SelfPose: 3D Egocentric Pose Estimation from a Headset Mounted Camera

Denis Tome, Thiemo Alldieck, Patrick Peluse, Gerard Pons-Moll
Lourdes Agapito, Hernan Badino, Fernando de la Torre

Abstract—We present a new solution to egocentric 3D body pose estimation from monocular images captured from a downward looking fish-eye camera installed on the rim of a head mounted virtual reality device. This unusual viewpoint leads to images with unique visual appearance, characterized by severe self-occlusions and strong perspective distortions that result in a drastic difference in resolution between lower and upper body. We propose a new encoder-decoder architecture with a novel multi-branch decoder designed specifically to account for the varying uncertainty in 2D joint locations. Our quantitative evaluation, both on synthetic and real-world datasets, shows that our strategy leads to substantial improvements in accuracy over state of the art egocentric pose estimation approaches. To tackle the severe lack of labelled training data for egocentric 3D pose estimation we also introduced a large-scale photo-realistic synthetic dataset. $\boldsymbol{x}$R-EgoPose offers 383K frames of high quality renderings of people with diverse skin tones, body shapes and clothing, in a variety of backgrounds and lighting conditions, performing a range of actions. Our experiments show that the high variability in our new synthetic training corpus leads to good generalization to real world footage and to state of the art results on real world datasets with ground truth. Moreover, an evaluation on the Human3.6M benchmark shows that the performance of our method is on par with top performing approaches on the more classic problem of 3D human pose from a third person viewpoint.

Index Terms—3D Human Pose Estimation, Egocentric, VR/AR, Character Animation

1 Introduction

The advent of $\boldsymbol{x}$R technologies (such as AR, VR, and MR) has led to a wide variety of applications in areas such as entertainment, communication, medicine, CAD design, art, and workspace productivity. These technologies mainly focus on immersing the user in a virtual space using a head mounted display (HMD) which renders the environment from the specific viewpoint of the user. However, current solutions have so far focused on the video and audio aspects of the user's perceptual system, leaving a gap in the touch and proprioception senses. Partial solutions to proprioception have been limited to the use of controller devices to track and render hand positions in real time. The 3D pose of the rest of the body is then inferred from inverse kinematics of the head and hand poses [1], but this often results in inaccurate estimates of the body configuration with a large loss of signal that impedes compelling social interaction [2] and even leads to motion sickness [3].

Fig. 1 illustrates the problem that this paper addresses: the goal is to infer 2D and 3D pose information, such as joint positions and rotations, from an egocentric camera perspective, necessary to transfer the motion from the original user to a *generic avatar* or to gather user pose information.

The monocular camera used in our configuration is mounted on the rim of a HMD (as shown in Fig. 1a), approximately $2cm$ away from an average size nose, looking down. Fig. 2 provides a more clear visualization of the unique visual appearance of the images that the camera sees for different body configurations — the top row shows which body parts would become self-occluded from an egocentric viewpoint. The continuous gradation from bright red to dark green encodes the increasing pixel resolution for the corresponding colored area.

There are several challenges that contribute to the difficulty of this problem: *(1)* Strong perspective distortions occur, due to the fish-eye lenses and the proximity of the camera to the face. This results in images with strong radial distortion and drastic difference in image resolution between the upper and lower body (as visible in Fig. 2 — bottom row). Due to this, state-of-the-art approaches for 2D body pose estimation [4] from a frontal or 360 degree yaw view, fail on this type of images; *(2)* There are many instances where body self-occlusion occurs, especially in the lower-body (see right images of Fig. 3), which demands strong spatial awareness of joint locations; *(3)* Egocentric 3D body pose estimation is a relatively unexplored problem in computer vision, hence the scarce availability of publicly accessible labeled datasets; *(4)* As shared by traditional 3D body pose estimation, natural ambiguity is present when lifting 2D joint positions in 3D.

The unusual visual egocentric appearance calls for a new approach and a new training corpus. This paper tackles both. Our novel neural network architecture encodes the difference in uncertainty between upper and lower body joints caused by the varying resolution, extreme perspective effects and self-occlusions. w

We conducted quantitative and qualitative evaluations on both synthetic and real-world benchmarks with ground truth 3D annotations, showing that our approach outperforms previous egocentric state-of-the-art Mo^2Cap^2 [5] by **more than 25%**. In addition, we achieve state-of-the-art performance on the more standard front-facing cameras 3D human pose reconstruction scenario, without any architec-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3029700, IEEE Transactions on Pattern Analysis and Machine Intelligence

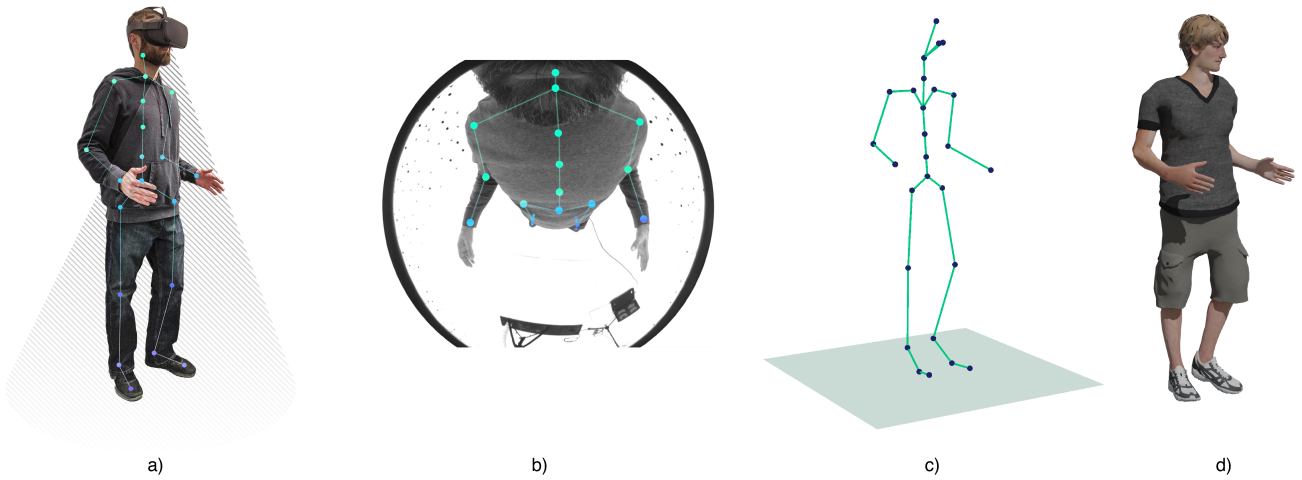

Fig. 1: Egocentric human pose estimation: driving an avatar from an egocentric camera perspective. *b)* Egocentric perspective of the pose visualized in *a)* from an external point of view; *c)* 3D joint locations predicted from the input RGB only-information shown in *b)*; *d)* synthetic character driven from the local joint rotations estimated alonside the 3D locations.

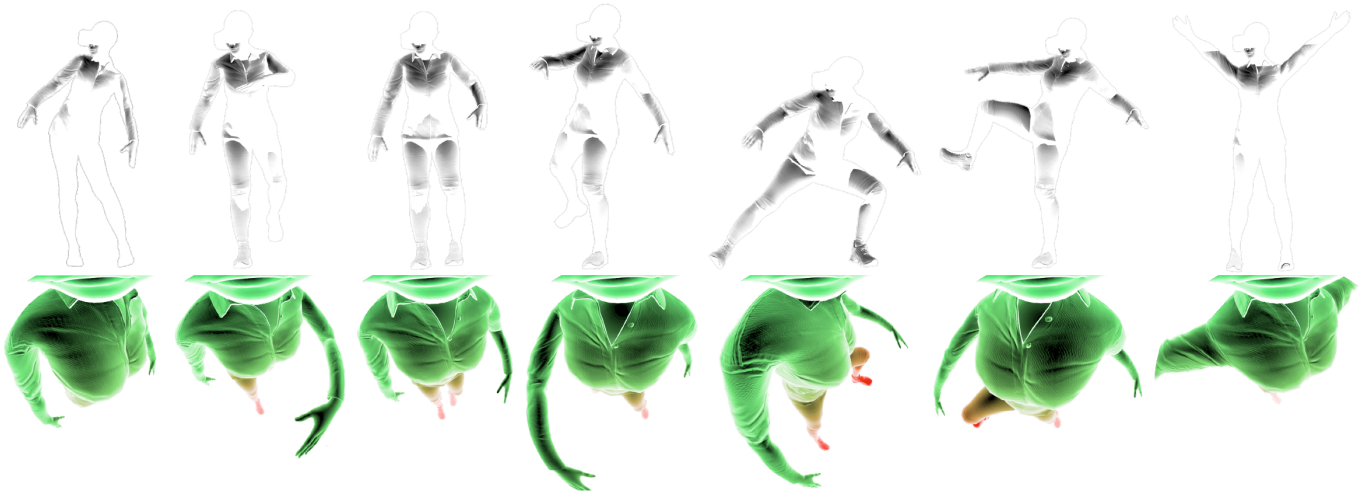

Fig. 2: Visualization of different poses with the same character. **Top:** poses rendered from an external camera viewpoint. White represents occlusion, which is body parts that would not be visible from the egocentric perspective. **Bottom:** poses rendered from the egocentric camera viewpoint. Color gradient indicates the density of image pixels for each area of the body: *green* is higher pixel density, whereas *red* is lower density. This figure illustrates the challenges faced in egocentric human pose estimation: severe self-occlusions, extreme perspective effects and lower pixel density for the lower body.

ture modifications, performing second best after [6] on the Human3.6M benchmark [7].

