψ and also used for creating multi-person 3D skeleton scenes \mathcal{D}_{syn} .

We first pretrain \mathcal{H} using \mathcal{L}_{distl} for about 15k iterations before imposing all losses. Our **phase-1** of training requires 450k iterations to converge. After training for 450k iterations, **phase-2** of our training is started. As discussed in main paper, in **phase-2** of our training, we impose only \mathcal{L}_{ss} and \mathcal{L}_{recon} while keeping \mathcal{G} frozen.

3 Artificial-pose-sampling

Artificial poses are created by sampling from joint-angle ranges specified by a biomechanic expert. These joint-angle limits are described in the local parent-relative system on the canonical pose representation (see Fig. 1). Therefore, the poses that are sampled from these angle limits provide us with diverse canonical poses. As described in Section 1, these poses can be used to train the AAE and to create the \mathcal{D}_{syn} , in a completely unsupervised setting where a 3D human pose dataset is inaccessible. In this section, we describe the sampling procedure and provide an analysis of the reliability of the *Artificial-pose-sampling*.

3.1 Sampling Procedure

We use the joint-angle limits defined per joint in the local coordinate system and visualize the limits in Fig. 3A. As shown in Fig. 3A, every joint can be completely described in a spherical coordinate system using two angle limits (azimuth and elevation). We represent the angles as a range in azimuth $[\theta_1, \theta_2]$ where $-180^\circ < \theta \leq 180^\circ$ and elevation $[\gamma_1, \gamma_2]$ where $0^\circ \leq \gamma \leq 180^\circ$. As described in the Section 1, certain joints, such as the right hip joint has only 1DoF while some joints such as the neck joint has 0DoF. Note that 3D keypoint locations of the left hip and the left shoulder joints can be inferred in canonical pose directly without sampling, because the pelvis joint and neck joint are the mid-points of the hip joints and shoulder joints respectively.

There is one limitation in describing joint angle ranges in the spherical coordinate system: angle limits for certain joints span beyond the 180° limit of θ . For such joints we propose to use angle ranges that span on the opposite side (beyond 180° into negative θ) of the spherical coordinate system. For example, the θ range for the right shoulder joint is 120° and spans from $\theta_1 = 120^\circ$, but θ_2 goes beyond the 180° . Therefore, we set θ_2 to a value that is equivalent to 240° (which is equal to -120°).

We create artificial single-person pose dataset by sampling from these joint angle limits for all joints applying bone lengths, followed by forward kinematics operation to construct a canonical pose. For obtaining a variety of root-relative poses, we apply random rotation transformation operations on canonical poses.

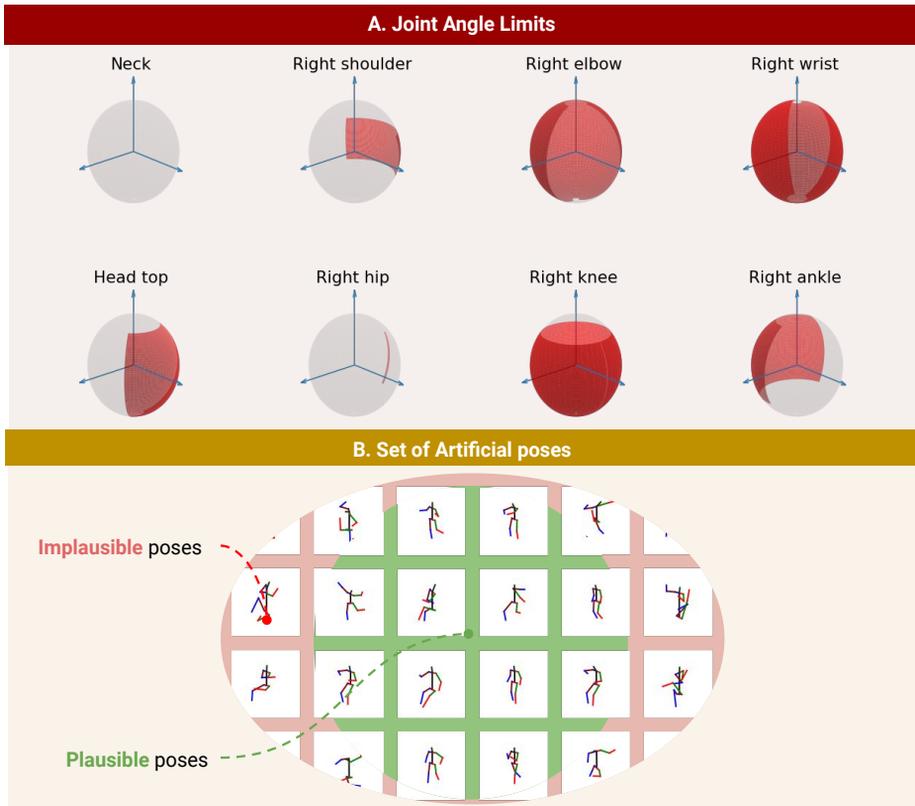


Fig. 3. Single-person artificial pose dataset is created by sampling uniformly from joint wise angle limits defined at local parent-relative coordinate system [1]. **A.** Since angle limits of left-body joints are symmetric to right-body joints, we present only right joints. Neck joint and right-hip joint have 0,1 DoF respectively. **B.** The artificial pose dataset subsumes all plausible poses and could contain a small fraction of implausible poses.

3.2 Analysis of artificial poses

Although sampled artificial poses may have certain degree of implausibility, because each joint angle is sampled independently of pose [1], we find that the artificial pose dataset subsumes all plausible poses [8,9] (see Fig. 3B). This ensures that the AAE learns rich representations in embedding space ϕ . Our experimental analysis shown in Section 4.2 (in the main paper) confirms that having a certain degree of implausibility does not adversely affect the performance. Hence, if we are not provided an access to any unpaired 3D poses, our approach would still perform reliably by *Artificial-pose-sampling*.

4 Additional results

a) Results on 3DPW dataset. The 3D-Poses-in-the-Wild (3DPW) [5] dataset consists of challenging outdoor in-the-wild video sequences. Compared to the MuPoTS-3D dataset, the 3DPW dataset contains larger volume of video sequences

and outdoor scenes. In order to evaluate the generalizability of our model, we evaluate on the test set containing 24 sequences and show the results under the protocol *All-Test-mode*. Note that, as per the *All-Test-mode* protocol, we do not use 3DPW train set and 3DPW validation set for training our model. We use Mean Per-Joint Position Error (MPJPE) and Procrustes Mean Per-Joint Position Error as error metric (PMPJPE). The MPJPE metric is obtained as the average Euclidean distance of joints from corresponding ground-truth joint locations. In PMPJPE, the predicted pose is Procrustes aligned with the ground-truth pose before averaging the error over all joints. Therefore, PMPJPE does not consider global orientation of the predicted pose.

Table 2. Evaluation on 3DPW test set under the protocol *All-Test-mode*. We report MPJPE (lower is better) and PMPJPE (lower is better).

Method	MPJPE	PMPJPE
<i>Ours-Fs</i>	100.7	77.6

b) 2D keypoint prediction. In this section, we extend the results presented in the Table 7 of the main paper. We present qualitative results in Fig. 4 to compare the 2D keypoint estimation for teacher model and student model (*Ours-Fs*) on MuPoTS-3D dataset [6]. The evaluation protocols used for 2D keypoint estimation are Intersection over Union (IoU), 2D-Mean Per-Joint Position Error (2D-MPJPE) and 2D-Percentage of correct keypoints (2D-PCK). IoU is the ratio of area of overlap between the predicted bounding box and the ground-truth bounding box to the area of union of the predicted bounding box and the ground-truth bounding box. 2D-MPJPE is average Euclidean distance between predicted 2D pose keypoints and ground-truth 2D pose keypoints. In 2D-PCK, a predicted keypoint is considered correct if it is present within a range of 25 pixels of ground-truth keypoint. All evaluations are done on keypoints that are shared by both teacher model and student model.

c) Additional qualitative results. We present additional qualitative results for MuPoTS-3D dataset (Fig. 6), MS-COCO 2D keypoints dataset (Fig. 7), and wild multi-person images from YouTube and other sources (Fig. 8). For MuPoTS-3D dataset, we estimate poses of all persons in the image even if ground truth annotation is absent. These results not only show that our model is able to correctly predict depth and pose of persons, but also show generalizability of our model on unseen images.