Our ablation studies show that the introduction of our novel *multi-branch* decoder to reconstruct the 2D input heatmaps and rotations, is responsible for the drastic improvements in 3D pose estimation. Furthermore, the contribution of each of the branches is analyzed, providing tools to control the level of uncertainty embedded in the latent space.

2 Related Work

We describe related work on monocular (single-camera) marker-less 3D human pose estimation focusing on two distinct capture setups: *outside-in* approaches where an external camera viewpoint is used to capture one or more subjects from a distance – the most commonly used setup; and *first person* or egocentric systems where a head-mounted camera observes the own body of the user. While our paper focuses on the second scenario, we build on recent advances in CNN-based methods for human 3D pose estimation. We also describe approaches that incorporate wearable sensors for first person human pose estimation.

Monocular 3D Pose Estimation from an External Camera Viewpoint: the advent of convolutional neural networks and the availability of large 2D and 3D training datasets [7], [8] has recently allowed fast progress in monocular 3D pose estimation from RGB images captured from external cameras. Two main trends have emerged: *(i)* fully supervised

regression of 3D joint locations directly from images [9], [10], [11], [12], [13], [14] and *(ii)* pipeline approaches that decouple the problem into the tasks of 2D joint detection followed by 3D lifting [15], [16], [17], [18], [19], [20], [21], [22], [23], [24]. Progress in fully supervised approaches and their ability to generalize has been severely affected by the limited availability of 3D pose annotations for in-the-wild images. This has led to significant efforts in creating photo-realistic synthetic datasets [25], [26] aided by the recent availability of parametric dense 3D models of the human body learned from body scans [27]. On the other hand, the appeal of two-step decoupled approaches comes from two main advantages: the availability of high-quality off-the-shelf 2D joint detectors [28], [29], [30], [31] that only require easy-to-harvest 2D annotations, and the possibility of training the 3D lifting step using 3D mocap datasets and their ground truth projections without the need for 3D annotations for images. Even simple architectures have been shown to solve this task with a low error rate [15]. Recent advances are due to combining the 2D and 3D tasks into a joint estimation [4], [32], [33], [34] and using weakly [35], [36], [37], [38], [39] or self-supervised losses [40], [41], [42], [43], [44] or mixing 2D and 3D data for training [6], [42], [45], [46].

First Person 3D Human Pose Estimation: while capturing users from an egocentric camera perspective for activity recognition has received significant attention in recent years [47], [48], [49], most methods detect, at most, only upper body motion (hands, arms or torso). Capturing full 3D body motion from head-mounted cameras is considerably more challenging. Some head-mounted capture systems are based on RGB-D input and reconstruct mostly hand, arm and torso motions [50], [51]. Jiang and Grauman [52] reconstruct full body pose from footage taken from a camera worn on the chest by estimating egomotion from the observed scene, but their estimates lack accuracy and have high uncertainty. Yuan *et al*. [53], [54] instead explores a different solution by moving away from kinematics-based representations and using a control-based representation of humanoid motion, commonly used in robotics. A step towards dealing with large parts of the body not being observable was proposed in [55] but for external camera viewpoints. Rhodin *et al*. [56] pioneered the first approach towards full-body capture from a helmet-mounted stereo fish-eye camera pair. The cameras were placed around 25 cm away from the user's head, using telescopic sticks, which resulted in a fairly cumbersome setup for the user but with the benefit of capturing large field of view images where most of the body was in view. A monocular head-mounted systems for full-body pose estimation has more recently been demonstrated by Xu *et al*. [5], who propose a real-time compact setup mounted on a baseball cap, although in this case the egocentric camera is placed a few centimeters further from the user's forehead than in our proposed approach. Our approach substantially outperforms Xu *et al*.'s method [5] by at least 20% on both indoor and outdoor sequences from their real world evaluation dataset. In this journal paper, we go beyond our previous conference paper [57]. First, we perform an extensive analysis on deep architectures for the task of egocentric pose estimation, and show that UNet architectures significantly outperform the originally proposed ResNet architecture [57], specifically for transfer learning from synthetic to real data. Second, we propose a new model which additionally predicts per part rotations. In contrast to [57], this allows us to animate virtual characters, which is necessary for many applications.

3D Pose Estimation from Wearable Devices: Inertial Measurement Units (IMUs) worn by the subject provide a camera-free alternative solution to first person human pose estimation. However, such systems are intrusive and complex to calibrate. While reducing the number of sensors leads to a less invasive configuration [58], [59] recovering accurate human pose from sparse sensor readings becomes a more challenging task. Video data can be fused with IMU [60], [61], [62], [63] to improve accuracy, but these approaches require line of sight with an external camera. An alternative approach, introduced by Shiratori *et al*. [64] consists of a multi-camera structure-from-motion (SFM) approach using 16 limb-mounted cameras. Still very intrusive, this approach suffers from motion blur, automatic white balancing, rolling shutter effects and motion in the scene, making it impractical in realistic scenarios.

3 xR-EgoPose Synthetic Dataset

Ego-3D posed estimation from HMC is a relatively new research problem in computer vision, and to the best of our knowledge there is only one dataset available to analyze the algorithms, see Fig. 3. Existing databases are not rich enough to provide statistical significant analysis due to the scarcity of data. This section describes a photo-realistic synthetic egocentric dataset with ground-truth data, that overcomes some of the limitations of existing approaches.

The design of this dataset focuses on scalability, with augmentation of characters, environments, and lighting conditions. A rendered scene is generated from a random selection of characters, environments, lighting rigs, and animation actions. The animations are obtained from mocap data. A small random displacement is added to the positioning of the camera on the headset to simulate the typical variation of the pose of the headset with respect to the head when worn by the user.

Characters: To improve the diversity of body types, from a single character, we generate additional *skinny short*, *skinny tall*, *full short*, and *full tall* versions. The height distribution of ranges from 155 cm to 189 cm.

Skin: color tones include *white* (Caucasian, freckles or Albino), *light-skinned European*, *dark-skinned European* (darker Caucasian, European mix), *Mediterranean or olive* (Mediterranean, Asian, Hispanic, Native American), *dark brown* (Afro-American, Middle Eastern), and *black* (Afro-American, African, Middle Eastern). Additionally, we built random skin tone parameters into the shaders of each character used with the scene generator.

Clothing: Clothing types include athletic pants, jeans, shorts, dress pants, skirts, jackets, T-Shirts, long sleeves, and tank tops. Shoes include sandals, boots, dress shoes, athletic shoes, crocs. Each type is rendered with different texture and colors.

Actions: the type of actions are listed in Table 1.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3029700, IEEE Transactions on Pattern Analysis and Machine Intelligence

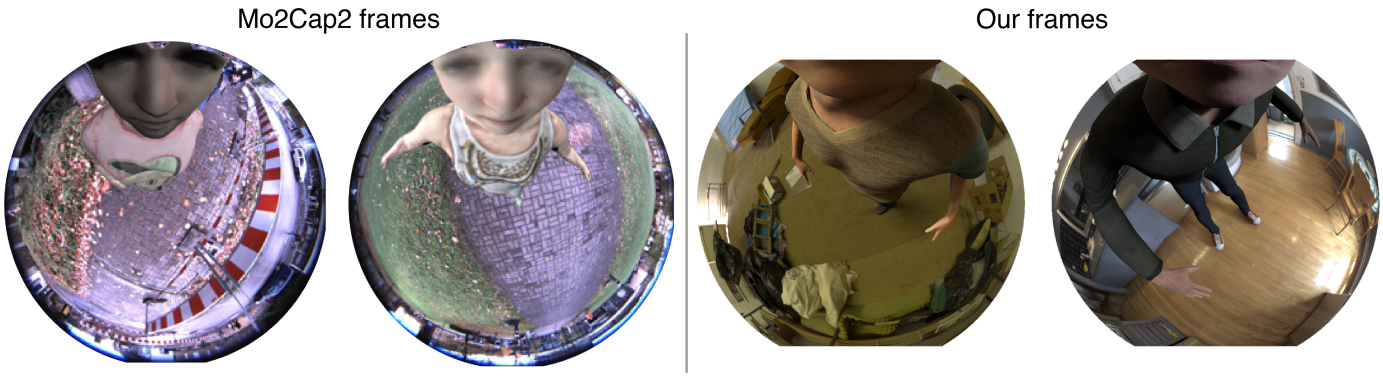

Fig. 3: Example images from our $\boldsymbol{x}$R-EgoPose Dataset compared with the competitor Mo2Cap2 dataset [5]. The quality of our frames is far superior than the randomly sampled frames from mo2cap2, where the characters suffer color matching with respect to the background light conditions.

Images: the images have a resolution of 1024×1024 pixels and 16-bit color depth. For training and testing, we downsample the color depth to 8 bit. The frame rate is 30 fps. *RGB*, *depth*, *normals*, *body segmentation*, and *pixel world position* images are generated for each frame, with the option for exposure control for augmentation of lighting. Metadata is provided for each frame including 3D joint positions, height of the character, environment, camera pose, body segmentation, and animation rig.

Render quality: Maximizing the photo-realism of the synthetic dataset was our top priority. Therefore, we animated the characters in Maya using actual mocap data [65], and used a standardized physically based rendering setup with V-Ray. The characters were created with global custom shader settings applied across clothing, skin, and lighting of environments for all rendered scenes.

3.1 Training, Test, and Validation Sets

The dataset has a total size of 383K frames, with 23 male and 23 female characters, divided into three sets: *Train-set:* 252K frames; *Test-set:* 115K frames; and *Validation-set:* 16K frames. The gender distribution is: *Train-set:* 13M/11F, *Test-set:* 7M/5F and *Validation-set:* 3M/3F. Table 1 provides a detailed description of the partitioning of the dataset according to the different actions.