5 Limitations of the proposed framework

a) Estimation of pelvis (root) location. As discussed in Section 3.1.2 of the main paper, the neural representation of multi-person 3D pose is interpretable only in presence of a pelvis at the corresponding grid location. Therefore, in some scenarios where more than one person shares the same grid location, our model predicts only one pose for all persons in that grid. In rare cases, our model is



Fig. 4. Comparison of teacher model and student model (*Ours-Fs*) results for the task of 2D keypoint estimation on MuPoTS-3D dataset. Erroneous predictions of the teacher model are highlighted using red ovals. Teacher model either fails to predict keypoint locations or fails to assign keypoint to the correct person. As the student model estimates 2D keypoints by projecting 3D pose, it does not involve any keypoint grouping operation usually employed in bottom-up methods, such as the teacher model. These results show that the our model is able to perform better than the teacher model.

unable to predict the root joint of some persons in a given image. This limitation is shown in Fig. 5(a) and Fig. 5(b). The problem of having two pelvises in the same grid cell can be eliminated either by estimating two poses per grid-cell in the neural-representation or by increasing resolution of the output spatial map discussed in the Section 3.1 of the main paper.

b) Rare and ambiguous poses. Fig. 5(c) shows erroneous prediction on rarely occurring poses like acrobatic flips. The model fails to identify correct global orientation of the pose due to left-right symmetry ambiguity in lifting 2D pose to 3D pose. This limitation is also attributed to visibility of body parts. As the face of the person is not visible in the image of the Fig. 5(c), the model is not able to estimate correct body orientation. Similar example of pose ambiguity is shown in Fig. 5(d). The model predicts an ambiguous pose for the person tagged with

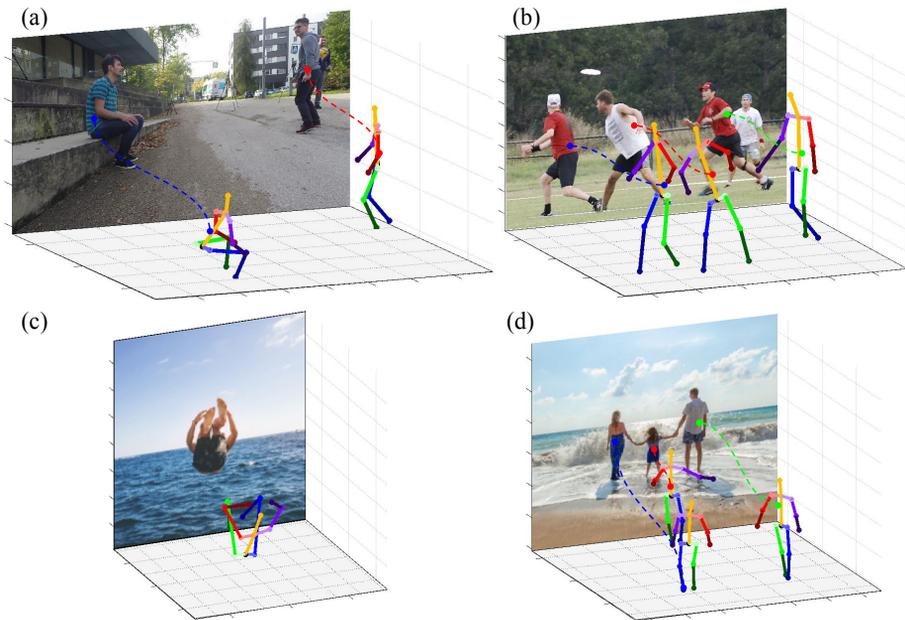


Fig. 5. Limitations of the proposed framework. (a) Multiple pelvises in the same grid cell, (b) Missed pelvis detection, (c) Ambiguous pose and (d) Prediction on small body-frame sized person (d) Ambiguous pose for person tagged with dashed blue line

a blue dashed line. In this case, the person’s 3D pose cues in the image, such as the feet and facial orientation, are not clearly visible because of the limited spatial information owing to low-resolution of the image.

c) Perception of depth based on bone lengths. As the proposed model is bone-length scale-invariant, it expects all 3D poses to be of the same size. Due to this, a person with small body-frame is assumed to be located far away from the camera. This drawback is illustrated in Fig. 5(d) wherein, a person tagged with dashed red line is assumed to be of the same body-frame size as that of remaining people in the image.