Action	N. Frames	Size Train	Size Test
Gaming	24019	11153	4684
Gesticulating	21411	9866	4206
Greeting	8966	4188	1739
Lower Stretching	82541	66165	43491
Patting	9615	4404	1898
Reacting	26629	12599	5104
Talking	13685	6215	2723
Upper Stretching	162193	114446	46468
Walking	34989	24603	9971

TABLE 1: Total number of frames per action and their distribution between train and test sets. Everything else not mentioned is validation data.

4 Architecture

This section describes the deep learning architecture for 3D pose estimation. The proposed architecture (Fig. 4), is a two step approach consisting of two main modules: *i)* the first module detects 2D heatmaps of the locations of the body joints in image space. We experiment with different standard architectures, please refer to Sec. 5 for details; *ii)* the second one takes as input the 2D heatmap predictions generated from the preceding module and regresses the 3D coordinates of the body joints, local joint rotations according to the skeleton hierarchy and reconstructed heatmap predictions, using a novel *multi-branch auto-encoder* architecture.

One of the most important advantages of this pipeline approach is that 2D and 3D modules can be trained independently according to the available training data. For instance, if a sufficiently large corpus of images with 3D annotations is not available, the 3D lifting module can be trained independently using 3D mocap data and its projected heatmaps. Once the two modules are pretrained the entire architecture can be fine-tuned end-to-end since it is fully differentiable. The *multi-branch* auto-encoder module gives also the ability of having multiple representations of the pose: e.g. joint positions, local rotations, etc. A further advantage of this architecture is that the second and third branches are only needed at training time (see Sec. 4.2) and can be removed at test time, guaranteeing the better performance and a faster execution.

4.1 2D Pose Detection

Given an RGB image $\mathbf{I} \in \mathbb{R}^{368\times368\times3}$ as input, the 2D pose detector infers 2D poses, represented as a set of heatmaps $\mathbf{HM} \in \mathbb{R}^{47\times47\times15}$, one for each of the body joints. For this task we have experimented with different standard architectures including *ResNet 50* [66] and *U-Net* [67]. For a detailed analysis, please refer to Sec. 5.

The models were trained using normalized input images, obtained by subtracting the mean value and dividing by the standard deviation, and using the mean square error of the difference between the ground truth heatmaps and the predicted ones as the loss:

$$\mathrm{L}_{2D} = \mathrm{mse}(\mathbf{HM}, \widehat{\mathbf{HM}}) \quad (1)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3029700, IEEE Transactions on Pattern Analysis and Machine Intelligence

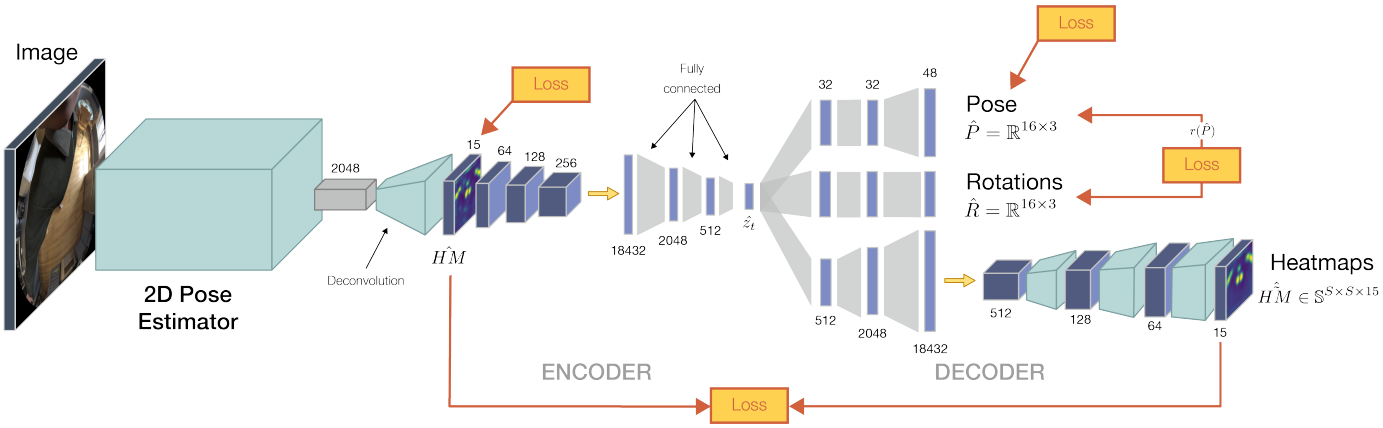

Fig. 4: Proposed architecture for egocentric 3D human pose estimation consisting of two modules: *a)* interchangeable 2D pose detector that predicts heatmaps from the input RGB image; *b)* multi-branch auto-encoder that finds a representation of poses which includes also a level of uncertainty of predictions per joint. Alongside the main branch, for 3D joint location prediction, two auxiliary branches as used at training-time to improve latent space distribution. Branch *ii)* estimates local joint rotations, forcing them to be consistent with those rotations extracted by the predicted pose from *i)*; branch *iii)* forces the latent space to include a level of uncertainty of the 2D joint locations by reconstructing the given predicted heatmaps from the pose embedding. These additional branches have demonstrated considerable improvements with respect to a standard AE architecture, as shown in Sec. 5.

4.2 2D-to-3D Mapping

The 3D pose module takes as input the 15 heatmaps computed by the first module and outputs the final 3D pose $\mathbf{P} \in \mathbb{R}^{16\times 3}$ as a set of joint locations. Note that the number of output 3D joints is 16 since we include the head which despite being out of the field of view it can be regressed in 3D.

In most pipeline approaches the *3D lifting module* usually is given as input the 2D joint pixel positions in the image of the detected the 3D position. Instead, similarly to Pavlakos *et al*. [39], our approach predicts the 3D pose from input heatmaps, not just 2D locations. The main advantage is that these heatmaps carry important information relative to the *uncertainty of the 2D* pose estimations. Furthermore, due to the unique architecture, it is possible to change the different levels or representation of a pose, afterwards.

The main novelty of the proposed architecture (see Fig. 4), is that we ensure that the uncertainty information expressed in the heatmap representations does not get lost but it is preserved in the pose embedding. While the encoder takes as input a set of heatmaps and encodes them into the embedding $\hat{\mathbf{z}}$, the decoder has multiple branches – *1st* regresses the 3D pose from $\hat{\mathbf{z}}$; *2nd* estimates the local joint rotations (with respect to the parent node); and *3rd* reconstructs the input heatmaps. The purpose of this branch is to force the latent vector to encode the probability density function of the estimated 2D heatmaps.

The overall loss function for the auto-encoder is expressed as

$$\begin{aligned} \mathrm{L_{AE}} \quad = \quad & \lambda_p(||\mathbf{P}-\hat{\mathbf{P}}||^2 + W(\mathbf{P},\hat{\mathbf{P}})) + \\ & \lambda_r||\hat{\mathbf{R}} - r(\hat{\mathbf{P}})||^2 + \\ & \lambda_{hm}||\widehat{\mathbf{HM}} - \widetilde{\mathbf{HM}}||^2 \end{aligned} \tag{2}$$

with $\mathbf{P}$ the ground truth; $\hat{\mathbf{R}}$ the predicted local joint rotations and $r(\hat{\mathbf{P}})$ the function that estimates local joint rotations from a given pose; $\widetilde{\mathbf{HM}}$ is the set of heatmaps regressed by the decoder from the latent space and $\widehat{\mathbf{HM}}$ are the heatmaps regressed by 2D pose estimator module (see Sec. 4.1). Different local joint rotation representations were tested and ultimately a Quaternion representation was chosen due to the stability of the rotations during training, leading to more robust models. The rotation branch also helps generating better results as shown in Sec. 5 with smoother transitions on consecutive frames on poses estimated frame-by-frame.

Finally W is the regularizer over the 3D poses

$$W(\mathbf{P},\hat{\mathbf{P}}) = \lambda_\theta \theta(\mathbf{P},\hat{\mathbf{P}}) + \lambda_L \mathrm{L}(\mathbf{P},\hat{\mathbf{P}})$$

with

$$\theta(\mathbf{P},\hat{\mathbf{P}}) = \sum_l^L \frac{\mathbf{P}_l \cdot \hat{\mathbf{P}}_l}{||\mathbf{P}|| * ||\hat{\mathbf{P}}_l||} \quad \mathrm{L}(\mathbf{P},\hat{\mathbf{P}}) = \sum_l^L ||\mathbf{P}_l - \hat{\mathbf{P}}_l||$$

corresponding to the cosine-similarity error and the limb-length error, with $\mathbf{P}_l \in \mathbb{R}^3$ the l^{th} limb of the pose. An important advantage of this loss is that the model can be trained on a mix of 3D and 2D datasets simultaneously: if an image sample only has 2D annotations then $\lambda_p = 0$ and $\lambda_r = 0$, such that only the heatmaps are contributing to the loss. In Section 5.7 we show how having a larger corpus of 2D annotations can be leveraged to improve final 3D body pose estimates.

4.3 Training Details

The model has been trained on the entire training set for 3 epochs, with a learning rate of $1e-3$ using batch normalization on a mini-batch of size 16. The deconvolutional layer used to identify the heatmaps from the features computed by *ResNet* has kernel size $=$ 3 and stride $=$ 2. The convolutional and deconvolutional layers of the encoder have kernel size $=$ 4 and stride $=$ 2. Finally, all the layers of

the encoder use leakly ReLU as activation function with 0.2 leakiness. The λ weights used in the loss function were identified through grid search and set to $\lambda_{hm} = 10^{-3}$, $\lambda_p = 10^{-1}$, $\lambda_r = 10^{-1}$ $\lambda_\theta = -10^{-2}$ and $\lambda_L = 0.5$. The model has been trained from scratch with Xavier weight initializer.

5 Experimental Evaluation

In the following, we thoroughly evaluate our proposed approach on our novel $\boldsymbol{x}$R-EgoPose dataset, we perform parameter and architecture ablations, and we evaluate on the real-world Mo^{2}Cap^2 test-set [5] which includes 2.7K frames of real images with ground truth 3D poses of two people captured in indoor and outdoor scenes. In addition, we show qualitative results on our controlled small-scale real-world dataset and demonstrate how our approach can be used to animate virtual characters for $\boldsymbol{x}$R telepresence. Finally, we evaluate quantitatively on the Human3.6M dataset to show that our architecture generalizes well without any modifications to the case of an external camera viewpoint.

Evaluation protocol: Unless otherwise mentioned, we report the Mean Per Joint Position Error - MPJPE:

$$E(\mathbf{P}, \hat{\mathbf{P}}) = \frac{1}{N_f} \frac{1}{N_j} \sum_{f=1}^{N_f} \sum_{j=1}^{N_j} ||\mathbf{P}_j^{(f)} - \hat{\mathbf{P}}_j^{(f)}||_2 \qquad (3)$$

where $\mathbf{P}_j^{(f)}$ and $\hat{\mathbf{P}}_j^{(f)}$ are the 3D points of the ground truth and predicted pose at frame f for joint j, out of N_f number of frames and N_j number of joints.

To ensure high reproducibility of our results on our novel synthetic $\boldsymbol{x}$R-EgoPose dataset, we first evaluate our method on a randomly initialized *ResNet 50*. We intentionally do not perform any pre-training strategies given that, as we show in Sec. 5.3, this affects the final results. Our goal is to establish our $\boldsymbol{x}$R-EgoPose dataset as a benchmark and therefore report reproducible numbers that have been computed using a standard network architecture, trained with a simple protocol, cf. Sec. 4.3.

5.1 Evaluation on our Egocentric Synthetic Dataset

Evaluation on $\boldsymbol{x}$R-EgoPose test-set: Firstly, we evaluate our approach on the test-set of our synthetic $\boldsymbol{x}$R-EgoPose dataset. We show qualitative results in Fig. 9. Unfortunately, it was not possible to establish a comparison on our dataset with state of the art monocular egocentric human pose estimation methods such as Mo^{2}Cap^2 [5] given that their code has not been made publicly available. Instead we compare with Martinez *et al.* [15], a recent state of the art method for a traditional external camera viewpoint. For a fair comparison, the training-set of our $\boldsymbol{x}$R-EgoPose dataset has been used to re-train the model of Martinez *et al.*. This way we can directly compare the performance of the 2D to 3D modules.

Table 2 reports the MPJPE (Eq. 3) for both methods showing that our approach (Ours-dual-branch) outperforms Martinez *et al.*'s by 36.4% in the upper body reconstruction, 60% in the lower body reconstruction, and 52.3% overall, showing a considerable improvement.

Reconstruction errors per joint type: Table 3 reports a decomposition of the reconstruction error into different individual joint types. The highest errors are in the hands and feet. This observation is in accordance with the fact that hands and feet are often not or only barely visible. Hands can go out of the camera field of view e.g. by lifting or stretching the arms or may be occluded by the body. Feet are only visible when the subject looks slightly down and always cover only a very small portion of the image, due to the strong distortion. Nevertheless, our method always predicts plausible poses, even for high occlusions as displayed in Fig. 6, Fig. 9 and Fig. 10.

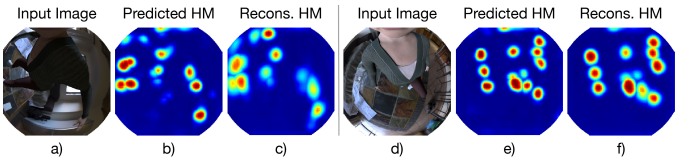

Fig. 5: Reconstructed heatmaps generated by the decoder branch which can reproduce the correct uncertainty of the 2D input predictions from the pose embedding.

Effect of the decoder branches: Table 2 reports an ablation study to compare the performance of three versions of our approach. We report results using: *i)* only 3D pose supervision only (Ours — p3d); *ii)* additional supervision on regressed rotations (Ours — p3d+rot); *iii)* and on regressed heatmaps (Ours – p3d+hm); finally for our novel *multi-branch* auto-encoder supervised on all three signals (Ours — p3d+hm+rot).

The overall average error of the single branch encoder is 130.4 mm, far from the 54.7 mm error achieved by our novel *multi-branch* architecture. The dual branch encoders produce an error of 91.2 mm and 58.2 mm, respectively. Ours results clearly demonstrate that all branches contribute to our final result. Both, forcing the network to encode uncertainty of the 2D joint estimates by regressing heatmaps, as well as preserving the limb orientation information by regressing rotations, helps to estimate better 3D poses.

Encoding uncertainty in the latent space: Figure 5 demonstrates the ability of our approach to encode the uncertainty of the input 2D heatmaps in the latent vector. Examples of input 2D heatmaps and those reconstructed by the second branch of the decoder are shown for comparison.

5.2 Character Animation Using Estimated Rotations

The pose embedding estimation generated by the *multi-branch* auto-encoder architecture contains the relevant essential information of a pose, which grants us the ability to change / add a representation based the a specific application. Specifically, the introduction of the rotation branch improves the overall reconstruction error, as demonstrated in Table 2, and it is a pose definition usable for character animation.

The joint rotations estimated by the rotation-branch are expressed as local-rotations of each joint with respect to the parent node according to the skeleton hierarchy. Several rotation representations have been tested, including Euler angles, Rotation Matrices, Quaternions and the approach proposed by Zhou *et al.* [68]. We have not noticed any relevant improvements between Quaternions and [68], however

Approach	Gaming	Gesticulating	Greeting	Lower Stretching	Patting	Reacting	Talking	Upper Stretching	Walking	All (mm)
Martinez [15]	109.6	105.4	119.3	125.8	93.0	119.7	111.1	124.5	130.5	122.1
Ours — p3d	138.3	108.5	100.3	133.3	117.8	175.6	93.5	129.0	131.9	130.4
Ours — p3d+rot	110.7	90.9	91.9	119.1	98.6	106.8	86.9	88.0	88.2	91.2
Ours — p3d+hm	56.0	50.2	44.6	51.1	59.4	60.8	43.9	53.9	57.7	58.2
Ours — p3d+hm+rot	60.4	54.6	44.7	56.5	57.7	52.7	56.4	53.6	55.4	**54.7**

TABLE 2: Quantitative evaluation with Martinez *et al.* [15], a state-of-the-art approach developed for front-facing cameras. Both upper and lower body reconstructions are shown as well. A comparison with our own architecture where different configurations are analyzed. Specifically, the impact of the additional branches is evaluated. Note how the competing approach fails consistently across different actions in lower body reconstructions. This experiment emphasizes how, even a state-of-the-art 3D lifting method developed for external cameras fails on this challenging task. It also emphasizes the contribution of encoding uncertainty for achieving low-reconstruction errors.

the latter demands a larger number of components per joint to express rotations.

Example frames showing the *driven character* compared against the original animation are sown in Fig. 6. Notice how the model is able to reliably estimate the correct rotations even for poses where the avatar's limbs fall outside of the camera's field-of-view. Furthermore, there is temporal consistency between poses in consecutive frames despite estimations being computed frame-by-frame.

Fig. 7 shows joint angle predictions, estimated from input images, through time. Specifically, joint angles are consistent with the ground truth. The rotations are smooth and limited "jittering" artefacts are introduced by the network in the predictions.

5.3 Heatmap Estimation: Architecture Ablation

So far, we have used the established *ResNet 50* [66] architecture in all our experiments. In order to study the effect of the heatmap estimation network, we experiment with different architectures and initialization strategies. Specifically, we experiment with *ResNet 50* [66] and *U-Net* [67]. We use *ResNet 50* in two variants: randomly initialized using Xavier initialization [69] and pre-trained on ImageNet [70]. The *U-Net* is composed from a *ResNet 18* backbone encoder, pre-trained on ImageNet, and a randomly initialized decoder. The *ResNet 50* consists of 24.2 million trainable parameters. The *U-Net* contains 18.3 million parameters. All variants produce the same heatmap resolution for better comparison. The lifting networks share the same architecture and number of parameters, but have been trained specifically for each 2D pose estimation network, to accommodate its unique heatmap properties. We additionally experimented with *ResNet 101* [66], *Convolutional Pose Machines* [28], and *Stacked Hourglass Network* [29]. These experiments resulted in comparable performance at a higher computational cost compared to *ResNet 50*, and are therefore not discussed in following.

Joint	Error (mm)	Joint	Error (mm)
Left Leg	34.33	Right Leg	33.85
Left Knee	62.57	Right Knee	61.36
Left Foot	70.08	Right Foot	68.17
Left Toe	76.43	Right Toe	71.94
Neck	6.57	Head	23.20
Left Arm	31.36	Right Arm	31.45
Left Elbow	60.89	Right Elbow	50.13
Left Hand	90.43	Right Hand	78.28

TABLE 3: Average reconstruction error per joint using Eq. 3, evaluated on the entire test-set (see Sec. 3) with model trained using only synthetic data.