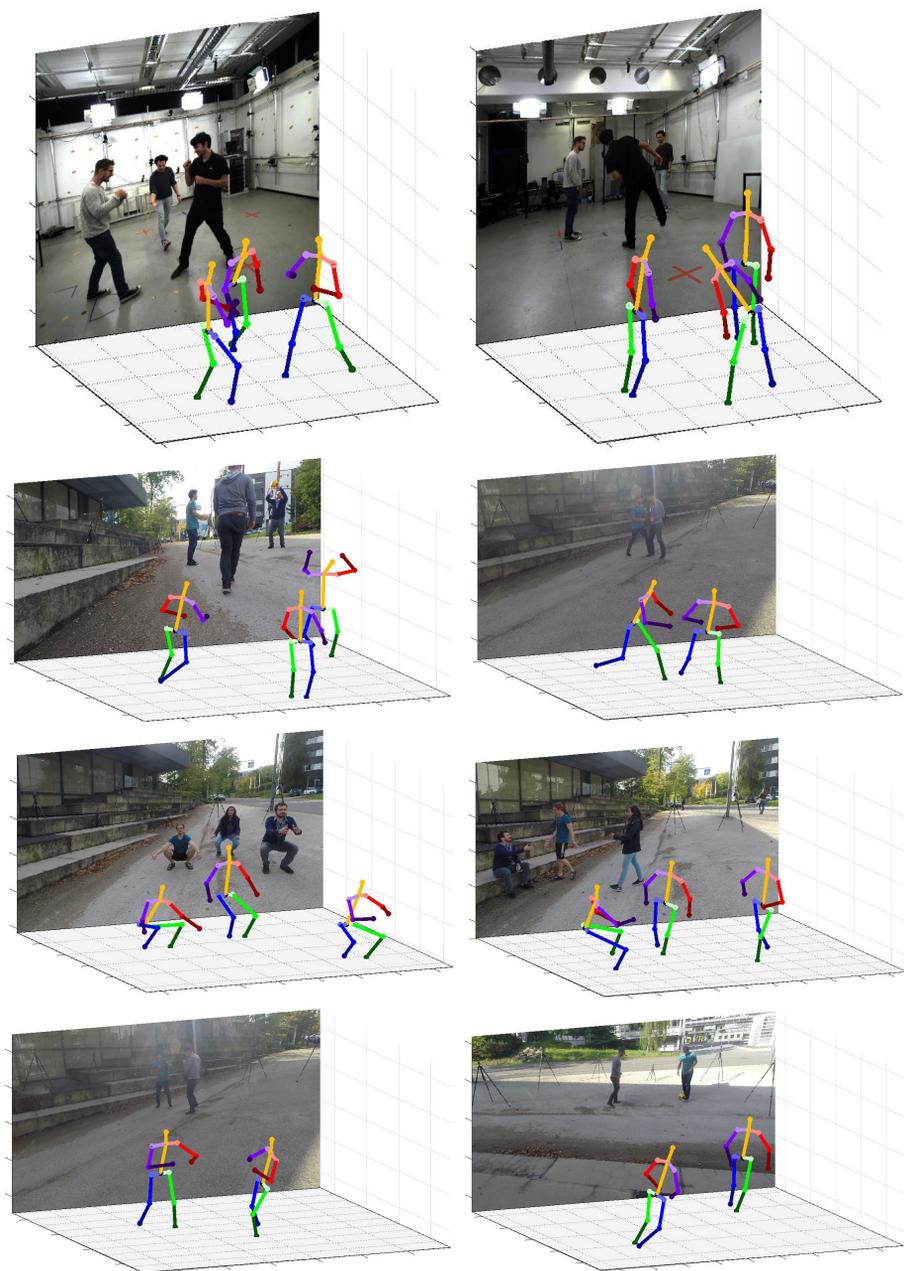


Fig. 6. Qualitative results on MuPoTS-3D dataset. Note that even if ground truth annotation is absent, we predict poses of all people in the image.