Our experiments suggest that pre-training helps. The full pipeline using a pre-trained *ResNet 50* improves the MPJPE error to 51.1 mm, compared to 54.7 for random initialization, see Tab. 4. While a recent work [71] suggests that pre-training usually is not necessary, the authors describe two aspects where pre-training does help. First, pre-training helps faster convergence. Second, for small datasets, pre-training helps to improve accuracy. While our synthetic dataset is large, it features less variability in scenes and subjects, compared to large real-world datasets like e.g. MPII [8].

In a next step, we experiment using a *U-Net* for 2D pose estimation. Using a *U-Net* architecture boosts the performance of our pipeline and significantly improves the MPJPE error to 41.0 mm. Empirically, we found that the *U-Net*-based 2D pose estimator also generalizes, to a certain extent, to real data, predicting plausible heatmaps for unseen data, while only having been trained on our synthetic dataset. The *Resnet 50*-based estimator fails without prior refinement. We hypothesize, that the improved performance, and the observed behavior on real images, demonstrate better generalization properties of the *U-Net*. To support our hypothesis, we perform an additional experiment. We add white Gaussian noise to the test images of our synthetic dataset and measure the performance of our pipeline using the different 2D pose estimation networks. In Fig. 8 we plot the MPJPE error under various levels of noise. Notably, the error of the *U-Net*-based pipeline increases slowly, while *Resnet 50*-based pipelines produce large errors already under small noise levels. This behavior supports our hypothesis that the *U-Net* architecture features better generalization properties.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2020.3029700, IEEE Transactions on Pattern Analysis and Machine Intelligence

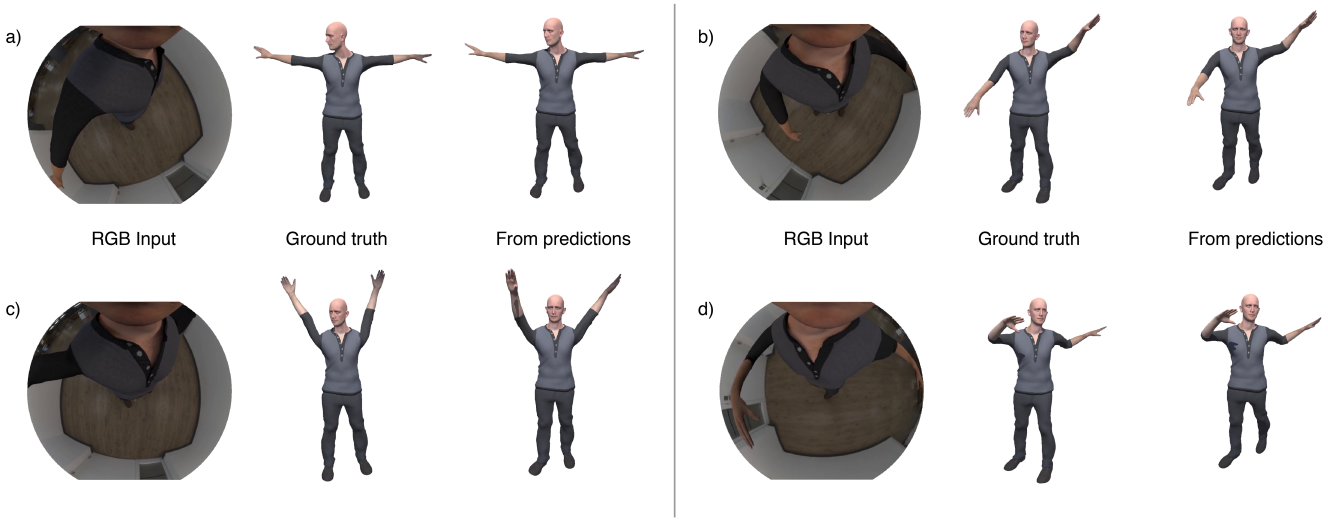

Fig. 6: Character animation from the joint local rotation predictions computed from the input image. Note how the model is able to retrieve most of the desired information even when limbs fall outside the camera field of view.

Configuration	Gaming	Gesticulating	Greeting	Lower Stretching	Patting	Reacting	Talking	Upper Stretching	Walking	All (mm)
ResNet 50	60.4	54.6	**44.7**	56.5	57.7	52.7	56.4	53.6	55.4	54.7
ResNet 50 (p)	**51.6**	**44.6**	64.6	52.4	**50.8**	**44.0**	46.5	51.4	52.8	51.1
U-Net (p)	52.5	49.2	72.0	**37.3**	53.0	44.4	**46.1**	**39.3**	**37.2**	**41.0**

TABLE 4: Performance analysis: different combinations of 2D pose detectors combined with the *multi-branch* lifting network. All variants have been trained and tested on the synthetic dataset. Variants with (p) have been pre-trained on ImageNet.

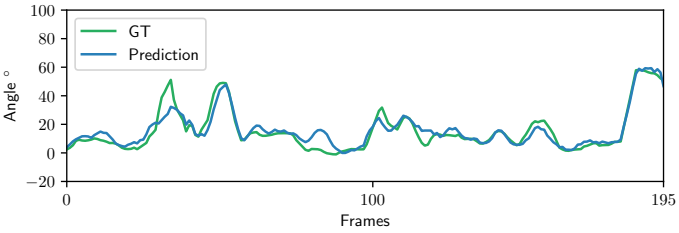

Fig. 7: Analysis of the angle predictions through time for the Righ Foot in sequence of the test-set.

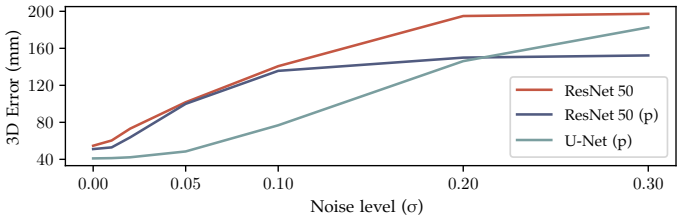

Fig. 8: Performance of our proposed pipeline using different 2D pose estimation networks under the influence of white Gaussian noise in the image domain. Networks with (p) have been pretrained on ImageNet.

5.4 Lifting Network: Parameter Ablation

In order to validate the architecture design choices of our *multi-branch* 3D pose lifting network, we perform an ablation study of two main parameters.

First, we find the optimal size of the embedding $\hat{\mathbf{z}}$, that encodes the 3D pose, the joint rotations, and the 2D pose uncertainty. Table 6 lists the MPJPE error using different sizes for $\hat{\mathbf{z}}$ for all three different heatmap estimation networks. Regardless of the choice of the heatmap estimation network, we find that $\hat{\mathbf{z}} \in \mathbb{R}^{50}$ produces the best results. Smaller embeddings produce significantly higher errors, while larger embeddings only slightly impair the results.

Further, we study how the dimensions of the regressed heatmaps $\widetilde{\mathbf{HM}}$ influence the results, see 5. Unsurprisingly, we find that regressing the full heatmap produces the best results. This is in accordance with the experiments in Sec. 5.5, where we show that encoding uncertainty via regressing heatmaps helps over using them only as input.

To contribute towards fostering fairness in Computer Vision and Machine Learning we analyze the performance of the proposed models on our diverse dataset based on different skin tones. A comparison is shown in Table 7.

5.5 Evaluation on Egocentric Real Datasets

Comparison with Mo^{2}Cap^2 [5]: We compare the results of our approach with those given by our direct competitor, Mo^{2}Cap^2, on their real world test set including both indoor and outdoor sequences. For a fair comparison, we train our model solely on their provided synthetic training data (cf. Fig. 3). Table 8 reports the MPJPE errors for both methods. Our dual-branch approach substantially outperforms

Mo^2Cap^2 [5] in both indoor and outdoor scenarios. Here again, our approach using the *U-Net* model pre-trained on ImageNet produces the best results. However, indoors in a more controlled setting, both our architecture variants are almost on par. Note that comparison with the stereo egocentric system EgoCap [56] on their dataset is not meaningful, due to the hugely different camera position relative to the head (their stereo cameras are 25 cm from the head).

Evaluation on $\boldsymbol{x}$R-EgoPose$^{\mathbf{R}}$: The $\sim$ 10K frames of our small scale real-world data set were captured from a fish-eye camera mounted on a VR HMD worn by three different actors wearing different clothes, and performing 6 different actions. The ground truth 3D poses were acquired using a custom mocap system. The network was trained on our synthetic corpus ($\boldsymbol{x}$R-EgoPose) and fine-tuned using the data from two of the actors. The test set contained data from the unseen third actor. $\boldsymbol{x}$R-EgoPose$^{\mathrm{R}}$ is too small for meaningful numerical evaluation. However, we show qualitative examples of the input views and the reconstructed poses in Fig. 10. These results show good generalization of the model (trained mostly on synthetic data) to real images.

5.6 Evaluation on Front-facing Cameras

Comparison on Human3.6M dataset: We show that our proposed approach is not specific for the egocentric case, but also provides excellent results in the more standard case of front-facing cameras. For this evaluation, we chose the Human3.6M dataset [7], [73]. We used two evaluation protocols. *Protocol 1* has five subjects (S1, S5, S6, S7, S8) used in training, with subjects (S9, S11) used for evaluation. The MPJPE error is computed on every 64th frame. *Protocol 2* contains six subjects (S1, S5, S6, S7, S8, S9) used for training, and the evaluation is performed on every 64th frame of Subject 11 (Procrustes aligned MPJPE is used for evaluation). The results are shown in Table 9 from where it can be seen that our approach is on par with state-of-the-art methods, scoring second overall within the non-temporal methods.

5.7 Mixing 2D and 3D Ground Truth Datasets

An important advantage of our architecture is that the model can be trained on a mix of 3D and 2D datasets simultaneously: if an image sample only has 2D annotations but no 3D ground truth labels, the sample can still be used, only the heatmaps will contribute to the loss. We evaluated the effect of adding additional images with 2D but no 3D labels on both scenarios: egocentric and front-facing cameras. In the egocentric case we created two subsets of the $\boldsymbol{x}$R-EgoPose test-set. The first subset contained 50% of all the available image samples with both 3D and 2D labels. The second contained 100% of the image samples with 2D labels, but only 50% of the 3D labels. Effectively the second subset contained twice the number of images with 2D annotations only. Table 10a compares the results between the subsets, where it can be seen that the final 3D pose estimate benefits from additional 2D annotations. Equivalent behavior is seen on the Human3.6M dataset. Table 10b shows the improvements in reconstruction error when additional 2D annotations from COCO [83] and MPII [8] are used.

$\hat{\mathbf{z}}$ size	Error (mm)		
	ResNet50	ResNet50 (p)	UNet (p)
10	70.6	61.0	45.8
20	67.3	52.5	45.3
50	**54.7**	**51.1**	**41.0**
70	55.7	54.5	41.6
100	58.9	54.2	41.3
500	61.0	56.0	41.2

TABLE 5: Average reconstruction error per joint using Eq. 3, evaluated on the entire test-set when the model architecture differs based on the size of the embedding $\hat{\mathbf{z}}$.

6 Conclusion

We have presented a solution to the problem of 3D body pose estimation from a monocular camera installed on a HMD. Given a single image, our fully differentiable network estimates heatmaps and uses them as an intermediate representation to regress 3D poses using a novel *multi-branch* auto-encoder. This new architecture design was fundamental for accurate reconstructions in our challenging dataset, with over 24% accuracy improvement on competitor datasets and that proves to generalize to the more generic 3D human pose estimation from front-facing cameras task with state-of-the-art performance. We have shown how the proposed architecture can be used to drive a virtual avatar directly from the estimations of the network, a fundamental step towards telepresence in virtual or augmented reality.

Finally, we have also introduced the $\boldsymbol{x}$R-EgoPose dataset, a new large scale photo-realistic synthetic dataset that was essential for training and will be made publicly available to

HM size	Error (mm)		
	ResNet50	ResNet50 (p)	UNet (p)
48	**54.7**	**51.1**	**41.0**
36	57.8	59.6	44.2
24	59.9	57.7	43.8
16	61.2	56.8	41.4
8	61.4	56.7	41.7

TABLE 6: Average reconstruction error per joint using Eq. 3, evaluated on the entire test-set for different heatmap (HM) reconstruction sizes. Notice how little uncertainty information still has dramatic impact on the reconstruction accuracy.

Skin tone	Error (mm)		
	ResNet50	ResNet50 (p)	UNet (p)
White	42.7	46.5	46.3
Light European	61.9	58.2	43.5
Dark European	63.6	52.0	35.6
Dark brown	22.5	28.7	27.5
Black	89.0	68.8	42.7

TABLE 7: Model evaluation based on skin tones.

INDOOR	walking	sitting	crawling	crouching	boxing	dancing	stretching	waving	total (mm)
3DV'17 [14]	48.76	101.22	118.96	94.93	57.34	60.96	111.36	64.50	76.28
VCNet [72]	65.28	129.59	133.08	120.39	78.43	82.46	153.17	83.91	97. 85
Xu [5]	38.41	70.94	94.31	81.90	48.55	55.19	99.34	60.92	61.40
Ours - ResNet 50	**38.39**	61.59	69.53	51.14	**37.67**	**42.10**	58.32	**44.77**	48.16
Ours - U-Net (p)	45.83	**47.24**	**47.35**	**45.15**	48.72	47.00	**46.15**	46.45	**46.61**
OUTDOOR	walking	sitting	crawling	crouching	boxing	dancing	stretching	waving	total (mm)
3DV'17 [14]	68.67	114.87	113.23	118.55	95.29	72.99	114.48	72.41	94.46
VCNet [72]	84.43	167.87	138.39	154.54	108.36	85.01	160.57	96.22	113.75
Xu [5]	63.10	85.48	96.63	92.88	96.01	68.35	123.56	61.42	80.64
Ours - ResNet 50	**43.60**	85.91	83.06	69.23	69.32	**45.40**	76.68	**51.38**	60.19
Ours - U-Net (p)	53.96	**52.24**	**55.50**	**55.65**	**54.38**	54.48	**54.46**	56.12	**54.61**

TABLE 8: Quantitative evaluation on Mo^2Cap^2 dataset [5], both indoor and outdoor test-sets. Our approach outperforms all competitors by more than **21.6%** (13.24 mm) on indoor data and more than **25.4%** (20.45 mm) on outdoor data when using only the provided synthetic data for training the model. Similarly to other experiments we provide in Sec 5, when using a pre-trained U-Net model with the configuration defined as in Sec 5.3, results improve even further: **24.9%** (14.79 mm) and **32.28%** (26.03 mm) respectively.

Protocol #1	Chen [74]	Hossain [75]*	Dabral [76]*	Tome [36]	Moreno [16]	Kanazawa [77]	Zhou [78]	Jahangiri [79]	Mehta [14]	Martinez [15]	Fang [80]	Sun [81]	Sun [6]	**Ours**
Errors (mm)	114.2	51.9	52.1	88.4	87.3	88.0	79.9	77.6	72.9	62.9	60.4	59.1	**49.6**	51.3
Protocol #2	Yasin [82]	Hossain [75]*	Dabral [76]*	Rogez [25]	Chen [74]	Moreno [16]	Tome [36]	Zhou [78]	Martinez [15]	Kanazawa [77]	Sun [81]	Fang [80]	Sun [6]	**Ours**
Errors (mm)	108.3	42.0	36.3	88.1	82.7	76.5	70.7	55.3	47.7	58.8	48.3	45.7	**40.6**	42.3

TABLE 9: Comparison with other state-of-the-art approaches on the Human3.6M dataset (front-facing cameras). Approaches with * make use of temporal information. No specific modifications have been applied to our architecture: UNet 2D pose detector pre-trained on ImageNet has been used to estimate joint-heatmaps fed through our dual-branch auto-encoder architecture, since rotation information is not available for these data.

promote research in this exciting area. While our results are state-of-the-art, there are a few failures cases due to extreme occlusion and the inability of the system to measure hands when they are out of the field of view. Adding additional cameras to cover more field of view and enable multi-view sensing is the focus of our future work.

REFERENCES

[1] https://medium.com/@DeepMotionInc/how-to-make-3-point-tracked-full-body-avatars-in-vr-34b3f6709782. (last accessed on 2019-03-19) How to make 3 point tracked full-body avatars in vr, https://medium.com/@deepmotioninc/how-to-make-3-point-tracked-full-body-avatars-in-vr-34b3f6709782. [Online]. Available: https://medium.com/@DeepMotionInc/how-to-make-3-point-tracked-full-body-avatars-in-vr-34b3f6709782

[2] U. Hess, K. Kafetsios, H. Mauersberger, C. Blaison, and C.-L. Kessler, "Signal and noise in the perception of facial emotion expressions: From labs to life," *Personality and Social Psychology Bulletin*, vol. 42, no. 8, pp. 1092–1110, 2016.

3D	2D	Error (mm)
50%	50%	68.04
50%	100%	63.98

(a) $\boldsymbol{x}$R-EgoPose

Training dataset	Error (mm)
H36M	67.9
H36M + COCO + MPII	53.4

(b) Human3.6M

TABLE 10: Having a larger corpus of 2D annotations can be leveraged to improve final 3D pose estimation

[3] J. T. Reason and J. J. Brand, *Motion sickness*. Academic press, 1975.

[4] G. Rogez, P. Weinzaepfel, and C. Schmid, "LCR-Net++: Multi-person 2D and 3D Pose Detection in Natural Images," *CoRR*, vol. abs/1803.00455v1, 2018.

[5] W. Xu, A. Chatterjee, M. Zollhoefer, H. Rhodin, P. Fua, H.-P. Seidel, and C. Theobalt, "Mo^2Cap^2 : Real-time mobile 3d motion capture with a cap-mounted fisheye camera," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2019.

[6] X. Sun, B. Xiao, F. Wei, S. Liang, and Y. Wei, "Integral human pose regression," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 529–545.

[7] C. Ionescu, D. 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*, vol. 36, no. 7, pp. 1325–1339, 2014.

[8] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2d human pose estimation: New benchmark and state of the art analysis," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[9] S. Li and A. B. Chan, "3d human pose estimation from monocular images with deep convolutional neural network," in *Asian Conference on Computer Vision*. Springer, 2014, pp. 332–347.

[10] S. Park, J. Hwang, and N. Kwak, "3d human pose estimation using convolutional neural networks with 2d pose information," in *European Conference on Computer Vision, Workshops*. Springer, 2016, pp. 156–169.

[11] B. Tekin, I. Katircioglu, M. Salzmann, V. Lepetit, and P. Fua, "Structured prediction of 3d human pose with deep neural networks," in *British Machine Vision Conference (BMVC)*, 2016.

[12] X. Zhou, X. Sun, W. Zhang, S. Liang, and Y. Wei, "Deep kinematic pose regression," in *European Conference on Computer Vision*.