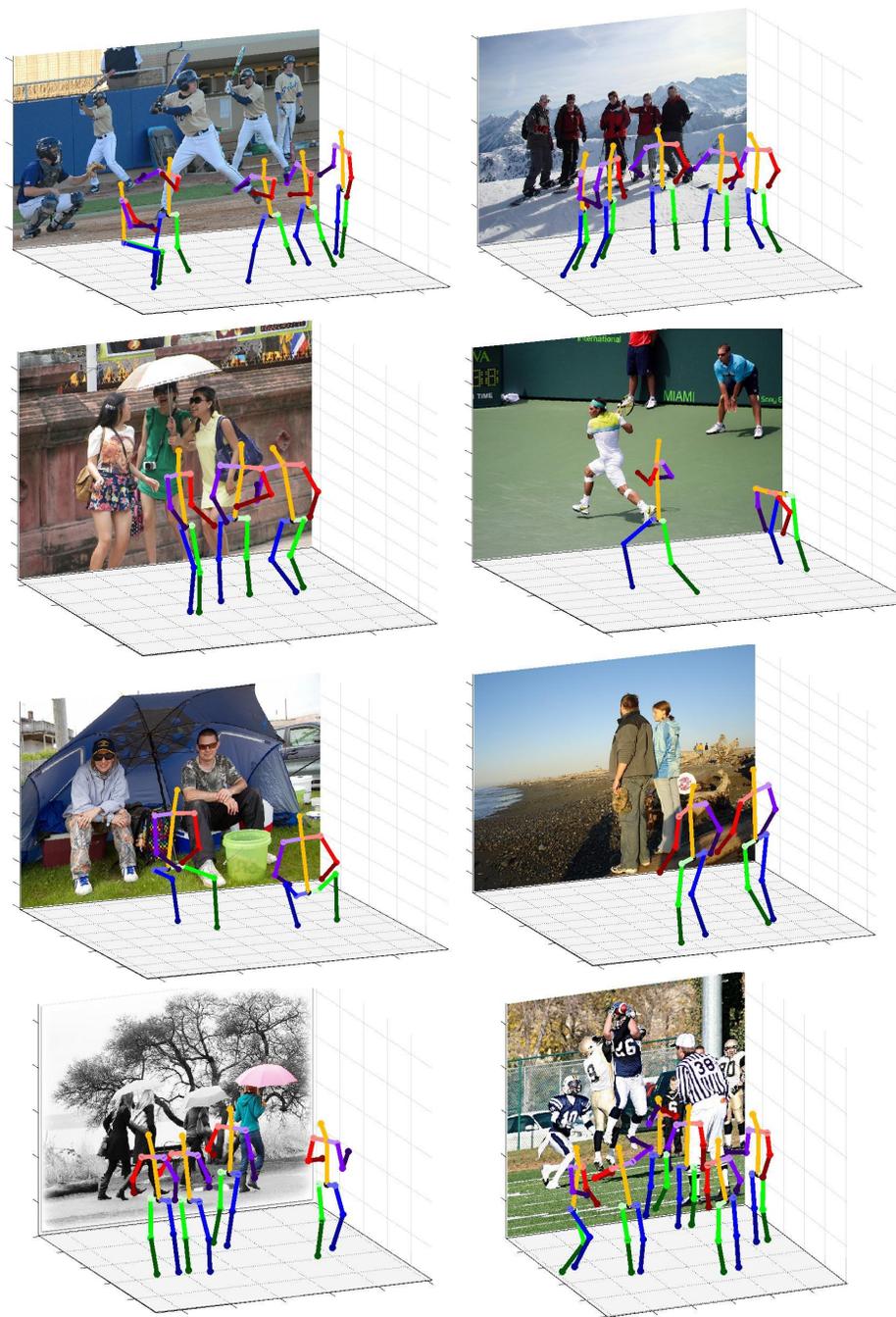


Fig. 7. Qualitative results on MS-COCO

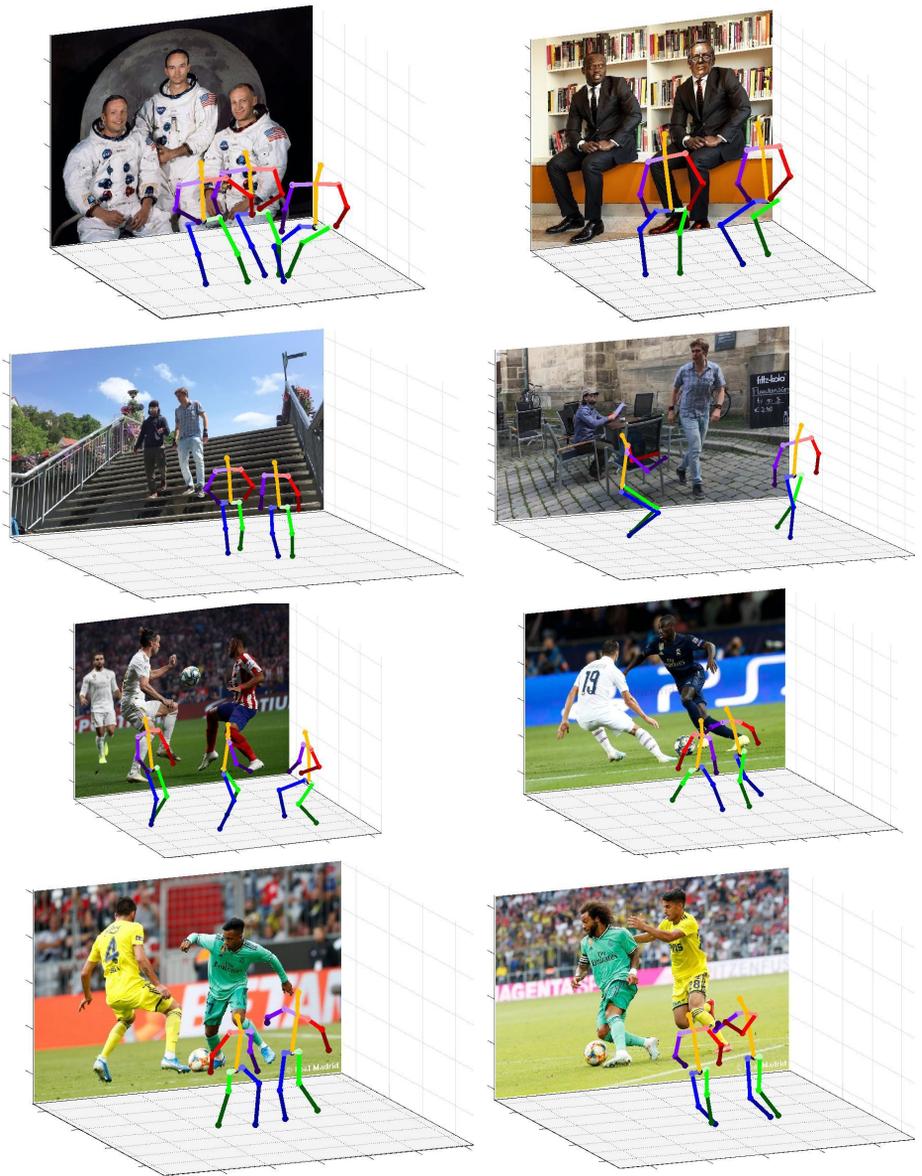


Fig. 8. Qualitative results on in-the-wild images

References

1. Akhter, I., Black, M.J.: Pose-conditioned joint angle limits for 3d human pose reconstruction. In: CVPR (2015) [2](#), [4](#), [6](#)
2. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: CVPR (2017) [3](#)
3. Du, Y., Wang, W., Wang, L.: Hierarchical recurrent neural network for skeleton based action recognition. In: CVPR (2015) [3](#)
4. Kundu, J.N., Gor, M., Uppala, P.K., Radhakrishnan, V.B.: Unsupervised feature learning of human actions as trajectories in pose embedding manifold. In: WACV (2019) [3](#)
5. von Marcard, T., Henschel, R., Black, M.J., Rosenhahn, B., Pons-Moll, G.: Recovering accurate 3d human pose in the wild using imus and a moving camera. In: ECCV (2018) [6](#)
6. Mehta, D., Sotnychenko, O., Mueller, F., Xu, W., Sridhar, S., Pons-Moll, G., Theobalt, C.: Single-shot multi-person 3d pose estimation from monocular rgb. In: 3DV (2018) [7](#)
7. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: CVPR (2016) [4](#)
8. Peng, X.B., Andrychowicz, M., Zaremba, W., Abbeel, P.: Sim-to-real transfer of robotic control with dynamics randomization. In: ICRA (2018) [6](#)
9. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain randomization for transferring deep neural networks from simulation to the real world. In: IROS (2017) [6](#)
10. Zhou, X., Sun, X., Zhang, W., Liang, S., Wei, Y.: Deep kinematic pose regression. In: ECCVW (2016) [3](#)