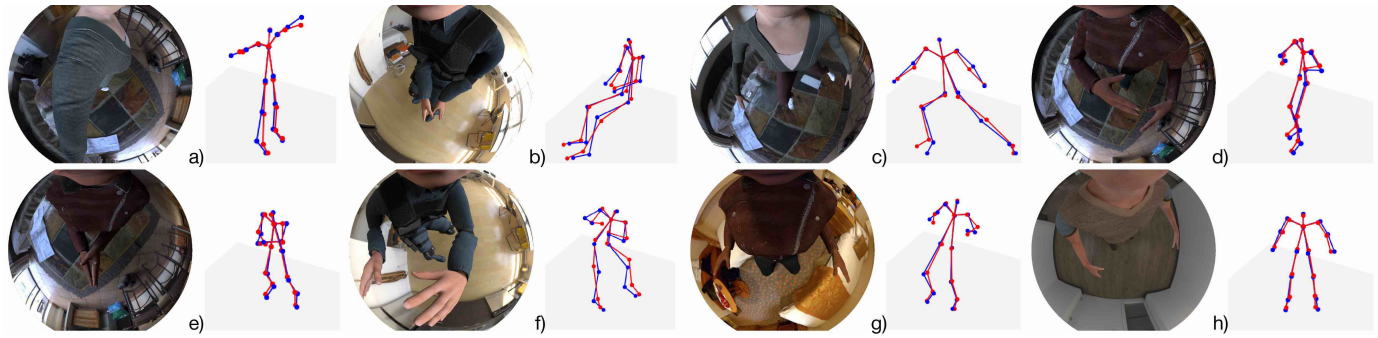

Fig. 9: Qualitative results on synthetic images from our synthetic test-set. Note that the poses are expressed with respect to the camera reference system. Blue poses represent ground truth, whereas poses in red correspond to predictions.

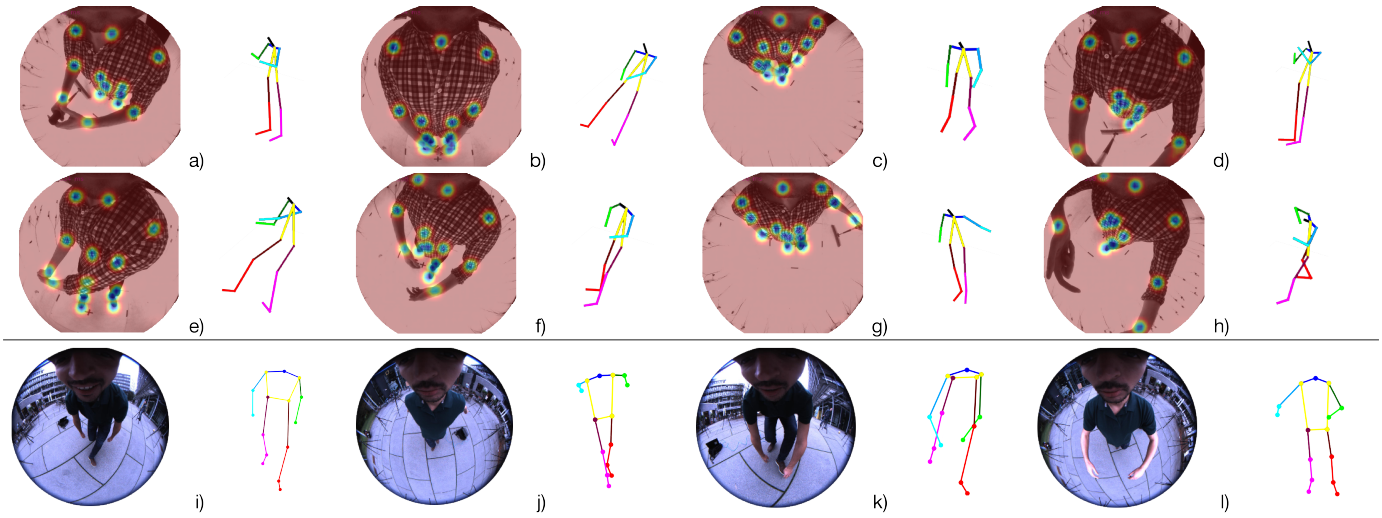

Fig. 10: Qualitative results on real images captured in a studio (top) and reconstructions of images in the wild from Mo^{2}Cap^2 [5] (bottom). Note that the poses are expressed with respect to the camera reference system.

Springer, 2016, pp. 186–201.

[13] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis, "Coarse-to-fine volumetric prediction for single-image 3d human pose," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 1263–1272.

[14] 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 *2017 International Conference on 3D Vision (3DV)*. IEEE, 2017, pp. 506–516.

[15] J. Martinez, R. Hossain, J. Romero, and J. J. Little, "A simple yet effective baseline for 3d human pose estimation," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2017.

[16] F. Moreno-Noguer, "3d human pose estimation from a single image via distance matrix regression," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 1561–1570.

[17] V. Ramakrishna, T. Kanade, and Y. Sheikh, "Reconstructing 3d human pose from 2d image landmarks," in *European Conference on Computer Vision*. Springer, 2012, pp. 573–586.

[18] I. Akhter and M. J. Black, "Pose-conditioned joint angle limits for 3d human pose reconstruction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1446–1455.

[19] X. Zhou, M. Zhu, S. Leonardos, and K. Daniilidis, "Sparse representation for 3d shape estimation: A convex relaxation approach," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 8, pp. 1648–1661, 2017.

[20] X. Zhou, M. Zhu, S. Leonardos, K. G. Derpanis, and K. Daniilidis, "Sparseness meets deepness: 3d human pose estimation from monocular video," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4966–4975.

[21] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it smpl: Automatic estimation of 3d human pose and shape from a single image," in *European Conference on Computer Vision*. Springer, 2016, pp. 561–578.

[22] M. Sanzari, V. Ntouskos, and F. Pirri, "Bayesian image based 3d pose estimation," in *European Conference on Computer Vision*. Springer, 2016, pp. 566–582.

[23] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 975–10 985.

[24] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll, "Video based reconstruction of 3d people models," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[25] G. Rogez and C. Schmid, "Mocap-guided data augmentation for 3d pose estimation in the wild," in *Advances in Neural Information Processing Systems*, 2016, pp. 3108–3116.

[26] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid, "Learning from synthetic humans," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, 2017.

[27] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 6, p. 248, 2015.

[28] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4724–4732.

[29] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European Conference on Computer Vision*. Springer, 2016, pp. 483–499.

[30] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele, "Deepcut: Joint subset partition and labeling for multi person pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4929–4937.

[31] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.

[32] G. Rogez, P. Weinzaepfel, and C. Schmid, "Lcr-net: Localization-classification-regression for human pose," in *CVPR 2017-IEEE Conference on Computer Vision & Pattern Recognition*, 2017.

[33] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, M. Elgharib, P. Fua, H.-P. Seidel, H. Rhodin, G. Pons-Moll, and C. Theobalt, "Xnect: Real-time multi-person 3d human pose estimation with a single rgb camera," *arXiv preprint arXiv:1907.00837*, 2019.

[34] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, S. Sridhar, G. Pons-Moll, and C. Theobalt, "Single-shot multi-person 3d pose estimation from monocular rgb," in *International Conference on 3D Vision (3DV)*, sep 2018.

[35] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman, "Single image 3d interpreter network," in *European Conference on Computer Vision*. Springer, 2016, pp. 365–382.

[36] D. Tome, C. Russell, and L. Agapito, "Lifting from the deep: Convolutional 3d pose estimation from a single image," *CVPR 2017 Proceedings*, pp. 2500–2509, 2017.

[37] H.-Y. F. Tung, A. Harley, W. Seto, and K. Fragkiadaki, "Adversarial inversion: Inverse graphics with adversarial priors," *arXiv preprint arXiv:1705.11166*, 2017.

[38] D. Drover, C.-H. Chen, A. Agrawal, A. Tyagi, and C. P. Huynh, "Can 3d pose be learned from 2d projections alone?" *arXiv preprint arXiv:1808.07182*, 2018.

[39] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis, "Learning to estimate 3d human pose and shape from a single color image," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[40] H.-Y. Tung, H.-W. Tung, E. Yumer, and K. Fragkiadaki, "Self-supervised learning of motion capture," in *Advances in Neural Information Processing Systems*, 2017, pp. 5242–5252.

[41] H. Rhodin, M. Salzmann, and P. Fua, "Unsupervised geometry-aware representation for 3d human pose estimation," *arXiv preprint arXiv:1804.01110*, 2018.

[42] M. Omran, C. Lassner, G. Pons-Moll, P. Gehler, and B. Schiele, "Neural body fitting: Unifying deep learning and model based human pose and shape estimation," in *International Conference on 3D Vision (3DV)*, sep 2018.

[43] A. Kanazawa, J. Y. Zhang, P. Felsen, and J. Malik, "Learning 3d human dynamics from video," in *Computer Vision and Pattern Regognition (CVPR)*, 2019.

[44] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *Computer Vision and Pattern Regognition (CVPR)*, 2018.

[45] T. Alldieck, M. Magnor, B. L. Bhatnagar, C. Theobalt, and G. Pons-Moll, "Learning to reconstruct people in clothing from a single RGB camera," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun 2019.

[46] B. L. Bhatnagar, G. Tiwari, C. Theobalt, and G. Pons-Moll, "Multi-garment net: Learning to dress 3d people from images," in *IEEE International Conference on Computer Vision (ICCV)*. IEEE, oct 2019.

[47] A. Fathi, A. Farhadi, and J. M. Rehg, "Understanding egocentric activities," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011.

[48] M. Ma, H. Fan, and K. M. Kitani, "Going deeper into first-person activity recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1894–1903, 2016.

[49] C. Cao, Y. Zhang, Y. Wu, H. Lu, and J. Cheng, "Egocentric gesture recognition using recurrent 3d convolutional neural networks with spatiotemporal transformer modules," *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[50] H. Yonemoto, K. Murasaki, T. Osawa, K. Sudo, J. Shimamura, and Y. Taniguchi, "Egocentric articulated pose tracking for action recognition," in *International Conference on Machine Vision Applications (MVA)*, 2015.

[51] G. Rogez, J. S. Supancic, and D. Ramanan, "First-person pose recognition using egocentric workspaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4325–4333.

[52] H. Jiang and K. Grauman, "Seeing invisible poses: Estimating 3d body pose from egocentric video," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 3501–3509.

[53] Y. Yuan and K. Kitani, "3d ego-pose estimation via imitation learning," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 735–750.

[54] ——, "Ego-pose estimation and forecasting as real-time pd control," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[55] M. Amer, S. V. Amer, and A. Maria, "Deep 3d human pose estimation under partial body presence," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2018.

[56] H. Rhodin, C. Richardt, D. Casas, E. Insafutdinov, M. Shafiei, H.-P. Seidel, B. Schiele, and C. Theobalt, "Egocap: egocentric marker-less motion capture with two fisheye cameras," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 162, 2016.

[57] D. Tome, P. Peluse, L. Agapito, and H. Badino, "xr-egopose: Egocentric 3d human pose from an hmd camera," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7728–7738.

[58] T. von Marcard, B. Rosenhahn, M. J. Black, and G. Pons-Moll, "Sparse inertial poser: Automatic 3d human pose estimation from sparse imus," in *Computer Graphics Forum*, vol. 36, no. 2. Wiley Online Library, 2017, pp. 349–360.

[59] Y. Huang, M. Kaufmann, E. Aksan, M. J. Black, O. Hilliges, and G. Pons-Moll, "Deep inertial poser learning to reconstruct human pose from sparseinertial measurements in real time," *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, vol. 37, no. 6, pp. 185:1–185:15, nov 2018.

[60] C. Malleson, J. Collomosse, and A. Hilton, "Real-time multi-person motion capture from multi-view video and imus," *International Journal of Computer Vision*, pp. 1–18, 2019.

[61] T. von Marcard, G. Pons-Moll, and B. Rosenhahn, "Human pose estimation from video and imus," *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, jan 2016.

[62] G. Pons-Moll, A. Baak, J. Gall, L. Leal-Taixe, M. Mueller, H.-P. Seidel, and B. Rosenhahn, "Outdoor human motion capture using inverse kinematics and von mises-fisher sampling," in *IEEE International Conference on Computer Vision (ICCV)*, nov 2011, pp. 1243–1250.

[63] T. von Marcard, R. Henschel, M. Black, B. Rosenhahn, and G. Pons-Moll, "Recovering accurate 3d human pose in the wild using imus and a moving camera," in *European Conference on Computer Vision (ECCV)*, sep 2018.

[64] T. Shiratori, H. S. Park, L. Sigal, Y. Sheikh, and J. K. Hodgins, "Motion capture from body-mounted cameras," in *ACM Transactions on Graphics (TOG)*, vol. 30, no. 4. ACM, 2011, p. 31.

[65] https://www.mixamo.com/. (last accessed on 2019-03-19) Animated 3d characters. [Online]. Available: https://www.mixamo.com/

[66] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[67] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[68] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753.

[69] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[70] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[71] K. He, R. Girshick, and P. Dollár, "Rethinking imagenet pre-training," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4918–4927.

[72] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt, "Vnect: Real-time 3d human pose estimation with a single rgb camera," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 44, 2017.

[73] C. S. Catalin Ionescu, Fuxin Li, "Latent structured models for human pose estimation," in *International Conference on Computer Vision*, 2011.
[74] C.-H. Chen and D. Ramanan, "3d human pose estimation= 2d pose estimation+ matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7035–7043.
[75] M. R. I. Hossain and J. J. Little, "Exploiting temporal information for 3d human pose estimation," in *European Conference on Computer Vision*. Springer, 2018, pp. 69–86.
[76] R. Dabral, A. Mundhada, U. Kusupati, S. Afaque, and A. Jain, "Structure-aware and temporally coherent 3d human pose estimation," *arXiv preprint arXiv:1711.09250*, 2017.
[77] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7122–7131.
[78] X. Zhou, M. Zhu, G. Pavlakos, S. Leonardos, K. G. Derpanis, and K. Daniilidis, "Monocap: Monocular human motion capture using a cnn coupled with a geometric prior," *IEEE transactions on pattern analysis and machine intelligence*, 2018.
[79] E. Jahangiri and A. L. Yuille, "Generating multiple diverse hypotheses for human 3d pose consistent with 2d joint detections," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 805–814.
[80] H.-S. Fang, Y. Xu, W. Wang, X. Liu, and S.-C. Zhu, "Learning pose grammar to encode human body configuration for 3d pose estimation," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
[81] X. Sun, J. Shang, S. Liang, and Y. Wei, "Compositional human pose regression," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2602–2611.
[82] H. Yasin, U. Iqbal, B. Kruger, A. Weber, and J. Gall, "A dual-source approach for 3d pose estimation from a single image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4948–4956.
[83] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

Denis Tome is a research scientist at *Epic Games* working at the intersection between computer vision and graphics. Since 2016 he is a Ph.D. candidate and member of the Vision and Imaging Science Group at University College London (UCL) under the supervision of Prof. Lourdes Agapito and Dr. Gabriel Brostow. His research focuses on 3D human pose reconstruction, sparse and dense, from simplistic configuration (monocular) to more complex ones (multi-view), with different applications from robotics to VR/AR for headset mounted camera systems.

Thiemo Alldieck is currently a research intern at *Facebook Reality Labs*. Since 2016 he is a Ph.D. candidate at the Computer Graphics Lab at TU Braunschweig, Germany. Since 2018, he is also affiliated with the "Real Virtual Humans" group at Max Planck for Informatics (MPII) in Saarbrücken, Germany. His work lies at the intersection between computer vision, graphics, and machine learning and focuses on human pose, shape, and clothing reconstruction from monocular images and video.

Patrick Peluse received the BFA degree in Visual Effects/Animation at the Academy Of Art University in 2008. He has worked in computer graphics for over 15 years with current interests in simulation to solve real world problems. Currently, he is a Technical Artist at Facebook Reality Labs where his focus is on both real-time and offline data generation for quality assurance and hardware prototyping. He has worked as video game programmer, UI/UX conceptual designer for AR/VR, and data generator / content creator with companies Autodesk M&E, Digital Domain, Circuits for Fun, Meta Vision, Magic Leap, and Oculus.

Gerard Pons-Moll is the head of the Emmy Noether independent research group "Real Virtual Humans", senior researcher at the Max Planck for Informatics (MPII) in Saarbrücken, Germany, and Junior Faculty at Saarland Informatics Campus. His research lies at the intersection of computer vision, computer graphics and machine learning – with special focus on analyzing people in videos, and creating virtual human models by "looking" at real ones". His work has received several awards including the prestigious Emmy Noether Grant (2018), a Google Faculty Research Award (2019), a Facebook Reality Labs Faculty Award (2018), and recently the German Pattern Recognition Award (2019), which is given annually by the German Pattern Recognition Society to one outstanding researcher in the fields of Computer Vision and Machine Learning. His work got Best Papers Awards at BMVC'13, Eurographics'17 and 3DV'18 and he served as Area Chair for ECCV'18, 3DV'19, SCA'18'19, FG'20 and will serve as Area Chair for CVPR'21, ECCV'20 and 3DV'20.

Lourdes Agapito is Professor of 3D Vision and Head of the Vision and Imaging Science Group in the Department of Computer Science at University College London (UCL). She is also co-founder of London-based startup Synthesia Technologies and was an ERC Grant holder (2008-14). She obtained her BSc, MSc and PhD from the Universidad Complutense (Madrid) in 1991, 1992 and 1996, and was a postdoctoral fellow at the University of Oxford's Robotics Research Group (1997-2001). Her research interests lie at the intersection of computer vision, graphics and machine learning; more specifically 3D reconstruction from video, 3D shape modelling, weakly supervised learning for 3D vision, human pose estimation and video synthesis. She has served as Program Chair for the top computer vision conferences (CVPR'16, ICCV'21), Workshops Chair for ECCV'14 and Area chair for CVPR (3x), ECCV (2x), ICCV (1x). She is associate editor of IEEE PAMI and IJCV. She was keynote speaker at ICRA'17.

Hernan Badino is a research scientist at Facebook Reality Labs. He received his PhD degree in Computer Sciences from the Goethe University Frankfurt in 2008. During his PhD, he worked with the Image Based Environment Perception Group at Daimler AG and joined the Carnegie Mellon University in 2009 as a postdoctoral fellow. In 2012, he was appointed a faculty position at the Robotics Institute at the Carnegie Mellon University where he worked on visual based localization, sensor-fusion, ego-pose estimation, object detection, and tracking, and real-time embedded solutions for visual-based pose estimation. He joined Facebook Reality Labs in 2015 and has since then been working at the intersection of computer vision, motion capture, and telepresence systems towards the goal of achieving social presence in artificial reality.

Fernando de la Torre received the BSc degree in telecommunications, and the MSc and PhD degrees in electronic engineering from the La Salle School of Engineering at Ramon Llull University, Barcelona, Spain in 1994, 1996, and 2002, respectively. He is an associate research professor in the Robotics Institute at Carnegie Mellon University. His research interests are in the fields of computer vision and machine learning. Currently, he is directing the Component Analysis Laboratory (http://ca.cs.cmu.edu) and the Human Sensing Laboratory (http://humansensing.cs.cmu.edu) at Carnegie Mellon University. He has more than 100 publications in refereed journals and conferences. He has organized and co-organized several workshops and has given tutorials at international conferences on the use and extensions of component analysis.